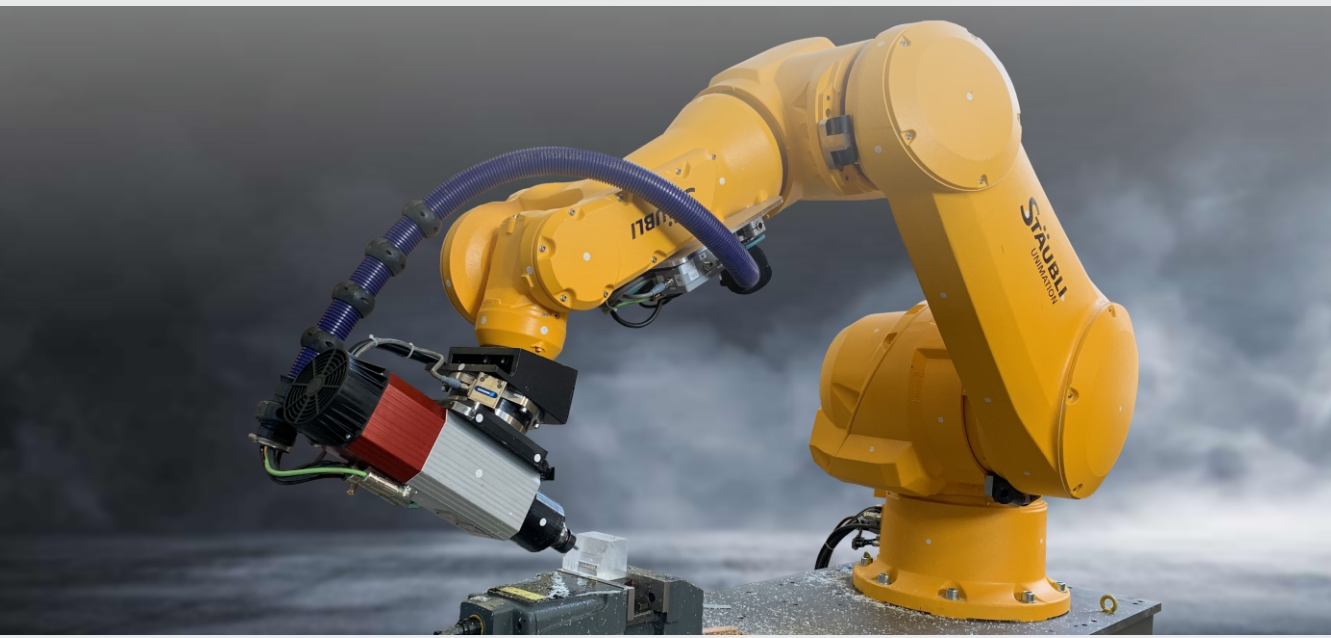
PhD Thesis
2019

ROBOTIC MACHINING

Development and validation of a numerical model of robotic milling in order to optimise the cutting parameters

Hoai Nam HUYNH



UMONS
Université de Mons

POLYTECH
MONS

UNIVERSITY OF MONS
FACULTY OF ENGINEERING

Department of Theoretical Mechanics, Dynamics and Vibrations
Boulevard Dolez, 31 | B - 7000 Mons (Belgium)

Thesis submitted in fulfilment
of the requirements for the degree of
Doctor in Engineering Sciences and Technology

Hoai Nam Huynh

Robotic machining

Development and validation of a numerical model of robotic milling to optimise the cutting parameters

Jury:

Prof. Sélim Datoussaïd — UMONS (Chairman)
Prof. François Ducobu — UMONS (Secretary)
Prof. Olivier Verlinden — UMONS (Supervisor)
Prof. Edouard Rivière-Lorphèvre — UMONS (Co-supervisor)
Prof. Keivan Ahmadi — University of Victoria
Prof. Olivier Brüls — ULiège
Prof. Vincent Gagnol — Institut Pascal - Université Clermont Auvergne
Prof. Alain Vande Wouwer — UMONS

September 2019

The only perfect model is the reality ...

Acknowledgments

I express my gratitude to everyone who supported me throughout the four years of this Ph.D. thesis.

I gratefully thank my supervisor, Prof. Olivier Verlinden, for his continuous support and guidance, careful listening, thorough proofreading and encouragements. His valuable advice largely contributed to the presented research work.

I also gratefully acknowledge my second supervisor, Prof. Edouard Rivière-Lorphèvre, for his relevant comments and availability. Always ready to help, his assistance was decisive during the first milling experiments with the robot and throughout the project.

I would like to express my very great appreciation to Dr. Keivan Ahmadi who welcomed me in his Dynamics and Digital Manufacturing research laboratory at the University of Victoria in Canada during three months. His invaluable suggestions and feedback strongly influenced the writing of this thesis. The assistance and hospitality of his research team illuminated views on a number of issues related to the project, special thanks to Anahita Habibian for bringing fun during the experimental modal analysis tests, to Hamed Assadi for his bright ideas regarding model updating and to Yaser Mohammadi for sharing his 3D digital ZOA code.

Many thanks to my colleagues from the Department of Theoretical Mechanics, Dynamics and Vibrations, especially to Bryan Olivier, for kindly reviewing the third chapter of this thesis, to Prof. Georges Kouroussis, for his great assistance in conducting experimental modal analyses on the robot and to Lassaad Ben Fékih, who shared his office with me.

I would also like to extend my thanks to the technicians, in particular to Benjamin Demaret, for his help during the tensile tests on the robot, to Kévin Nis, for his technical support regarding the measuring devices, to Laurent Vergari, for the set up of the industrial robot and to Régis Berton, for the manufacturing of the test specimens.

I wish to acknowledge Lander Debruyne and his colleagues for letting me familiarise with robotic milling in his company, CNC Solutions - VDS. I would also like to thank Benjamin Parent and Laurent Mattheys for their great support leading to a close collaboration with Stäubli.

Finally, I would like to thank my parents for their faithful support and perpetual encouragements during all my education.

The research leading to these results has received complete funding from the National Scientific Research Fund (F.R.S.-FNRS) under a FRIA (*Fund for Research training in Industry and Agriculture*) grant.

Abstract

Using industrial robots as machine tools is targeted by many industrials for their lower cost and larger workspace. Nevertheless, performance of industrial robots is limited due to their serial mechanical structure involving rotational joints with a lower stiffness. As a consequence, vibration instabilities, known as chatter, are more likely to appear in industrial robots than in conventional machine tools. Commonly, chatter is avoided by using stability lobe diagrams which determine the stable combinations of axial depth of cut and spindle speed. Although the computation of stability lobes in conventional machine tools is a well-studied subject, developing them in robotic milling is challenging because of the lack of accurate dynamic multibody models involving joint compliance to predict the posture-dependent dynamics of the robot.

In this work, the stability lobe diagrams of milling operations are computed in the time domain using a dynamic multibody model of the Stäubli TX200 robot, which is a six-axis serial robot. Since past studies revealed that the flexibility of industrial robots mainly originated from its joints, the multibody model of the Stäubli TX200 robot comprises joints with torsional and transversal compliances which represent the transmission and bearing flexibilities, respectively. Variants of the multibody model are developed in order to assess the influence of the link and controller flexibilities. A dynamic milling model is coupled to the simulated robot allowing the computation of the cutting forces and a representation of the virtual machined surface.

Once the inertia characteristics of the robot are identified, its elastic parameters are fitted to modal measurements obtained via experimental modal analysis. In particular, a straightforward identification method, relying on the curve fitting of the tool tip frequency response functions, is proposed to determine the elastic parameters of the multibody model. Parameters pertaining to the flexible links are determined by matching finite element models while the control parameters are settled based on the modal measurements. Cutting force coefficients are classically identified through milling experiments.

Using the robotic milling simulator validated in stable cutting conditions, stability lobe diagrams are simulated and experimentally validated. The influence of the feed direction and the aforementioned flexibilities is appraised on the system stability. The robotic milling simulator could reasonably capture the overall stability limits in all feed directions in aluminium and in steel. Appending the flexibility of the controller or the links did not lead to significant modification in the prediction of the stability charts. However, their consideration is important when dealing with static deflections. It was indeed shown that the consideration of flexible links induced 20 to 30 % of additional deflections while the ones introduced by the controller were almost negligible. Since robot structural modes depend on its posture, it appeared that stability limits were affected by the feed direction. It was also observed that in low-speed milling, mostly the low-frequency pose-dependent robot modes chatter while in high-speed milling robot wrist modes trigger the instability.

Contents

Notations[1]

Latin letters

a_{e}	Radial depth of cut
a_{p}	Axial depth of cut
$\mathbf{a}_{\mathrm{i}}$	Vector of translational acceleration of body i or node i
A	Area of the cross section of a beam element
$\mathbf{C}$	Damping matrix
$\mathbf{C}_{\mathrm{i}}$	Vector locating the centre of mass of link i in frame $O_{\text{i-1}}$ in the augmented link model
$d_{x\|y\|z,\mathrm{i}}$	Joint damping around the X, Y or Z axis of joint i
$\mathbf{d}^{\mathrm{i,j}}$	Vector of partial contribution of $\dot{q}_{\mathrm{j}}$ in the linear velocity of body i
$\mathbf{D}$	Virtual damping matrix
E	Young's modulus
$f()$	Direct kinematics function
f_{c}	Chatter frequency
f_n	Natural frequency of mode n
$\mathrm{f_z}$	Feed per tooth
$\mathbf{F}_{\mathrm{ext}}$	Vector of external forces
$\mathbf{F}_{\mathrm{c}}$	Vector of cutting forces
$\mathbf{F}_{\mathrm{s}}$	Static friction matrix
$\mathbf{F}_{\mathrm{v}}$	Viscous friction matrix
$\mathbf{g}_0$	Gravity vector
G	Coulomb's modulus
$\boldsymbol{h}$	Vector of dynamic chip thickness
$\mathbf{H}(\omega)$	Frequency response function matrix
k_{i}	Reduction ratio of gearbox for joint i
$k_{x\|y\|z,\mathrm{i}}$	Joint stiffness around the X, Y or Z axis of joint i
$\mathbf{k}$	Diagonal matrix containing all the reduction ratios
$\mathbf{K}$	Stiffness matrix
$K_{\mathrm{t\|r\|a\ c}}$	Cutting force coefficients with t the tangential, r the radial and a the axial directions
$K_{\mathrm{t\|r\|a\ e}}$	Edge force coefficients with t the tangential, r the radial and a the axial directions
$\mathbf{K}_{\mathrm{c}}$	Cutting force coefficient matrix
L	Length of a beam element
L_{Beam}	Total length of the beam
m_{i}	Mass of link i

[1] Vector and matrix are printed in bold text.

$m_{\mathrm{m_i}}$	Mass of motor i
$\mathbf{M}$	Mass matrix
$\boldsymbol{\mathcal{M}}_{\mathrm{G,i}}$	Vector of resultant moment of all applied torques exerted at the centre of mass of body i
n	Number of degrees of freedom for the considered multibody model
n_{B}	Number of rigid body(ies) included in the multibody model
n_{F}	Number of flexible body(ies) included in the multibody model
n_θ	Number of joint(s) of the robotic manipulator
N	Number of node(s) belonging to one flexible body
$\mathrm{N_s}$	Number of slices discretising the cutting tool
$\mathrm{N_z}$	Number of teeth pertaining to the cutting tool
O_{Base} (x_{Base}, y_{Base}, z_{Base})	Base frame
O_{i} (x_{i}, y_{i}, z_{i})	Mobile frame i
$\mathbf{p}$ (p_x, p_y, p_z)	Position vector
$\mathbf{q}$	Vector of instantaneous degrees of freedom of the selected model
$\mathbf{q}_{\mathrm{f}}$	Vector of instantaneous degrees of freedom representing the flexibility of all links
$\mathbf{q}_{\mathrm{f.g}}$	Vector of instantaneous degrees of freedom representing the flexibility of node g of a flexible body f
$\mathbf{q}_{\mathrm{m}}$	Vector of degrees of freedom representing the instantaneous position of the motor shaft seen from the motor side
$\mathbf{q}_{\mathrm{s}}$	Vector of degrees of freedom representing the instantaneous position of the motor shaft seen from the joint side
$\mathbf{q}_{x\|y}$	Vector of instantaneous degrees of freedom representing the flexibility orthogonal to the motion axis
$\mathbf{q}_z$	Vector of instantaneous degrees of freedom representing the flexibility around the motion axis
$\mathbf{R}_{\mathrm{i,j}}$	Rotation matrix of frame j in frame i
$\boldsymbol{\mathcal{R}}_{\mathrm{i}}$	Vector of resultant force of all applied forces exerted at the centre of mass of body i
t	Time
$\mathbf{T}_{\mathrm{i,j}}$	Homogeneous transformation matrix of frame j in frame i
$\mathcal{T}$	Kinetic energy
$\mathcal{U}$	Potential energy
$\mathbf{v}_{\mathrm{i}}$	Vector of translational velocity of body i or node i
$\mathbf{w}^{\mathrm{i,j}}$	Vector of partial contribution of $\dot{q}_{\mathrm{j}}$ in the rotational velocity of body i

$\mathbf{x}_\mathrm{i}$	Vector describing the Cartesian pose of frame i

Greek letters

α_Damping	Mass proportional Rayleigh damping coefficient
β_Damping	Stiffness proportional Rayleigh damping coefficient
$\mathbf{\Gamma}$	Combination of actuator torques $\boldsymbol{\tau}$ and external torques $\boldsymbol{\tau}_\mathrm{ext}$
ϵ	Phase shift
ε	Dynamic part of chip thickness
ζ_n	Damping ratio of mode n
θ_i	Desired motor angle of joint i seen from the joint side
$\theta_\mathrm{m,i}$	Desired motor angle of joint i seen from the motor side
$\mathbf{\Theta}$	Vector of desired joint angles seen from the joint side
ν	Poisson's ratio
ρ	Density
$\boldsymbol{\tau}$	Vector of actuator torques
$\boldsymbol{\tau}_\mathrm{ext}$	Vector of external torques
Υ	Angle orienting the spindle axis
φ_j	Instantaneous angular immersion of tooth j
ϕ	Component of the inertia tensor
$\boldsymbol{\phi}$	Orientation vector
$\mathbf{\Phi}_\mathrm{G,i}$	Central inertia tensor at the centre of mass of body i
$\mathbf{\Phi}_\mathrm{G,m_i}$	Tensor of inertia of motor i
$\boldsymbol{\omega}_\mathrm{i}$	Vector of rotational velocity of body i or node i
Ω	Spindle speed

Set of symbols

$(\alpha_\mathrm{EE}, \beta_\mathrm{EE}, \gamma_\mathrm{EE})$	XYZ Euler angle orienting the end effector
$(a_\mathrm{i}, \alpha_\mathrm{i}, d_\mathrm{i}, \theta_\mathrm{i})$	Standard Denavit-Hartenberg parameters for joint i
$(d_\mathrm{i\text{-}1}, \alpha_\mathrm{i\text{-}1}, r_\mathrm{i}, \theta_\mathrm{i})$	Modified Denavit-Hartenberg parameters for joint i

List of abbreviations

Back EMF	Back Electromotive Force
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CNC	Computer Numerical Control
DDE	Delay Differential Equation
DH	Denavit-Hartenberg
DOF	Degree Of Freedom

DVA	Dynamic Vibration Absorber
EE	End Effector
EMA	Experimental Modal Analysis
FFT	Fast Fourier Transform
FRF	Frequency Response Function
HPM	High Performance Machining
LSCE	Least Squares Complex Exponential
LSE	Least Squares Estimation
MBS	Multibody System
MDOF	Multiple-Degree-Of-Freedom
MIMO	Multi-Input Multi-Output
MRR	Material Removal Rate
ODE	Ordinary Differential Equation
PC	Personal Computer
PMSM	Permanent Magnet Synchronous Motor
rev/min	Revolution per minute
RMSE	Root Mean Square Error
SDm	Semi-Discretisation method
SDOF	Single-Degree-Of-Freedom
TCP	Tool Centre Point
VJM	Virtual Joint Modelling
ZOA	Zero-Order Approximation

Chapter 1

Introduction

1.1 Context, issues and motivations

The increasing demand of high added value manufactured parts is a key factor in industrial development. The current trend aims at reducing the number of parts in mechanical assemblies by selecting advanced materials and enhanced geometries. Providing strengthened mechanical properties for the final product, the production of optimised parts introduces more complexity in the manufacturing process. Hence, industrial processes are constantly revised to satisfy the modern constraints and requirements. Specifically in machining, required high accuracy and high productivity drive the industrials to reconsider the design and programming of their manufacturing cells.

Machining denotes any process involving a raw blank material being cut into a desired shape while controlling the material removal process. One of the main objectives consists in increasing the material removal rate without altering the quality of the product and with a reasonable tool wear. While CNC (Computer Numerical Control) machine tools offer the best accuracy and productivity, the use of industrial robots can turn out to be an economic asset for workpiece with complicated geometry and large dimensions. This new technology, encouraged by increasing process automation, looks very promising knowing that industrial robots are less expensive than CNC machining centres with the same operational volume. Moreover, due to easier set-up, robots can be easily integrated in existing production flow. Figure 1.1 illustrates the concept of robotic milling. An industrial robot performs a deburring operation on a car door to trim the edge to its nominal dimensions.

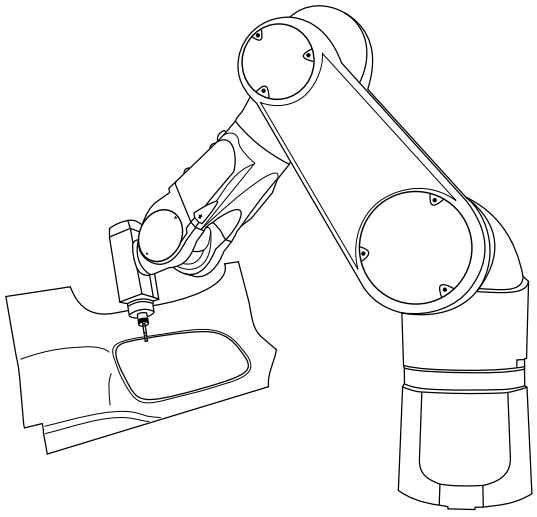


Figure 1.1: Deburring of a car door using an industrial robot

However, industrial robots are significantly less stiff than conventional machine tools and therefore cannot be used in all machining applications. Indeed, most of industrial robots are designed as open serial chains with rotational joints which lower their rigidity. Consequently their static and dynamic deflections are likely to cause dimensional errors and poor surface finish. Milling forces applied at the tool tip can lead to significant trajectory deviation resulting in unacceptable dimensional tolerance violations. Although compensation techniques exist at the finishing and roughing stages for the static deflections, the productivity of the milling process is inherently limited by self-excited vibrations, better known as *chatter*, causing, in severe cases, waves and asperities on the surface of the workpiece, a shorter tool life and damages to the spindle. Figure 1.2 depicts the undesirable effects of chatter on workpiece surfaces: Figure 1.2a in chatter free conditions and Figure 1.2b in unstable conditions. Succinctly, chatter arises when one of the dominant structural modes of the mechanical system is excited by the process forces.

(a) Stable conditions: smooth lateral surface [1]

(b) Chatter conditions: wavy lateral surface [1]

Figure 1.2: Effects of chatter on machined surfaces

While extensive research has been carried out in machining to predict chatter for CNC machine tools, only a few studies have focussed on its origin in robotic-based milling process. Chatter prediction methods have been developed in the frequency and time domains and are mainly based on the dynamic response of the mechanical system at the tool tip. Traditionally, the frequency response functions (FRFs), providing the dynamic behaviour of the mechanical system, are measured at the tool tip and are assumed to be constant in the entire workspace of the machine tool. Then, in combination with the identified parameters of the cutting force model, cutting conditions leading to chatter can be predicted and avoided. Typically, so-called *stability lobe diagrams* are computed using the chatter prediction methods aiming at delivering a quick overview of cutting conditions generating unstable behaviours. Figure 1.3 shows a typical stability lobe diagram. The latter is the stability boundary and separates stable chip thickness-spindle speed combinations (below the boundary, marked as a circle) from unstable pairs (above the boundary, marked as a cross) for milling operations. These diagrams are very useful in machine shops to determine the cutting conditions leading to the best productivity while maintaining the part quality and avoiding chatter conditions. Large axial depths of cut and high spindle speeds are preferably used to increase the productivity, under the constraints of the machine tool limitations.

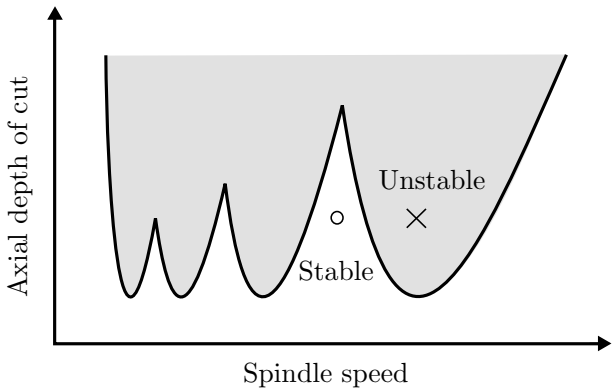


Figure 1.3: Stability lobe diagram concept

In robotic milling however, the computation of stability lobe diagrams is still challenging. Due to the higher compliance of industrial robots, frequency response functions can no longer be assumed constant at the tool tip and are therefore dependent on the posture of the manipulator. As it was observed that those highly flexible and pose-dependent structural modes can cause chatter, they need to be considered in the stability prediction for milling applications. In order to model the pose-dependent structural modes, multi-body dynamic model of robot can be developed to predict its dynamic behaviour in its entire workspace. Although such a model would be ideal to predict the varying dynamic behaviour of the robot, its development involves the identification of many parameters often enviously kept by robot manufacturers. In particular, since the robot flexible joints are mostly accountable of its compliance, emphasis should be brought to a detailed model of the transmission to be able to predict the natural frequencies and damping of the robot in any posture. With such a model, one could theoretically predict the stability behaviour of any robotic-based milling operation.

1.2 Objective

The main objective of this research project is to develop and validate a numerical model of robotic milling to optimise the cutting parameters. Dynamic simulations of robotic-based milling operations need to be computed to predict the stability of the process. The resulting stability lobe diagrams thus allow selecting optimised cutting conditions ensuring a trade-off between productivity, stability and accuracy. Such a model is naturally an asset since the delicate choice of cutting conditions is carried out off-line, without the need to run actual milling experiments except when identifying the cutting coefficients. At present, off-line robot-oriented computer-aided manufacturing (CAM) software are only limited to kinematic simulations to generate the tool path.

Developing realistic dynamic simulations of robotic milling involves three major steps:

1. The development of a model of the robotic milling process in which the spindle is manipulated by the robot while the workpiece is fixed. It involves the definition of dynamic equations describing the manipulator dynamics as well as the milling

process.

2. The identification of the model parameters, either for the robot model or for the milling process, by means of model identification techniques or from manufacturer's data when available.

3. The validation of the robotic machining model on the basis of measurements. Typical motions are performed on the real robot to validate the identified dynamic parameters and milling experiments are achieved to compare the stability limits of the model with the measured stability lobe diagrams.

In this thesis, these three steps are carried out for the Stäubli TX200 industrial robot.

1.3 Contributions

Although the modelling and identification of industrial robots and the understanding of milling phenomena are individually studied in the literature, the thesis brings in several new contributions which can be listed as follows:

- The complete modelling of an industrial robot is carried out using the multibody approach. The kinematic model of the robot is first settled in addition to its equations of motion taking into account the inertia effects, the non-linear terms from the Coriolis, centrifugal and gyroscopic effects and the gravity. Moreover, as the structural compliance of industrial robots is a key factor in milling stability, the main flexibility sources are integrated in the model, namely the joint flexibility, the link flexibility and the controller.

- A new model is developed to take into consideration the flexibility of the links. The new model was developed in accordance with the type of coordinates describing the kinematics of the multibody model. It can accurately describe the flexibility of beam elements for small deformations.

- The multibody dynamic model of the robot is coupled with a pre-developed milling model. As a verification, the coupled environment is successfully used to reproduce results obtained through milling experiments with conventional machine tools.

- The identification of robot parameters is achieved by using dedicated methods either for the link inertia or for the joint flexibility. Model parameters are compared with manufacturer's data when available and the prediction of posture-dependent modes is addressed. A new straightforward method is developed in order to quickly identify the joint stiffness and joint damping of any dynamic multibody model. Also, the parameters of the cutting force model are identified through milling experiments.

- A database of measurements from robotic milling experiments is constituted using an industrial robot. Measurements include recordings of motion and force measured during the milling operations as well as metrological analyses of the resulting machined workpieces.

- The stability lobe diagrams are computed for different materials and feed directions using the robotic machining simulator and later experimentally validated. The diagrams are compared and discussed with their counterpart obtained with the classical approaches developed for machine tools.

- Finally, a study regarding the consideration of individual sources of flexibility is discussed. The idea is to determine how detailed the model has to be in order to properly capture all the important phenomena related to robotic milling.

1.4 Outline of the thesis

This research work comprises three main parts. Starting with a summary of the state of the art of the robotic machining technology (Chapter 2), the first part (Chapters 3 and 4) presents the model behind the robotic machining simulator. The modelling of the industrial robot is treated separately from the milling model. The different modelling options are analysed and choices are justified keeping in mind the main objective of this work, the prediction of stability lobe diagrams in robotic milling. In the second part (Chapter 5), the identification of model parameters is addressed through experiments using an industrial robot equipped with a spindle, either for the robot model or the milling model. Lastly, the third part (Chapters 6 and 7) concerns the validation of the robotic machining simulator. All the modelling options are assessed and stability lobe diagrams are derived in several milling cases and for different materials.

Chapter 2 gives an overview of the state of the art in robotic machining. Even though the technology is quite new, it has been the subject of a strong activity in recent years thanks to the industrial demand. The chapter goes through the history of robots in manufacturing technologies. An overview of robotic-based milling architectures is presented as well as the current industrial applications. Various areas of research related to robotic machining are listed and briefly presented.

The equations of motion of the multibody dynamic model representing the robot are derived in Chapter 3. They are constructed according to the minimal coordinates and involve the link inertia, the Coriolis, centrifugal and gyroscopic effects and the gravity. All the modelling options are eventually exposed ranging from the consideration of the joint and link flexibilities to the controller compliance. The development of the robot model is carried out in a released multibody framework called `EasyDyn`.

The modelling of the milling process is treated in Chapter 4. Both time domain and frequency domain approaches leading to the determination of stability lobe diagrams are presented. In particular, the cutting force model is derived without losing sight of the workpiece and tool representations. A concern of prime importance is also covered in this chapter. It is in regard to the detection of intersections between the tool cutting edges and the workpiece in order to estimate the chip thickness and therefore simulate the material removal process in the time domain. Milling simulations in the time domain

are based on a pre-developed routine called `Dystamill`.

In Chapter 5, identification methods are implemented in order to determine the model parameters of the multibody dynamic model and the milling model. A rigid body identification method is used to appraise the inertia parameters of the robot links. Joint torques, recorded on optimised joint trajectories, permit to estimate, by Least Squares Estimation (LSE), the combinations of inertial parameters that let the model faithfully reproduce the experimental results. In order to identify the joint stiffness and joint damping, experimental modal analysis techniques are used to measure natural frequencies, mode shapes and an estimation of the modal damping of the actual robot. A new fitting method relying on a FRF fitting is presented so as to determine the joint stiffness and joint damping of any multibody model. The prediction of the posture-dependent modes is addressed by comparing the simulated modes in different postures with their experimental counterparts. Thereafter, the elastic parameters related to the modelling of flexible links are identified on the basis of finite element models. Controller gains are set by considering the modal behaviour of the actual robot. Lastly, regarding the parameters of the milling model, they are determined on the basis of a fitting of the cutting force measurements.

Chapter 6 is devoted to the validation of the robotic machining environment. The coupling between the multibody dynamic model and the milling model is discussed. In particular, the interaction of the two models during the numerical process is detailed. Under the influence of the cutting forces applied on the robot and the progression in the material, workpiece geometry changes, which implies a computation of the cutting forces from the updated geometry of the workpiece. The validation of the coupled environment is carried out by replicating results obtained through experiments with conventional machine tools. Once the coupling verified, milling experiments are carried out with the actual robot. Cutting forces, vibrations and metrological features of the machined workpiece are discussed and compared with the simulated counterparts for different materials and milling cases.

Chapter 7 is dedicated to the stability analysis in robotic milling. The stability of the robotic milling system is solved in both frequency and time domains. Frequency domain solution is provided using the measured FRFs at the tool tip for a defined robot posture. Coupled multibody dynamic model with the milling process offers the possibility to generate the stability lobe diagrams from time domain simulations involving tool motions in the material. Simulated stability charts are then compared with their experimental counterparts. Lastly, the incorporation of the different flexibility sources in the model is assessed by superimposing the resulting stability lobe diagrams for different materials and feed directions.

A summary of the research work is provided in the last chapter and recommendations for future work are issued. The outline of the thesis is illustrated in Figure 1.4.

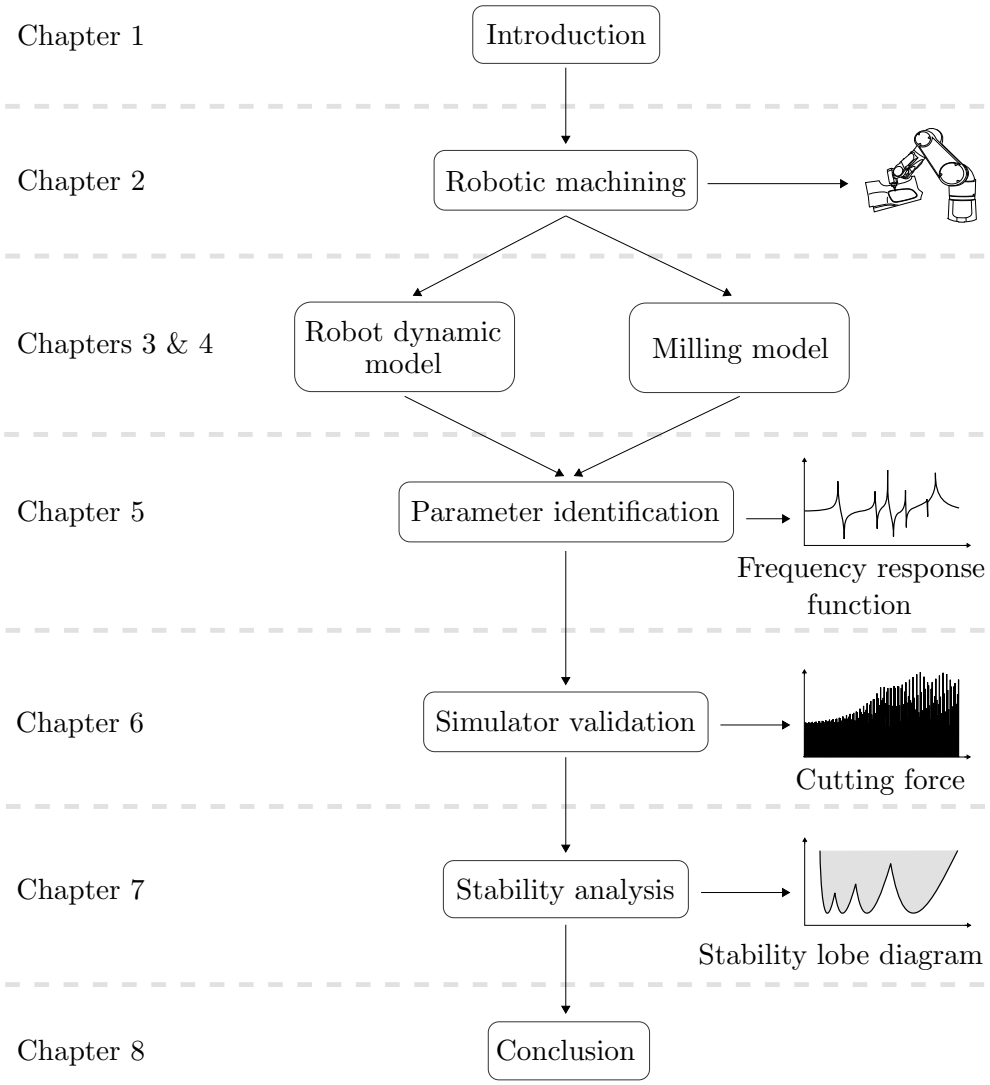


Figure 1.4: Thesis outline

Chapter 2

Robotic machining

Robotic machining is sometimes considered as a hobby but the technology constantly convinces more and more industrials and researchers. The growing interest aligns with the exponential trend of robot integration in factories. Versatile, multifunctional, reprogrammable, robots are the ideal candidates to boost productivity. In manufacturing, robots can be asked to perform repetitive tasks with a constant quality and reliability, programmed for continuous production, customised to execute complex functions and used to work in hazardous environments or achieve unsafe duties. Typical applications of robots include welding, painting, assembling, handling, packaging, labelling, palletizing, inspecting, testing and pick and place. Even though robots are considered as a keystone of today's competitive manufacturing, there are still challenges to solve for manufacturers to efficiently respond to changing consumer behaviour and global shifts in competitiveness.

Focussing on robotic machining, this chapter aims at providing an up-to-date review on the topic through its industrial establishment, the common applications, the current research status and the issues. Robotics is a branch of technology that deals with the design, construction, operation and application of robots. It therefore refers to anything involving robots. By definition, robots are programmable machines which are able to carry out a series of actions autonomously or semi-autonomously. They interact with the physical world by means of sensors and actuators and since they are programmable, multitude of various tasks can be carried out. As a result, robots are used as a flexible way to automate a physical task or process. On the other hand, machining remains a staple of manufacturing technology for mechanical parts with tight tolerances. It is still competitive thanks to steady progress in the machine design, tooling, control and materials. Machining is a manufacturing process that involves cutting a piece of raw material into a desired final shape by a controlled material-removal process. In contrast with the additive manufacturing technology, machining belongs to the category of subtractive manufacturing in which portions of material are removed to shape the part, commonly called *workpiece*. Metallic materials are especially targeted by machining operations which often aim at rendering the final dimensions to the parts previously cast, forged, stamped, welded, assembled or obtained via additive techniques. As it is often the case in the aerospace industry, lightweight and high-precision parts are obtained from a block of raw material from which up to 95 % of material are being cut out (Figure 2.1). In order to remove such quantity of material in acceptable cycle time, high performance machining (HPM) appears to be the suitable solution for high removal rate and reduced cutting forces.

Figure 2.1: Milling of a centrifugal impeller [2]

Limiting the considerations to material removal operations by chip cutting, machining operations are classified into three major processes, namely milling, turning and drilling. Other operations fall into miscellaneous categories and include (but are not limited to) polishing, deburring, grinding, chamfering, trimming, cutting, threading and boring. All the above operations are frequently attempted using robotic systems and are therefore worth to be defined. Figure 2.2 illustrates an overview of machining operations that can be carried out by robots.

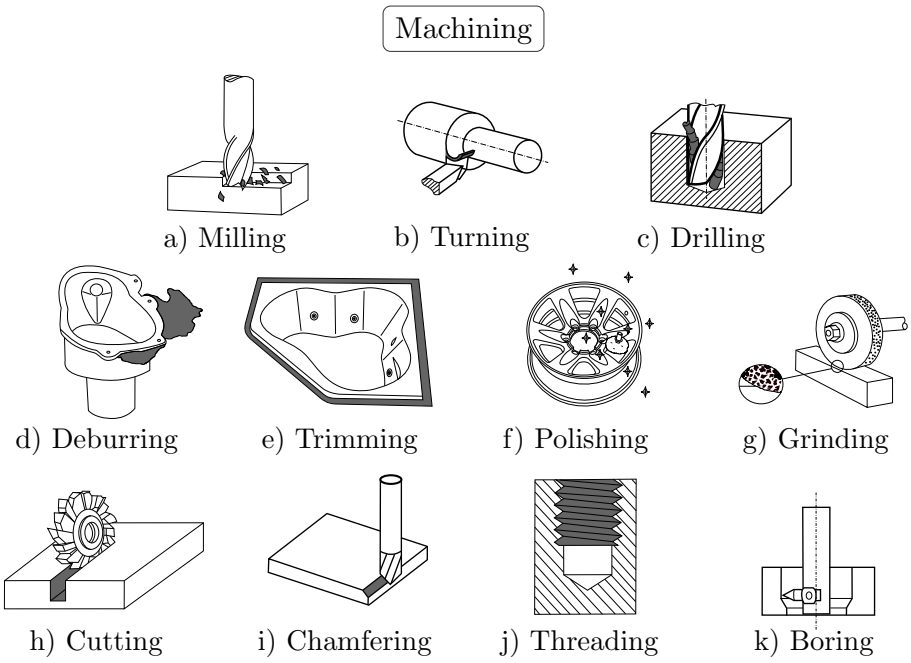


Figure 2.2: Machining operations suitable for robotic systems

Ranked in order of demand:

a) Milling: it is a cutting process using a milling cutter to remove material from the surface of a workpiece. The rotary milling tool, often constituted of several teeth, performs the cutting motion by rotating around its revolution axis. The cutter usually moves perpendicularly to its axis so that cutting occurs on the circumference of the tool. The speed at which the cutter advances through the workpiece is called *feed rate*, or simply *feed*. As the milling cutter enters the workpiece, the cutting edges of the tool periodically cut into and exit from the material, tearing away chips from the workpiece at each pass. The chip formation involves a shear deformation.

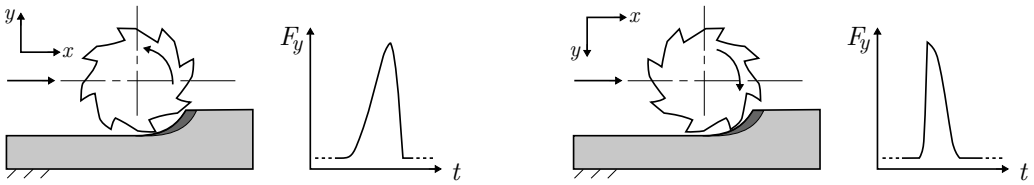
There are two major forms of milling (Figure 2.3):

(a) Peripheral milling (b) Face milling

Figure 2.3: Milling operations

- Peripheral milling: the cutting action occurs primarily around the circumference of the cutter (Figure 2.3a).
- Face milling: the main cutting action takes place at the tool tip of the milling cutter (Figure 2.3b).

Other important considerations in milling concern the engagement direction of the milling cutter relatively to the workpiece. More specifically, it is related to the direction of rotation of the cutter against the feed. Two cases are possible (Figure 2.4):



(a) Up milling direction (b) Down milling direction

Figure 2.4: Milling directions

- Up milling (conventional) direction (Figure 2.4a): the tool rotates against the direction of feed. As the tool moves into the material, the chip thickness increases with the cutter rotation. It means that the chip load on teeth (or uncut chip thickness) gradually increases from zero at the entry point to maximum at the exit point. Cutting forces follow the same trend since they are approximately proportional to the chip thickness. As a result, cutting edges can rub against the surface before engaging in the cut. Hence, heat generation and work hardening can occur and the surface finish is usually poor. Back in the days, up milling direction was preferred in conventional machine tools for backlash compensation.
- Down milling (climbing) direction (Figure 2.4b): the tool rotates in the same direction as the feed. As the tool advances into the workpiece, the chip thickness decreases with the cutter rotation. The chip load on teeth gradually decreases from maximum at the point of engagement to zero at disengagement. Cutting

forces are directed downwards and tend to press the workpiece in its fixture. No rubbing action takes place and therefore material hardening is reduced. Better surface finish can be generated using the down milling direction.

b) Turning: it is a cutting process in which the workpiece continuously rotates while a cutter, fixed or moving, removes material on its surface. It is used to generate rotational parts.

c) Drilling: it is a cutting process in which a rotating drill bit enters into the workpiece to make a hole. As opposed to milling in which the tool moves perpendicularly to its axis, the drill bit advances along its rotation axis.

d) Deburring: the deburring operation is usually carried out in foundry industry to remove excess of material after the casting. Sprues, runners, flashes and other material excess are cut out using a cutting tool.

e) Trimming (contouring): in the trimming operation, process excess of material is removed and the part is set to its final dimensions. The difference with deburring is that the cut is continuous all along the contour.

f) Polishing: it is a finishing process for smoothing the surface of workpiece by using an abrasive wheel. It is usually used to beautify the appearance of a part and to prevent contamination or corrosion.

g) Grinding: it is a finishing operation that improves the surface quality. An abrasive wheel rotates and cuts small chip from the workpiece via shear deformation. Hence, high accuracy of shape and dimensions are achieved.

h) Cutting: it is a cutting process in which a rotating circular cutter blade separates workpiece into some parts.

i) Chamfering: it is a finishing process in which a cutting tool cuts sharp edges to usually create 45° angle.

j) Threading (tapping): it is a cutting process which aims at producing a helical ridge from an existing hole in a workpiece. The resulting thread is used to fasten screws.

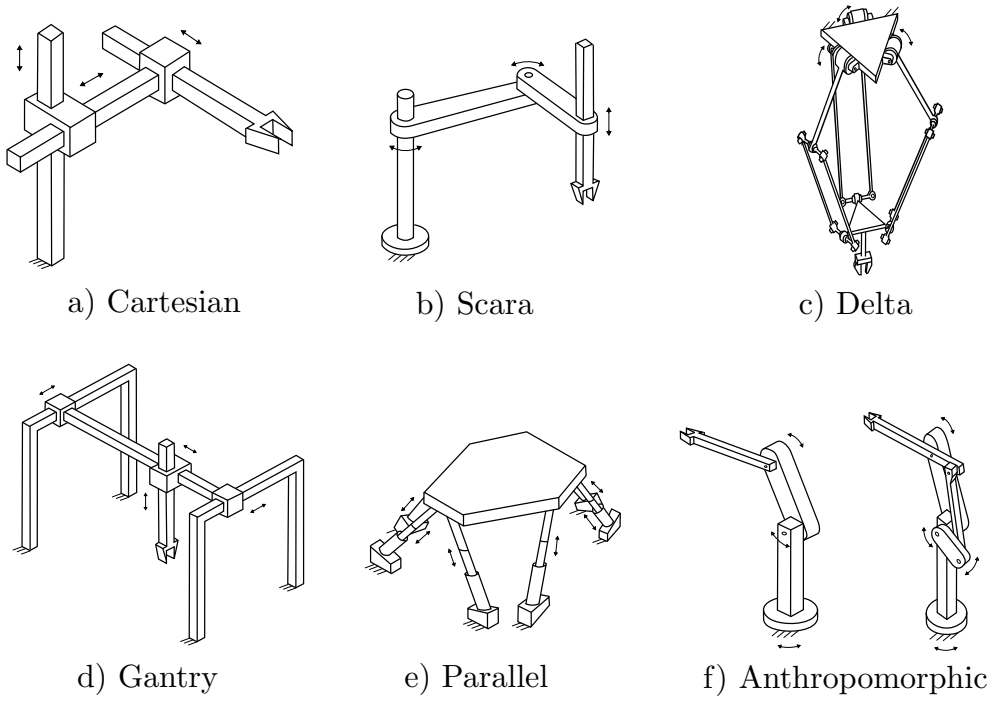
k) Boring: it is a finishing process in which the cutting tool enlarges an existing hole to its nominal dimensions.

Having defined the robotic machining technology, the first section will provide an overview of its evolution over the past decades.

2.1 Background

What differentiates one robot from another one is its mechanical structure. Whether its base is mobile or not, robots can be categorised into robot manipulators or mobile robots. Focussing on robot manipulators for robotic machining applications, their mechanical

structure consists of a sequence of rigid bodies, so-called *links*, interconnected by means of translational or revolute articulations, so-called *joints*. Looped sequence of links forms a closed kinematic chain whereas open kinematic chains are called serial structures. The mechanical structure, called *arm*, aims at positioning a wrist so that the end-effector can be oriented in the workspace. Depending on the arrangement of the degrees of freedom of the arm, different architectures of manipulator can be designed [3]. Excluding the cylindrical and spherical manipulators, the most common robot structures are depicted in Figure 2.5.



a) Cartesian b) Scara c) Delta

d) Gantry e) Parallel f) Anthropomorphic

Figure 2.5: Various architectures of manipulator [3]

a) Cartesian manipulator: it is comprised of three prismatic joints whose axes are usually mutually orthogonal. Straight motions are achieved with such manipulator which offers good mechanical stiffness.

b) Scara manipulator: Scara manipulator, standing for *Selective Compliance Assembly Robot Arm*, encloses two revolute joints and one prismatic joint. The arrangement offers high stiffness to vertical loads and flexibility in the horizontal plane.

c) Delta manipulator: the structure is made of three arms connected to the base by means of universal joints. The orientation of the end effector remains constant thanks to the use of parallelograms in the arm.

d) Gantry manipulator: for larger volume and higher stiffness, Cartesian manipulator can be designed as Gantry manipulator whose base is securely grounded.

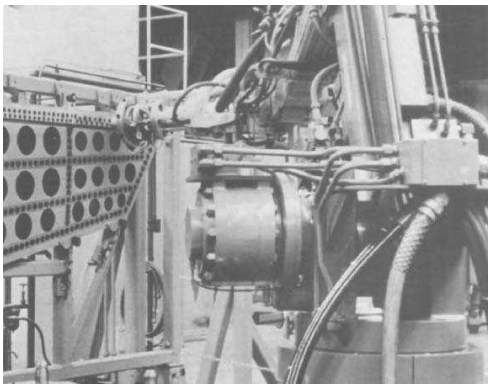
e) Parallel manipulator: high structural stiffness is realised with a parallel geometry for which multiple closed-chains connect the base to the end-effector. Obviously, a reduced workspace is the main drawback.

f) Anthropomorphic manipulator: the most common structure found in the industry resembles the arm of a human being since the second joint is called the shoulder joint and the third joint the elbow, respectively connecting the links called the arm and forearm. The most dexterous structure also has the larger workspace compared to its floor footprint. Figure 2.5f shows two types of anthropomorphic manipulator, without or with a parallel arm for enhanced rigidity.

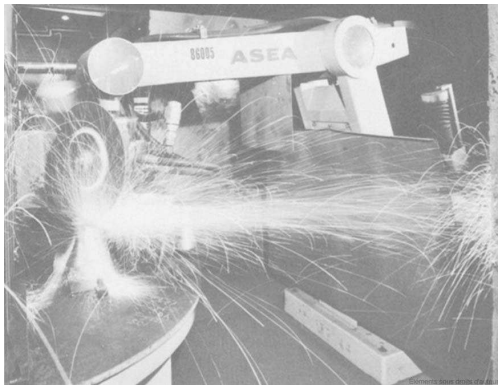
Combinations of mechanical structures also exist and are designated as *hybrid*.

2.1.1 Industrial situation

Gaining in popularity over the past few years, the first signs of the use of robots for machining operations were actually found in 1975 in the aerospace industry. At that time, the idea was still to alleviate the work load for repetitive, exhausting and dangerous tasks. Hydraulically actuated, the Cincinnati Milacron T3-566 robotic arm (Figure 2.6a) was introduced for drilling operations at the Military Aircraft Division of British Aerospace [4]. The robot had a load capacity of 45 kg at 254 mm from its base and a workspace of 2591 mm. Even with its hydraulic actuators, it was too compliant for the drilling task and arrangements of metallic or elastomeric springs had to be added to stiffen the structure during the machining operation. Another notable milestone in robotic machining history was discovered in the foundry industry in the early 1980s. At the Kohlswa Steelworks in Sweden, an ASEA (now ABB) IRB-60 robot, with a load capacity of 60 kg and a workspace of 2288 mm, was used to trim steel flash after a casting operation (Figure 2.6b). Because it is such a noxious environment, the foundry seemed an obvious place for robots demoted to repetitive and fatiguing tasks. In addition, cast parts, often large and with complex shapes, allow exploiting their full dexterity. Indeed, industrial robots are usually anthropomorphic imparting the possibility to follow complicated contours with a dexterity close to human beings [5, 6].



(a) Drilling with the Cincinnati Milacron T3-566 robotic arm in 1975 [7]



(b) Robot trimming steel castings at Kohlswa Steelworks, Sweden [6]

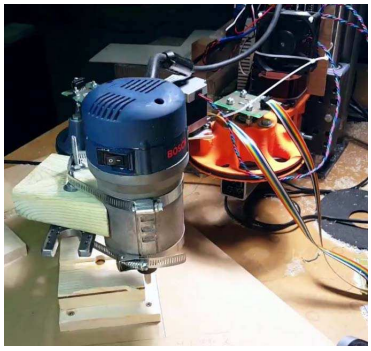
Figure 2.6: First signs of robotic machining in history

As witnessed, robots were usually carrying out a secondary machining task after a primary shaping operation such as casting, forming or machining. The current

applications, though diverse, have not changed significantly. Robotic machining is still relegated to roughing, semi-finishing or finishing operations. Recent progress in robotics and in technology allowed robotic machining to continuously spread in industry. Starting with the aerospace and foundry industries, robotic machining extended to automotive, energy, plastics, prototyping, stone, wood and glass sectors.

Almost all robot architectures have been at least tried for robotic machining applications. Hobbyists recover Cartesian and Scara robots for prototyping applications on soft materials. In Figure 2.7a, a Cartesian robot deburs metal parts while in Figure 2.7b a Scara robot cuts wood material [8, 9].

(a) Deburring of metal parts with a Cartesian robot [8]



(b) Wood milling using a Scara robot [9]

Figure 2.7: Robotic machining using Cartesian and Scara robots

Much more often used for pick and place applications because of their execution speed, robot manufacturer Fanuc demonstrates that delta robots could be part of the robotic machining technology. Delta robot Fanuc M-3iA/6A with the combination of a force sensor carries out a deburring operation on a metallic cast part in Figure 2.8 [10].

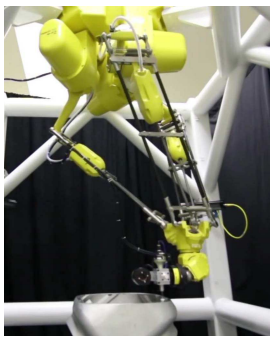


Figure 2.8: Deburring with Fanuc delta robot [10]

In another vein, serious milling applications can be achieved with stiffer robot architectures such as the Gantry and the parallel robots. Gantry robots for machining applications are mainly addressed to energy producers for the manufacturing of medium or large parts. Productivity increase can also be obtained when completing the machining phase in one single set up, avoiding the transfer to other machines. Turning, milling, drilling, tapping, boring and grinding operations are possible using Gantry robots. They constitute common solutions for machining of generators, turbine parts or nuclear units (Figure 2.9a) [11]. Aside from the cost of large machines for large parts, they also occupy massive swathes

of shopfloor. An alternative vision for the manufacture of large components is the use of smaller robots brought to the part. Research conducted by the nuclear Advanced Manufacturing Research Centre (AMRC) in UK developed techniques for robotic machining with a parallel robot. Machining parameters can be chosen using a model to minimise vibration and chatter (Figure 2.9b) [12].

(a) Milling of rotor for power plant generator [11]

(b) Parallel robot for nuclear manufacturing applications [12]

Figure 2.9: Machining applications with Gantry and parallel robots

The most popular robot architecture found in industry is the anthropomorphic arm. It is also the case for machining applications. Being multifunctional and cost-effective, anthropomorphic manipulators or industrial robots can be easily integrated in current applications, even in machining. A fascinating example of integration of robotic machining lies in the production of Kuka robots themselves. Besides loading and unloading robot links to the machine tool shown in the background in Figure 2.10, the robot also achieves finishing operations. Once the main milling task is completed by the machine tool, the robot clamps the part on a table with a multifunctional tool comprising a gripper. It then moves to a tool changer exposing various milling tools. Operations such as drilling, tapping, trimming and polishing are eventually carried out by the robot. Compared with the previous situation in which an operator had to load, process and unload the parts, Kuka observed an increase in productivity of 10 % which means 300 more components per year [13].

Figure 2.10: Kuka robot KR Quantec milling parts in its own production [13]

Since the document concerns robotic milling using industrial robots, next sections are dedicated to the aforementioned manipulator.

2.1.2 Economic impact

Not directly oriented towards robotic machining, it was thought advisable to hand over some numbers regarding the current sales of industrial robots. Since 2010, the demand for industrial robots has sharply accelerated due to the ongoing trend toward automation and technical improvements in industrial robots. After the banner year of 2017, 2018 was another record year but only just (Figure 2.11). Some reasons indicate that car sales were down 3 % as well as global smartphones 5 %. Despite some clouds overhead, sales are still expected to grow to increase manufacturing flexibility [14]. In 2012, it was said that robotic machining products and services constituted less than 5 % of industrial robot sales [15]. Nevertheless, in 2008, an estimation of 2.5 % was given for industrial robots involved in milling, drilling and cutting operations [16].

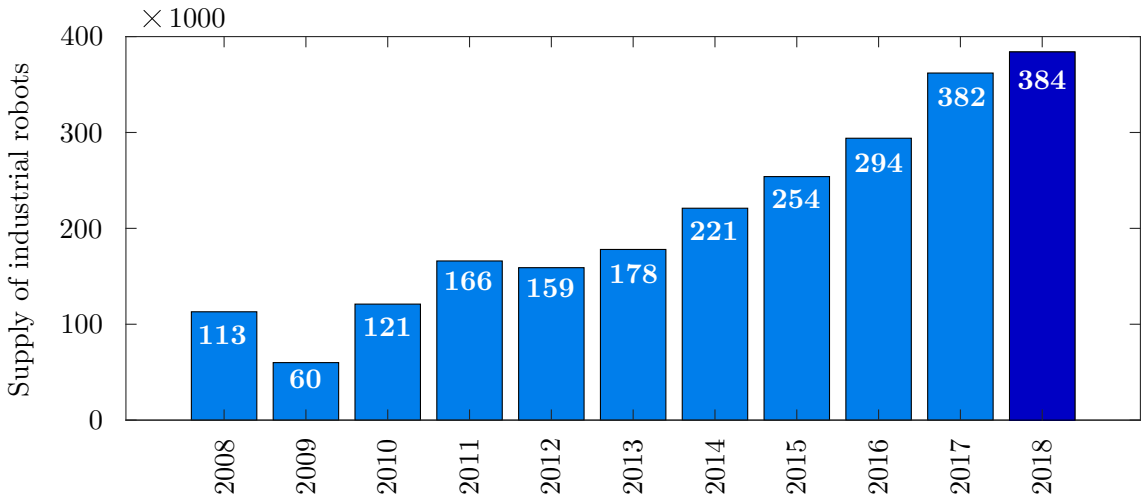
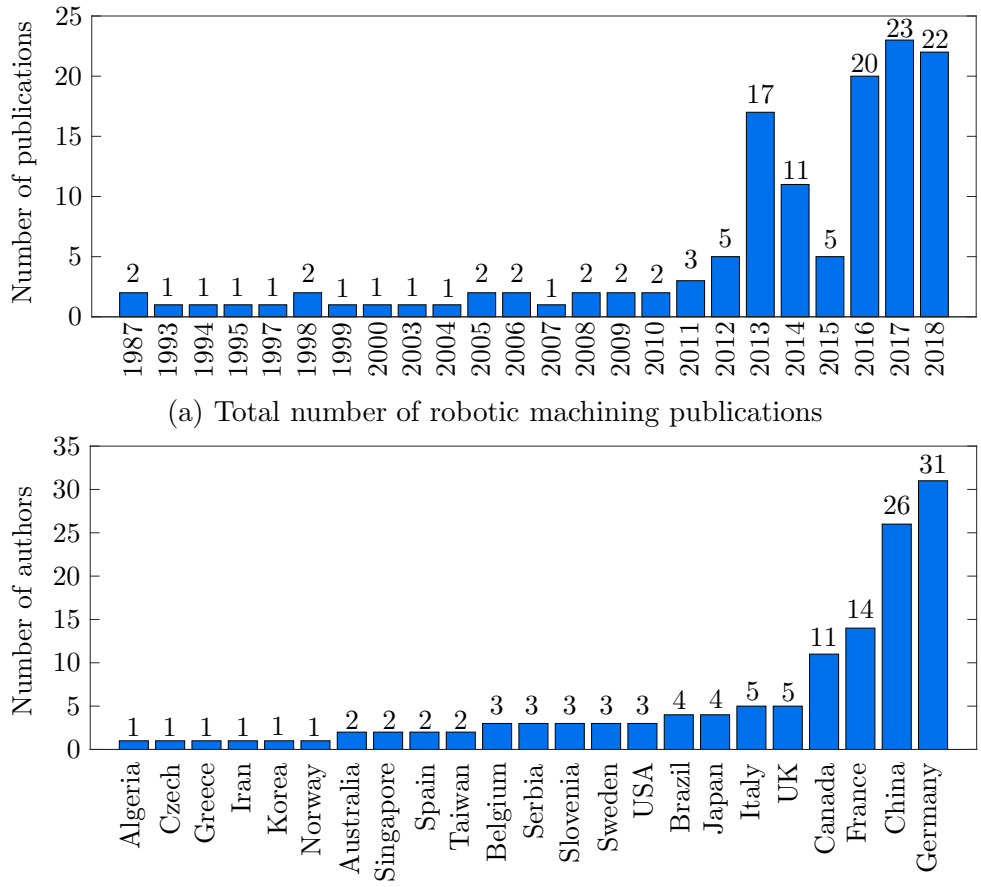


Figure 2.11: Industrial robot sales from 2008 to 2018 [14]

2.1.3 Research interest

First research in the field of robotic machining began in the 1980s focussing on the feasibility of deburring, grinding, polishing and chamfering, often with the aid of a force sensor [17–22]. Researchers were episodically interested in the technology before 2011. After having recovered from the economic crisis of 2008, the topic rose in popularity, coinciding with the increase in industrial robot sales. Since then, a constant growing number of researchers were attracted by the subject. Figure 2.12a shows the total number of publications related to robotic machining from 1987 to 2018 as presented by Ji et al. [23]. From a geographical standpoint, authors originating from 23 countries published papers about robotic machining (Figure 2.12b). The most fruitful countries are Germany, China, France and Canada (Belgium is now represented thanks to this project).



(a) Total number of robotic machining publications

(b) Countries of authors publishing most robotic machining papers

Figure 2.12: Research interest in robotic machining from 1987 to 2018 [23]

2.2 Machining with articulated robots

Since industrial robots are predominant in industry for their attractive cost, it is natural that most research focuses on machining applications with this articulated structure. Composed of revolute joints, they were not intended to be used for machining applications from their design. As opposed to CNC machine tools comprising linear actuators, revolute joints are less rigid and thus lower the structural stiffness of robots. In general, the stiffness of industrial robots is about 1 N/μm whereas for CNC machine tools it is much higher, often more than 50 N/μm. As a result, first robot natural frequencies are located around 10 Hz to 20 Hz. It is much lower than the CNC machine tools, which are several hundred or thousand Hz. Moreover, robot structural dynamics vary over the workspace which is not the case for CNC machine tools, complicating the modelling task. The low rigidity of robots can lead to large deviations errors, and in severe cases, to overall system vibrations or even chatter [24]. Hence their static deflections can easily violate the dimensional and surface finish tolerances in high-material-removal-rate operations. Having considered the main limitations of articulated robots for machining applications, it is clear that only operations requiring low force and tolerance are targeted. Besides, the use of industrial robots is coherent with the actual trend of industry 4.0 aiming at more automation and data exchange in manufacturing technologies. The

future of manufacturing is characterised by high customisation. Being multifunctional, easy to set up and reconfigurable, industrial robots can be readily integrated in existing process and take over operations necessitating multiple operators. As mentioned with the illustrative example in Kuka factory, one single robot was assigned to load, unload and machine parts undergoing multiple milling operations. Beyond the time saving, monitoring would still be required. Open structure also benefits from large workspace. Their available workspace ratio to footprint on shop floor is very high. However, in order to fully exploit the available workspace, mass programming work is demanded from operators to move the robot along complex path for complicated geometries. It is still considered as a difficulty for low-material-removal-rate operations since some knowledge in robotics is required to overcome problems such as collision and singularity [15, 25, 26].

All considered, robotic machining can be seen as a viable alternative to CNC machine tools thanks to the high flexibility of tool positioning, the available workspace and the purchase price. It is said that a cost reduction of 30 % up to 50 % can be expected [24]. Practically, a CNC machine tool roughly costs half a million USD while a fully equipped robotic machining cell is about 200,000 USD. Table 2.1 attempts at providing a summary of the key comparison points between industrial robots for machining applications and CNC machine tools. Strengths towards a technology are marked with a "+". Ideally, if industrial robots were as stiff as CNC machine tools, the latter would be superseded.

	Industrial robot	CNC machine tool
Cost	+	-
Structural stiffness	-	+
Varying dynamics	-	+
Accuracy	-	+
Material removal rate	-	+
Multifunctional	+	-
Integration	+	-
Set-up	+	-
Workspace	+	-
Footprint	+	-
Programming	-	+
Portability	+	-
Complex geometry	+	-

Table 2.1: Industrial robot versus CNC machine tool

2.2.1 Parallel versus serial structure

Referring to articulated robots, two typical architectures are found in industry for machining applications (Figure 2.5f). Aware of the lack of stiffness of serial manipulators, robot manufacturers modified their structure by shortening the length of the links and increasing their sections. In addition, a parallelogram loop was appended to enhance the stiffness along with increased gear ratios and actuator powers. However, adding those

features make industrial robots less versatile and more expensive [27]. Their available workspace becomes smaller and the parallelogram loop only increases the axial stiffness in the plane of the robot (formed by its arm and forearm) which is not the most critical direction of deflection. It is still said that they are more accurate than the serial structure much more popular in industry for machining applications.

Table 2.2 exposes the main differences between the parallel and serial architectures [28]. Advantageous features towards one structure are marked with "+".

	Parallel	Serial
Cost	-	+
Stiffness	+	-
Weight	+	-
Workspace	-	+
Accuracy	+	-

Table 2.2: Parallel versus serial structure

In 2018, Airbus unveiled the latest employees helping put together the A320 passenger plane in Hamburg. The robots are able to drill almost 80 percent of holes on the upper side of the sections of the plane. The line features two seven-axis robots to drill with higher accuracy and ensure a constant level of production quality resulting in less rework.

The parallel robots, secured on a mobile platform, navigate autonomously along the fuselage through dynamic laser tracking (Figure 2.13).

Figure 2.13: Parallel articulated robot [29]

2.2.2 Applications with serial robots

Limiting the study to articulated robots with serial structure, an overview of contemporary industrial applications is presented from the most common configurations to the unusual ones.

1. Drilling of aerospace parts

In 2016, Boeing Aerostructures Australia revealed the use of serial robots for the drilling and riveting of resin-based aerospace parts. The latter form the movable trailing edge for the Boeing 787 wings. The robot is also able to move along the large part via a mobile platform (Figure 2.14).

Figure 2.14: Drilling of aerospace parts [30]

2. Gigantic pipe milling

Since 2016, Asset International in Wales uses serial robots to mill pipes measuring up to 3.5 m in diameter (Figure 2.15).

These oversized pipes are used for such applications as dewatering, subterranean canals and low-pressure applications. KUKA 120 R2700 robot equipped with a milling spindle allowed increasing flexibility, precision and reproducibility of the process.

Figure 2.15: Milling of gigantic pipe [31]

3. Composite trimming

Another common application refers to the trimming of complex shapes made of composite materials that are lightweight and resistant, such as carbon fiber. The use of robot is especially beneficial as the effect of carbon dust can be hazardous to health. In the background in Figure 2.16, a laser scanner allows calibrating the robot by removing the systematic errors due to static deflections.

Figure 2.16: Carbon fiber trimming [32]

4. Mould drilling

Car manufacturer Audi is taking a step towards Industry 4.0 in mould manufacture by using the Stäubli TX200 robot to drill vent holes. Previously, radial drilling machines were used but it was costly in terms of time and labour. The project has been implemented since 2017 at the Audi Competence Center in Germany (Figure 2.17).

Figure 2.17: Mould drilling for Audi [33]

5. Stone milling

Aside industry, the restoration of stone ornament can be carried out with the robotic machining technology. If the sculpture or the ornament was heavily damaged, a digital copy of the existing work of art is first created using photogrammetry. The digital model is materialised in high-density polyurethane foam and a craftsman rebuilds the damaged area. Then, the carving is digitised again and the final relief is milled from stone using a robot (Figure 2.18).

Figure 2.18: Stone ornament restoration [34]

6. Carried part deburring

Deburring operation can be achieved in a different way. The part can be manipulated by the robot with a gripper while the spindle is grounded (Figure 2.19). The advantage is a constant pressure between the tool and the workpiece. Previously, complex paths had to be taught to the robot using conventional approach to clean castings. Nowadays, automatic path learning is widespread in combination with force control.

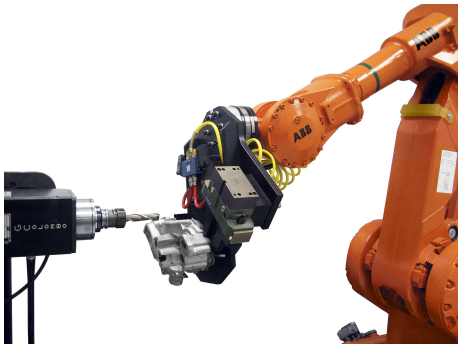


Figure 2.19: Deburring of a carried part [35]

7. Carried part polishing

Still in the same context, an industrial robot moves workpieces to allow the polishing of multiple surfaces in a single operation using an abrasive radial wheel (Figure 2.20).

Figure 2.20: Polishing of a carried part [36]

8. Milling of rotating part

If the workpiece shows revolution patterns, it can be clamped on a turning table. This additional axis allows using the robot as a lathe. In addition, conventional milling operations can be carried out without changing the set-up and all areas are easily reachable using the extra axis (Figure 2.21).

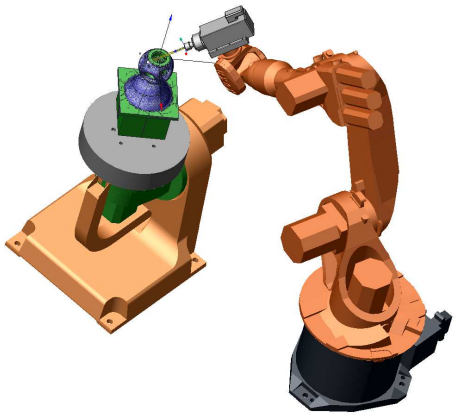


Figure 2.21: Additional rotary axis [37]

9. Milling with robot on linear axis

Similarly, linear axis can be supplemented to traditional articulated robot in order to enlarge their workspace. In Figure 2.22, a wood milling operation to design an architectural work of art is shown.

Figure 2.22: Additional linear axis [38]

10. Milling with industrial robot on mobile platform

A mobile platform for industrial robots was designed in order to provide the desired degrees of freedom for positioning the machining system on components.

The mobile platform is firmly secured on the floor for the machining operation with the aid of three supports. They prevent any destabilisation. To navigate, the platform extends its three wheels. The set-up was successfully tested on CFRP (Carbon Fiber Reinforced Plastics) vertical tail plane shell of an Airbus A320 (Figure 2.23) [39].

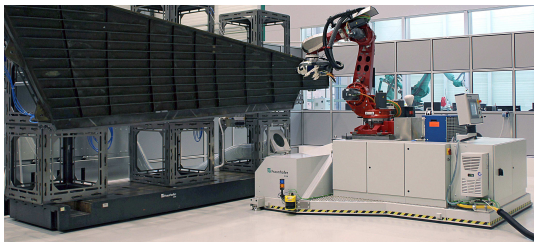


Figure 2.23: Milling on a mobile platform [39]

It is clear that the developed applications involving robotic machining are diverse and affect many different industrial sectors. Updated from [40], Table 2.3 presents an overview of the current end-user specific processes, manufactured products and materials.

Industrial sector	Process	Material	Product
All segments	Milling	Resin, foam, wood	Rapid prototyping
Aerospace	Grinding, polishing, drilling, cutting	Composite, alumnium	Turbine blades, bulkheads, insulation, wing segments, fuselage sections
Automotive	Deflashing, grinding, drilling, milling, cutting	Composite, alumnium, steel, gray cast iron	Engines, truck frames, body panels, door knobs, bumpers, stamping dies, sand cores, forming moulds
Fashion	Milling, sanding	Resin, foam, wood	Mannequin moulds, mannequins
Foundries	Deburring, milling, drilling, finishing	Aluminium, steel	Moulds, castings
Glass	Milling, deburring, cutting	Glass	Flask, glass
Medical	Milling, grinding, polishing	Resin, foam	Prosthesis
Nautical	Milling, drilling	Resin, foam, wood, aluminium	Boat hulls
Entertainment	Milling	Resin, foam, wood, stone	Movie set props, amusement park scenery, work of art, sculpture
Plastics	Milling	Plastics, polystyrene	Moulds, helmets
Woodworking	Milling	Wood	Hot-tub moulds, furniture, trim, banisters, modelling board

Table 2.3: Products, process and materials related to robotic machining in different industries

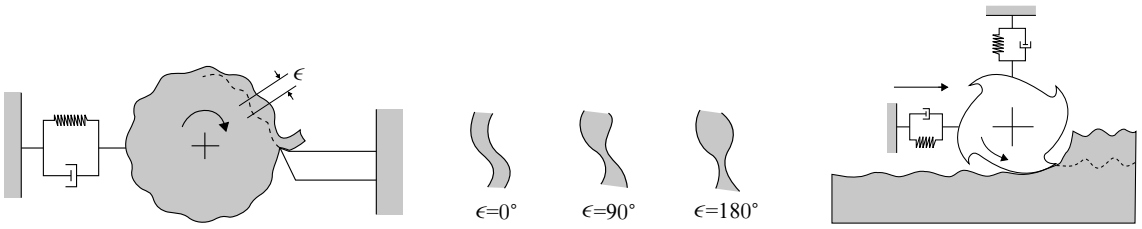
2.3 Chatter in robotic machining

Glimpsed in the introduction, chatter or self-excited vibration is caused by the most flexible, dominant structural modes of the machine-workpiece system. Adverse effects range from premature tool wear, poor surface finish to damages on the spindle mounted on the robot. Overall, the mechanical system can be represented as a combination of bodies that possess mass, damping and elasticity. The resulting vibrations of bodies can be divided into three main categories: free, forced, and self-excited vibration. Free vibration results from initial conditions imposed on the system, e.g. a displacement from its equilibrium position. If the system is damped, vibrations occur at natural frequencies and vanish following a decaying exponential envelope. More commonly in machining, forced vibration takes place when a continuous, external periodic excitation produces a response with the same frequency as the forcing function. The periodic tooth passing in the material is the primary source of forced vibration. Since forced vibration is in phase with the tooth passing frequency, their impact on the surface finish can be considered as negligible. However, in self-excited vibration, the periodic input force is also modulated at one of the system natural frequencies leading to instability [41]. Self-excited vibration is difficult to control and thus is preferably avoided. In robotic machining, two major sources of chatter have been identified: the regenerative effect and the self-excited mode coupling.

2.3.1 Regenerative chatter

Regenerative chatter is considered to be the most common cause of machining instability in machine tool [42,43]. The simplest way to illustrate the regenerative chatter phenomenon is to consider the turning operation. Since the tool vibrates as it removes material, the vibrations are necessarily imprinted on the workpiece surface as a wavy profile. Figure 2.24a shows an exaggerated view of the wavy profile after one revolution. At the beginning of the second revolution, the vibrating tool encounters the wavy surface generated during the first pass. From that instant, chip thickness depends both on the current tool deflection and the workpiece surface from the previous revolution. The variable chip thickness will then provoke variable cutting forces since a proportionality is commonly assumed between both. Finally, the variable cutting forces are looped back to the tool deflection leading to the regenerative effect. In other words, the cutting operation does not only depend on the actual state, but also depends on the past states. From a mathematical viewpoint, regenerative chatter is described by delay-differential equations (DDEs) [44].

Depending on the cutting conditions and the rigidity of the mechanical system, two different situations can arise. If the wavy surfaces between two successive revolutions are in phase, even though vibration is present during material removal, the variation of the chip thickness will be negligible. Hence, no instability will be triggered and a stable cut will be accomplished (phase shift ϵ=0°). In contrast, since the tool is likely to vibrate at its natural frequency, the wavy surfaces can also be out of phase (ϵ=180°) thus leading



(a) Chatter in turning (b) Chip thickness variation (c) Chatter in milling

Figure 2.24: Regenerative chatter concept

to significant variation in the chip thickness (Figure 2.24b). Unstable cutting conditions produce large force variations and important tool deflections damaging the surface finish and the machine. The concept of regenerative chatter can be extrapolated to milling by considering a rotating cutting tool with several teeth (Figure 2.24c) [45]. In this case, regenerative chatter results from the interaction between successive teeth through the generated machined profile.

2.3.2 Mode coupling chatter

Unlike regenerative chatter which happens locally at either the spindle or the workpiece, when mode coupling chatter arises, the entire robot structure experiences severe vibrations [26]. Mode coupling chatter occurs when the tool tip of the mechanical system vibrates simultaneously in two or more directions that interact with each other [46]. Without any regenerative effect, the structure vibrates at the same time in different directions with the same frequency and phase shift. The mode coupling alters the path followed by the tool tip also leaving wavy marks on the workpiece surface. For a planar case, the unstable motion of the tool tip relative to the workpiece results in an ellipse [47]. Figure 2.25 depicts the situation in which it is assumed that the tool tip stiffnesses along the X and Y axes closely match. As a result, preferential direction of deflection is oriented with angle γ along the feed direction leading to mode coupling chatter and the elliptic tool motion (Figure 2.25). The stability of robotic milling operations therefore depends on the relative angle existing between the cutting force vector and the directions of the tool tip compliance.

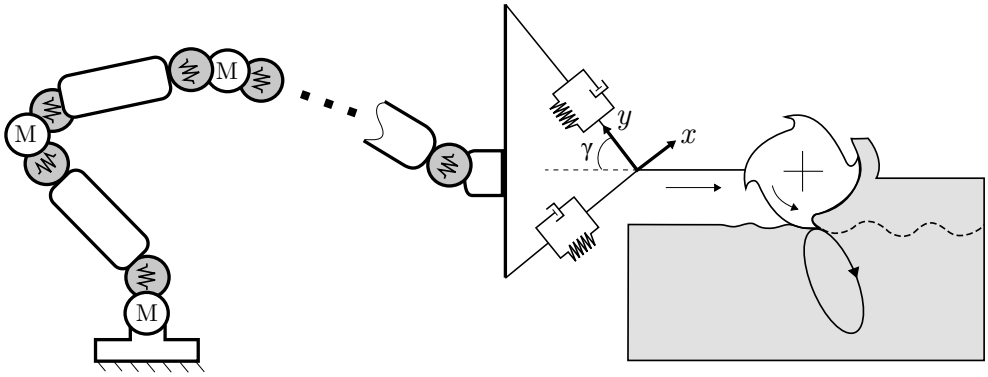


Figure 2.25: Mode coupling chatter concept

From the energy point of view, unstable conditions arise if the energy supplied to the

system by the cutting forces is not totally dissipated by damping. If the tool rotates counterclockwise, the unstable mode corresponds to an ellipse running clockwise i.e. the depth of cut is higher along the forward path of the ellipse. Hence, a net amount of energy enters the system, increasing the amplitude of the ellipse, thus giving rise to vibratory phenomena [48].

Pan et al. concluded that mode coupling chatter was the major source of instability in robotic milling [49, 50]. They indeed experienced low frequency chatter at high spindle speed. Since the low natural frequencies of the robot are unlikely to cause regenerative chatter at high spindle speed, mode coupling chatter was assumed. As a matter of fact, articulated structures have lower stiffness and their tool tip Cartesian stiffness matrix is highly pose dependent. Robot stiffnesses can therefore easily interact in various directions. These considerations are prone to trigger mode coupling chatter. Hence, selecting optimised robot posture can turn out to be an asset in order to avoid coupled self-excited vibrations.

2.4 Research topics

As suggested in [23], there are essentially two fields of research in robotic machining. The partition is closely related to the considered machining operation, and more specifically to the material removal rate (MRR). In low-MRR operations, the main difficulty lies in the static deflections of industrial robots leading to a poor accuracy. Such deviations can be accepted in roughing or compensated, to a certain extent, in finishing. In high-MRR operations, the productivity is limited by self-excited vibrations, the chatter phenomenon. Consequences of self-excited vibrations range from a shorter tool life, poor surface finish to a defective spindle. To address the two issues, there have been numerous research publications as well as technical reports for more than three decades. Within each of the categories, the issue is processed by means of various approaches. Before succinctly detailing solutions proposed by the authors, Figure 2.26 outlines the main approaches to either compensate the deflections induced at the tool tip or to avoid chatter.

Deflection compensation (low MRR)	Chatter avoidance (high MRR)
- Machining error source characterisation	- Workpiece placement
- Stiffness identification	- Robot posture-dependent dynamics
- Robot trajectory planning	- Structural dynamics modelling
- Monitoring and compensation	- Cutting parameters
- Robot control optimisation	- Stability analysis

Figure 2.26: Research topics

2.4.1 Deflection compensation

The low-MRR operations mainly relate to human-like operations necessitating a high dexterity but requiring low forces and accuracy: deburring, polishing, grinding and chamfering. The major research is devoted to the compensation of static deflections by modifying the trajectory followed by the tool tip, using sensors for on-line compensation, identifying models for off-line compensation, determining the sources of inaccuracy and

optimising the control law.

In order to compensate the deviation errors in robotic machining, it is first necessary to identify their source and quantify their magnitude. Schneider et al. [51] analysed the sources of errors in robotic machining and characterised them in amplitude and frequency. Experiments in machining and experiments in free space motion showed that joint compliance and backlash are the most dominant sources. However, when trying to achieve an accuracy below 100 μm, the disturbances from the environment and errors from cell calibration also need to be taken into account. Barnfather et al. [52] considered the error identification from another perspective by assessing the accuracy and precision of machining robots from a procedural and statistical standpoint. They found out that there was no standard available for robotic machining performance evaluation. Therefore, it was proposed to adapt standards from machine tool performance evaluation for static and dynamic assessments. Standard artefacts such as the NAS 979 (Figure 2.27) encompasses various prismatic features to efficiently expose the deviation errors in machine tool. However, applying the same procedure to milling robots hardly leads to the identification of the error sources causing deviation.

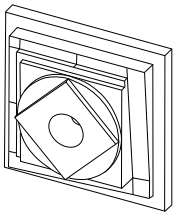


Figure 2.27: NAS 979 workpiece for error identification in five-axis machine tool [52]

Once the error sources are somehow identified, a flexible model of the robot can be derived in order to predict the deviation errors at the tool tip. The parameters of the model must be determined beforehand. Dumas et al. [53] developed a robust and fast procedure for joint stiffness identification around the axes of motion. The proposed method is based on the conservative congruence transformation [54] giving the Cartesian stiffness matrix. Translational forces and torques need to be applied at the end effector of the robot in order to compute the joint stiffness around their respective axis of motion. Set-up and tooling are presented in Figure 2.28. Although easy to implement, the method requires expensive equipment such as a laser tracker, as it is also the case in [55–57].

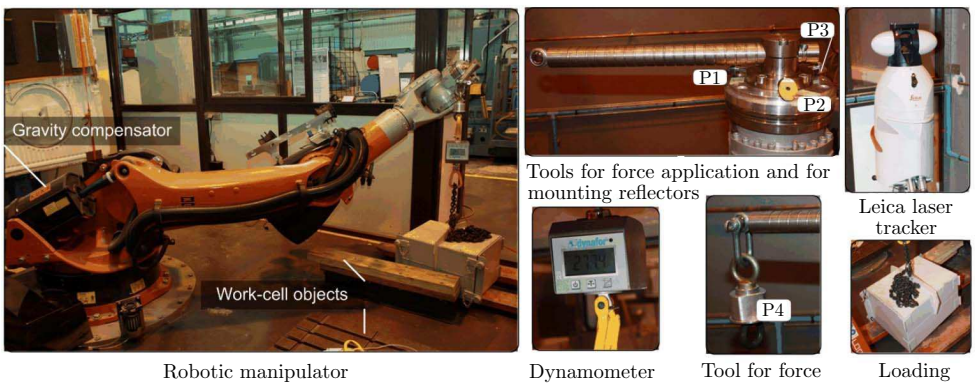


Figure 2.28: Experimental set-up for manipulator elastostatic calibration [58]

The set-up was later improved by Cordes et al. [59] such that no integration of a force sensor is required in the robot structure, and still all the joints (especially axis 1) can be loaded. Instead an external cutting force sensor is bolted to the ground and then connected to the robot tool tip. The tensile force is applied via a tension device located between the force sensor and the tool tip as presented in Figure 2.29. The method is thus independent from the periphery of the robot and does not influence any existing absolute calibration. The method needs five force-deflection measurements in 15 different robot postures in order to identify the joint stiffness of all the motion axes.

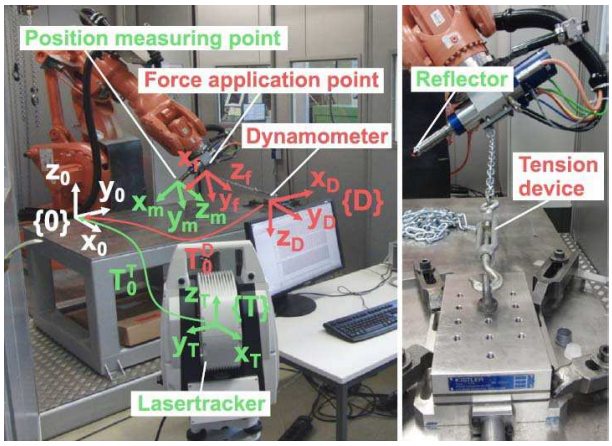


Figure 2.29: Experimental set-up for joint stiffness identification [59]

Doukas et al. [60] developed a finite element model to empirically tune the joint stiffness parameters around the axes of motion of the robot with inclusion of the link flexibility. A more spectacular method was carried out by Lehmann et al. [61]. The so-called clamping method was applied to an ABB IRB2400 robot to identify the joint stiffness around its motion axes. The advantage is that no arm-side encoders or optical tracking systems is needed, which reduces the cost of the set-up. The end effector of the robot is clamped to a force sensor. Each axis of the robot is then moved separately using its own motors and forces and torques are measured with the force sensor. Besides joint stiffness, the method is also able to evaluate joint backlash and friction.

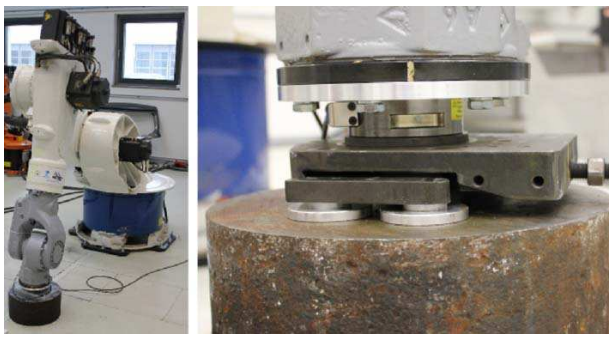


Figure 2.30: Clamping of the end-effector for stiffness identification [61]

With an estimation of the joint stiffness, the planned robot trajectory can be modified off-line to better follow the desired milling trajectory. Reinl et al. used the path mirroring technique which consists in following the reversed path against the one predicted by the model. As a result, the systematic error due to static deflections are compensated by following a fake trajectory (Figure 2.31) [62–64]. Similarly, Slavkovic et al. developed a method for off-line compensation of modelled cutting force-induced errors in robotic machining by tool path modification [65].

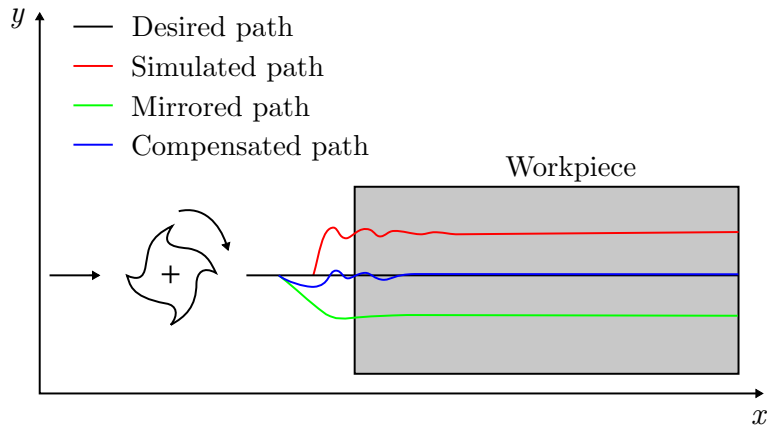


Figure 2.31: Concept of path mirroring

Another possibility to off-line optimise the accuracy of the tool path of milling robot consists in using some features pertaining to CAM (Computer Aided Manufacturing) software. The approach presented in [66] uses global interpolation of the required surface by a virtual surface composed from tool trajectories. Atmosudiro et al. [67] proposed a joint space path planning.

It is also possible to compensate the deviation errors on-line either by additional actuation, by on-line monitoring or by adopting a new control law. In [68], Ilyukhin et al. relied on the motor current to estimate the load on each joint due to the milling forces and consequently adapted the path velocity for a better tracking. In contrast, Lehmann et al. [69, 70] used a force/torque sensor mounted between the robot end effector and the spindle to measure the process force. Using the difference between the predicted forces and the measured forces, the deviation is calculated and looped back as an offset to the robot controller. Schneider et al. [71, 72] used a 3D-piezo compensation mechanism which is able of fast adaptation of the spindle position relatively to the workpiece which is carried by the robot and located by a laser tracker (Figures 2.32 and 2.33). Their research was conducted under the COMET project which is later presented. Extra sensor such a second encoder after the gearbox can also be installed to measure the joint deflection and achieve an on-line compensation [39].

In [74], Bo et al. implemented an adaptive impedance controller so that the tool always applies the same amount of force on the workpiece in finishing operations. A similar controller was developed by Sörnmo et al. in [75] in which the robot carried the part.

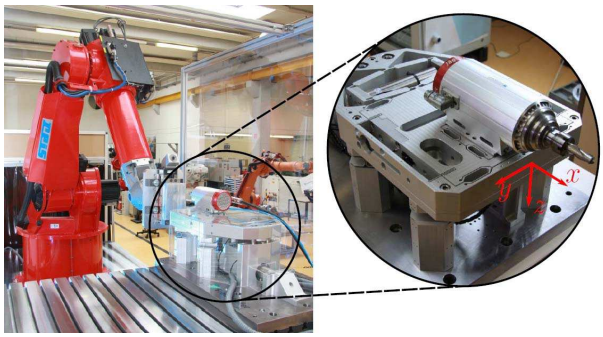


Figure 2.32: Real-time compensation of positioning errors using a 3D-piezo actuator on the spindle side [73]

Tian et al. [76] developed a fuzzy PID controller to regulate the normal polishing force in polishing operation. The results showed that the proposed regulated platform had the ability of effectiveness and feasibility for polishing on curved surfaces, and was able of achieving a mirror effect surface while keeping a good global uniformity. Other controller implementations are found in [77, 78].

To the best of author's knowledge, four major projects related to deflection compensation in robotic machining were financed by European funds in recent years:

1. COMET project: it aimed at developing plug-and-produce *CO*mponents and *MET*hods for adaptive control of industrial robots enabling cost effective and high precision manufacturing in factories of the future. The COMET project addressed the robotic machining challenge and developed a modular and configurable platform able to enhance the machining accuracy of standard industrial robots enabling cost-effective and first-time-right robotic machining operations [79]. Figure 2.33 shows the objectives of the project while Figure 2.32 presented the final demonstrator (duration: 2.5 years until 30^{th} June 2013, budget: 5.3M €, leader: Dr. C. Lehmann from the Brandenburg Technical University).

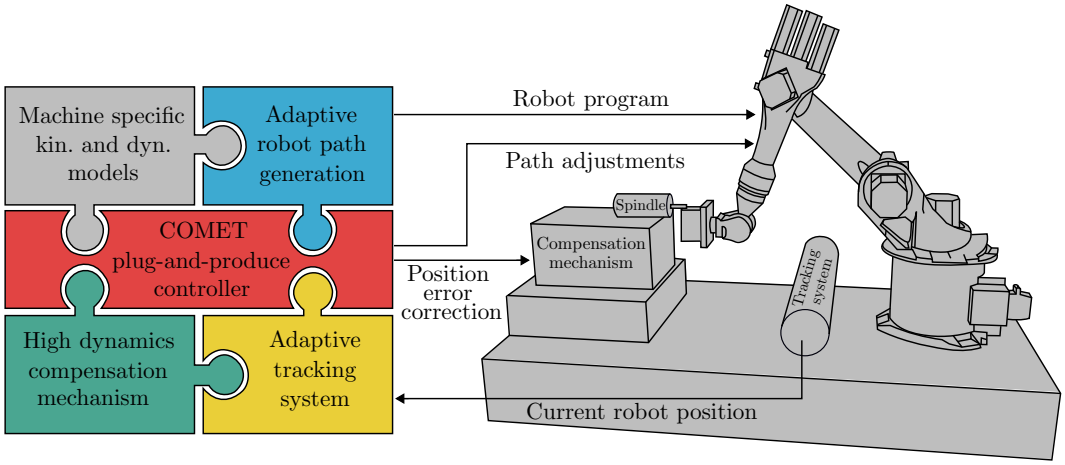


Figure 2.33: Objectives of the COMET project [79]

2. Hephestos project: with the focus on developing sophisticated methods in robotic manufacturing, the Hephestos project aimed at rising cost-efficient solutions in hard materials machining for small-batch production of highly customised products through the application of industrial robots [80]. The project resulted in a robotic machining demonstrator including planning and programming tools, control, human-interaction and advanced sensory system (duration: 3 years until 31^{st} October 2015, budget: 3.3M €, leader: Prof. E. Gambao from the Technical University of Madrid).

3. Flexicast project: it aimed at developing innovative robotic machining systems that are flexible, reliable and predictable with an average of 30 % cost efficiency savings in comparison with machine tools. The main developments were in the field of process aware path planning, machining strategies and axis control, optimal tool design, active damping devices, for deburring and polishing operations [81]. Main outcomes concerned the development of a robotic cell improving the finishing operations of cast parts. Namely, three control strategies were implemented enhancing the deburring operation by either maintaining a constant execution velocity, either slowing down the velocity according to measured process forces or imitating the human motions (duration: 4 years until 31^{st} October 2016, budget: 9.2M €, leader: Dr. E. C. Maciá from the Technical University of Catalonia).

4. Coroma project: it proposes to develop a cognitively enhanced robot that can execute multiple tasks for the manufacturing of metal and composite parts. Targeted applications concern the aerospace, energy and naval industry. One of the objectives is the development of a mobile industrial robot for polishing operations on boat hull. The project also addresses the communication issues between hardware, software and human operators [82]. So far, four demonstrators involving robots are operational for machining, grinding and sanding thanks to a robot mounted on an automated guided vehicle (AGV) and for synchronising the operations with a machine tool (duration: 3 years and ongoing until October 2019, budget: 7.3M €, leader: Dr. A. Barrios from research centre IK4-IDEKO).

2.4.2 Chatter avoidance

In high-MRR operations, the low robot stiffness turns into a serious productivity limiting factor. As pointed out above, past research essentially focussed on the compensation of static deflections for low-MRR and man-like operations in soft materials. However, for robotic milling or drilling, self-excited vibrations are likely to occur if not well anticipated. In order to avoid chatter, either the robot stiffness can be enhanced through optimised workpiece placement and robot posture or stability limits can be predicted via modelling.

The stiffness of robots mainly originates from its joints. It is therefore understandable that depending on the posture of the robot, the Cartesian stiffness at the tool tip will change, which makes the prediction of its modal behaviour in the entire workspace

challenging. Naturally, the Cartesian stiffness at the tool tip will be higher if the robot machines near its anchor point rather than working with an outstretched arm. Caro et al. [83] proposed a methodology to determine the optimum placement of any workpiece to be machined knowing the cutting forces exerted on the tool and the elastostatic model of the robot. A machining quality criterion was developed and the problem was formulated as a constrained mono-objective optimisation. Figure 2.34 presents the resulting optimum workpiece placement with respect to the robot. A similar study was recently conducted in 2019 [84]. Li et al. [85] analysed the effect of the cutter path and the workpiece clamping position on the stability in milling. By means of milling experiments in six different clamping positions relative to the robot, it was discovered that the surface quality was better if the feed direction was oriented perpendicularly to the robot plane. Slamani et al. [86] compared the surface roughness quality obtained by high speed CNC trimming and high speed robotic trimming for CFRP laminate. Depending on the orientation of the composite ply, similar surface quality could be obtained from either process.

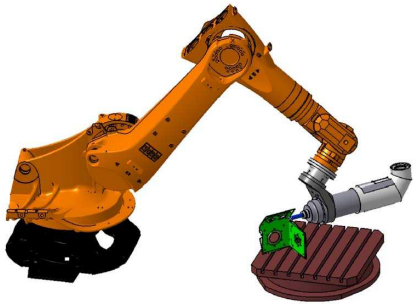


Figure 2.34: Optimum workpiece placement for KUKA KR270-2 robot [87]

In order to improve the posture of robots during robot-assisted machining operations, authors developed new criteria, so-called performance indexes, based on manipulator kinematic properties. Zargarbashi et al. [88] proposed a new robot performance index which does not only account for the posture dependency but also for the task. The concept is applicable to common industrial robot architectures, i.e. composed of six revolute joints. Similarly, a set of performance evaluation indexes were proposed by Lin et al. [89, 90] to optimise the robot posture. The indexes involved kinematic performance, body stiffness and deformation evaluation indexes. The indexes were experimentally validated on a Comau Smart5 NJ 220-2.7 robot demonstrating the feasibility and effectiveness of the present posture optimisation methodology (Figure 2.35).

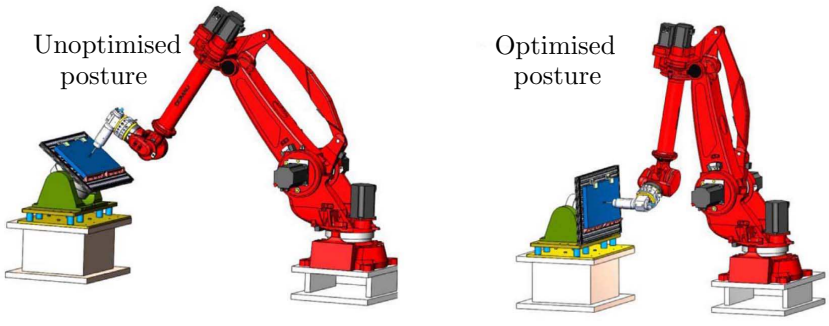
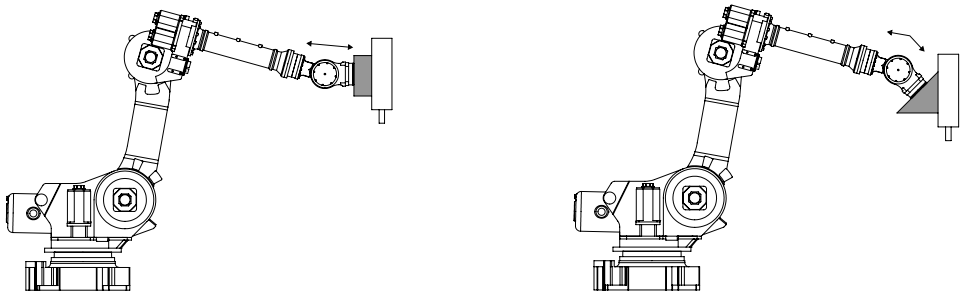


Figure 2.35: Robot optimum posture for drilling operation [89]

Xiong et al. [91] introduced a performance index to evaluate the stiffness of the robot at a certain posture. The posture optimisation problem was then formulated in consideration of the constraints of joint limits, singularity avoidance and trajectory smoothness. Simulations and experiments on an industrial robot showed that the machining accuracy could be efficiently improved by the proposed method. In [92], Shen et al. suggested a practical method to improve hole position accuracy for a drilling operation. Besides the implementation of a compensation method for the end effector deflections, they also optimised the orientation of the spindle mounted on the robot wrist. An optimum tilting angle of 45° was computed for the considered drilling operation (Figure 2.36). In addition to enhancing the rigidity, it is also known that orienting the spindle allows avoiding manipulator singularity. Indeed, rotation axes at the wrist are unlikely to be aligned.



(a) Unoptimised spindle orientation (b) Spindle tilted with an angle of 45°

Figure 2.36: Optimisation of the spindle orientation [92]

Guo et al. [93] recommended a posture optimisation for the drilling and boring operations of connecting holes on aircraft fuselage. The resulting optimum posture is shown in Figure 2.37. It is interesting to note that the same pose was used to drill holes on the trailing edge for the Boeing 787 wings in Figure 2.14. Further studies on robot posture optimisation can be found in [94, 95].

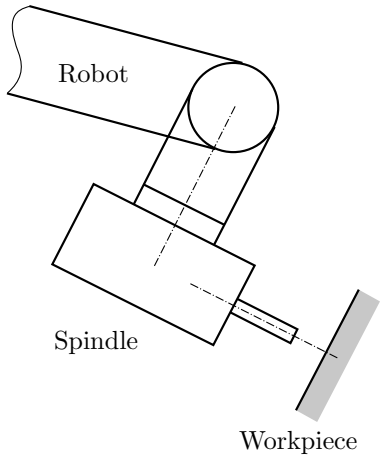


Figure 2.37: Optimum pose for drilling and boring [93]

The above optimisations often required a simplified stiffness model, only concentrated on a single point, the tool tip, via the Cartesian stiffness matrix [54, 96–98]. Such a model is useful for deflection compensation or stiffness enhancement for posture or workpiece

optimisation. However, no information is provided regarding the moving masses or the structural damping. It is therefore unable to predict the stability limits of a milling operation. The dynamics of a multiple-degree-of-freedom (MDOF), such as a robotic manipulator, at its tool tip are described, for small displacements about a reference configuration, by the differential equations of motion as

$$\mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{C}\dot{\mathbf{q}}(t) + \mathbf{K}\mathbf{q}(t) = \mathbf{F}(t), \tag{2.1}$$

where $\mathbf{M} \in \mathbb{R}^{n\times n}$, $\mathbf{C} \in \mathbb{R}^{n\times n}$ and $\mathbf{K} \in \mathbb{R}^{n\times n}$ denote the system mass, damping and stiffness matrices, respectively; $\mathbf{q}(t) \in \mathbb{R}^n$ is the displacement vector, $\mathbf{F}(t) \in \mathbb{R}^n$ is the cutting force vector and n is the number of degrees of freedom.

In [99], Cen et al. extended the stiffness model at the tool tip with a Cartesian mass matrix and a Cartesian damping matrix. The effect of robot dynamics on the machining forces could therefore be studied to improve the cutting conditions.

More generally, the dynamic behaviour in a static robot posture can be measured using an impact hammer test i.e. an experimental modal analysis (EMA). The impact test allows exciting a wide spectrum of frequencies. An accelerometer is usually glued at the tool tip and the hammer stroke is given at the same point (Figure 2.38). Direct FRFs can thus be measured when the hammer stroke is in the same direction as the measuring one (H_{xx}, H_{yy} and H_{zz}). If a multi-axis accelerometer is used, cross-FRFs can also be captured from the orthogonal directions (H_{xy}, H_{xz}, H_{yx}, H_{yz}, H_{zx} and H_{zy}).

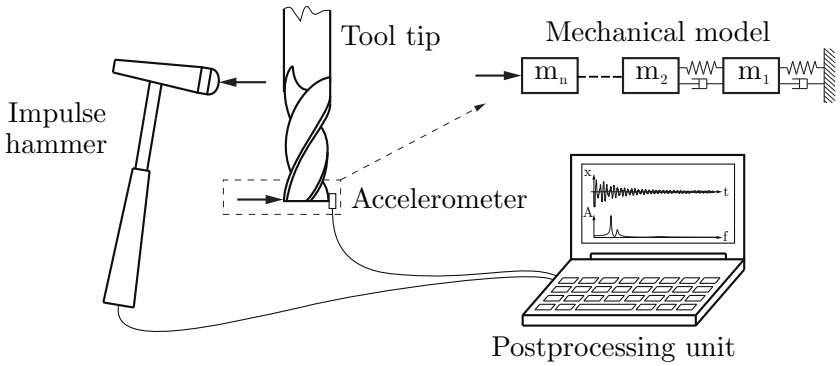


Figure 2.38: FRF measurements at the tool tip [100]

Tunc and Stoddart [101] directly used the measured FRFs at the tool tip to compute the stability limits of their milling robot. In this way, they could identify preferential feed directions. Wang et al. [102, 103] studied the chatter mechanism for robotic-assisted boring operations. They observed chatter at low spindle speed and predicted the stability by introducing a new approach still requiring the measured FRFs of the mechanical system. Similarly, Cordes et al. [104] predicted and experimentally validated stability lobe diagrams in aluminium and in titanium for an ABB IRB 6660-205/1.9 robot. For this purpose, they measured the frequency response function matrix along the X, Y and Z axes

including the cross FRFs for one defined robot posture (Figure 2.39). It can be observed that, in addition to be pose dependent, FRFs exhibit a significant coupling and cross FRFs are slightly non-symmetric. As a matter of fact, it is well known that articulated manipulators are non-linear structures meaning that tool tip FRFs somehow depend on the level of the input force.

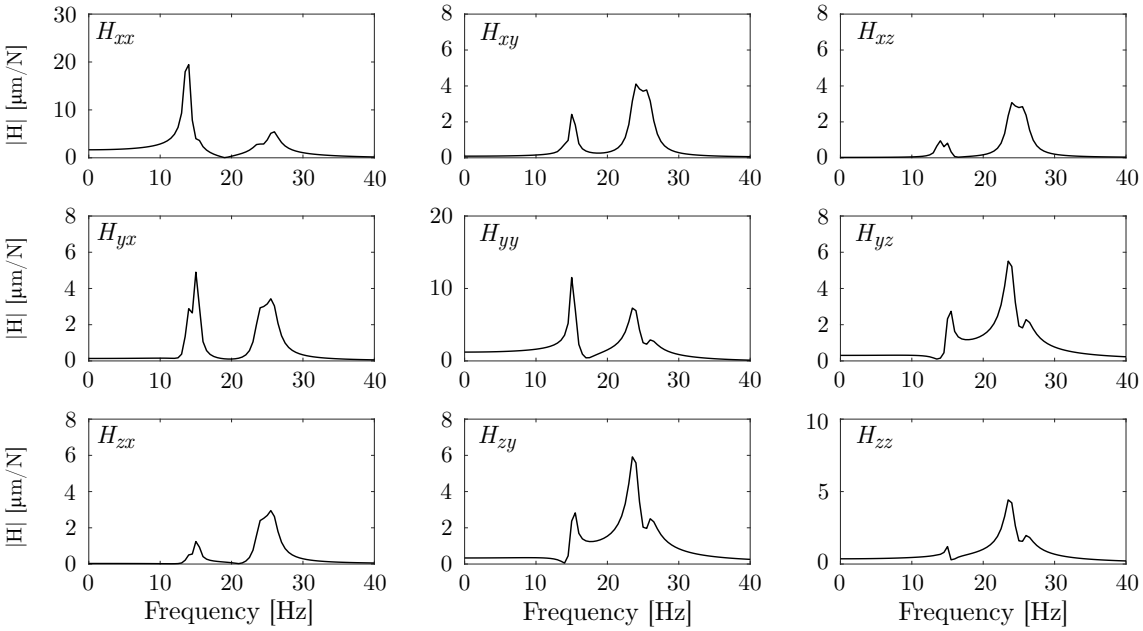


Figure 2.39: Modelled FRF matrix at tool tip for the ABB IRB 6660-205/1.9 robot [104]

Mejri et al. [28, 105], besides carrying out the classical stability analysis with a hammer test at the tool tip, performed an operational modal analysis (OMA) too. The latter method was adopted to compute the FRFs while the tool tip was rotating into the material for a milling operation, therefore using the cutting forces as the sole excitation. The resulting stability lobes showed a wider region of stability when the FRFs from the operational modal analysis were used (Figure 2.40). According to the authors, stability lobes obtained from OMA give a better stability prediction. However, they only tested one particular combination of spindle speed and axial depth of cut to experimentally verify their stability lobes. In addition, by measuring FRFs in different directions, they clearly asserted that robot pose-dependent dynamics needed to be considered in the stability prediction for milling applications.

Continuing the work of Mejri et al., Mousavi et al. [106–109] developed a dynamic multibody model of the robot including flexible joints and links. After identification and updating of their model through experimental modal analysis, stability lobe diagrams were computed in various robot postures. As a result, they could plan stable trajectories by exploiting the task redundancies in the kinematic chain of the manipulator. Analogously, Chen et al. [110] addressed the posture-dependent stability prediction by measuring tool tip FRFs in different configurations.

It appears that modelling the robotic system using the dynamic multibody approach

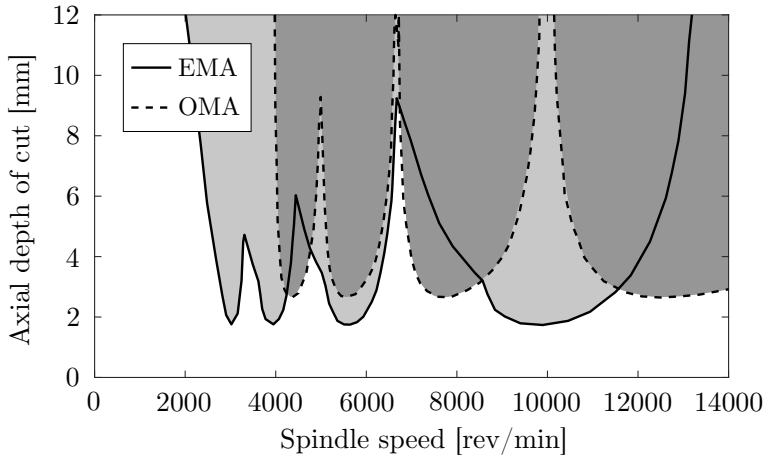


Figure 2.40: Stability lobe diagrams obtained from EMA and OMA [28]

seems an efficient way to capture the pose-dependent modal behaviour. However, depending on the complexity of the multibody model, the identification of its parameters can be challenging. Baglioni et al. [111] developed a multibody model of a KUKA robot assigned to milling operations. Link mass and inertia were commonly retrieved from the CAD (Computer Aided Design) models provided by the robot manufacturer. On the other hand, joint stiffness values around the robot axes of motion were read from catalogues and joint damping values were empirically chosen. Hence, they could correctly simulate the robot dynamic behaviour for some specific cases. Klimchik et al. [112] extended the traditional robot multibody modelling by considering multiple directions of deflection at the joint, representing all the drive and transmission compliances. At most, each joint comprises six degrees of freedom and can therefore deform in all directions. The non actuated deformations are treated as virtual joint and are only submitted to the gravity forces, hence the so-called Virtual Joint Modelling (VJM) method (Figure 2.41). Usually modelled with linear/torsional springs and viscous dampers, joint stiffness and damping values can be determined through modal tests. In view of the large number of parameters to identify, Klimchik et al. [113] limited their study to the joint stiffness identification so as to compensate the static deflections. Likewise, Reinl et al. [62] developed a similar multibody model but limited to virtual rotational motions. Nevertheless, since the identified joint damping was questionable, their study was also limited to static compensations.

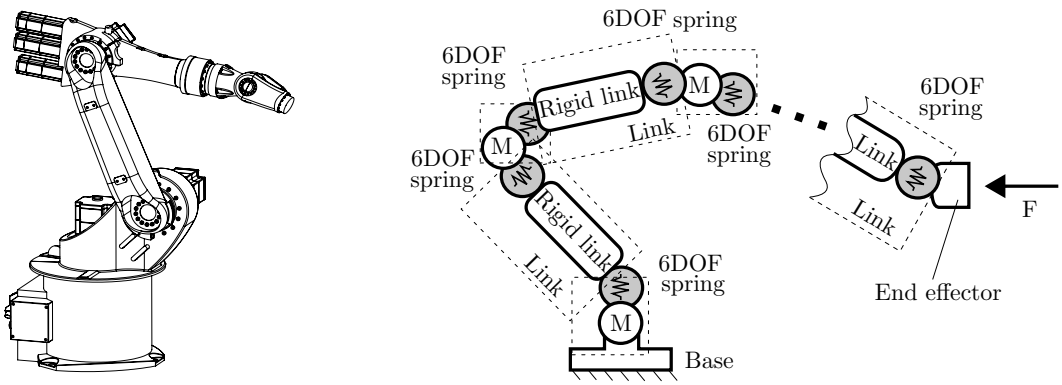


Figure 2.41: Virtual Joint Modelling (VJM) method [112]

Whatever the followed approach used to model the robot dynamic behaviour, stability lobe diagrams eventually aim at choosing optimised cutting conditions to improve the productivity.

2.5 Discussion

Although considered as a new technology, first investigations related to robotic machining actually started in the 1980s. Indeed, machining with robots offers larger workspace and demands less investment compared with conventional CNC machine tools. Robotic machining gained in popularity from both academics and industry only after recovering from the economic crisis of 2008. Due to their higher compliance, industrial robots were successfully adopted for machining operations only requiring low force and tolerance. Common applications are found in the aerospace industry for the drilling of holes in the fuselage and in the foundry industry for the deburring of cast parts. Globally, robotic machining affects multiple sectors of activity such as the automotive, the energy, the wood, the glass or the stone industry. Rather focussed on the cutting of soft materials, past research mainly covered the identification of error sources, the development of methods to measure the joint stiffness and the off-line and on-line compensations of the static deflections of robots. In order to deal with harder material such as steel, inconel or composite materials, models and methods are being developed to avoid and monitor self-excited vibrations limiting the productivity of machining operations. However, such modellings are challenging to develop since the robot compliance cannot be considered as constant in its large workspace. Following the usual approach in conventional machine tools, some researchers directly used the measured FRFs at the tool tip in order to derive stability lobe diagrams allowing the selection of optimised cutting conditions. Repeating the procedure in different postures, they could eventually determine regions of higher stiffness in the robot workspace. Nevertheless, this time-consuming procedure would be impractical from an industrial viewpoint. Others modelled industrial robots as a succession of bodies linked together through flexible joints using the multibody approach. Although convenient to predict the robot pose-dependent modes, the parameter identification of flexible robot model proves to be challenging since the mass, the damping and the stiffness properties must be determined. Generally, link inertia is roughly estimated using the CAD models of robot manufacturers and elastic parameters are fitted from measured FRFs at the tool tip. Robot multibody models are often limited to the consideration of the joint flexibility around the axes of motion. In rare cases, researchers modelled the drive and joint flexibility using elements deformable in multiple directions but could not identify the joint damping properly, depriving them from the computation of stability lobe diagrams. The above considerations therefore suggest the interest of developing models involving all the phenomena likely to influence on the one hand the posture dependent modes and on the other hand the interaction with the workpiece during milling operations. Among eligible phenomena, joint flexibility (in axial and transversal directions), link flexibilities, friction and controller dynamics can be cited.

Chapter 3
Robot dynamic model

In order to better understand the mechanisms underlying chatter in robotic machining, models able to cope with the pose-dependent nature and non-linearity of robots must be developed. Without a satisfactory modelling of the whole process, improvements will remain poor. As past studies suggested, multibody models seem appropriate to describe the varying dynamics of such system in their workspace. Accordingly, the multibody dynamic modelling of an industrial robot is developed in this chapter. The different elements composing the robotic machining cell are first described before focussing on the modelling of the industrial robot. The model of a typical serial manipulator with six actuated joints is presented. The kinematic model is derived from its kinematic chain connecting the links by means of joints. The clamped end of the chain is called *base* while the moving and controlled tip is termed *end-effector*. The positioning of the end effector results from the composition of the elementary displacements of each link with respect to the preceding one. Hence, it is necessary to characterise the end-effector position and orientation in order to manipulate the cutting tool. Besides, forces and torques responsible for the motion must be computed through the dynamic modelling with the derivation of the equations of motion. The first dynamic model considers the motion of rigid links connected with rigid joints. Thereafter, the dynamic model is extended by including various sources of compliance such as the joints, the links, the actuators and the controllers. Joint flexibility around and normal to the motion axes are introduced and a new approach to take into account the link deformations is derived. The control of the flexible manipulator is eventually addressed with the actuator modelling and two control schemes. The consideration of a gravity compensation system, reducing the actuator load, is also covered.

3.1 Robotic machining cell

A robotic machining cell was recently set up at the University of Mons as a result of the joint collaboration of two departments: the Department of Theoretical Mechanics, Dynamics and Vibrations and the Department of Machine Design and Production Engineering.

3.1.1 Overview

Figure 3.1 depicts the actual robotic machining cell. The dedicated area spans over 5 $\times$ 6 m and is surrounded by safety fences. The core component is of course the industrial robot whose location is slightly offset in the workspace. The considered robot is the Stäubli TX200 robot outlining a serial architecture with six rotational joints. Its nominal

payload is 1000 N and has a workspace of 2194 mm. The weight of the unloaded robot is about 1000 kg. The foot of the robot is clamped to a steel slab on which a bench vice is attached. Unsure of the ground foundation quality, the steel slab was inserted for ease of fastening to the robot. In addition, the bench vice is at the correct height to manipulate the workpiece to machine. Prior to the installation, a new concrete floor slab was poured to level the ground and secure the steel slab. Electrical outlets are also available within the cell. Nevertheless, a robot alone would be unable to move without a controller. Therefore, a controller cabinet is located outside the cell gathering the robot controller sending the commanded positions to the robot joints and amplifiers magnifying the controller signals for the servo motors. Furthermore, it provides the hardware for the robot safety system, such as emergency stops and power supply of servo amplifiers, including the motor brakes. Another cabinet is disposed next to the controller and is mainly dedicated to the completion of milling operations. The electrical cabinet was designed by the integrator so that operators can easily start milling operations with robots. It possesses a tactile screen to monitor the milling operation and connecting ports to load the end effector trajectory and cutting conditions. A pressurised air circuit is also included to cool down the spindle attached to the robot end effector.

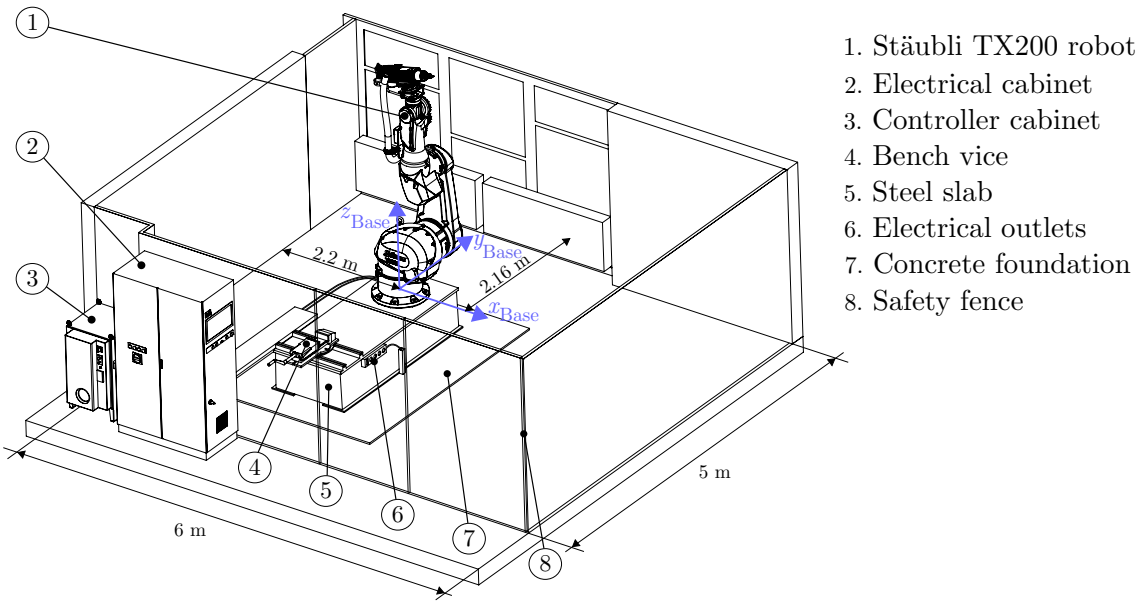


Figure 3.1: Robotic machining cell at the University of Mons

A closer look at the robot is provided in Figure 3.2 introducing the designation of all its components. The foot of the robot is commonly named the *base* of the robot. The other links are named after the close resemblance to the human arm. The first joint is located between the robot base and its *shoulder*, it rotates around a vertical axis. The shoulder link encloses both servo motors to actuate joints 1 and 2. Axis of the second motor is perpendicular to the previous one so that the second joint rotates around a horizontal axis. The longest link is connected to the shoulder and is known as the *arm*. This hollowed limb do not enclose any servo motor but encases the gravity compensator system instead, which aims at balancing the weight of the cantilevered payload. The *elbow* connects the arm and the *forearm* and encloses both servo motors to move joints 3 and 4 which are also

perpendicular to each other. The space inside the forearm is used to hold the fifth servo motor that actuates joint 5 driving the *wrist*. Lastly, the last servo motor is enclosed inside the wrist and moves the robot *flange* through joint 6. Without any payload, the centre of the flange top surface designates the robot end effector. Overall, robot manufacturer guarantees a repeatability of ± 0.06 mm for the end effector positioning under its nominal load.

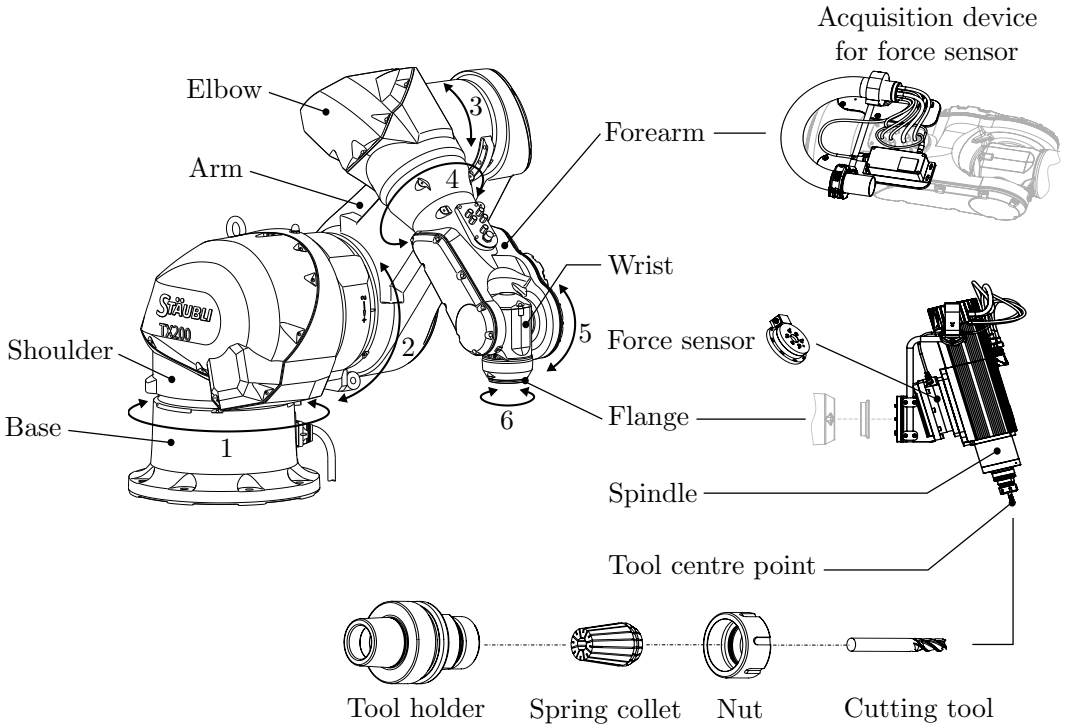


Figure 3.2: Stäubli TX200 robot at the University of Mons

For machining applications, an adequate tooling is fastened to the robot flange. Besides supporting a spindle, the installed Stäubli TX200 robot carries a force/torque sensor. Transducer ATI Omega160 is able to measure forces along the X, Y and Z directions and also the torques around the aforementioned axes. It is located just before the spindle to better perceive the effects of the cutting forces. Force measurement ranges from 1000 N along the planar directions to 2500 N along the axial axis while the torque scope is rather constant at 120 Nm. On the other side, the bottom plate of the force sensor is mounted on a support fixed to the robot flange. The support introduces an angle orienting the spindle axis. As mentioned in the literature review, such angle is often embedded in robot tooling to avoid singularity issues. Besides the sensor, its acquisition system is directly fixed to the robot forearm which delivers both electrical and pneumatic outlets. It converts the force and torque loads into electrical signals but adds some weight to the forearm. Behind the force sensor, the spindle rotates a tool holder in which the cutting tool is firmly clamped. The Teknomotor ATC71 spindle has a power of 7.5 kW and keeps a nominal torque of 6 Nm up to 12,000 rev/min. Its maximum speed is 24,000 rev/min. The attachment system for the cutting tool is made of three components: the tool holder, the spring collet and the nut. The cutting tool is inserted into a spring collet. The latter is put in the tool holder and the tightening of the nut around the spring collet allows

clamping the cutting tool. The conical portion of the tool holder is finally tightened into the spindle. Note that the distance between the tool tip and the flat surface just after the conical portion must be measured in order to know the position of the tool centre point (TCP).

3.1.2 Programming environment

There are mainly three ways to off-line program the presented robot for a machining operation. Presented in Figure 3.3, they can be described as follow:

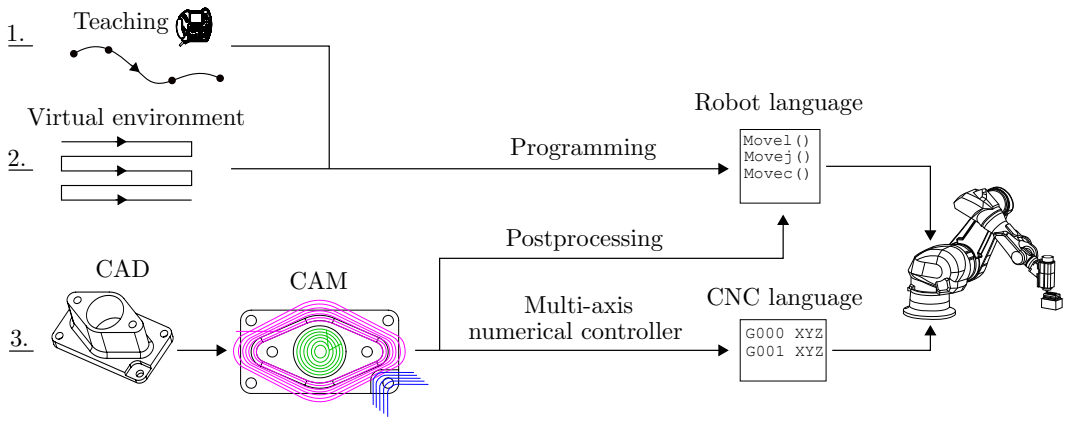


Figure 3.3: Overview of the off-line programming process

1. A teach pendant is usually supplied with the industrial robot. It looks like a remote controller to command the robot and gathers all the functionalities to write short programs. The robot can be moved on the demand by pressing buttons, joint per joint or in the Cartesian space. In this way, operator can manually teach series of points in the space through which the robot end effector will pass. As it is a manual operation, the positioning of the taught points is not accurate but might be useful to design trajectories for complex shape parts to deburr. With this method, the robot is directly programmed with its native language.

2. Another possibility of off-line programming relies on robot-oriented virtual environment also supplied by robot manufacturer. Operators write robot applications using all the dedicated functions to program the robot. The positioning of the points is accurate in the virtual environment. However, operators need to program all tool paths for all machining operations which can be time-consuming. The robot-oriented virtual environment proposed by Stäubli is named *Stäubli Robotics Suite* (SRS).

3. The last option is based on the purchase of CAD and CAM software. Operator designs the part to machine in a CAD environment before importing the resulting 3D model to CAM software. In CAM software, tool paths are automatically generated from the requested machining operations. Cutting parameters can also be entered to estimate the process duration. However, as no robot dynamic model is implemented, there is no way to optimise the cutting parameters. Once the tool

paths are generated, CAM software generates a program written in a programming language common to CNC machine tools, the so-called G-code. Two options are then possible: either translate the program into the robot programming language (available option is some CAM software) or directly use the G-code to move the robot. Multi-axis numerical controller is able to deal with the CNC code to actuate each joint accordingly.

On the other hand, on-line programming is also possible in combination with an external sensory system, e.g. a laser tracker or force sensor. Sensors react to any change in the desired motion and correct the trajectory accordingly.

In this work, the third way of off-line programming is used to achieve milling tests. The part to machine is first drawn in a CAD environment and tool paths are automatically generated using a CAM software. The generated G-code is then sent to the multi-axis numerical controller in order to actuate each joint. In other words, the kinematics of each joint is treated separately.

3.2 Kinematics of the Stäubli TX200 robot

The entry point in the robot modelling starts with the kinematics description of the manipulator. The Stäubli TX200 robot is a serial robot with six revolute joints. It is an anthropomorphic manipulator with a spherical wrist whose general representation is illustrated in Figure 3.4. By convention, O_{Base} is the orthonormal reference frame and x, y, z are the frame axes. The beginning of the open kinematic chain is given by frame O_{Base} while the end effector is referenced by frame O_{EE}. Note that direction z_{EE} is usually pointing outwards along the last rotation axis. Let θ_{i} be the **desired** motor angle (from the controller viewpoint) but seen from the joint side while $\theta_{\text{m,i}} = k_{\text{i}}\theta_{\text{i}}$ is the desired motor angle, with k_{i} the reduction ratio of the gearbox corresponding to motor i.

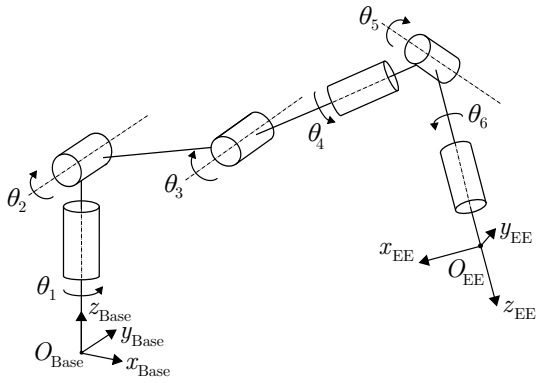
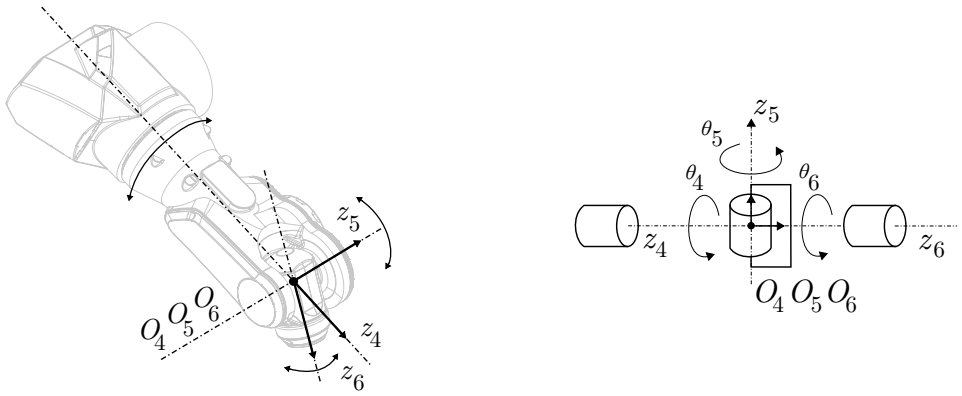


Figure 3.4: Anthropomorphic arm with spherical wrist [3]

The term “spherical wrist” refers to the arrangement of the last three axes of the manipulator. Since all joints are revolute, they all exhibit a rotation axis. In the particular case of a spherical wrist, rotation axes of the last three joints intersect at a single point. In particular for the Stäubli TX200 robot, the last three axes intersect inside the wrist as depicted in Figure 3.5a. Their arrangement follows the ZYZ Euler angles convention with

the first and last axes which are parallel and the second one being perpendicular to the two others. Such arrangement of the last three axes contributes to the orientation of the end effector in the reference frame. The first three joints are dedicated to the positioning of the intersection point of the spherical wrist. Hence, the pose of frame O_6, defining its orientation and the positioning, can be decoupled between the first and last three joints. This axis disposition is the most common one for industrial robots.



(a) Intersections of axes 4, 5 and 6 (b) Spherical wrist singularity

Figure 3.5: Spherical wrist of the Stäubli TX200 robot

However, the particular disposition of the axes also leads to particular situations commonly called *singularities*. A singular pose designates a situation in which the manipulator mobility is reduced, corresponding to a loss of degree of freedom. Singularities are classified into two categories: the *boundary* singularities happen when the manipulator is outstretched or retracted and the *internal* singularities which arise inside the reachable workspace. While boundary singularity can be easily avoided, internal singularities are generally caused by the alignment of two or more axes of motion or in particular postures [114]. Aside from the boundary singularities, the Stäubli TX200 robot can come across two forms of internal singularities. The most notable one is the wrist singularity (Figure 3.5b). In the particular situations in which $\theta_5 = 0$ or multiples of π, axes 4 and 6 are aligned. There is a loss of mobility since rotations of equal magnitude about opposite directions on θ_4 and θ_6 do not produce any end effector rotation. The other internal singularity is the shoulder singularity and happens when frame O_6 crosses the Z axis of base frame O_{Base}. In this situation, rotations of equal magnitude about opposite directions on θ_1 and θ_6 do not change the end effector pose. As a consequence, a particular attention must be considered in programming of the end effector motion to avoid such situations.

3.2.1 Geometric model

The robot geometric model describes the pose of the frame attached to the end effector with respect to the reference frame. In a robot manipulator, since only the joint angles are measured by means of encoder, the geometric model must be expressed in terms of (desired) joint variables θ_{i}. The space in which joint variables vary is named the *joint space*. The $(n_\theta \times 1)$ vector of desired joint variables is defined as

$$\boldsymbol{\Theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \end{bmatrix}, \tag{3.1}$$

with n_θ being the number of joints of the manipulator. In the particular case of the Stäubli TX200 robot n_θ=6. However, tasks achieved by robots are often defined in the Cartesian space with respect to the reference frame, e.g. moving the robot end effector parallel to axis x_{Base}. Hence, a task-oriented space is defined and called *operational space*. The end effector pose $\mathbf{x}_{\text{EE}}$ is composed of its position $\mathbf{p}_{\text{EE}}$ and its orientation $\boldsymbol{\phi}_{\text{EE}}$, which are functions of time if a trajectory is imposed. The position is easily given by coordinates along the X, Y and Z directions with respect to the base frame. For the rotation, the minimal representation using the XYZ Euler angles is commonly chosen

$$\mathbf{x}_{\text{EE}} = \begin{bmatrix} \mathbf{p}_{\text{EE}} \\ \boldsymbol{\phi}_{\text{EE}} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} p_{\text{EE},x} \\ p_{\text{EE},y} \\ p_{\text{EE},z} \end{bmatrix} \\ \begin{bmatrix} \alpha_{\text{EE}} \\ \beta_{\text{EE}} \\ \gamma_{\text{EE}} \end{bmatrix} \end{bmatrix}. \tag{3.2}$$

Figure 3.6 represents the end effector pose with the adopted convention. Vector $\mathbf{p}_{\text{EE}}$ defines the actual position of the end effector through its components $[p_{\text{EE},x}, p_{\text{EE},y}, p_{\text{EE},z}]^T$. The final orientation of the end effector is determined by three successive rotations around its current local axes. Given the order of rotations, the first rotation is achieved around x'''_{EE}. The second rotation takes place around new axis y''_{EE}. The final rotation occurs around new axis z'_{EE} leading to the resulting end effector pose in O_{EE}.

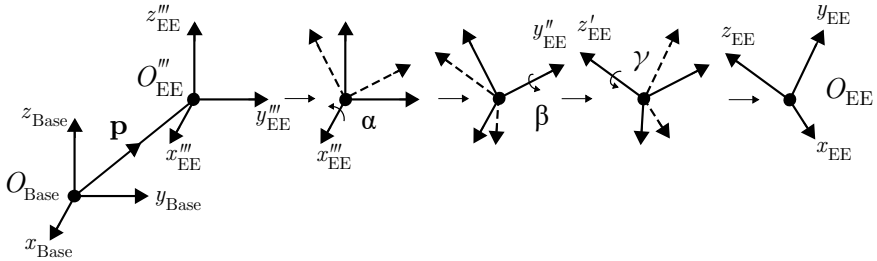


Figure 3.6: End effector pose [115]

Since joint angles determine the pose of the end effector, it means that there is a kinematic relationship between both which can be expressed in terms of joint variables as

$$\mathbf{x}_{\text{EE}} = f(\boldsymbol{\Theta}). \tag{3.3}$$

Kinematics function f refers to the *direct kinematics* and allows computing the end effector pose in the operational space from the knowledge of the joint space variables. For a set of

joint variables, the end effector pose is uniquely defined. Nevertheless, the task is generally not specified in joint space. It is in fact not convenient to position the end effector in the operational space from the joint angles. Therefore, the task trajectory is rather converted in desired joint angles. The latter form the set points that each joint must reach in order to complete the requested task. In other words, the *inverse kinematics* problem consists in the determination of the joint variables corresponding to a given end effector position and orientation. The solution to this problem is of fundamental importance in order to transform the motion specifications that are assigned to the end effector in the operational space, into the corresponding joint space motions that allow execution of the desired motion [3]. The inverse kinematics problem can be stated as follows

$$\mathbf{\Theta} = f^{-1}(\mathbf{x}_{\mathrm{EE}}). \tag{3.4}$$

The geometric modelling of a robot manipulator is classically handled with the Denavit-Hartenberg convention using the so-called *DH parameters* [116]. The convention proposes a systematic and general method to define the relative position and orientation of two consecutive links. Succinctly, the problem is solved by determining two frames attached to two successive links and computing the coordinate transformation between them (Figure 3.7). Historically, there are two different conventions: the standard Denavit-Hartenberg convention and the modified Denavit-Hartenberg convention which was proposed by Khalil and Kleinfinger for a better ease of application [117]. Since the standard DH parameters are still widely used, the geometric model of the Stäubli TX200 is based on the original parameters.

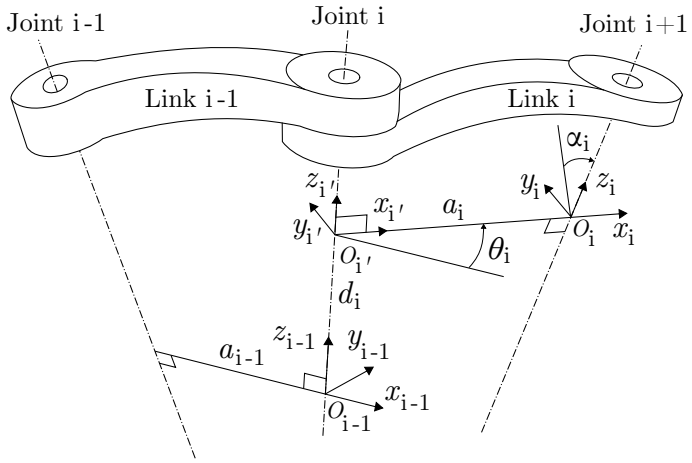


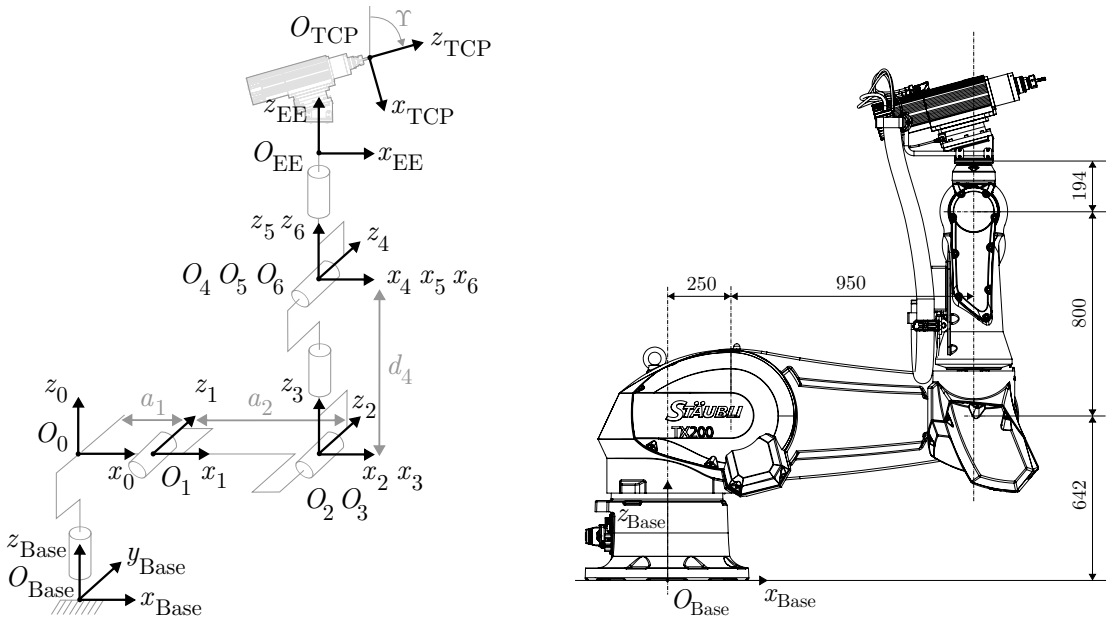
Figure 3.7: Standard Denavit-Hartenberg convention [3]

To each robot joint i is assigned a reference frame O_i for which the Z direction is commonly chosen along the rotation axis in the case of revolute joints. The defined axis i connects link i-1 to link i. Regarding the X direction, it is often chosen along the longest link dimension. Using the standard DH convention, a set of four parameters describes the location of each robot joint with respect to the previous one. The coordinate transformation from frame $O_{i\text{-}1}$ to frame O_i is completely described using the four following parameters:

1. θ_i: angle representing the motor angle from the joint side. It is defined from axis $x_{i\text{-}1}$ to axis x_i around axis $z_{i\text{-}1}$.

3. a_i: distance between transitional frame $O_{i'}$ and frame O_i which represents the common normal to their rotation axes. Rotation axis passing through O_{i-1} is aligned with the one passing through $O_{i'}$. The origin of frame $O_{i'}$ is accordingly defined at the intersection of the common normal and the rotation axis passing through $O_{i'}$.
2. d_i: distance between frame O_{i-1} and transitional frame $O_{i'}$ along the common rotation axis.
4. α_i: angle describing the orientation offset between two successive rotation axes. It is defined from axis $z_{i'}$ to z_i around axis x_i

Having defined the DH convention, it can be readily applied to the geometric definition of the Stäubli TX200 robot. Figure 3.8 depicts the application of the standard DH convention to the Stäubli TX200 robot by highlighting the joint frames in relation with the actual robot.



(a) Geometric pose (b) Relevant dimensions [mm]

Figure 3.8: Standard Denavit-Hartenberg convention applied to the Stäubli TX200 robot

Since the main application regards milling operations, the spindle with a cutting tool is appended to the end effector (Figure 3.8a). Of course, direct and inverse kinematics problems are solved for the tool centre point position since the milling task refers to the tool tip of the cutting tool. Consequently, subscripts EE indicating the end effector in equations 3.2 to 3.4 are updated with subscripts TCP. Vector of joint variables $\boldsymbol{\Theta}$ is not affected since the coordinate transformation between O_{EE} and O_{TCP} is constant. The orientation of frame O_0 is positioned on the rotation axis of the first joint at the same level as the second joint. The orientation of the X axis of frame O_0 is generally chosen in the opposite direction to the cable outlet of the robot base. Then, the Z axis is made vertical and pointing upwards thus defining the orientation of the Y axis since O_0 is an orthonormal frame. The base frame of the cell O_{Base} is defined in regards to Figure 3.8b. Its centre is positioned at the bottom of the robot base and aligned with the rotation

axis of the first joint. Its orientation is the same as frame O_0. Relevant dimensions of the robot to define its geometric model are also found in Figure 3.8b. Note that the code name of the robot (TX200) relates to the length of the arm when it is outstretched horizontally. There are indeed 200 centimetres between the first axis and the centre point of the spherical wrist for the Stäubli TX200 robot.

i	a_i [m]	α_i [rad]	d_i [m]	θ_i [rad]
1	0.25	-$\pi/2$	0	θ_1
2	0.95	0	0	θ_2
3	0	$\pi/2$	0	θ_3
4	0	-$\pi/2$	0.8	θ_4
5	0	$\pi/2$	0	θ_5
6	0	0	0	θ_6

Table 3.1: Stäubli TX200 standard Denavit-Hartenberg parameters

The standard Denavit-Hartenberg parameters are provided in Table 3.1 for the Stäubli TX200 robot. Since the representation starts from frame O_0, the location of frame O_1 is defined on the second joint at a distance a_1 measured in O_0 along x_0. Its orientation must be changed so that z_1 is aligned with the rotation axis of the second joint. Frame O_2 is found at a distance a_2 from frame O_1 along x_1 without orientation change. Then, there is an orientation change to define frame O_3 so that the rotation axis of the fourth joint matches the direction of z_3. Finally, the centre point of the spherical wrist is located at a distance d_4 separating frames O_3 and O_4 along the rotation axis of the fourth joint. Axes z_4 and z_5 are oriented such that they correspond to the axes of rotation of joints five and six, respectively. Frame O_6 is made coincident with frame O_5.

O_i	x [m]	y [m]	z [m]	Υ [rad]
Base	0	0	0.642	0
EE	0	0	0.194	0
TCP	0.277	0	0.326	$\frac{5}{12}\pi$

Table 3.2: Tool and base coordinate transformations

The standard DH convention begins from robot frame O_0 and stops at the location of the spherical wrist centre point. From there, the end effector frame O_EE is determined from frame O_6 using conventional coordinates x, y and z. Its orientation is typically the same as frame O_0 in the robot posture in which the DH parameters are identified. Similarly, the tool centre point frame O_TCP is eventually found with respect to the end effector frame O_EE with angle Υ giving the orientation of the rotation axis of the spindle, therefore defining z_TCP. Likewise, the location of base frame O_Base is given by a vertical offset separating frame O_0 along the first rotation axis z_0. Table 3.2 summarises the tool and base coordinate transformations.

The schematic representation of the kinematic chain in Figure 3.8a is called the robot *geometric pose* and allows defining the DH parameters of the manipulator $\mathbf{\Theta}_\mathrm{Geometric} =$

$[0\,0\,0\,0\,0\,0]^T$. However, such postures usually illustrate robot configurations where the joint variable values are different from zero in the encoder readings $\boldsymbol{\Theta}$ proposed by the robot manufacturers, i.e. such values differ from the null references used for robot manipulator programming. Hence, it is necessary to add constant contributions $\boldsymbol{\Theta}_{\text{Offset}}$ to the values of the joint variables in the geometric pose in order to match the ones measured by the robot encoders. The posture in which all the joint variable values are null from the encoder readings is called the *encoder pose*. For the Stäubli TX200 robot, the encoder pose showcases a robot vertically outstretched $\boldsymbol{\Theta} = [0\,0\,0\,0\,0\,0]^T$ (Figure 3.9). Consequently, offset values are added to the geometric pose to match the encoder pose such as

$$\boldsymbol{\Theta} = \boldsymbol{\Theta}_{\text{Geometric}} + \boldsymbol{\Theta}_{\text{Offset}}, \tag{3.5}$$

in which $\boldsymbol{\Theta}_{\text{Offset}} = [0\ -\frac{\pi}{2}\ \frac{\pi}{2}\ 0\,0\,0]^T$ for the Stäubli TX200 robot.

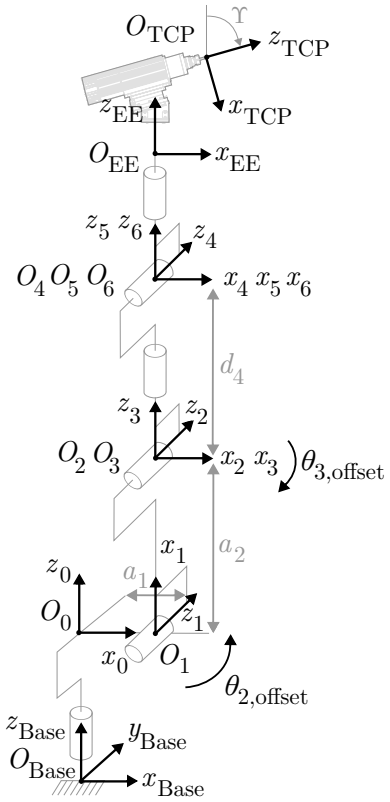


Figure 3.9: Encoder pose with null references for all joint variables $\boldsymbol{\Theta}$

Throughout the rest of the document, values of joint variables $\boldsymbol{\Theta}$ are expressed in the encoder pose, i.e. $\boldsymbol{\Theta} = [0\,0\,0\,0\,0\,0]^T$ in the vertical posture presented in Figure 3.9. Now that the reference robot configuration is defined, ranges of motion for each joint can be provided and are delivered in Table 3.3. Direction of rotations is settled using the trigonometric convention. In addition, the maximum velocity of each joint is shown.

3.2.2 Direct kinematics

Having defined the geometric model of the Stäubli TX200 robot situating the joint frames with respect to the base frame, the direct kinematics problem is addressed. As a

	Axis 1	Axis 2	Axis 3	Axis 4	Axis 5	Axis 6
Motion range [°]	± 180	± 120	+145/−140	± 270	± 120	± 270
Maximum velocity [°/s]	150	150	150	260	260	400

Table 3.3: Joint motion range and maximum velocity of the Stäubli TX200 robot

reminder, it consists in determining function $f(\boldsymbol{\Theta})$, which is usually a non-linear function in the joint variables, to express the tool centre point pose in the base frame such as $\mathbf{x}_{\text{TCP}} = f(\boldsymbol{\Theta})$.

A convenient way to express the relationship between the coordinates of the same point in two different frames consists in the usage of *homogeneous transformation matrix*. It offers a compact representation to define the position and orientation of a frame. Homogeneous transformation matrix $\mathbf{T}_{\text{i,j}}$ gives the position and orientation of frame j with respect to frame i. The coordinate transformation can be written as a (4×4) matrix

$$\mathbf{T}_{\text{i,j}} = \begin{bmatrix} \mathbf{R}_{\text{i,j}} & \{\mathbf{p}_{\text{j}}\}_{\text{i}} \\ 0\ 0\ 0 & 1 \end{bmatrix}, \tag{3.6}$$

in which $\{\mathbf{p}_{\text{j}}\}_{\text{i}}$ is the position vector of frame j projected[1] in frame i and $\mathbf{R}_{\text{i,j}}$ reflecting the orientation of frame j relatively to frame i[2]. The rotation matrix is interpreted as a matrix operator that allows rotating a vector by a given angle about an arbitrary axis. It is written as a (3×3) matrix. In regards to the XYZ Euler angles presented in Figure 3.6, the corresponding rotation matrix is expressed in terms of the direction cosines as follows[3]

$$\mathbf{R}_{\text{Base,EE}} = \begin{bmatrix} c_\beta c_\gamma & -c_\beta s_\gamma & s_\beta \\ s_\alpha s_\beta c_\gamma + c_\alpha s_\gamma & -s_\alpha s_\beta s_\gamma + c_\alpha c_\gamma & -s_\alpha c_\beta \\ -c_\alpha s_\beta c_\gamma + s_\alpha s_\gamma & c_\alpha s_\beta s_\gamma + s_\alpha c_\gamma & c_\alpha c_\beta \end{bmatrix} = \begin{bmatrix} S_x & N_x & A_x \\ S_y & N_y & A_y \\ S_z & N_z & A_z \end{bmatrix}. \tag{3.7}$$

For the sake of readability, notations expressing elementary motions from frame i to frame j are introduced. Notation $\mathbf{T}_{\text{disp}}(p_x, p_y, p_z)$ returns a homogeneous transformation matrix corresponding to a displacement without rotation by coordinates p_x, p_y and p_z and $\mathbf{T}_{\text{rotx}}(\alpha)$ corresponds to a rotation around the X axis by angle α, such as

$$\mathbf{T}_{\text{disp}}(p_x, p_y, p_z) = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad \mathbf{T}_{\text{rotx}}(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_\alpha & -s_\alpha & 0 \\ 0 & s_\alpha & c_\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.8}$$

[1]Projection in a frame is denoted by a subscript such that vector $\bullet$ projected in base frame O_{Base} reads $\{\bullet\}_{\text{Base}}$.

[2]Origin of frame (O) is dropped in subscript such that rotation matrix of frame O_{TCP} relatively to base frame O_{Base} is denoted by $\mathbf{R}_{\text{Base,TCP}}$.

[3]Abbreviations $c_\bullet$ and $s_\bullet$ stands for $\sin(\bullet)$ and $\cos(\bullet)$, respectively. Note that if the name of the angle is not specified, angle θ is assumed as for example c_1 is equivalent to $\cos(\theta_1)$.

Similarly, $\mathbf{T}_{\text{roty}}(\alpha)$ and $\mathbf{T}_{\text{rotz}}(\alpha)$ correspond to rotations around the Y and Z axes by angle α, respectively, such as

$$\mathbf{T}_{\text{roty}}(\alpha) = \begin{bmatrix} c_\alpha & 0 & s_\alpha & 0 \\ 0 & 1 & 0 & 0 \\ -s_\alpha & 0 & c_\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad \mathbf{T}_{\text{rotz}}(\alpha) = \begin{bmatrix} c_\alpha & -s_\alpha & 0 & 0 \\ s_\alpha & c_\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.9}$$

The standard Denavit-Hartenberg convention can be expressed in terms of homogeneous transformation matrices. The relative position and orientation of two consecutive links is provided by the successive multiplication of four elementary homogeneous transformation matrices. More specifically, the coordinate transformation from frame $O_{\text{i-1}}$ to O_{i} using the standard DH parameters reads

$$\mathbf{T}_{\text{i-1,i}} = \mathbf{T}_{\text{rotz}}(\theta_{\text{i}}) \cdot \mathbf{T}_{\text{disp}}(0, 0, d_{\text{i}}) \cdot \mathbf{T}_{\text{disp}}(a_{\text{i}}, 0, 0) \cdot \mathbf{T}_{\text{rotx}}(\alpha_{\text{i}}). \tag{3.10}$$

Since the multiplication of elementary homogeneous transformation matrices leads to the definition of complex transformation matrix thanks to the following statement

$$\mathbf{T}_{\text{i,k}} = \mathbf{T}_{\text{i,j}} \cdot \mathbf{T}_{\text{j,k}}, \tag{3.11}$$

the direct kinematics problem is solved by successively applying the property. As a result, homogeneous transformation matrix $\mathbf{T}_{0,6}(\boldsymbol{\Theta})$, describing the pose of frame O_6 with respect to robot frame O_0 in terms of the joint variables, can be decomposed as follows

$$\mathbf{T}_{0,6}(\boldsymbol{\Theta}) = \mathbf{T}_{0,1}(\theta_1) \cdot \mathbf{T}_{1,2}(\theta_2) \cdot \mathbf{T}_{2,3}(\theta_3) \cdot \mathbf{T}_{3,4}(\theta_4) \cdot \mathbf{T}_{4,5}(\theta_5) \cdot \mathbf{T}_{5,6}(\theta_6). \tag{3.12}$$

The elementary homogeneous transformation matrices needed to build the direct kinematics of the Stäubli TX200 robot are provided below in terms of the standard Denavit-Hartenberg parameters such as

$$\begin{aligned}
\mathbf{T}_{0,1} &= \begin{bmatrix} c_1 & 0 & -s_1 & a_1 c_1 \\ s_1 & 0 & c_1 & a_1 s_1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, & \mathbf{T}_{1,2} &= \begin{bmatrix} c_2 & -s_2 & 0 & a_2 c_2 \\ s_2 & c_2 & 0 & a_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
\mathbf{T}_{2,3} &= \begin{bmatrix} c_3 & 0 & s_3 & 0 \\ s_3 & 0 & -c_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, & \mathbf{T}_{3,4} &= \begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
\mathbf{T}_{4,5} &= \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, & \mathbf{T}_{5,6} &= \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
\end{aligned} \tag{3.13}$$

Considering that the standard Denavit-Hartenberg convention begins from frame O_0 and stops at the centre point of the spherical wrist in frame O_6, the complete direct kinematics from base frame O_{Base} to TCP frame O_{TCP} leads to the following expression

$$\mathbf{T}_{\text{Base,TCP}}(\boldsymbol{\Theta}) = \mathbf{T}_{\text{Base,0}} \cdot \mathbf{T}_{0,6} \cdot \mathbf{T}_{6,\text{EE}} \cdot \mathbf{T}_{\text{EE,TCP}}, \tag{3.14}$$

with $\mathbf{T}_{\text{Base,0}}$, $\mathbf{T}_{6,\text{EE}}$ and $\mathbf{T}_{\text{EE,TCP}}$ being constant transformations from base frame O_{Base} to frame O_0, from frame O_6 to end effector frame O_{EE} and from end effector frame O_{EE} to TCP frame O_{TCP}, respectively. Joint angle offsets defined in Eq. 3.5 are applied to recover the encoder pose. Using parameters defined in Table 3.2, the base and tool transformations are also provided in terms of homogeneous transformation matrices such as

$$\mathbf{T}_{\text{Base,0}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & z_{\text{Base}} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.15}$$

$$\mathbf{T}_{6,\text{EE}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & z_{\text{EE}} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad \mathbf{T}_{\text{EE,TCP}} = \begin{bmatrix} c_\Upsilon & 0 & s_\Upsilon & x_{\text{TCP}} \\ 0 & 1 & 0 & 0 \\ -s_\Upsilon & 0 & c_\Upsilon & z_{\text{TCP}} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.16}$$

3.2.3 Inverse kinematics

The inverse kinematics problem aims at determining the set of joint variables $\boldsymbol{\Theta}$ corresponding to the desired end effector (or TCP) pose $\mathbf{x}_{\text{TCP}}$ such that $\boldsymbol{\Theta} = f^{-1}(\mathbf{x}_{\text{TCP}})$. In contrast with the direct kinematics problem in which a set of joint variables $\boldsymbol{\Theta}$ defines a unique TCP pose, the inverse problem generally leads to multiple solutions. In other words, for a desired TCP pose, several sets of joint variables are possible still fulfilling the pose requirements both in terms of position and orientation. In general, for 6-axis manipulators with an open kinematic chain, the number of solutions to the inverse problem can reach up to 16 solutions. However, knowing that the considered robot owns a spherical wrist decoupling position from rotation, the number of solutions reduces to eight for one desired TCP pose. It is typically the case for most industrial robots with six axes as they usually possess a spherical wrist.

For an anthropomorphic manipulator with six axes as the Stäubli TX200 robot, there is an analytical solution to the inverse geometric problem which allows determining the eight possible sets of joint variables $\boldsymbol{\Theta}$. Several methods exist in the literature among which two of them are quite popular: the Paul's method and the Pieper's method [57, 114, 118, 119]. The Pieper's method is the most general one and can handle the inverse kinematics problem for manipulators which do not have a spherical wrist.

Hence, it is computationally less efficient than the Paul's method which is solely dedicated to manipulators with spherical wrist. Paul's method takes advantage of the decoupling between position and orientation to determine the pose of the tool centre point. First, knowing the TCP pose, the first three joint angles θ_1, θ_2 and θ_3 that define the position of the centre point of the spherical wrist are computed. Then, the orientation of the TCP pose is processed and the last three joint angles θ_4, θ_5 and θ_6 are found. Since, the Paul's method proved to be numerically faster, it was retained for solving the inverse kinematics problem. Detail of the Paul's method implementation is found in Appendix A.

The eight possible robot configurations are derived by the method and each of them can be identified by three identifiers:

1. *Righty* and *lefty*: term is related to the orientation of the first joint with respect to the position of tool centre point. In a righty configuration, the robot arm is located on the right-hand side with respect to the TCP. For a lefty configuration, the robot arm is naturally positioned on the left-hand side of the task.

2. *Up* and *down*: term is referring to the orientation of the second joint. If the robot arm is said up, it is therefore pointing to an upwards direction otherwise, it is pointing to a downwards direction (down).

3. *Flip* or *no flip*: term describes the orientation of the fourth joint which influences the rotations of joints 5 and 6. The wrist is naturally in a flipped configuration e.g. in the encoder pose. If there is an offset of 180° for the fourth joint, the wrist is said not flipped.

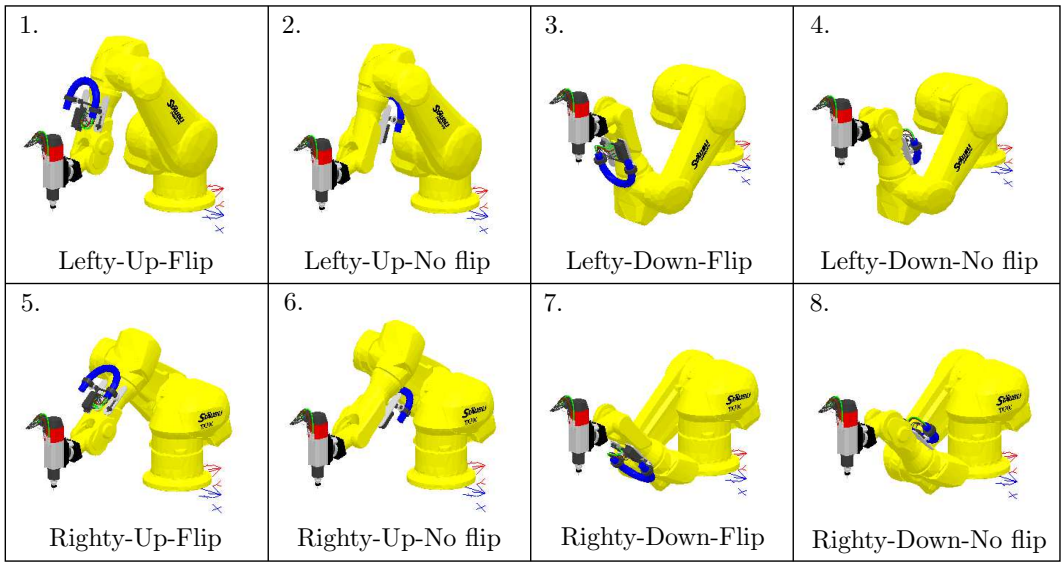


Figure 3.10: The eight solutions to the inverse kinematics problem for the Stäubli TX200 robot

The visual effect on the robot configuration is depicted in Figure 3.10 in the case of the Stäubli TX200 robot. The same TCP pose $\mathbf{x}_{\text{TCP}}$ is achieved for the eight possible

configurations. The most commonly used posture for all kind of applications is the lefty-up-flip configuration. The latter is also used in the context of milling operations and in this work. The configurations showcasing a robot arm pointing downwards are excluded since it would interfere with the steel slab of the robotic machining cell. The remaining no flipped configurations are also rejected as they would damage the cables powering the force sensor and the spindle. Finally, the righty-up-flip configuration is not investigated due to its peculiar and uncommon aspect; usage of this posture was not reported in the literature.

3.2.4 Differential kinematics

The differential kinematics consists in expressing the velocities and accelerations of the joint variables ($\dot{\boldsymbol{\Theta}}$, $\ddot{\boldsymbol{\Theta}}$) and of the operational variables ($\dot{\mathbf{x}}_{\text{TCP}}$, $\ddot{\mathbf{x}}_{\text{TCP}}$) and the relationships between both. In other words, it is desired to express the TCP linear velocity $\dot{\mathbf{p}}_{\text{TCP}}$ and acceleration $\ddot{\mathbf{p}}_{\text{TCP}}$ and angular velocity $\boldsymbol{\omega}_{\text{TCP}}$ and acceleration $\dot{\boldsymbol{\omega}}_{\text{TCP}}$ as a function of the joint variables and their time derivatives. In the same way as for the geometric modelling, the direct and the inverse problems are solved.

Regarding the direct kinematics model, it can be demonstrated that the TCP velocity is linear in the joint velocity $\dot{\boldsymbol{\Theta}}$ such as

$$\begin{bmatrix} \{\dot{\mathbf{p}}_{\text{TCP}}\}_{\text{Base}} \\ \{\boldsymbol{\omega}_{\text{TCP}}\}_{\text{Base}} \end{bmatrix} = [\mathbf{J}(\boldsymbol{\Theta})]_{\text{Base}}\dot{\boldsymbol{\Theta}}, \tag{3.17}$$

with $\mathbf{J}(\boldsymbol{\Theta})$ the *geometric Jacobian* of the manipulator depending on the robot configuration via the joint variables contained in vector $\boldsymbol{\Theta}$. As suggested in Eq. 3.17, it is common to project the velocities of the TCP in base frame O_{Base}. The link with $\{\dot{\mathbf{x}}_{\text{TCP}}\}_{\text{Base}}$ is settled knowing that $\{\boldsymbol{\omega}_{\text{TCP}}\}_{\text{Base}} = \boldsymbol{\Xi}\ \{\dot{\boldsymbol{\phi}}_{\text{TCP}}\}_{\text{Base}}$, with matrix $\boldsymbol{\Xi}$ a function of the angles of the chosen orientation representation (α_{TCP}, β_{TCP} and γ_{TCP}). Geometric Jacobian $\mathbf{J}(\boldsymbol{\Theta})$ is a $(6 \times n_\theta)$ matrix since $\{\dot{\mathbf{x}}_{\text{TCP}}\}_{\text{Base}}$ gathers the TCP linear velocities $\{\dot{\mathbf{p}}_{\text{TCP}}\}_{\text{Base}}$ along the X, Y and Z axes and the angular velocities $\{\dot{\boldsymbol{\phi}}_{\text{TCP}}\}_{\text{Base}}$ around the same axes.

The expression of the TCP acceleration $\{\ddot{\mathbf{x}}_{\text{TCP}}\}_{\text{Base}}$ involves the derivative of the geometric Jacobian $[\dot{\mathbf{J}}]_{\text{Base}}$ which is also a $(6 \times n_\theta)$ matrix. The relationship is not solely linear in joint accelerations and reads

$$\begin{bmatrix} \{\ddot{\mathbf{p}}_{\text{TCP}}\}_{\text{Base}} \\ \{\dot{\boldsymbol{\omega}}_{\text{TCP}}\}_{\text{Base}} \end{bmatrix} = [\mathbf{J}]_{\text{Base}}\ddot{\boldsymbol{\Theta}} + [\dot{\mathbf{J}}]_{\text{Base}}\dot{\boldsymbol{\Theta}}. \tag{3.18}$$

Even if the direct kinematics relationships allow expressing the TCP velocities and accelerations ($\dot{\mathbf{x}}_{\text{TCP}}$, $\ddot{\mathbf{x}}_{\text{TCP}}$), the joint velocities and accelerations ($\dot{\boldsymbol{\Theta}}$, $\ddot{\boldsymbol{\Theta}}$) are often unknown. In contrast, task requirements often impose a specific kinematics for its execution e.g. a defined feed rate for a milling operation. TCP velocities and accelerations ($\dot{\mathbf{x}}_{\text{TCP}}$, $\ddot{\mathbf{x}}_{\text{TCP}}$) being usually known, the inverse relationships read

$$\dot{\boldsymbol{\Theta}} = [\mathbf{J}]^{-1}_{\text{Base}}(\boldsymbol{\Theta}) \begin{bmatrix} \{\dot{\mathbf{p}}_{\text{TCP}}\}_{\text{Base}} \\ \{\boldsymbol{\omega}_{\text{TCP}}\}_{\text{Base}} \end{bmatrix}, \tag{3.19}$$

$$\ddot{\boldsymbol{\Theta}} = [\mathbf{J}]^{-1}_{\text{Base}}(\begin{bmatrix} \{\ddot{\mathbf{p}}_{\text{TCP}}\}_{\text{Base}} \\ \{\dot{\boldsymbol{\omega}}_{\text{TCP}}\}_{\text{Base}} \end{bmatrix} - [\dot{\mathbf{J}}]_{\text{Base}}\dot{\boldsymbol{\Theta}}). \tag{3.20}$$

The knowledge of the joint velocities and accelerations ($\dot{\boldsymbol{\Theta}}$, $\ddot{\boldsymbol{\Theta}}$) is useful when the controller implements a feedforward action. Additional contributions comprising the joint velocities and accelerations are added to the control command to compensate dynamic effects. It ensures better performance and reduction of the tracking error along the trajectory. More insights regarding the derivation of the differential kinematics can be found in [3].

3.3 Dynamics of industrial robots with rigid links

The purpose of the dynamic modelling is to build the equations of motion of the manipulator representing its dynamic behaviour. The general form of the equations of motion is expressed as follows

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}(t) + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}(t) + \mathbf{K}\mathbf{q}(t) = \mathbf{F}(t), \tag{3.21}$$

where $\mathbf{M} \in \mathbb{R}^{n\times n}$, $\mathbf{C} \in \mathbb{R}^{n\times n}$ and $\mathbf{K} \in \mathbb{R}^{n\times n}$ denote the system mass, damping and stiffness matrices, respectively, with n the number of degrees of freedom. The degrees of freedom at the position level are given by vector $\mathbf{q}(t) \in \mathbb{R}^n$ and $\mathbf{F}(t) \in \mathbb{R}^n$ is the vector of applied forces.

Regarding the dynamic modelling, direct and inverse problems are also considered. Direct or forward dynamics refer to the description of the simulated motion when the manipulator is subjected to joint torques. When the trajectory of the TCP is known, the inverse kinematics procedure allows the computation of joint positions, velocities and accelerations. In addition, if the forces applied on the TCP are also known, inverse dynamics allow the computation of the joint torques to obtain the desired motion [3, 120]. It is typically the adopted procedure when a task is assigned to a manipulator. The same applies for milling operations. Hence, the overall idea behind the dynamic modelling of the considered robot is the constitution of equations of motion representing its dynamic behaviour to, on the one hand compute the joint torques from the command input, and on the other hand, solve the forward dynamics to simulate the resulting motion. Both problems are solved together to conduct a milling task.

Throughout the remaining of the chapter, the complete dynamic model of the Stäubli TX200 robot is incrementally built. A first dynamic model is derived with the multibody approach using the minimal coordinates. Then, the classical dynamic model of the motorised manipulator comprising rigid joints and rigid links is presented. The structural

flexibility is progressively introduced with a flexible joint model and a new method to deal with flexible bodies in minimal coordinates. Other compliance sources are investigated at the end of the chapter. These flexibilities of various origins can be seen as modelling options.

3.3.1 Multibody modelling with minimal coordinates

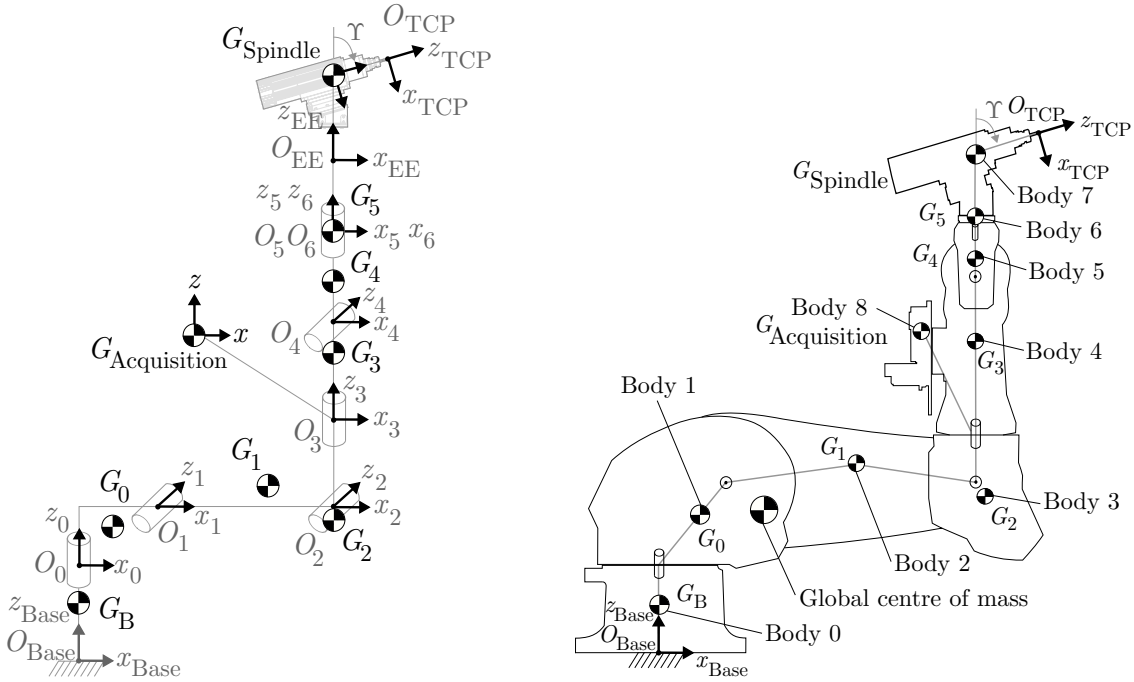
In practice, the building of a multibody system starts with the selection of a set of configuration parameters $\mathbf{q}$ as function of which the situation of each body is defined. In particular, each homogeneous transformation matrix $\mathbf{T}_{\text{Body i}}$, giving the situation of Body i with respect to base frame O_{Base}, is expressed as function of the set of configuration parameters $\mathbf{q}$. Depending on the selected coordinates, the number of configuration parameters can be equal to or larger than the number of degrees of freedom. If the number is larger, it means that the configuration parameters are not independent and are therefore subjected to constraint equations expressing the dependency. The most notable types of coordinates are the *relative*, *absolute*, *minimal*, *natural* or *nodal* coordinates, the first three being the most popular [121]. The choice naturally influences the principle used to build the equations of motion and consequently the numerical methods expected to deal with the resulting equations. The adoption of minimal coordinates is an appropriate choice to bypass the numerical resolution of constraint equations. Instead, the constraints are handled in the first place when selecting the configuration parameters expressing the kinematics of the system. Therefore, in the case of minimal coordinates, the configuration parameters are chosen freely as far as they are independent and that their number is equal to the number of degrees of freedom of the system [122]. Throughout the remaining of the document, the denomination in terms of degrees of freedom or configuration parameters is treated equivalently.

The minimal coordinates are applied to the building of the kinematics of the Stäubli TX200 robot. The basic case is first treated with the sole consideration of the motion around the robot axes as degrees of freedom. The model will be upgraded later on with the inclusion of the motors and the flexibilities. Comparatively with the kinematics derived with the joint variables in Section 3.2, the purpose is no longer to locate the joint axes but the centre of mass of each body. Homogeneous transformation matrices are still used to express the situation of each body with respect to base frame O_{Base}. Regarding the basic dynamic model, since the Stäubli TX200 robot has six joints, the number of degrees of freedom reduces to six as well with n=6. On the assumption of rigid transmissions seen from the joint side, vector of degrees of freedom $\mathbf{q}_{\text{s}}$ (subscript s refers to the motor shaft under control) is expressed as

$$\mathbf{q}_{\text{s}} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 & q_6 \end{bmatrix}^T. \tag{3.22}$$

Afterwards, the objective is to locate the frame corresponding to the centre of mass of each body. Property consisting in multiplying elementary homogeneous transformation matrices (Eq. 3.11) is of particular interest to express the pose of each centre of mass

frame. Illustrations presented in Figure 3.11 depict a representation of the multibody dynamic model of the robot pointing out the locations of the centres of mass for each body. A total of nine bodies are accounted in this basic dynamic model ($n_{\mathrm{B}} = 9$). Starting from index 0 to 6, they follow the robot kinematic chain from bottom to top, Body 0 being the robot base and Body 6 being the robot flange. Two additional bodies are appended to take into account the payloads: the spindle mounted on the robot flange (Body 7) and the acquisition device for the force sensor (Body 8).



(a) Joint frame definition (b) Centre of gravity location

Figure 3.11: Dynamic modelling of the Stäubli TX200 robot

Focussing on Figure 3.11a, it is observed that the kinematic definition of the multibody model is carried out in the same posture as the one used to define the standard DH parameters of the robot (Figure 3.8a). It allows keeping the same notations between both representations for an easier understanding of the modelling. Hence, the orientation of the joint frames also remains unchanged (joint angle offset defined in Eq. 3.5 needs to be applied once the dynamic modelling is completed). However, the joint locations from the DH representation are slightly shifted to better coincide with the physical locations of the joints (Figure 3.11b). The kinematics definition of the multibody model starts from base frame O_{Base} up to the TCP frame O_{TCP}. The provided distances to locate the centre of mass frames and the joint frames are expressed in the joint frame of the previous rotation axis $O_{\text{i-1}}$. For instance, distance O_0O_{1x} locating joint frame O_1 is defined in frame O_0 and projected along the x_0 axis. The same applies for the centre of mass frame G_0 whose spacing is given relatively to frame O_0 without orientation change. As a consequence, the pose of the centre of mass frame of Body 0 (the robot base) is expressed as follows relatively to the base frame

$$\mathbf{T}_{\mathrm{Base},G_{\mathrm{B}}} = \mathbf{T}_{\mathrm{Body\ 0}} = \mathbf{T}_{\mathrm{disp}}(O_{\mathrm{Base}}G_{\mathrm{B}x}, O_{\mathrm{Base}}G_{\mathrm{B}y}, O_{\mathrm{Base}}G_{\mathrm{B}z}), \tag{3.23}$$

where $\mathbf{T}_{\text{Body i}}$ locates the centre of mass of Body i with respect to base frame O_{Base}. As for example in this case, $O_{\text{Base}}G_{\text{B}x}$ is the distance between frames O_{Base} and G_{B} projected in O_{Base} along x_{Base}. All distance values are provided in Chapter 5 to define the multibody model of the Stäubli TX200 robot.

The pose of the centre of mass frame of Body 1 (the robot shoulder) involves the multiplication of three elementary homogeneous transformation matrices and the first degree of freedom q_1 as

$$\begin{aligned}\mathbf{T}_{G_{\text{B}},G_0} =&\mathbf{T}_{\text{disp}}(O_{\text{Base}}O_{0x} - O_{\text{Base}}G_{\text{B}x}, O_{\text{Base}}O_{0y} - O_{\text{Base}}G_{\text{B}y}, O_{\text{Base}}O_{0z} - O_{\text{Base}}G_{\text{B}z})\cdot\\ &\mathbf{T}_{\text{rotz}}(q_1)\cdot\mathbf{T}_{\text{disp}}(O_0G_{0x}, O_0G_{0y}, O_0G_{0z}).\end{aligned} \tag{3.24}$$

Its projection in base frame O_{Base} is obtained by pre-multiplying Eq. 3.24 by Eq. 3.23 as

$$\mathbf{T}_{\text{Base},G_0} = \mathbf{T}_{\text{Body 1}} = \mathbf{T}_{\text{Base},G_{\text{B}}}\cdot\mathbf{T}_{G_{\text{B}},G_0}. \tag{3.25}$$

Similarly, the pose of the centre of mass frame of Body 2 (the robot arm) is given by the multiplication of three elementary homogeneous transformation matrices and the second degree of freedom q_2 as

$$\begin{aligned}\mathbf{T}_{G_0,G_1} =&\mathbf{T}_{\text{disp}}(O_0O_{1x} - O_0G_{0x}, O_0O_{1y} - O_0G_{0y}, O_0O_{1z} - O_0G_{0z})\cdot\\ &\mathbf{T}_{\text{rotx}}(\frac{-\pi}{2})\cdot\mathbf{T}_{\text{rotz}}(q_2)\cdot\mathbf{T}_{\text{disp}}(O_1G_{1x}, O_1G_{1y}, O_1G_{1z}).\end{aligned} \tag{3.26}$$

As can be seen, joint frames are oriented such that rotation around a motion axis is always defined with a $\mathbf{T}_{\text{rotz}}$ matrix. Projection of $\mathbf{T}_{G_0,G_1}$ in base frame O_{Base} is expressed as

$$\mathbf{T}_{\text{Base},G_1} = \mathbf{T}_{\text{Body 2}} = \mathbf{T}_{\text{Base},G_{\text{B}}}\cdot\mathbf{T}_{G_{\text{B}},G_0}\cdot\mathbf{T}_{G_0,G_1}. \tag{3.27}$$

The pose of the other bodies $\mathbf{T}_{\text{Body i}}$, given by their centre of mass location, with respect to the base frame O_{Base} is derived in the same way. A total of nine homogeneous transformation matrices are computed similarly from $\mathbf{T}_{\text{Body 0}}$ to $\mathbf{T}_{\text{Body 8}}$.

From a multibody viewpoint, the kinematics of the Stäubli TX200 robot is therefore defined at the position level by locating all the centres of mass of the bodies. Velocity $\dot{\mathbf{x}}_{\text{i}}$ and acceleration $\ddot{\mathbf{x}}_{\text{i}}$ of the centre of mass of each Body i with respect to base frame O_{Base} are obtained by derivation of the position information. Same relations as the ones presented to derive the differential kinematics of the TCP (Eqs. 3.17 and 3.18) are used except that they are applied to each body such that

$$\mathbf{v}_{\text{i}} = \sum_{\text{j}=1}^{n}\mathbf{d}^{\text{i,j}}\cdot\dot{q}_{\text{j}}, \qquad \boldsymbol{\omega}_{\text{i}} = \sum_{\text{j}=1}^{n}\mathbf{w}^{\text{i,j}}\cdot\dot{q}_{\text{j}}, \tag{3.28}$$

in which index i is relative to the number of bodies n_{B} and index j is relative to the number of degrees of freedom n; $\mathbf{v}_{\mathrm{i}}$ (similar to previously defined $\dot{\mathbf{p}}_{\mathrm{i}}$) is the linear velocity of Body i and $\boldsymbol{\omega}_{\mathrm{i}}$ the rotational velocity of Body i; and $\mathbf{d}^{\mathrm{i,j}}$ and $\mathbf{w}^{\mathrm{i,j}}$ the partial contributions of $\dot{q}_{\mathrm{j}}$ in the translational and rotational velocities of Body i, respectively. It is like defining a Jacobian matrix for each Body i, still projected in the base frame, as

$$\begin{bmatrix} \{\mathbf{v}_{\mathrm{i}}\}_{\mathrm{Base}} \\ \{\boldsymbol{\omega}_{\mathrm{i}}\}_{\mathrm{Base}} \end{bmatrix} = \begin{bmatrix} \{\mathbf{d}^{\mathrm{i},1}\}_{\mathrm{Base}} \{\mathbf{d}^{\mathrm{i},2}\}_{\mathrm{Base}} \dots \{\mathbf{d}^{\mathrm{i},n}\}_{\mathrm{Base}} \\ \{\mathbf{w}^{\mathrm{i},1}\}_{\mathrm{Base}} \{\mathbf{w}^{\mathrm{i},2}\}_{\mathrm{Base}} \dots \{\mathbf{w}^{\mathrm{i},n}\}_{\mathrm{Base}} \end{bmatrix} \cdot \dot{\mathbf{q}} = [\mathbf{J}_{\mathrm{i}}(\mathbf{q})]_{\mathrm{Base}} \cdot \dot{\mathbf{q}}. \tag{3.29}$$

Acceleration $\ddot{\mathbf{x}}_{\mathrm{i}}$ of the centre of mass of each Body i is expressed in the same way as Eq. 3.18 as

$$\begin{bmatrix} \{\mathbf{a}_{\mathrm{i}}\}_{\mathrm{Base}} \\ \{\dot{\boldsymbol{\omega}}_{\mathrm{i}}\}_{\mathrm{Base}} \end{bmatrix} = [\mathbf{J}_{\mathrm{i}}]_{\mathrm{Base}} \cdot \ddot{\mathbf{q}} + [\dot{\mathbf{J}}_{\mathrm{i}}]_{\mathrm{Base}} \cdot \dot{\mathbf{q}}, \tag{3.30}$$

with $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ the first and second time derivatives of $\mathbf{q}$, respectively, $\mathbf{a}_{\mathrm{i}}$ the linear acceleration of body i, $\dot{\boldsymbol{\omega}}_{\mathrm{i}}$ the time derivative of the rotational velocity of body i and $\dot{\mathbf{J}}_{\mathrm{i}}$ being the derivative of the Jacobian matrix of Body i. The complete kinematics of the multibody system is consequently constructed.

As the minimal coordinates were selected, the two popular principles used to build the equations of motion are the Jourdain's and d'Alembert's principles. In this case, the d'Alembert's principle is applied and states that "*the total virtual work of the applied forces plus the inertial forces vanishes for reversible displacements*". The n equations of motion of a system owning n_{B} bodies and n degrees of freedom are built according to the principle such as

$$\sum_{\mathrm{i}=1}^{n_{\mathrm{B}}} \left[\mathbf{d}^{\mathrm{i,j}} \cdot (\boldsymbol{\mathcal{R}}_{\mathrm{i}} - m_{\mathrm{i}}\mathbf{a}_{\mathrm{i}}) + \mathbf{w}^{\mathrm{i,j}} \cdot (\boldsymbol{\mathcal{M}}_{\mathrm{G,i}} - \boldsymbol{\Phi}_{\mathrm{G,i}}\dot{\boldsymbol{\omega}}_{\mathrm{i}} - \boldsymbol{\omega}_{\mathrm{i}} \times \boldsymbol{\Phi}_{\mathrm{G,i}}\boldsymbol{\omega}_{\mathrm{i}})\right]_{\mathrm{Base}} = 0 \quad \mathrm{j} = 1, \dots, n, \tag{3.31}$$

in which $\boldsymbol{\mathcal{R}}_{\mathrm{i}}$ and $\boldsymbol{\mathcal{M}}_{\mathrm{G,i}}$ are the resultant force and moment, at the centre of mass G_{i}, of all applied forces exerted on Body i, respectively. All the involved quantities are supposed to be projected in base frame O_{Base}. Adopting the commonly lumped approach to characterise the inertia properties of rigid body i, its mass m_{i} and central inertia tensor $\boldsymbol{\Phi}_{\mathrm{G,i}}$ are clustered at its centre of mass frame (Figure 3.12).

Tensor $\boldsymbol{\Phi}_{\mathrm{G,i}}$ is symmetric and constant when projected in the centre of mass frame of rigid body i. Its six different components are explicitly exposed in a matrix form as

$$\boldsymbol{\Phi}_{\mathrm{G,i}} = \begin{bmatrix} \phi_{\mathrm{G},xx_{\mathrm{i}}} & \phi_{\mathrm{G},xy_{\mathrm{i}}} & \phi_{\mathrm{G},xz_{\mathrm{i}}} \\ \phi_{\mathrm{G},yx_{\mathrm{i}}} & \phi_{\mathrm{G},yy_{\mathrm{i}}} & \phi_{\mathrm{G},yz_{\mathrm{i}}} \\ \phi_{\mathrm{G},zx_{\mathrm{i}}} & \phi_{\mathrm{G},zy_{\mathrm{i}}} & \phi_{\mathrm{G},zz_{\mathrm{i}}} \end{bmatrix}. \tag{3.32}$$

The n equations of motion are therefore built in a systematic fashion using the d'Alembert's principle. Their resolution, also depending on the choice of coordinates, is

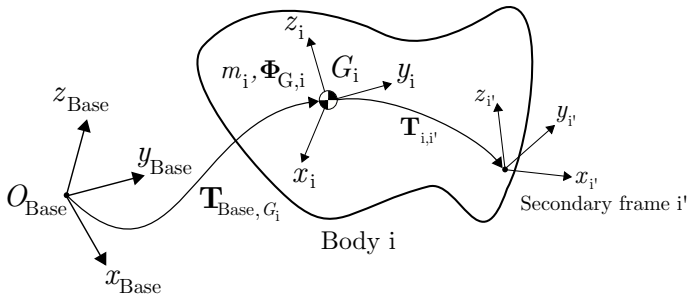


Figure 3.12: Situation of the centre of mass frame of rigid body i with respect to base frame O_{Base}

addressed in Chapter 6 concerning the validation of the robotic machining simulator.

Note as well that a rigid body can also gather so-called secondary frame(s) as illustrated in Figure 3.12, rigidly attached to Body i through a constant homogeneous transformation matrix $\mathbf{T}_{\mathrm{i,i'}}$. For instance, TCP frame O_{TCP} can be seen as a secondary frame relative to centre of mass G_{Spindle}. It is indeed interesting to have access to the position, velocity and information of the task frame during the simulation.

3.3.2 Manipulator with rigid joints

The main objective of this section is the derivation of the dynamic model of the Stäubli TX200 robot. As for now, the model is bounded to the consideration of rigid joints and rigid links and its vector of degrees of freedom $\mathbf{q}$ represents the motion around its six rotation axes, n=6. Hence, in this model $\mathbf{q} = \mathbf{q}_{\mathrm{s}}$ represents the vector of instantaneous positions of the motor shafts seen from the joint side. The rotor of the servo motors is now included in the modelling.

Figure 3.13 illustrates the considered dynamic model for the Stäubli TX200 robot. Each rotor is located at joint frame O_{i} as defined in Figure 3.11a, i = 0, ..., 5. At the interface between the rotor output shaft and the rigid link, i.e. coincident to frame O_{i}, is situated a rigid gearbox with reduction ratio k_{i}. The position of the rotor, relatively to the defined degrees of freedom, is given by $\mathbf{q}_{\mathrm{m}} = \mathbf{k}\,\mathbf{q}$ with diagonal matrix $\mathbf{k}$ containing the gear ratio of each gearbox as

$$\mathbf{k}_{\mathrm{i}} = \begin{bmatrix} k_1 & & 0 \\ & \ddots & \\ 0 & & k_6 \end{bmatrix}. \tag{3.33}$$

Regarding the multibody modelling, the inclusion of rotors is handled by simply considering each rotor as an additional body with a mass $m_{\mathrm{m_i}}$ and a central tensor of inertia $\mathbf{\Phi}_{\mathrm{G,m_i}}$ considered to be diagonal $\mathbf{\Phi}_{\mathrm{G,m_i}}$=diag($\phi_{\mathrm{m},xx_{\mathrm{i}}}$, $\phi_{\mathrm{m},yy_{\mathrm{i}}}$, $\phi_{\mathrm{m},zz_{\mathrm{i}}}$). As a result, the number of bodies becomes n_{B}=15. The incorporation of the rotors is carried out easily by deriving the expressions of the homogeneous transformation matrices reflecting their po-

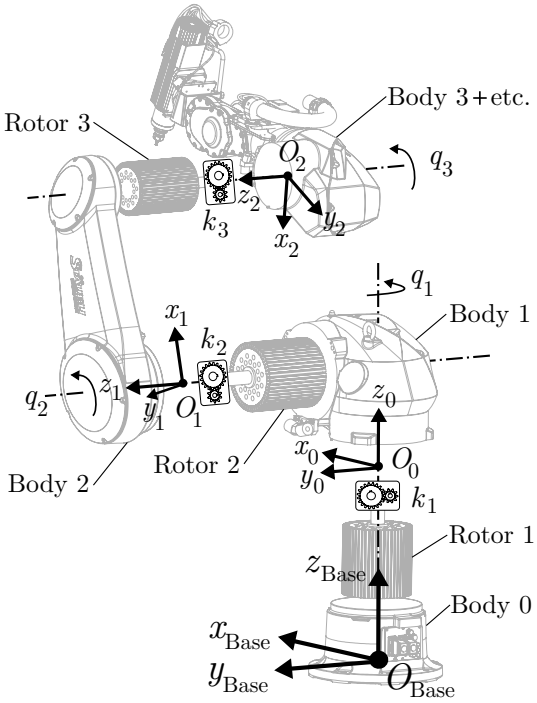


Figure 3.13: Dynamic model of the Stäubli TX200 robot with rigid joints and rigid links

sition. For instance, the homogeneous transformation matrix related to the position of the first rotor with respect to base frame O_{Base} is given by

$$\begin{aligned}\mathbf{T}_{\text{Base,Rotor}_1} =& \mathbf{T}_{\text{Body 0}} \cdot \mathbf{T}_{\text{disp}}(O_{\text{Base}}O_{0x} - O_{\text{Base}}G_{\text{B}x}, O_{\text{Base}}O_{0y} - O_{\text{Base}}G_{\text{B}y}, \\ & O_{\text{Base}}O_{0z} - O_{\text{Base}}G_{\text{B}z}) \cdot \mathbf{T}_{\text{rotz}}(k_1 q_1).\end{aligned} \tag{3.34}$$

The pose of the other rotors is derived in a similar way relatively to base frame O_{Base}. It leads to the definition of the pose of six additional bodies from $\mathbf{T}_{\text{Base,Rotor}_1}$ to $\mathbf{T}_{\text{Base,Rotor}_6}$ whose kinematics is computed using Eqs. 3.28 to 3.31. The d'Alembert's principle used to build the equations of motion is still applicable.

Equivalent to the equations of motion derived in Eq. 3.31 in the base frame, their general form is often expressed in a matrix form referring to the so-called *joint space dynamic model* [3] with

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_{\text{v}}\dot{\mathbf{q}} + \mathbf{F}_{\text{s}}\text{sgn}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \boldsymbol{\tau}_{\text{ext}}, \tag{3.35}$$

given from the joint side. Without the consideration of any flexibility, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{6\times6}$, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{6\times6}$ and $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^{6\times1}$ denote the system mass, damping and gravity matrices, respectively. Diagonal matrices $\mathbf{F}_{\text{v}}$ and $\mathbf{F}_{\text{s}} \in \mathbb{R}^{6\times6}$ account for the viscous and static frictions mainly originating from the motors and the transmissions. On the right-hand side, vector $\boldsymbol{\tau} \in \mathbb{R}^{6\times1}$ gathers the motor torques computed from the joint side and vector $\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^{6\times1}$ encloses the external torques applied to the joints. External torque $\boldsymbol{\tau}_{\text{ext}}$ comprises the milling forces $\mathbf{F}_{\text{c}}$ and the torque from the gravity compensation system $\boldsymbol{\tau}_{\text{g}}$.

The mass matrix encompasses the contribution of several effects that can be exposed as follows

$$\mathbf{M}(\mathbf{q}) = \mathbf{M}_{\mathrm{L}}(\mathbf{q}) + \mathbf{M}_{\mathrm{R}}(\mathbf{q}) + \mathbf{S}(\mathbf{q}) + \begin{bmatrix} k_1^2\phi_{\mathrm{m},zz_1} & & 0 \\ & \ddots & \\ 0 & & k_6^2\phi_{\mathrm{m},zz_6} \end{bmatrix}, \tag{3.36}$$

with $\mathbf{M}_{\mathrm{L}}(\mathbf{q})$ the link inertia matrix, $\mathbf{M}_{\mathrm{R}}(\mathbf{q})$ the rotor inertia matrix and $\mathbf{S}(\mathbf{q})$ the inertial coupling matrix. While the meaning of the first two matrices is evident, $\mathbf{S}(\mathbf{q})$ expresses the inertial couplings between the rotors and the previous links in the robot chain and comprises terms in $k_{\mathrm{i}}\phi_{\mathrm{m},zz_{\mathrm{i}}}$ [120]. Contributions in $\mathbf{S}(\mathbf{q})$ can be neglected in case of high reduction ratios but are kept here for the sake of completeness. Note that contributions that appear in $k_{\mathrm{i}}^2\phi_{\mathrm{m},zz_{\mathrm{i}}}$ represent the most significant part of the complete mass matrix $\mathbf{M}(\mathbf{q})$. From the formulation of the Lagrange equations, the computation of the kinetic energy leads to the expression of mass matrix $\mathbf{M}(\mathbf{q})$. Each of its components $m_{\mathrm{j,k}}$ is computed from the inertia properties and the partial contributions of $\dot{q}$ in the velocity of Body i such as

$$\begin{aligned} m_{\mathrm{j,k}} = \sum_{\mathrm{i}=1}^{n_{\mathrm{B}}} \big[& m_{\mathrm{i}} \left(\{\mathbf{d}^{\mathrm{i,j}}\}_{\mathrm{Base}} \cdot \{\mathbf{d}^{\mathrm{i,k}}\}_{\mathrm{Base}} \right) + \\ & \{\mathbf{w}^{\mathrm{i,j}}\}_{\mathrm{Base}} \cdot \left(\mathbf{R}_{\mathrm{Base,i}} \mathbf{\Phi}_{\mathrm{G,i}} \mathbf{R}_{\mathrm{Base,i}}^{T} \{\mathbf{w}^{\mathrm{i,k}}\}_{\mathrm{Base}} \right) \big], \end{aligned} \tag{3.37}$$

with subscripts j and k relative to the number of degrees of freedom n and all the quantities are projected in base frame O_{Base}.

Since the choice of matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is not unique, a particular choice leads to the computation of the so-called *Christoffel symbols* of the first type. For a manipulator comprising n degrees of freedom, the elements of matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ projected in the base frame are computed as follows

$$c_{\mathrm{j,k}} = \sum_{\mathrm{l}=1}^{n} c_{\mathrm{j,k,l}}\, \dot{q}_{\mathrm{l}}, \tag{3.38}$$

where coefficients $c_{\mathrm{j,k,l}}$ are the Christoffel symbols of the first type computed by partial derivation of the mass matrix elements $m_{\mathrm{j,k}}$ as

$$c_{\mathrm{j,k,l}} = \frac{1}{2} \left(\frac{\partial m_{\mathrm{j,k}}}{\partial q_{\mathrm{l}}} + \frac{\partial m_{\mathrm{j,l}}}{\partial q_{\mathrm{k}}} - \frac{\partial m_{\mathrm{k,l}}}{\partial q_{\mathrm{j}}} \right). \tag{3.39}$$

The *Christoffel* matrix combines the effects of Coriolis, centrifugal and gyroscopic forces.

Finally, if the links of the manipulator are assumed to be rigid, the potential energy is only due to gravitational forces and gravity vector $\mathbf{g}(\mathbf{q})$ is given as the sum of the contributions for all the bodies for the multibody system such as

$$g_{\mathrm{j}} = -\sum_{\mathrm{i}=1}^{n_{\mathrm{B}}} m_{\mathrm{i}} \left(\{\mathbf{g}_0\}_{\mathrm{Base}} \cdot \{\mathbf{d}^{\mathrm{i,j}}\}_{\mathrm{Base}} \right), \tag{3.40}$$

with subscript j relative to the number of degrees of freedom n and $\{\mathbf{g}_0\}_{\mathrm{Base}}$ being the gravity vector $[0\ 0\ g_z]^T_{\mathrm{Base}}$ in the base frame if Z is the vertical axis (g_z = -9.81 m/s^2 if Earth's attraction is assumed).

3.3.3 Manipulator with joint axial flexibility

As pointed out in the literature [53], the main compliance originates from the robot joints so that it is necessary to include the joint flexibilities in the dynamic model. A joint flexibility is first accounted only around each of the motion axes of the robot. As the Stäubli TX200 robot comprises six motion axes n_θ=6, six flexible elements are appended to the dynamic model developed in the previous subsection. In other words, all the rigid joints are replaced by flexible elements. The popular model consists in representing such flexibility using torsional springs and viscous dampers. Elasticity at joint i is modelled by a spring of torsional stiffness $k_{z,\mathrm{i}}$ >0 and by a damper of torsional damping $d_{z,\mathrm{i}}$ >0 (subscript z is adopted as all motion axes are oriented along the Z direction). Joint stiffness and damping are assumed to be constant. Resulting deflection at joint i is represented with an additional degree of freedom $q_{z,\mathrm{i}}$ relative to the angles of motor shaft contained in $\mathbf{q}_{\mathrm{s}}$ seen from the joint side. Hence, the dynamic robot model accounting for the flexibility around all the motion axes includes a total of twelve degrees of freedom n=12: q_1 to q_6 for the rotor motions from the joint side and $q_{z,1}$ to $q_{z,6}$ for the joint axial flexibility. Such a model is illustrated in Figure 3.14 for the third joint of the Stäubli TX200 robot. The joint axial flexibility is inserted just after the rigid gearbox, simply modelled with reduction ratio k_3. The desired motor angle is denoted by $k_3\theta_3$. Since the rotor is under control, variable q_3 accounts for the current position of the motor output shaft after reduction. If a perfect controller is assumed, $q_3 = \theta_3$. Variable $q_{z,3}$ accounts for any deflection coming from the torsional spring $k_{z,3}$ while $\dot{q}_{z,3}$ is affected by joint damping $d_{z,3}$.

Adoption of such joint model implies the three standard assumptions [120]:

1. Joint deflections are small so that flexibility effects are limited to the domain of linear elasticity.
2. The rotors are modelled as uniform bodies having their centre of mass on the rotation axis.
3. Each motor is located on the robot arm in a position preceding the driven link.

From a modelling viewpoint, the multibody model derived from Eq. 3.34 and including the rotor effects needs to be modified to consider the joint flexibility around the motion axes. Referring to Figure 3.14, the homogeneous transformation matrix giving the position of the third rotor $\mathbf{T}_{\mathrm{Base,Rotor_3}}$ with respect to base frame O_{Base} reads

$$\begin{aligned}\mathbf{T}_{\mathrm{Base,Rotor_3}} =& \mathbf{T}_{\mathrm{Body\ 2}} \cdot \mathbf{T}_{\mathrm{disp}}(O_1O_{2x} - O_1G_{1x}, O_1O_{2y} - O_1G_{1y}, O_1O_{2z} - O_1G_{1z}) \cdot \\ & \mathbf{T}_{\mathrm{rotz}}(k_3 q_3).\end{aligned} \tag{3.41}$$

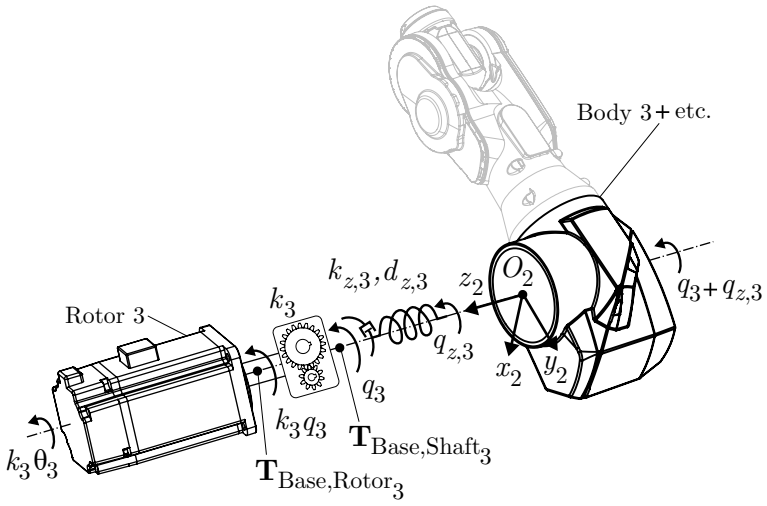


Figure 3.14: Joint axial flexibility model

Its expression remains unchanged compared with Eq. 3.34. In order to include the joint axial flexibility, it is decided to decompose the position of the centre of mass of Body 3 (robot elbow) into two homogeneous transformation matrices. The first homogeneous transformation matrix $\mathbf{T}_{\text{Base,Shaft}_3}$ is a frame, without inertia properties, located after the rigid gearbox (Figure 3.14). It can be interpreted as the position of the output motor shaft after gear reduction and can be expressed as

$$\begin{aligned}\mathbf{T}_{\text{Base,Shaft}_3} =& \mathbf{T}_{\text{Body 2}} \cdot \mathbf{T}_{\text{disp}}(O_1O_{2x} - O_1G_{1x}, O_1O_{2y} - O_1G_{1y}, O_1O_{2z} - O_1G_{1z}) \cdot \\ & \mathbf{T}_{\text{rotz}}(q_3).\end{aligned} \tag{3.42}$$

Then, the position of the centre of mass of Body 3 is given with the introduction of degree of freedom $q_{z,3}$ dealing with the joint axial flexibility such as

$$\mathbf{T}_{\text{Body 3}} = \mathbf{T}_{\text{Base,Shaft}_3} \cdot \mathbf{T}_{\text{rotz}}(q_{z,3}) \cdot \mathbf{T}_{\text{disp}}(O_2G_{2x}, O_2G_{2y}, O_2G_{2z}). \tag{3.43}$$

The complete kinematics of the multibody model can be derived as before using Eqs. 3.28 to 3.30 and equations of motion are built using the d'Alembert's principle (Eq. 3.31). Under the assumption of a perfect controller $\mathbf{\Theta} = \mathbf{q}_{\text{s}} = [q_1 \dots q_6]^T$ i.e. controller stiffness and damping are infinite, resultant torque $\{\boldsymbol{\mathcal{M}}_{\text{G,i}}\}_{\text{Base}}$ applied at the centre of mass of Body i and projected in base frame O_{Base}, only arising from the joint deflections, reads

$$\{\boldsymbol{\mathcal{M}}_{\text{G,i}}\}_{\text{Base}} = \left(k_{z,\text{j}} \cdot (0 - q_{z,\text{j}}) + d_{z,\text{j}} \cdot (0 - \dot{q}_{z,\text{j}})\right) \mathbf{R}_{\text{Base,i}} \mathbf{u}_z, \tag{3.44}$$

with subscript j indicating the number of the robot joint to which the flexibility is associated, $\mathbf{R}_{\text{Base,i}}$ rotation matrix of Body i projected in the base frame and unit vector $\mathbf{u}_z$ defined as $\mathbf{u}_z = [0\,0\,1]^T$. The reaction torque is of course accounted on Body i-1 having the same amplitude but the opposite sign.

For better readability, the derived equations of motion can be expressed in a matrix form as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} + \mathbf{F}_{\mathrm{v}}\dot{\mathbf{q}} + \mathbf{F}_{\mathrm{s}}\mathrm{sgn}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\Gamma}, \tag{3.45}$$

given from the joint side with $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{12\times12}$, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) \in \mathbb{R}^{12\times12}$, $\mathbf{K} \in \mathbb{R}^{12\times12}$ and $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^{12\times1}$ which denote the system mass, damping, stiffness and gravity matrices, respectively. Diagonal matrices $\mathbf{F}_{\mathrm{v}}$ and $\mathbf{F}_{\mathrm{s}} \in \mathbb{R}^{12\times12}$ account for the viscous and static frictions originating from the motors and the transmissions. On the right-hand side, vector $\boldsymbol{\Gamma} \in \mathbb{R}^{12\times1}$ gathers the combination of the actuation and external torques $\boldsymbol{\Gamma} = [\boldsymbol{\tau}\ \boldsymbol{\tau}_{\mathrm{ext}}]^T$. Vector of degrees of freedom $\mathbf{q}$ now comprises the ones accounting for the rotor rotations from the joint side $\mathbf{q}_{\mathrm{s}} \in \mathbb{R}^{6\times1}$ and the ones relative to the spring deflections $\mathbf{q}_z \in \mathbb{R}^{6\times1}$ as $\mathbf{q} = [\mathbf{q}_{\mathrm{s}}\ \mathbf{q}_z]^T = [q_1 \dots q_6\ q_{z,1} \dots q_{z,6}]^T$.

The different terms contributing to the mass matrix can again be exposed such as

$$\mathbf{M}(\mathbf{q}) = \mathbf{M}_{\mathrm{L}}(\mathbf{q}) + \mathbf{M}_{\mathrm{R}}(\mathbf{q}) + \mathbf{S}(\mathbf{q}) + \begin{bmatrix} k_1^2\phi_{\mathrm{m},zz_1} & & & & & 0 \\ & \ddots & & & & \\ & & k_6^2\phi_{\mathrm{m},zz_6} & & & \\ & & & 0 & & \\ & & & & \ddots & \\ 0 & & & & & 0 \end{bmatrix}, \tag{3.46}$$

with $\mathbf{M}_{\mathrm{L}}(\mathbf{q})$ the link inertia matrix, $\mathbf{M}_{\mathrm{R}}(\mathbf{q})$ the rotor inertia matrix and $\mathbf{S}(\mathbf{q})$ the inertial coupling matrix pertaining to $\mathbb{R}^{12\times12}$. Rotor inertias multiplied by the square of reduction ratio only affect the diagonal of the first six rows of the last matrix.

Equation 3.45 can be developed as

$$\begin{aligned} \begin{bmatrix} \boldsymbol{\tau} \\ \boldsymbol{\tau}_{\mathrm{ext}} \end{bmatrix} =& \mathbf{M}(\mathbf{q}) \begin{Bmatrix} \ddot{\mathbf{q}}_{\mathrm{s}} \\ \ddot{\mathbf{q}}_z \end{Bmatrix} + \underbrace{\mathbf{C}_{\mathrm{Christoffel}}(\mathbf{q},\dot{\mathbf{q}}) + \begin{bmatrix} \mathbf{K}_{\mathrm{D}}\left(\dot{\boldsymbol{\Theta}} - \dot{\mathbf{q}}_{\mathrm{s}}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{d}_z\left(\dot{\mathbf{q}}_{\mathrm{s}} - \dot{\mathbf{q}}_z\right) \end{bmatrix}}_{\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}} + \\ & \underbrace{\begin{bmatrix} \mathbf{K}_{\mathrm{P}}\left(\boldsymbol{\Theta} - \mathbf{q}_{\mathrm{s}}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{k}_z\left(\mathbf{q}_{\mathrm{s}} - \mathbf{q}_z\right) \end{bmatrix}}_{\mathbf{K}\mathbf{q}} + \mathbf{F}_{\mathrm{v}}\dot{\mathbf{q}} + \mathbf{F}_{\mathrm{s}}\mathrm{sgn}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}), \end{aligned} \tag{3.47}$$

with $\mathbf{C}_{\mathrm{Christoffel}}(\mathbf{q},\dot{\mathbf{q}}) \in \mathbb{R}^{12\times12}$ the Christoffel matrix accounting for the Coriolis, centrifugal and gyroscopic effects of the manipulator comprising 12 degrees of freedom. Global damping matrix $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ is composed of two contributions: the Christoffel matrix $\mathbf{C}_{\mathrm{Christoffel}}$ and the virtual damping matrix $\mathbf{D}$. Virtual damping matrix $\mathbf{D} \in \mathbb{R}^{12\times12}$ is a diagonal matrix whose first six rows account for the controller damping $\mathbf{K}_{\mathrm{D}}$ while the last rows are related to the damping of the joint damper around the motion axes $\mathbf{d}_z$. Developed expression of the virtual damping matrix reads

$$\mathbf{D} = \begin{bmatrix} \mathbf{K}_{\mathrm{D}} \left(\dot{\boldsymbol{\Theta}} - \dot{\mathbf{q}}_{\mathrm{s}} \right) & \mathbf{0} \\ \mathbf{0} & \mathbf{d}_z \left(\dot{\mathbf{q}}_{\mathrm{s}} - \dot{\mathbf{q}}_z \right) \end{bmatrix}, \tag{3.48}$$

with $\mathbf{K}_{\mathrm{D}} \in \mathbb{R}^{6\times6}$ enclosing the controller damping as $\mathbf{K}_{\mathrm{D}} = \mathrm{diag}[K_{\mathrm{D},1} \dots K_{\mathrm{D},6}]$ and matrix $\mathbf{d}_z \in \mathbb{R}^{6\times6}$ comprising the joint damping $\mathbf{d}_z = \mathrm{diag}[d_{z,1} \dots d_{z,6}]$. Similarly, stiffness matrix $\mathbf{K} \in \mathbb{R}^{12\times12}$ is a diagonal matrix whose first six rows account for the controller stiffness $\mathbf{K}_{\mathrm{P}}$ while the last rows are related to the stiffness of the joint spring around the motion axes $\mathbf{k}_z$. Developed expression of the stiffness matrix reads

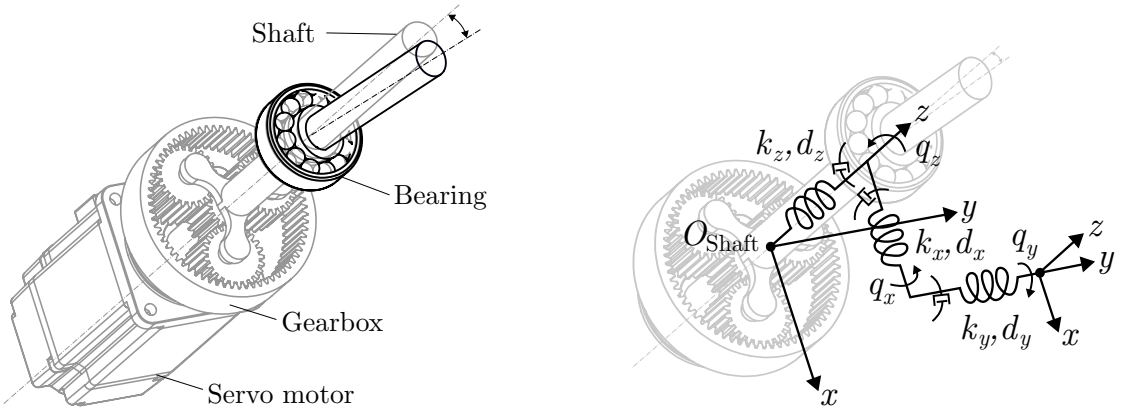
$$\mathbf{Kq} = \begin{bmatrix} \mathbf{K}_{\mathrm{P}} \left(\boldsymbol{\Theta} - \mathbf{q}_{\mathrm{s}} \right) & \mathbf{0} \\ \mathbf{0} & \mathbf{k}_z \left(\mathbf{q}_{\mathrm{s}} - \mathbf{q}_z \right) \end{bmatrix}, \tag{3.49}$$

with $\mathbf{K}_{\mathrm{P}} \in \mathbb{R}^{6\times6}$ enclosing the controller stiffness as $\mathbf{K}_{\mathrm{P}} = \mathrm{diag}[K_{\mathrm{P},1} \dots K_{\mathrm{P},6}]$ and matrix $\mathbf{k}_z \in \mathbb{R}^{6\times6}$ comprising the joint stiffness $\mathbf{k}_z = \mathrm{diag}[k_{z,1} \dots k_{z,6}]$. The presented controller action is provided in a caricatural manner as it can be perceived as a regulated flexibility around the motor axes. Further information regarding the controller implementation is covered in Section 3.8.

3.3.4 Manipulator with joint tri-axial flexibility

Ideally, joint and drive flexibilities should be modelled with flexible elements able to deform in all directions. In Chapter 2, such a model was designated as the virtual joint modelling (VJM) method and illustrated in Figure 2.41 [112]. All flexible elements gathered six degrees of freedom either at the drive or at the joint. However, it would imply the identification of 12 stiffness coefficients and 12 damping coefficients per joint which seems unaffordable. A reduced model to account of such flexibility, which is commonly accepted in the literature [62, 123, 124], consists in collecting all the flexibilities at the joint location with three orthogonal and torsional springs and dampers. The reduced model can be seen as an extension of the joint model with an axial flexibility, where two torsional springs and dampers are orthogonally appended. Henceforth, it is called the *joint tri-axial flexibility* model.

Concentrating all the flexibilities at the joint with torsional elements implies the following assumption. In Figure 3.15a is represented the driving chain with the joint bearing from which most of the perpendicular deflections come from. In fact, since it is a revolving part, the flexibility perpendicular to the motion axis should be equally distributed all around the circumference of the bearing. Whatever the orientation of the motor shaft, the seen flexibility should be the same. Therefore, parallel orthogonal springs and dampers would represent such behaviour. However, the kinematic formulation with homogeneous transformation matrices involves the definition of successions of elementary motions. As a result, the orthogonal flexibilities, also termed as *virtual flexibilities*, are modelled using torsional springs and dampers in series (Figure 3.15b). Deflections from the virtual flexibilities for all the joints are modelled using vector of degrees of freedom



(a) Amplified view of bearing deflections (b) Torsional springs in series

Figure 3.15: Model of the joint bearing

$\mathbf{q}_{x|y} \in \mathbb{R}^{12\times 1}$. Virtual joint stiffness and damping around the X axis of joint i are denoted by $k_{x,\text{i}}$ and $d_{x,\text{i}}$ and virtual joint stiffness and damping around the Y axis of joint i are referenced by $k_{y,\text{i}}$ and $d_{y,\text{i}}$. Such a representation is generally a good approximation of the real behaviour as long as the flexibility properties are close to each other in the orthogonal directions.

The multibody modelling with joint axial flexibilities presented in Subsection 3.3.3 is continued by considering the orthogonal deflections to the motion axes. The only modification is made to homogeneous transformation matrices locating the centres of mass of the links. As an illustrative example, derived homogeneous transformation matrix $\mathbf{T}_{\text{Body 3}}$ (Eq. 3.43) is expanded in order to include the virtual deflections such as

$$\begin{aligned}\mathbf{T}_{\text{Body 3}} =& \mathbf{T}_{\text{Base,Shaft}_3} \cdot \mathbf{T}_{\text{rotz}}(q_{z,3}) \cdot \mathbf{T}_{\text{rotx}}(q_{x,3}) \cdot \mathbf{T}_{\text{roty}}(q_{y,3}) \cdot \\ & \mathbf{T}_{\text{disp}}(O_2G_{2x}, O_2G_{2y}, O_2G_{2z}).\end{aligned} \tag{3.50}$$

Note that the order of virtual deflections is selected as a first rotation around the X axis followed by a second rotation around the Y axis. Of course, the modification is carried out for all the robot joints to derive the new kinematics of the multibody model.

Tri-axial deflections not only arise at the joint level but can also appear at the connecting point between the steel slab and the robot base. The latter are modelled using virtual springs and dampers (Figure 3.16). Intuitively, it is clear that the stiffness along directions X and Y and around the Z direction are high since the robot is clamped with M24 screws. Consequently, virtual deflections can be limited to the rotations around the X and Y directions and the translational motion along the Z direction. Allotted degrees of freedom to represent the virtual flexibilities between the robot base and the steel slab are enclosed in vector $\mathbf{q}_{\text{ground}} \in \mathbb{R}^{3\times 1}$. Stiffness and damping properties are respectively denoted by diagonal matrices $\mathbf{k}_{\text{g}}$ and $\mathbf{d}_{\text{g}} \in \mathbb{R}^{3\times 3}$.

The introduction of the ground virtual flexibilities modifies the expression of homogeneous

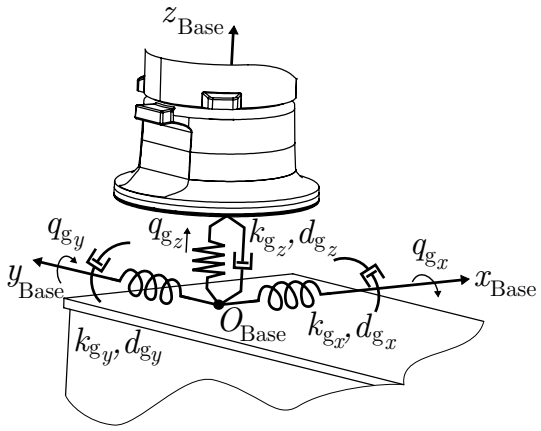


Figure 3.16: Modelled compliance between the robot base and the steel slab

transformation matrix $\mathbf{T}_{\text{Base},G_\text{B}}$ expressing the pose of the centre of mass frame of the robot base with respect to base frame O_{Base} such as

$$\begin{aligned}\mathbf{T}_{\text{Base},G_\text{B}} =& \mathbf{T}_{\text{rotx}}(q_{g_x}) \cdot \mathbf{T}_{\text{roty}}(q_{g_y}) \cdot \mathbf{T}_{\text{disp}}(0,0,q_{g_z}) \cdot \\ & \mathbf{T}_{\text{disp}}(O_{\text{Base}}G_{\text{B}x}, O_{\text{Base}}G_{\text{B}y}, O_{\text{Base}}G_{\text{B}z}).\end{aligned} \tag{3.51}$$

Having completely defined the kinematics with all the virtual flexibilities, the resulting multibody model of the Stäubli TX200 robot gathers a total of 27 degrees of freedom n=27 with $\mathbf{q} \in \mathbb{R}^{27\times 1}$ ($\mathbf{q}_\text{s} \in \mathbb{R}^{6\times 1}$ instantaneous rotor position from the joint side, $\mathbf{q}_z \in \mathbb{R}^{6\times 1}$ instantaneous joint deflection around the Z direction of the joint frame, $\mathbf{q}_{x|y} \in \mathbb{R}^{12\times 1}$ instantaneous joint virtual deflections around the X and Y directions of the joint frame and $\mathbf{q}_{\text{ground}} \in \mathbb{R}^{3\times 1}$ instantaneous virtual deflections around the X and Y directions and along the Z direction of the frame between the base frame and the robot base). From the position of all the bodies, their velocity and acceleration are still expressed using Eqs. 3.28 to 3.30 and the application of the d'Alembert's principle (Eq. 3.31) leads to the equations of motion. It should be pointed out that since the motion of the links is no longer restricted to a rotation around the motion axes, the complete tensor of inertia $\mathbf{\Phi}_{\text{G,i}}$ is required for all the bodies. Translational force $\{\boldsymbol{\mathcal{R}}_\text{i}\}_{\text{Base}}$ and moment $\{\boldsymbol{\mathcal{M}}_{\text{G,i}}\}_{\text{Base}}$ projected in base frame O_{Base} yielded by the virtual flexibilities and applied at the centre of mass of Body i are respectively expressed by

$$\{\boldsymbol{\mathcal{R}}_\text{i}\}_{\text{Base}} = \left(k_{g_z} \cdot (0 - q_{g_z}) + d_{g_z} \cdot (0 - \dot{q}_{g_z})\right) \mathbf{R}_{\text{Base,i}} \mathbf{u}_z, \tag{3.52}$$

and

$$\{\boldsymbol{\mathcal{M}}_{\text{G,i}}\}_{\text{Base}} = \left(k_{x|y} \cdot (0 - q_{x|y}) + d_{x|y} \cdot (0 - \dot{q}_{x|y})\right) \mathbf{R}_{\text{Base,i}} \mathbf{u}_{x|y}, \tag{3.53}$$

with unit vectors $\mathbf{u}_x$ and $\mathbf{u}_y$ defined as $\mathbf{u}_x = [1\,0\,0]^T$ and $\mathbf{u}_y = [0\,1\,0]^T$, respectively. The reaction torques are also accounted on Body i-1 having the same amplitudes but the opposite signs.

Lastly, the resulting equations of motion can be expressed in a matrix form as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} + \mathbf{F}_{\mathrm{v}}\dot{\mathbf{q}} + \mathbf{F}_{\mathrm{s}}\mathrm{sgn}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\Gamma}, \tag{3.54}$$

given from the joint side with $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{27\times27}$, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) \in \mathbb{R}^{27\times27}$, $\mathbf{K} \in \mathbb{R}^{27\times27}$ and $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^{27\times1}$ which denote the system mass, damping, stiffness and gravity matrices, respectively. Diagonal matrices $\mathbf{F}_{\mathrm{v}}$ and $\mathbf{F}_{\mathrm{s}} \in \mathbb{R}^{27\times27}$ account for the viscous and static frictions originating from the motors and the transmissions. On the right-hand side, vector $\boldsymbol{\Gamma} \in \mathbb{R}^{27\times1}$ gathers the combination of the actuation and external torques $\boldsymbol{\Gamma} = [\boldsymbol{\tau}\ \boldsymbol{\tau}_{\mathrm{ext}}]^T$. Vector of degrees of freedom $\mathbf{q} \in \mathbb{R}^{27\times1}$ is partitioned such that $\mathbf{q} = [\mathbf{q}_{\mathrm{s}}\ \mathbf{q}_z\ \mathbf{q}_{x|y}\ \mathbf{q}_{\mathrm{ground}}]^T = [q_1 \dots q_6\ q_{z,1} \dots q_{z,6}\ q_{x,1}q_{y,1} \dots q_{x,6}q_{y,6}\ q_{\mathrm{g}_x}q_{\mathrm{g}_y}q_{\mathrm{g}_z}]^T$. While matrix $\mathbf{M}(\mathbf{q})$ can still be decomposed as presented in Eq. 3.46, the partitioning of global damping matrix $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ and stiffness matrix $\mathbf{K}$ is given.

The general form of the global damping matrix $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ comes from the summation of the Christoffel matrix $\mathbf{C}_{\mathrm{Christoffel}}$ and the virtual damping matrix $\mathbf{D}$ such as

$$\begin{aligned}\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} =& \mathbf{C}_{\mathrm{Christoffel}}(\mathbf{q},\dot{\mathbf{q}}) + \\ &\underbrace{\begin{bmatrix} \mathbf{K}_{\mathrm{D}}\left(\dot{\boldsymbol{\Theta}} - \dot{\mathbf{q}}_{\mathrm{s}}\right) & & & \mathbf{0} \\ & \mathbf{d}_z\left(\dot{\mathbf{q}}_{\mathrm{s}} - \dot{\mathbf{q}}_z\right) & & \\ & & \mathbf{d}_{x|y}\left(\mathbf{0} - \dot{\mathbf{q}}_{x|y}\right) & \\ \mathbf{0} & & & \mathbf{d}_{\mathrm{g}}\left(\mathbf{0} - \dot{\mathbf{q}}_{\mathrm{ground}}\right) \end{bmatrix}}_{\mathbf{D}},\end{aligned} \tag{3.55}$$

and the global stiffness matrix $\mathbf{K}$ is expressed in a similar way as

$$\mathbf{K}\mathbf{q} = \begin{bmatrix} \mathbf{K}_{\mathrm{P}}\left(\boldsymbol{\Theta} - \mathbf{q}_{\mathrm{s}}\right) & & & \mathbf{0} \\ & \mathbf{k}_z\left(\mathbf{q}_{\mathrm{s}} - \mathbf{q}_z\right) & & \\ & & \mathbf{k}_{x|y}\left(\mathbf{0} - \mathbf{q}_{x|y}\right) & \\ \mathbf{0} & & & \mathbf{k}_{\mathrm{g}}\left(\mathbf{0} - \mathbf{q}_{\mathrm{ground}}\right) \end{bmatrix}. \tag{3.56}$$

3.4 Flexible body modelling

In order to deal with the link flexibility of the robot, a new approach was developed regarding the modelling of flexible bodies using the minimal coordinates [125]. Section 3.3 presented the derivation of the dynamic model of the Stäubli robot constituted of rigid links. Using rigid bodies, it was stated that their motion was naturally described through the one of a coordinate system attached to it i.e. their centre of mass. When considering the motion of flexible bodies, it is therefore intuitively understood that its motion should be defined by the one of its nodes. The situation of the rigid body presented in Figure 3.12 with respect to base frame O_{Base} is transposed to the consideration of a flexible body in Figure 3.17. For the sake of illustration, two nodes are depicted and their respective position is still provided by a homogeneous transformation matrix with respect to base

frame O_{Base}. Homogeneous transformation matrix $\mathbf{T}_{\text{Base,f.g}}$ gives the situation of node g belonging to flexible body f with respect to base frame O_{Base}.

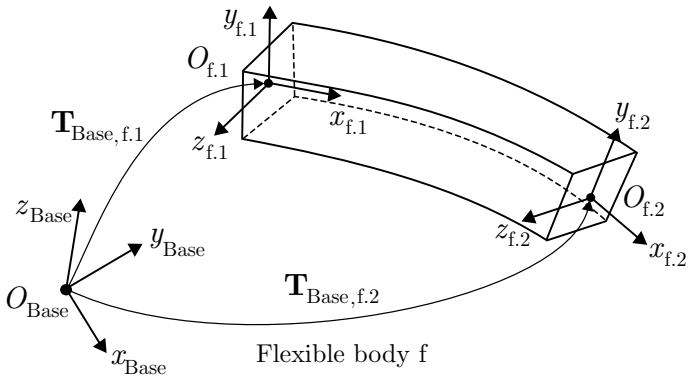


Figure 3.17: Kinematics of the nodes of flexible body f with respect to base frame O_{Base}

Generally, two main approaches are found in the literature to express the motion of a flexible body:

1. The *full finite element* approach: the motion of a flexible body is expressed directly from the translational and rotational motion of its nodes, which assumes a consistent parametrisation of the rotations and shape functions valid for large displacements.
2. The *floating reference frame* approach [126]: the motion of a flexible body results from the superimposition of a rigid body (referenced by its body reference frame) motion and a deformation, generally assumed small, with respect to the latter. Usually, different coordinates are used to describe the rigid and the flexible motions using this approach. Figure 3.18 illustrates the situation in which frame O_{f} is used as the body reference frame represented by the dashed undeformed configuration.

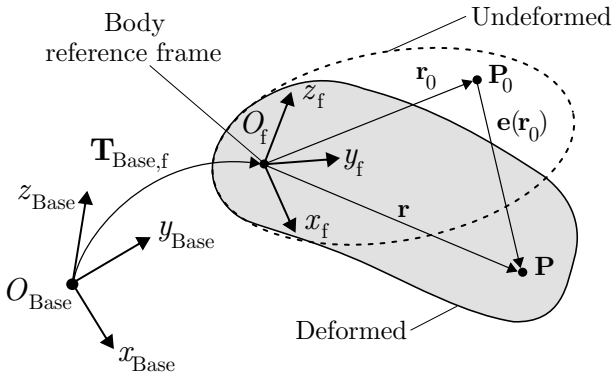


Figure 3.18: Concept of the floating reference frame approach

The pose of the reference frame is given by homogeneous transformation matrix $\mathbf{T}_{\text{Base,f}}$ as it was the case for a rigid body motion. As a result, local position of point $\mathbf{P}_0$ attached to the undeformed body is given by vector $\mathbf{r}_0$ (subscript 0 refers to the undeformed configuration). If deformation occurs, position of point $\mathbf{P}$ is given by vector $\mathbf{r}$ expressed as

$$\mathbf{r} = \mathbf{r}_0 + \mathbf{e}(\mathbf{r}_0), \tag{3.57}$$

with $\mathbf{e}(\mathbf{r}_0)$ the model of elastic deformation with respect to the undeformed configuration. This model can be expressed as the sum of pre-defined deformations, also called mode components $\mathbf{\Psi}(\mathbf{r}_0)$, such as

$$\mathbf{e}(\mathbf{r}_0) = \sum_{\mathrm{c}=1}^{n_\mathrm{d}} \mathbf{\Psi}_\mathrm{c}(\mathbf{r}_0) \cdot \boldsymbol{\lambda}_\mathrm{c}, \tag{3.58}$$

with n_d the number of deformation mode and $\boldsymbol{\lambda}$ being a vector of weighting coefficients related to the component modes, the elastic degrees of freedom.

Although the two approaches are popular, they were not convenient to model the robot flexible links in an efficient way suitable for the minimal coordinates. On the one hand, the full finite element approach writes equations of motion for specific translation and rotation parameters dedicated to one particular element (beams, plates, tetrahedrons, bricks, etc.). Since the shape of the link is a priori arbitrary, a transposition of the full finite element method for each type of element would make the foreseen approach inefficient. On the other hand, the approach based on the floating reference frame was also discarded as it imposes the knowledge of information regarding the component modes $\mathbf{\Psi}(\mathbf{r}_0)$ used to express the deformation, again for all types of element.

Consequently, the wish to keep freedom in the choice of coordinates led to the implementation of the so-called *corotational* formulation and to its adaptation to the minimal coordinates [127–129]. Succinctly, it allows keeping the selected coordinates and the formulation is somehow versatile to model the flexibility of elements with arbitrary shape.

3.4.1 Corotational formulation

The concept of the corotational formulation consists in representing the motion of one complete flexible body only on the basis of reduced information being the pose of some of its nodes. The nodes can be interpreted as interface nodes connected to the rest of the mechanical system even though internal nodes can be inserted to enhance the modelling. The deformation of the flexible body is computed from the displacement of its nodes. The corotational formulation implements a particular type of superelement in the sense that an additional frame, so-called the *corotational frame*, is appended to the existing nodes. It represents the average motion of the nodes. As its name tells, the **co**rotational frame moves with the motion of the flexible body as it was the case with the body reference frame. Elastic deformations are eventually computed as the difference between the pose of the interface nodes with respect to the corotational frame. Figure 3.19 exposes the concept of the corotational formulation with the depiction of the node deformations. Homogeneous transformation matrix $\mathbf{T}_{\mathrm{Base,f}*}$ locates the corotational frame of flexible body f while nodes $\mathbf{T}_{\mathrm{Base,f}*.1}$ and $\mathbf{T}_{\mathrm{Base,f}*.2}$ refer to the their pose in the undeformed configuration. Initially [127], the corotational frame was attached to one of the nodes. More recently [128], Cardona proposed to use instead a floating corotational frame obtained by averaging the translational and rotational parameters of the body nodes. Deformed poses of the nodes are given by homogeneous transformation matrices

$\mathbf{T}_{\text{Base,f.1}}$ and $\mathbf{T}_{\text{Base,f.2}}$. Finally, vectors of deformation are noted $\mathbf{e}_{\text{f.1}}$ and $\mathbf{e}_{\text{f.2}}$. Homogeneous transformation matrices are all expressed in base frame O_{Base}.

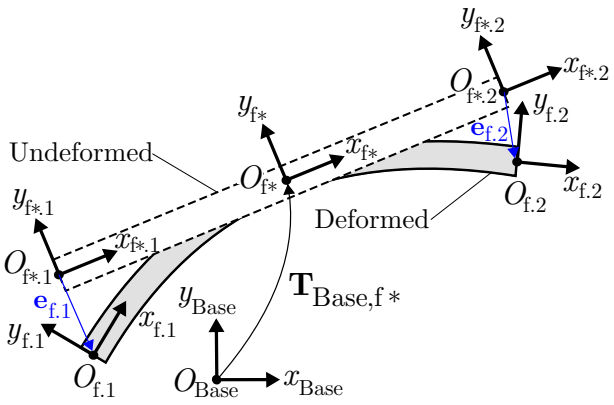


Figure 3.19: Concept of the corotational formulation

The magic idea of the corotational formulation lies in the expression of the kinetic energy for a N-node flexible body. The kinetic energy of superelement f comprising N nodes and based on the corotational formulation is expressed as

$$\mathcal{T}_{\text{f}} = \frac{1}{2} \begin{bmatrix} \{\mathbf{v}_{\text{f.1}}\}_{\text{f}*} \\ \{\boldsymbol{\omega}_{\text{f.1}}\}_{\text{f}*} \\ \vdots \\ \{\mathbf{v}_{\text{f.N}}\}_{\text{f}*} \\ \{\boldsymbol{\omega}_{\text{f.N}}\}_{\text{f}*} \end{bmatrix}^T \cdot \underbrace{\begin{bmatrix} \mathbf{M}_{\text{f,T}_1,\text{T}_1} & \mathbf{M}_{\text{f,T}_1,\text{R}_1} & \vdots & \mathbf{M}_{\text{f,T}_1,\text{T}_\text{N}} & \mathbf{M}_{\text{f,T}_1,\text{R}_\text{N}} \\ \mathbf{M}_{\text{f,R}_1,\text{T}_1} & \mathbf{M}_{\text{f,R}_1,\text{R}_1} & \vdots & \mathbf{M}_{\text{f,R}_1,\text{T}_\text{N}} & \mathbf{M}_{\text{f,R}_1,\text{R}_\text{N}} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{M}_{\text{f,T}_\text{N},\text{T}_1} & \mathbf{M}_{\text{f,T}_\text{N},\text{R}_1} & \vdots & \mathbf{M}_{\text{f,T}_\text{N},\text{T}_\text{N}} & \mathbf{M}_{\text{f,T}_\text{N},\text{R}_\text{N}} \\ \mathbf{M}_{\text{f,R}_\text{N},\text{T}_1} & \mathbf{M}_{\text{f,R}_\text{N},\text{R}_1} & \vdots & \mathbf{M}_{\text{f,R}_\text{N},\text{T}_\text{N}} & \mathbf{M}_{\text{f,R}_\text{N},\text{R}_\text{N}} \end{bmatrix}_{\text{f}*}}_{\{\mathbf{M}_{\text{f}}\}_{\text{f}*}} \cdot \begin{bmatrix} \{\mathbf{v}_{\text{f.1}}\}_{\text{f}*} \\ \{\boldsymbol{\omega}_{\text{f.1}}\}_{\text{f}*} \\ \vdots \\ \{\mathbf{v}_{\text{f.N}}\}_{\text{f}*} \\ \{\boldsymbol{\omega}_{\text{f.N}}\}_{\text{f}*} \end{bmatrix}, \tag{3.59}$$

with $\{\mathbf{v}_{\text{f.g}}\}_{\text{f}*}$ the vector of translational velocity of node g of flexible body f and $\{\boldsymbol{\omega}_{\text{f.g}}\}_{\text{f}*}$ the vector of rotational velocity of node g of flexible body f, both projected in the corotational frame, denoted by $O_{\text{f}*}$, which follows or rotates with flexible body f. Exposed mass matrix $\mathbf{M}_{\text{f}}$ comes from a finite element model of flexible body f and is composed of sub-matrices $\mathbf{M}_{\text{f,T}_\text{g},\text{T}_\text{h}}$, $\mathbf{M}_{\text{f,T}_\text{g},\text{R}_\text{h}}$, $\mathbf{M}_{\text{f,R}_\text{g},\text{T}_\text{h}}$ and $\mathbf{M}_{\text{f,R}_\text{g},\text{R}_\text{h}}$ that can be considered as (3×3) tensors which are constant in corotational frame $O_{\text{f}*}$. Subscripts T and R are respectively related to the translational and rotational motions of the node. Indexes g and h refer to the node number. The only condition to generate a mass matrix in accordance with the corotational formulation is that the finite element model involves the 3 displacements and the 3 rotations of each node. Dimensions are thus consistent with Eq. 3.59.

Note that a rigid body f can be considered as a one-node element for which the sub-matrices would take the following form if the node was located at the centre of mass

$$[\mathbf{M}_{\text{f,T}_1,\text{T}_1}]_{\text{f}*} = m_{\text{f}}\mathbf{I}, \qquad [\mathbf{M}_{\text{f,T}_1,\text{R}_1}]_{\text{f}*} = [\mathbf{M}_{\text{f,R}_1,\text{T}_1}]_{\text{f}*} = \mathbf{0}, \qquad [\mathbf{M}_{\text{f,R}_1,\text{R}_1}]_{\text{f}*} = \boldsymbol{\Phi}_{\text{G,f}}, \tag{3.60}$$

with $\mathbf{I}$ the (3×3) identity matrix.

The main advantage of the approach is that any kind of flexible body can be modelled as soon as its finite element model is available and respect the aforementioned condition

regarding its mass matrix $\mathbf{M}_\mathrm{f}$. The same condition applies to its stiffness matrix $\mathbf{K}_\mathrm{f}$ which is expressed in a similar way as

$$\{\mathbf{K}_\mathrm{f}\}_{\mathrm{f}*} = \begin{bmatrix} \mathbf{K}_{\mathrm{f},\mathrm{T}_1,\mathrm{T}_1} & \mathbf{K}_{\mathrm{f},\mathrm{T}_1,\mathrm{R}_1} & : & \mathbf{K}_{\mathrm{f},\mathrm{T}_1,\mathrm{T}_\mathrm{N}} & \mathbf{K}_{\mathrm{f},\mathrm{T}_1,\mathrm{R}_\mathrm{N}} \\ \mathbf{K}_{\mathrm{f},\mathrm{R}_1,\mathrm{T}_1} & \mathbf{K}_{\mathrm{f},\mathrm{R}_1,\mathrm{R}_1} & : & \mathbf{K}_{\mathrm{f},\mathrm{R}_1,\mathrm{T}_\mathrm{N}} & \mathbf{K}_{\mathrm{f},\mathrm{R}_1,\mathrm{R}_\mathrm{N}} \\ : & : & : & : & : \\ \mathbf{K}_{\mathrm{f},\mathrm{T}_\mathrm{N},\mathrm{T}_1} & \mathbf{K}_{\mathrm{f},\mathrm{T}_\mathrm{N},\mathrm{R}_1} & : & \mathbf{K}_{\mathrm{f},\mathrm{T}_\mathrm{N},\mathrm{T}_\mathrm{N}} & \mathbf{K}_{\mathrm{f},\mathrm{T}_\mathrm{N},\mathrm{R}_\mathrm{N}} \\ \mathbf{K}_{\mathrm{f},\mathrm{R}_\mathrm{N},\mathrm{T}_1} & \mathbf{K}_{\mathrm{f},\mathrm{R}_\mathrm{N},\mathrm{R}_1} & : & \mathbf{K}_{\mathrm{f},\mathrm{R}_\mathrm{N},\mathrm{T}_\mathrm{N}} & \mathbf{K}_{\mathrm{f},\mathrm{R}_\mathrm{N},\mathrm{R}_\mathrm{N}} \end{bmatrix}_{\mathrm{f}*}, \tag{3.61}$$

with sub-matrices $\mathbf{K}_{\mathrm{f},\mathrm{T}_\mathrm{g},\mathrm{T}_\mathrm{h}}$, $\mathbf{K}_{\mathrm{f},\mathrm{T}_\mathrm{g},\mathrm{R}_\mathrm{h}}$, $\mathbf{K}_{\mathrm{f},\mathrm{R}_\mathrm{g},\mathrm{T}_\mathrm{h}}$ and $\mathbf{K}_{\mathrm{f},\mathrm{R}_\mathrm{g},\mathrm{R}_\mathrm{h}}$ that can also be considered as (3×3) tensors constant in corotational frame $O_{\mathrm{f}*}$. However, the approach is not fully consistent as Eq. 3.59 implements an interpolation of the velocity field through the shape functions of the superelement. To understand the possible discrepancies in this interpolation, velocity of point $\mathbf{v}_\mathrm{P}$ of flexible body f in Figure 3.18 using the floating reference frame approach is developed such as

$$\mathbf{v}_\mathrm{P} = \mathbf{v}_\mathrm{f} + \boldsymbol{\omega}_\mathrm{f} \times (\mathbf{r}_0 + \mathbf{e}(\mathbf{r}_0)) + \dot{\mathbf{e}}(\mathbf{r}_0), \tag{3.62}$$

where $\mathbf{v}_\mathrm{f}$ and $\boldsymbol{\omega}_\mathrm{f}$ are the velocity of the origin of the floating reference frame and the rotational velocity of the floating reference frame, respectively, $\mathbf{r}_0$ the local position vector of point P in the undeformed configuration and $\mathbf{e}(\mathbf{r}_0)$ its elastic displacement. In detail, the previous equation can be expressed as

$$\mathbf{v}_\mathrm{P} = \underbrace{\mathbf{v}_\mathrm{f} + \boldsymbol{\omega}_\mathrm{f} \times \mathbf{r}_0}_{\mathbf{v}_\mathrm{rigid}} + \boldsymbol{\omega}_\mathrm{f} \times \mathbf{e}(\mathbf{r}_0) + \dot{\mathbf{e}}(\mathbf{r}_0) = \mathbf{v}_\mathrm{rigid} + \boldsymbol{\omega}_\mathrm{f} \times \mathbf{e}(\mathbf{r}_0) + \dot{\mathbf{e}}(\mathbf{r}_0), \tag{3.63}$$

where $\mathbf{v}_\mathrm{rigid}$ represents the rigid body contribution. The discrepancies that may arise using the corotational formulation are related to term $\boldsymbol{\omega}_\mathrm{f} \times \mathbf{e}(\mathbf{r}_0)$. Usually, the shape functions of the finite element model are able to span rigid body displacement fields. Hence, term $\mathbf{v}_\mathrm{rigid}$ is extrapolated exactly by the corotational formulation. Likewise, term $\dot{\mathbf{e}}(\mathbf{r}_0)$ is also interpolated in the same way as the displacement field and keeps the inherent accuracy of the finite element model. On the contrary, term $\boldsymbol{\omega}_\mathrm{f} \times \mathbf{e}(\mathbf{r}_0)$ is orthogonal to the displacement field and is not necessarily interpolated exactly between the nodes. This explains the well-known limitations of the corotational formulation which turns out to be only suitable when deformations and rotation speeds are small. It is indeed the case for milling operations since the imposed feed rate is usually small. The results are also expected to be better if the corotational frame is located near the centre of mass of the deformed body as reported in [128].

Damping can also be included to the corotational formulation. If a proportional damping is assumed, damping matrix $\mathbf{C}_\mathrm{f}$ of flexible body f projected in corotational frame $O_{\mathrm{f}*}$ reads

$$\{\mathbf{C}_\mathrm{f}\}_{\mathrm{f}*} = \alpha_\mathrm{Damping} \{\mathbf{M}_\mathrm{f}\}_{\mathrm{f}*} + \beta_\mathrm{Damping} \{\mathbf{K}_\mathrm{f}\}_{\mathrm{f}*}, \tag{3.64}$$

with α_Damping and β_Damping, the mass and stiffness proportional Rayleigh damping coefficients.

3.4.2 Contribution of a flexible body to the equations of motion

The purpose of this subsection is to develop the equations of motion of a multibody system comprising flexible bodies in a form as close as possible to Eq. 3.31 derived from the application of the d'Alembert's principle for rigid bodies.

The final form of the equations of motion, for a multibody system comprising n_{F} flexible bodies with N nodes and relying on the corotational formulation, reads

$$\begin{aligned}\sum_{\mathrm{f}=1}^{n_{\mathrm{F}}}\sum_{\mathrm{g}=1}^{\mathrm{N}}\Bigg[&\{\mathbf{d}^{\mathrm{f.g,j}}\}_{\mathrm{Base}}\cdot(\underbrace{\boldsymbol{\mathcal{R}}_{\mathrm{f.g}}(-m\mathbf{a})}_{\text{Inertia}}+\underbrace{\boldsymbol{\mathcal{R}}_{\mathrm{f.g}}(\mathbf{g})}_{\text{Gravity}}+\underbrace{\boldsymbol{\mathcal{R}}_{\mathrm{e,f.g}}}_{\text{Elastic}}+\underbrace{\boldsymbol{\mathcal{R}}_{\mathrm{c,f.g}}}_{\text{Damping}}+\underbrace{\boldsymbol{\mathcal{R}}_{\mathrm{f.g}}}_{\text{External forces}})+\\ &\{\mathbf{w}^{\mathrm{f.g,j}}\}_{\mathrm{Base}}\cdot(\underbrace{\boldsymbol{\mathcal{M}}_{\mathrm{f.g}}(-m\mathbf{a})}_{\text{Inertia}}+\underbrace{\boldsymbol{\mathcal{M}}_{\mathrm{f.g}}(\mathbf{g})}_{\text{Gravity}}+\underbrace{\boldsymbol{\mathcal{M}}_{\mathrm{e,f.g}}}_{\text{Elastic}}+\underbrace{\boldsymbol{\mathcal{M}}_{\mathrm{c,f.g}}}_{\text{Damping}}+\underbrace{\boldsymbol{\mathcal{M}}_{\mathrm{f.g}}}_{\text{External moments}})\Bigg]+\\ &\{\mathbf{w}^{\mathrm{f*,j}}\}_{\mathrm{Base}}\cdot\underbrace{\boldsymbol{\mathcal{M}}_{\mathrm{f*}}(-m\mathbf{a})}_{\text{Inertia moments}}=0\quad \mathrm{j}=1,\ldots,n\quad ,\end{aligned}\tag{3.65}$$

with subscripts f, g, j related to the flexible body number, the node number and the current degree of freedom, respectively, and, $\mathbf{d}^{\mathrm{f.g,j}}$ and $\mathbf{w}^{\mathrm{f.g,j}}$ the partial contributions of $\dot{q}_{\mathrm{j}}$ in the translational and rotational velocities of node g of flexible body f projected in base frame O_{Base}. Similarly, $\mathbf{w}^{\mathrm{f*,j}}$ is the partial contributions of $\dot{q}_{\mathrm{j}}$ in the rotational velocity of corotational frame $O_{\mathrm{f*}}$ of flexible body f projected in the base frame such as

$$\{\boldsymbol{\omega}_{\mathrm{f*}}\}_{\mathrm{Base}}=\sum_{\mathrm{j}=1}^{n}\{\mathbf{w}^{\mathrm{f*,j}}\}_{\mathrm{Base}}\cdot\dot{q}_{\mathrm{j}},\tag{3.66}$$

with $\boldsymbol{\omega}_{\mathrm{f*}}$ the rotational velocity of corotational frame $O_{\mathrm{f*}}$ of flexible body f.

The different contributions in Eq. 3.65 relative to the resultant forces $\boldsymbol{\mathcal{R}}$ and moments $\boldsymbol{\mathcal{M}}$ applied on node g of flexible body f and on its corotational frame $O_{\mathrm{f*}}$ are developed hereafter as well as their projection. They are successively related to the inertia reaction forces (suffix $(-m\mathbf{a})$), the gravity forces (suffix $(\mathbf{g})$), the elastic forces (subscript e), the damping forces (subscript c) and the external forces.

1. Contribution of the inertia forces

The contribution of the inertia forces projected in the corotational frame ($\{\boldsymbol{\mathcal{R}}_{\mathrm{f.g}}(-m\mathbf{a})\}_{\mathrm{f*}}$, $\{\boldsymbol{\mathcal{M}}_{\mathrm{f.g}}(-m\mathbf{a})\}_{\mathrm{f*}}$ and $\{\boldsymbol{\mathcal{M}}_{\mathrm{f*}}(-m\mathbf{a})\}_{\mathrm{f*}}$) in the equations of motion for flexible bodies can also be obtained via the Lagrange formulation. The kinetic energy (Eq. 3.59) of superelement f comprising N nodes can be reformulated as follows

$$\mathcal{T}_{\mathrm{f}} = \frac{1}{2}\sum_{\mathrm{f}=1}^{n_{\mathrm{f}}}\sum_{\mathrm{g}=1}^{\mathrm{N}}\sum_{\mathrm{h}=1}^{\mathrm{N}}\big[\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\cdot[\mathbf{M}_{\mathrm{f},\mathrm{T_g},\mathrm{T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\cdot[\mathbf{M}_{\mathrm{f},\mathrm{T_g},\mathrm{R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\cdot[\mathbf{M}_{\mathrm{f},\mathrm{R_g},\mathrm{T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\cdot[\mathbf{M}_{\mathrm{f},\mathrm{R_g},\mathrm{R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\big], \tag{3.67}$$

with sub-matrices $\mathbf{M}_{\mathrm{f},\mathrm{T_g},\mathrm{T_h}}$, $\mathbf{M}_{\mathrm{f},\mathrm{T_g},\mathrm{R_h}}$, $\mathbf{M}_{\mathrm{f},\mathrm{R_g},\mathrm{T_h}}$ and $\mathbf{M}_{\mathrm{f},\mathrm{R_g},\mathrm{R_h}}$ projected in corotational frame $O_{\mathrm{f}*}$ in which they are constant.

From the application of the Lagrange formulation, it can be shown that the equivalent inertia reaction force exerted on node g of flexible body f is given by

$$\begin{aligned}\{\boldsymbol{\mathcal{R}}_{\mathrm{f.g}}(-m\mathbf{a})\}_{\mathrm{f}*} = -\sum_{\mathrm{g}=1}^{\mathrm{N}}\sum_{\mathrm{h}=1}^{\mathrm{N}}\big[&[\mathbf{M}_{\mathrm{f},\mathrm{T_g},\mathrm{T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}\times\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ &[\mathbf{M}_{\mathrm{f},\mathrm{T_g},\mathrm{R_h}}]_{\mathrm{f}*}(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ &\boldsymbol{\omega}_{\mathrm{f}*}\times([\mathbf{M}_{\mathrm{f},\mathrm{T_g},\mathrm{T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + [\mathbf{M}_{\mathrm{f},\mathrm{T_g},\mathrm{R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*})\big],\end{aligned} \tag{3.68}$$

while the torque can be computed according to

$$\begin{aligned}\{\boldsymbol{\mathcal{M}}_{\mathrm{f.g}}(-m\mathbf{a})\}_{\mathrm{f}*} = -\sum_{\mathrm{g}=1}^{\mathrm{N}}\sum_{\mathrm{h}=1}^{\mathrm{N}}\big[&[\mathbf{M}_{\mathrm{f},\mathrm{R_g},\mathrm{T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}\times\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ &[\mathbf{M}_{\mathrm{f},\mathrm{R_g},\mathrm{R_h}}]_{\mathrm{f}*}(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ &(\boldsymbol{\omega}_{\mathrm{f}*} - \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\times \\ &([\mathbf{M}_{\mathrm{f},\mathrm{R_g},\mathrm{T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + [\mathbf{M}_{\mathrm{f},\mathrm{R_g},\mathrm{R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*})\big].\end{aligned} \tag{3.69}$$

In the same way, the inertia forces bring a contribution equivalent to a torque $\{\boldsymbol{\mathcal{M}}_{\mathrm{f}*}(-m\mathbf{a})\}_{\mathrm{f}*}$ exerted on corotational frame $O_{\mathrm{f}*}$ given by

$$\begin{aligned}\{\boldsymbol{\mathcal{M}}_{\mathrm{f}*}(-m\mathbf{a})\}_{\mathrm{f}*} = -\sum_{\mathrm{g}=1}^{\mathrm{N}}\sum_{\mathrm{h}=1}^{\mathrm{N}}\big[&\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times([\mathbf{M}_{\mathrm{f},\mathrm{T_g},\mathrm{T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ &\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times([\mathbf{M}_{\mathrm{f},\mathrm{R_g},\mathrm{R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ &\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times([\mathbf{M}_{\mathrm{f},\mathrm{R_g},\mathrm{T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ &\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times([\mathbf{M}_{\mathrm{f},\mathrm{T_g},\mathrm{R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*})\big].\end{aligned} \tag{3.70}$$

The complete derivation of the contribution of the inertia forces using the corotational formulation is provided in Appendix B.

2. Contribution of the gravity forces

Contribution of gravity projected in the corotational frame ($\{\boldsymbol{\mathcal{R}}_{\mathrm{f.g}}(\mathbf{g})\}_{\mathrm{f}*}$ and $\{\boldsymbol{\mathcal{M}}_{\mathrm{f.g}}(\mathbf{g})\}_{\mathrm{f}*}$) is naturally included in the equations of motion through the inertia terms by replacing in Eqs. 3.68 and 3.69 the acceleration $\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*}$ by $\{\mathbf{a}_{\mathrm{f.h}} - \mathbf{g}_0\}_{\mathrm{f}*}$, $\mathbf{g}_0$ being the

gravity vector $[0\ 0\ g_z]^T_{\text{Base}}$ in the base frame. The equivalent force and moment on node g of flexible body f correspond to

$$\{\boldsymbol{\mathcal{R}}_{\text{f.g}}(\mathbf{g})\}_{\text{f}*} = \left[\sum_{\text{g}=1}^{\text{N}}\sum_{\text{h}=1}^{\text{N}}[\mathbf{M}_{\text{f,T}_\text{g},\text{T}_\text{h}}]_{\text{f}*}\right]\{\mathbf{g}_0\}_{\text{f}*}, \tag{3.71}$$

$$\{\boldsymbol{\mathcal{M}}_{\text{f.g}}(\mathbf{g})\}_{\text{f}*} = \left[\sum_{\text{g}=1}^{\text{N}}\sum_{\text{h}=1}^{\text{N}}[\mathbf{M}_{\text{f,R}_\text{g},\text{T}_\text{h}}]_{\text{f}*}\right]\{\mathbf{g}_0\}_{\text{f}*}. \tag{3.72}$$

3. Contribution of the elastic forces

Assuming small deformations, the elastic forces are computed from the finite element stiffness matrix of the superelement considered in the axes of the corotational frame. The elastic force $\{\boldsymbol{\mathcal{R}}_{\text{e,f.g}}\}_{\text{f}*}$ and moment $\{\boldsymbol{\mathcal{M}}_{\text{e,f.g}}\}_{\text{f}*}$ applied on node g of flexible body f, projected in corotational frame $O_{\text{f}*}$, are given by

$$\{\boldsymbol{\mathcal{R}}_{\text{e,f.g}}\}_{\text{f}*} = -\sum_{\text{g}=1}^{\text{N}}\sum_{\text{h}=1}^{\text{N}}\left[[\mathbf{K}_{\text{f,T}_\text{g},\text{T}_\text{h}}]_{\text{f}*}\{\Delta\mathbf{p}_{\text{f.h}}\}_{\text{f}*} + [\mathbf{K}_{\text{f,T}_\text{g},\text{R}_\text{h}}]_{\text{f}*}\{\Delta\boldsymbol{\phi}_{\text{f.h}}\}_{\text{f}*}\right], \tag{3.73}$$

$$\{\boldsymbol{\mathcal{M}}_{\text{e,f.g}}\}_{\text{f}*} = -\sum_{\text{g}=1}^{\text{N}}\sum_{\text{h}=1}^{\text{N}}\left[[\mathbf{K}_{\text{f,R}_\text{g},\text{T}_\text{h}}]_{\text{f}*}\{\Delta\mathbf{p}_{\text{f.h}}\}_{\text{f}*} + [\mathbf{K}_{\text{f,R}_\text{g},\text{R}_\text{h}}]_{\text{f}*}\{\Delta\boldsymbol{\phi}_{\text{f.h}}\}_{\text{f}*}\right], \tag{3.74}$$

with $\mathbf{K}_{\text{f,T}_\text{g},\text{T}_\text{h}}$, $\mathbf{K}_{\text{f,T}_\text{g},\text{R}_\text{h}}$, $\mathbf{K}_{\text{f,R}_\text{g},\text{T}_\text{h}}$ and $\mathbf{K}_{\text{f,R}_\text{g},\text{R}_\text{h}}$ the (3×3) sub-matrices of stiffness matrix $\mathbf{K}_{\text{f}}$ of superelement f corresponding to translational and rotational components projected in corotational frame $O_{\text{f}*}$ as defined in Eq. 3.61. The elastic displacements $\Delta\mathbf{p}_{\text{f.h}} = \mathbf{e}_{\text{f.h}}$ and rotations $\Delta\boldsymbol{\phi}_{\text{f.h}}$ are computed from the difference between the actual position of node g and its position on the undeformed body attached to corotational frame $O_{\text{f}*}$ as depicted in Figure 3.19 ($\Delta\mathbf{p}_{\text{f.h}} = \mathbf{p}_{\text{f.h}} - \mathbf{p}_{\text{f}*.\text{h}}$). For the sake of simplicity, the computation of the elastic rotations is performed from angles between axes, assuming small relative rotations, using the XYZ Euler angles ($\Delta\boldsymbol{\phi}_{\text{f.h}} = \boldsymbol{\phi}_{\text{f.h}} - \boldsymbol{\phi}_{\text{f}*.\text{h}}$) (Figure 3.6).

4. Contribution of the damping forces

The contribution of the damping forces projected in the corotational frame ($\{\boldsymbol{\mathcal{R}}_{\text{c,f.g}}\}_{\text{f}*}$ and $\{\boldsymbol{\mathcal{M}}_{\text{c,f.g}}\}_{\text{f}*}$) in the equations of motion are computed in the similar way as for the elastic forces such as

$$\{\boldsymbol{\mathcal{R}}_{\text{c,f.g}}\}_{\text{f}*} = -\sum_{\text{g}=1}^{\text{N}}\sum_{\text{h}=1}^{\text{N}}\left[[\mathbf{C}_{\text{f,T}_\text{g},\text{T}_\text{h}}]_{\text{f}*}\{\Delta\dot{\mathbf{p}}_{\text{f.h}}\}_{\text{f}*} + [\mathbf{C}_{\text{f,T}_\text{g},\text{R}_\text{h}}]_{\text{f}*}\{\Delta\boldsymbol{\omega}_{\text{f.h}}\}_{\text{f}*}\right], \tag{3.75}$$

$$\{\boldsymbol{\mathcal{M}}_{\text{c,f.g}}\}_{\text{f}*} = -\sum_{\text{g}=1}^{\text{N}}\sum_{\text{h}=1}^{\text{N}}\left[[\mathbf{C}_{\text{f,R}_\text{g},\text{T}_\text{h}}]_{\text{f}*}\{\Delta\dot{\mathbf{p}}_{\text{f.h}}\}_{\text{f}*} + [\mathbf{C}_{\text{f,R}_\text{g},\text{R}_\text{h}}]_{\text{f}*}\{\Delta\boldsymbol{\omega}_{\text{f.h}}\}_{\text{f}*}\right], \tag{3.76}$$

with $\mathbf{C}_{\mathrm{f,T_g,T_h}}$, $\mathbf{C}_{\mathrm{f,T_g,R_h}}$, $\mathbf{C}_{\mathrm{f,R_g,T_h}}$ and $\mathbf{C}_{\mathrm{f,R_g,R_h}}$ the (3×3) sub-matrices of damping matrix $\mathbf{C}_{\mathrm{f}}$ of superelement f corresponding to translational and rotational components projected in corotational frame $O_{\mathrm{f}*}$. The elastic velocities in translation $\Delta\dot{\mathbf{p}}_{\mathrm{f.h}}$ are computed from the difference between the actual velocity of node g and its velocity on the undeformed body attached to corotational frame $O_{\mathrm{f}*}$ using a velocity field ($\Delta\dot{\mathbf{p}}_{\mathrm{f.h}} = \mathbf{v}_{\mathrm{f.h}} + \boldsymbol{\omega}_{\mathrm{f}*} \times \Delta\mathbf{p}_{\mathrm{f.h}}$). The elastic velocities in rotation $\Delta\boldsymbol{\omega}_{\mathrm{f.h}}$ are computed from the difference between the actual rotational velocity of node g and its rotational velocity on the undeformed body attached to corotational frame $O_{\mathrm{f}*}$ ($\Delta\boldsymbol{\omega}_{\mathrm{f.h}} = \boldsymbol{\omega}_{\mathrm{f.h}} - \boldsymbol{\omega}_{\mathrm{f}*.\mathrm{h}}$).

5. Contribution of the external forces

The external force $\{\boldsymbol{\mathcal{R}}_{\mathrm{f.g}}\}_{\mathrm{Base}}$ and moment $\{\boldsymbol{\mathcal{M}}_{\mathrm{f.g}}\}_{\mathrm{Base}}$ on node g of flexible body f projected in the base frame are expressed as

$$\{\boldsymbol{\mathcal{R}}_{\mathrm{f.g}}\}_{\mathrm{Base}} = F_{\mathrm{ext}}\mathbf{R}_{\mathrm{Base,f.g}}\mathbf{u}_{x|y|z}, \tag{3.77}$$

$$\{\boldsymbol{\mathcal{M}}_{\mathrm{f.g}}\}_{\mathrm{Base}} = \tau_{\mathrm{ext}}\mathbf{R}_{\mathrm{Base,f.g}}\mathbf{u}_{x|y|z}, \tag{3.78}$$

where F_{ext} and τ_{ext} are scalar force and torque, respectively, applied along or around the X, Y or Z direction using unit vector $\mathbf{u}_{x|y|z}$.

3.4.3 Application to the robot links

The corotational formulation was used to model the flexibility of some of the links of the Stäubli TX200 robot although W.J. Book proposed to consider their deflection via transformation matrices from a summation of their modal shapes in [130]. Clearly, it is understandable that the main contributions to the link flexibility are likely to arise in the robot arm and forearm. The other links are either too short or internally reinforced to significantly contribute to the link flexibility. In contrast, the robot arm and forearm present elongated and hollow shapes prone to deformations. As their structure suggests, their flexibility is modelled using flexible beam elements. In order to apply the corotational formulation to the particular case of flexible beams, the corresponding mass and stiffness matrices need to be provided in their 3D format. As a result, a flexible beam element is represented with two end nodes which can move freely in space i.e. the motion of each node involves 3 displacements and 3 rotations. The flexible beam element is thus able to respond to flexural, tensile and torsional stresses. The 3D mass and stiffness matrices for a flexible beam element are provided in Appendix C. Conceptual examples are found and concern the application of the corotational formulation to the modelling of flexible beam by following two approaches.

Focussing on the modelling of the robot arm and forearm with flexible beam elements, Figure 3.20 illustrates the positions of the nodes and the corotational frame. For the sake

of computing efficiency, it was decided to simply model either the robot arm or forearm, by one single beam element. It comprises two nodes, one at each end $O_{1.1}$ and $O_{1.2}$, and corotational frame O_{1*} was located on the first node for simplicity. The neutral axis of the beam is represented by local X axis and the Y and Z directions are the equatorial axes.

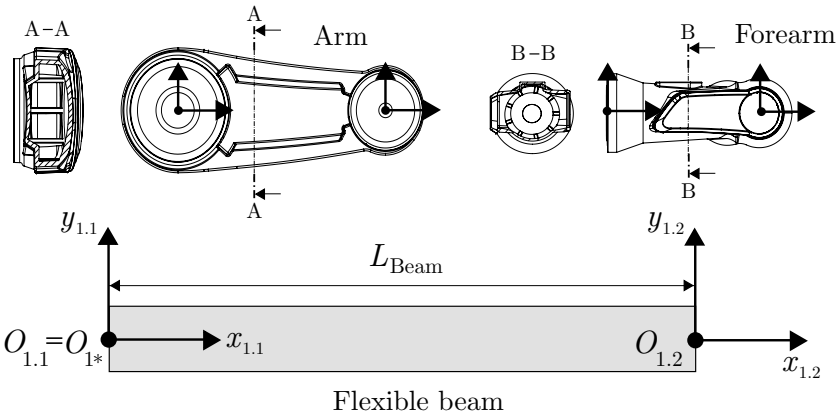


Figure 3.20: Modelling of the robot link flexibility

The situation of each of the nodes and the corotational frame is still given by homogeneous transformation matrices. The modelling of the flexible robot arm is addressed as an illustrative example. Practically, the robot arm is attached to the second joint after its gearbox. From relationship 3.42, the connecting point with the flexible beam is given by homogeneous transformation matrix $\mathbf{T}_{\text{Base,Shaft}_2}$. If tri-axial deflections are included to model the joint flexibility, the situation of the first node $O_{1.1}$ with respect to base frame O_{Base} reads

$$\mathbf{T}_{\text{Base,1.1}} = \mathbf{T}_{\text{Base,Shaft}_2} \cdot \mathbf{T}_{\text{rotz}}(q_{z,2}) \cdot \mathbf{T}_{\text{rotx}}(q_{x,2}) \cdot \mathbf{T}_{\text{roty}}(q_{y,2}). \tag{3.79}$$

Node $O_{1.1}$ is assumed to be clamped just after the consideration of the joint tri-axial flexibility i.e. no deformation from the flexible beam is accounted at that position. Assuming that the length of the arm is equal to length L_{Beam}, the situation of the second node is provided by

$$\begin{aligned} \mathbf{T}_{\text{Base,1.2}} =& \mathbf{T}_{\text{Base,1.1}} \cdot \mathbf{T}_{\text{disp}}(L_{\text{Beam}} + q_{1.2,x}, q_{1.2,y}, q_{1.2,z}) \cdot \\ & \mathbf{T}_{\text{rotx}}(q_{1.2,\text{Rx}}) \cdot \mathbf{T}_{\text{roty}}(q_{1.2,\text{Ry}}) \cdot \mathbf{T}_{\text{rotz}}(q_{1.2,\text{Rz}}), \end{aligned} \tag{3.80}$$

with $q_{1.2,x}$, $q_{1.2,y}$ and $q_{1.2,z}$ the translational degrees of freedom of the second node of the first flexible beam element and $q_{1.2,\text{Rx}}$, $q_{1.2,\text{Ry}}$ and $q_{1.2,\text{Rz}}$ its rotational degrees of freedom. Detailed information regarding the distribution of degrees of freedom per node is found in Appendix C. The location of corotational frame O_{1*} is the same as the first node and reads

$$\mathbf{T}_{\text{Base,1*}} = \mathbf{T}_{\text{Base,1.1}}. \tag{3.81}$$

Hence the nodes relative to the corotational frame in the undeformed configuration and from which the elastic forces are computed are given by $\mathbf{T}_{\text{Base},1*.1} = \mathbf{T}_{\text{Base},1.1}$ and $\mathbf{T}_{\text{Base},1*.2} = \mathbf{T}_{\text{Base},1.1} \cdot \mathbf{T}_{\text{disp}}(L_{\text{Beam}}, 0, 0)$. Similar expressions are derived to define the situations of nodes $O_{2.1}$ and $O_{2.2}$ and corotational frame O_{2*} of the flexible beam element representing the robot forearm. In total, twelve degrees of freedom account for the link deformations of the robot arm and forearm with $\mathbf{q}_{\text{f}} \in \mathbb{R}^{12\times 1}$.

Once the situations of each node and corotational frames are known, their partial contributions and accelerations are computed using Eqs. 3.28 to 3.30 and Eq. 3.66. Since the multibody model of the robot is built using rigid and flexible elements, the equations of motion are computed using Eqs. 3.31 and 3.65 from the application of the d'Alembert's principle.

The equations of motion of the flexible robot can still be expressed in a matrix form as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} + \mathbf{F}_{\text{v}}\dot{\mathbf{q}} + \mathbf{F}_{\text{s}}\text{sgn}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \mathbf{\Gamma}, \tag{3.82}$$

given from the joint side with $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{39\times 39}$, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{39\times 39}$, $\mathbf{K} \in \mathbb{R}^{39\times 39}$ and $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^{39\times 1}$ which denote the system mass, damping, stiffness and gravity matrices, respectively. Diagonal matrices $\mathbf{F}_{\text{v}}$ and $\mathbf{F}_{\text{s}} \in \mathbb{R}^{39\times 39}$ only account for the viscous and static frictions originating from the motors and the transmissions. On the right-hand side, vector $\mathbf{\Gamma} \in \mathbb{R}^{39\times 1}$ gathers the combination of the actuation and external torques $\mathbf{\Gamma} = [\boldsymbol{\tau}\ \boldsymbol{\tau}_{\text{ext}}]^T$. Vector of degrees of freedom $\mathbf{q} \in \mathbb{R}^{39\times 1}$ is partitioned such that $\mathbf{q} = [\mathbf{q}_{\text{s}}\ \mathbf{q}_z\ \mathbf{q}_{x|y}\ \mathbf{q}_{\text{ground}}\ \mathbf{q}_{\text{f}}]^T = [q_1 \dots q_6\ q_{z,1} \dots q_{z,6}\ q_{x,1}q_{y,1} \dots q_{x,6}q_{y,6}\ q_{\text{g}_x}q_{\text{g}_y}q_{\text{g}_z}\ q_{1.1,x} \dots q_{2.2,\text{Rz}}]^T$. While matrix $\mathbf{M}(\mathbf{q})$ can still be decomposed as presented in Eq. 3.46 by including the degrees of freedom related to the flexible beam elements $\mathbf{q}_{\text{f}}$, the partitioning of global damping matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ and stiffness matrix $\mathbf{K}$ is provided.

The general form of the global damping matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ comes from the summation of the Christoffel matrix $\mathbf{C}_{\text{Christoffel}}$, including the effects of the Rayleigh's damping, and the virtual damping matrix $\mathbf{D}$ such as

$$\begin{aligned}\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} =& \mathbf{C}_{\text{Christoffel}}(\mathbf{q}, \dot{\mathbf{q}}) + \\ &\underbrace{\begin{bmatrix} \mathbf{K}_{\text{D}}\left(\dot{\mathbf{\Theta}} - \dot{\mathbf{q}}_{\text{s}}\right) & & & & \mathbf{0} \\ & \mathbf{d}_z\left(\dot{\mathbf{q}}_{\text{s}} - \dot{\mathbf{q}}_z\right) & & & \\ & & \mathbf{d}_{x|y}\left(\mathbf{0} - \dot{\mathbf{q}}_{x|y}\right) & & \\ & & & \mathbf{d}_{\text{g}}\left(\mathbf{0} - \dot{\mathbf{q}}_{\text{ground}}\right) & \\ \mathbf{0} & & & & \mathbf{0} \end{bmatrix}}_{\mathbf{D}}\end{aligned} \tag{3.83}$$

and the global stiffness matrix $\mathbf{K}$ is expressed such as

$$\mathbf{Kq} = \begin{bmatrix} \mathbf{K}_{\mathrm{P}}\left(\boldsymbol{\Theta} - \mathbf{q}_{\mathrm{s}}\right) & & & & & \mathbf{0} \\ & \mathbf{k}_z\left(\mathbf{q}_{\mathrm{s}} - \mathbf{q}_z\right) & & & & \\ & & \mathbf{k}_{x|y}\left(\mathbf{0} - \mathbf{q}_{x|y}\right) & & & \\ & & & \mathbf{k}_{\mathrm{g}}\left(\mathbf{0} - \mathbf{q}_{\mathrm{ground}}\right) & & \\ \mathbf{0} & & & & & \mathbf{K}(\mathbf{q}_{\mathrm{f}}) \end{bmatrix}. \quad (3.84)$$

with $\mathbf{K}(\mathbf{q}_{\mathrm{f}})$ the resulting stiffness matrix, obtained from the application of the corotational formulation, which depends on the degrees of freedom $\mathbf{q}_{\mathrm{f}}$ related to the flexible beam elements.

3.5 Dynamic vibration absorber

As it will be seen in Chapter 5 regarding the identification of robot parameters, results obtained via modal analysis technique showed that a dynamic vibration absorber (DVA) could possibly be embedded in the mechanical structure of the Stäubli TX200 robot. A dynamic vibration absorber is a tuned damper-spring-mass system which reduces or neutralises harmonic vibrations. Its effect is characterised by a sharp antiresonance, thus breaking into two parts a former peak. For the Stäubli TX200 robot, two peaks separated by a sharp antiresonance are found around 23 Hz in the measured FRFs. Corresponding mode shapes of the robot showcase a rotating motion around X axis of local joint frame O_0 (Figure 3.13). In order to model the behaviour of the dynamic vibration absorber, one more degree of freedom q_{DVA} is added to the developed model around the x_0 axis. Consequently, the final robot model utilised to simulate robotic machining operations contains 40 degrees of freedom n=40.

In contrast to classical DVAs, as mode shapes showed a rotating motion, the considered DVA is a system comprising a tuned inertia and torsional spring and damper. Hence, the DVA can be incorporated in the multibody model as an additional body with a negligible mass but, with an inertia and one degree of freedom q_{DVA}. Continuing the multibody model, homogeneous transformation matrix representing the DVA can be expressed as

$$\mathbf{T}_{\mathrm{Base,DVA}} = \mathbf{T}_{\mathrm{Body\ 1}} \cdot \mathbf{T}_{\mathrm{rotx}}(q_{\mathrm{DVA}}), \quad (3.85)$$

which indicates that it is located at the centre of mass of Body 1, namely the robot shoulder. The inclusion of the DVA does not modify the form of the equations of motion which are still obtainable from the d'Alembert's principle. Their matrix form is also the same such as

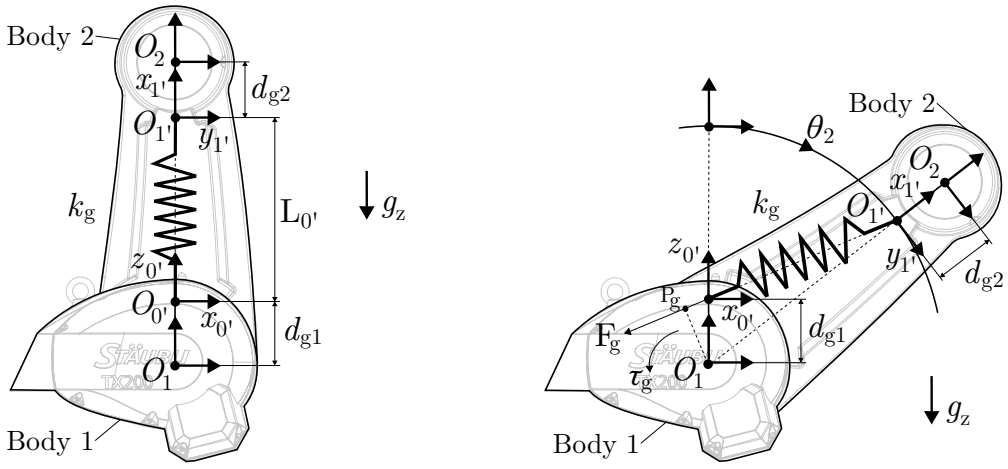
$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{Kq} + \mathbf{F}_{\mathrm{v}}\dot{\mathbf{q}} + \mathbf{F}_{\mathrm{s}}\mathrm{sgn}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\Gamma}, \quad (3.86)$$

given from the joint side with $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{40\times40}$, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{40\times40}$, $\mathbf{K} \in \mathbb{R}^{40\times40}$ and $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^{40\times1}$ which denote the system mass, damping, stiffness and gravity matrices, respectively. Diagonal matrices $\mathbf{F}_{\mathrm{v}}$ and $\mathbf{F}_{\mathrm{s}} \in \mathbb{R}^{40\times40}$ only account for the viscous

and static frictions originating from the motors and the transmissions. On the right-hand side, vector $\mathbf{\Gamma} \in \mathbb{R}^{40\times1}$ is the vector of applied torques. Vector of degrees of freedom $\mathbf{q} \in \mathbb{R}^{40\times1}$ is partitioned such that $\mathbf{q} = [\mathbf{q}_\mathrm{s}\ \mathbf{q}_z\ \mathbf{q}_{x|y}\ \mathbf{q}_\mathrm{ground}\ \mathbf{q}_\mathrm{f}\ q_\mathrm{DVA}]^T = [q_1 \dots q_6\ q_{z,1} \dots q_{z,6}\ q_{x,1}q_{y,1} \dots q_{x,6}q_{y,6}\ q_{\mathrm{g}_x}q_{\mathrm{g}_y}q_{\mathrm{g}_z}\ q_{1.1,x} \dots q_{2.2,\mathrm{Rz}}\ q_\mathrm{DVA}]^T$. One more row and one more column are added to matrices $\mathbf{K}$ and $\mathbf{D}$ to account for the DVA damping and stiffness. Neglecting friction components, the equations of motion presented in Eq. 3.86 are eventually used to simulate robotic machining operations.

3.6 Gravity compensator modelling

The Stäubli TX200 robot comprises a gravity compensation spring located inside the robot arm. Its role is to reduce the load of the motor of the second joint that supports the cantilevered arm and the payload which are subjected to gravity. It means that the linear spring generates a torque balancing the gravity effect on the second motor. The linear gravity spring, of stiffness k_g and with its preload force $\mathrm{F}_\mathrm{Preload}$, is mounted between the second joint and the third joint of the robot. As depicted in Figure 3.21, the lower end of the spring is fixed at the shoulder (Body 1) and the upper end is clamped inside the robot arm (Body 2). In order to produce the compensating torque to balance the gravity effects, the lower end of the spring is eccentric with respect to the location of the second joint O_1. The lower end point is seen as a secondary frame belonging to the robot shoulder and is denoted by frame $O_{0'}$ in Figure 3.21a. The orientation of frame $O_{0'}$ is the same as joint frame O_0 defined with the robot schematic of the dynamic model (Figure 3.11). The distance between frames O_1 and $O_{0'}$ is $d_{\mathrm{g}1}$. The upper end point of the spring is located by frame $O_{1'}$ and follows the orientation of joint frame O_1. The distance between frames O_2 and $O_{1'}$ is $d_{\mathrm{g}2}$. As a result, the spring deforms between frames $O_{0'}$ and $O_{1'}$ as second joint θ_2 moves (Figure 3.21b).



(a) Gravity spring at rest (b) Elongated gravity spring

Figure 3.21: Gravity compensator of the Stäubli TX200 robot

The torque generated by the gravity compensation spring τ_g can be expressed in a

general way such as

$$\tau_{\mathrm{g}} = \mathrm{F_g}\ |O_1\mathrm{P_g}|(\theta_2), \tag{3.87}$$

with $|O_1\mathrm{P_g}|(\theta_2)$ the lever arm of linear force $\mathrm{F_g}$ which is a non-linear function of angle θ_2. The linear force generated by the gravity compensation spring reads

$$\mathrm{F_g} = k_{\mathrm{g}}\ |O_{0'}O_{1'}|(\theta_2) + \mathrm{F_{Preload}}, \tag{3.88}$$

with $|O_{0'}O_{1'}|(\theta_2)$ another non-linear function of angle θ_2. Knowing distances d_{g1} and d_{g2} locating the anchor points of the linear spring, non-linear functions $|O_1\mathrm{P_g}|$ and $|O_{0'}O_{1'}|$ can be determined as a function of angle θ_2 using geometric constructions.

Using homogeneous transformation matrices, generic expressions of the forces and torques applied on bodies and generated by linear springs can be derived. In the case of the gravity compensation spring of the Stäubli TX200 robot, the locations of the two anchor points of the spring are expressed with position vectors $\mathbf{p}_{d_{\mathrm{g1}}}$=[0 0 d_{g1}]^T and $\mathbf{p}_{d_{\mathrm{g2}}}$=[$d_{\mathrm{g2}}$ 0 0]^T in frames O_0 and O_1, respectively. Hence, the varying length between the anchor points is expressed such as

$$d_{O_{0'}O_{1'}} = \left\| \left(\mathbf{R}_{\text{Body 2}}\mathbf{p}_{d_{\mathrm{g2}}} + \mathbf{p}_{\text{Body 2}}\right) - \left(\mathbf{R}_{\text{Body 1}}\mathbf{p}_{d_{\mathrm{g1}}} + \mathbf{p}_{\text{Body 1}}\right) \right\|, \tag{3.89}$$

with $\mathbf{R}_{\text{Body 1}}$ and $\mathbf{R}_{\text{Body 2}}$ the rotation matrices associated to Body 1 and Body 2 relative to the base frame, respectively. Under the effect of the preload force, the natural length of the linear spring can be computed as

$$\mathrm{L_{0'}} = \mathrm{L_0} - \frac{\mathrm{F_{Preload}}}{k_{\mathrm{g}}}, \tag{3.90}$$

with $\mathrm{L_0}$, the natural length of the linear spring. In the vertical configuration of the arm depicted in Figure 3.21a, the linear spring is already subjected to the preload force. Finally, the forces and torques applied on Body 1 are expressed as follows

$$\{\boldsymbol{\mathcal{R}}_{\text{Body 1}}\}_{\text{Base}} = k_{\mathrm{g}}\left(d_{O_{0'}O_{1'}} - \mathrm{L_{0'}}\right)\mathbf{u}_{O_{0'}O_{1'}}, \tag{3.91}$$

$$\{\boldsymbol{\mathcal{M}}_{\text{Body 1}}\}_{\text{Base}} = \left(\mathbf{R}_{\text{Body 1}}\mathbf{p}_{d_{\mathrm{g1}}}\right) \times \{\boldsymbol{\mathcal{R}}_{\text{Body 1}}\}_{\text{Base}}, \tag{3.92}$$

with $\mathbf{u}_{O_{0'}O_{1'}}$ the unit vector giving the direction of the linear spring force computed as $\mathbf{u}_a = \frac{\mathbf{a}}{\|\mathbf{a}\|}$ with arbitrary vector $\mathbf{a}$ (computed from Eq. 3.89). Reaction forces and torques with equal amplitudes but opposite directions are applied on Body 2. Damping can also be added using similar derivations.

3.7 Actuator modelling

Once the mechanical model is completed, actuators are introduced to drive the robot on the desired trajectory. The Stäubli TX200 robot actuates its joints through electric servo motors. This is the case of the vast majority of industrial robots, though hydraulic servo motors are used for large size manipulators. Using the robot dynamic model developed from Sections 3.2 to 3.6, current TCP position $\mathbf{x}_{\mathrm{TCP,c}}$ is known from the instantaneous degrees of freedom $\mathbf{q}$. In Figure 3.22, the developed model of the robot is represented by a block whose output is the current TCP position (Robot dynamics block). Including the gearbox block with diagonal matrix $\mathbf{k}$ of the reduction ratios, the robot model is actuated through torque $\boldsymbol{\tau}_{\mathrm{m}}$ in an open-loop fashion i.e. no feedback loop is inserted. Torques are directly applied on the bodies representing the rotors from the motor side (forward dynamic scheme) such as

$$\{\boldsymbol{\mathcal{M}}_{\mathrm{Rotor_i}}\}_{\mathrm{Base}} = \tau_{\mathrm{m,i}}\mathbf{R}_{\mathrm{Base,Rotor_i}}\mathbf{u}_z, \tag{3.93}$$

with $\tau_{\mathrm{m,i}}$ the scalar motor torque associated with joint i and $\mathbf{R}_{\mathrm{Base,Rotor_i}}$ the rotation matrix of rotor i with respect to the base frame.

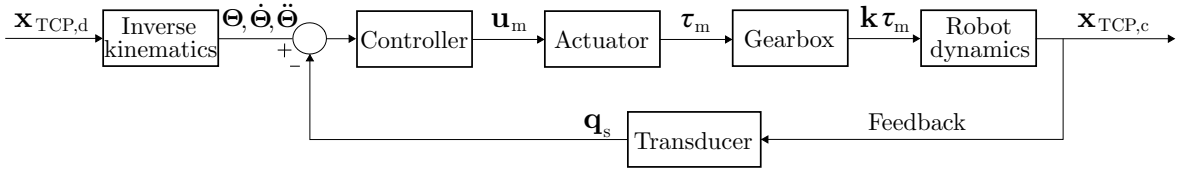


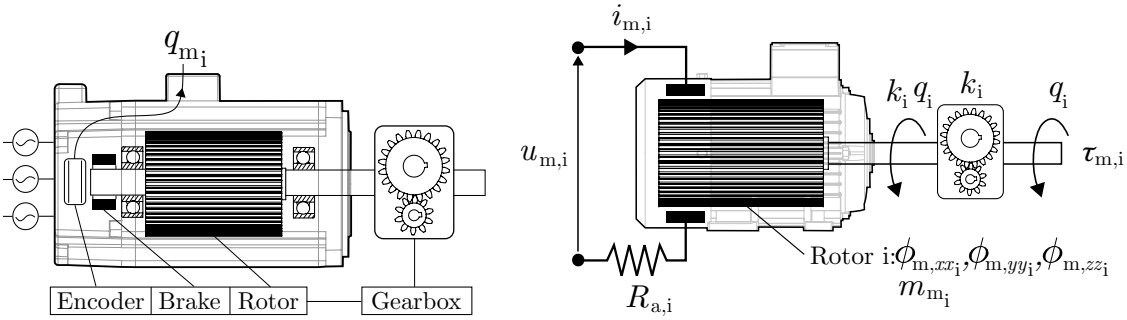
Figure 3.22: Closed-loop robot model

Of course, in order to follow a desired trajectory $\mathbf{x}_{\mathrm{TCP,d}}$, the system must be able to correct the current TCP position from deviation error or disturbance. It is achieved by feeding back the current TCP position to the controller block via a transducer, and thus closing the loop. The role of the controller unit is to compute command input u_{m} for each of the actuators[4]. Command input $\mathbf{u}_{\mathrm{m}}$ is determined on the basis on the difference between the actual rotor positions $\mathbf{q}_{\mathrm{s}}$ and the desired joint angles $\boldsymbol{\Theta}$. The desired joint positions $\boldsymbol{\Theta}$ are usually obtained using the inverse kinematics from desired trajectory $\mathbf{x}_{\mathrm{TCP,d}}$. Differential kinematics can also be used to derive the desired joint velocities $\dot{\boldsymbol{\Theta}}$ and accelerations $\ddot{\boldsymbol{\Theta}}$ in order to include a feedforward action. Finally, command input $\mathbf{u}_{\mathrm{m}}$ is supplied to the actuator block that converts it into torque $\boldsymbol{\tau}_{\mathrm{m}}$ transmitted to the robot links via the gearbox.

The energy conversion between command input $\mathbf{u}_{\mathrm{m}}$ (generally a voltage or a current) into torque $\boldsymbol{\tau}_{\mathrm{m}}$ takes place in the actuator block. The actual servo motors are three phase AC servo motors comprising a built-in encoder, a brake and a rotor (Figure 3.23a). AC servo motors usually refer to three phase synchronous motor whose rotor field is excited by

[4]Signals generated by controllers are generally of low amplitudes and must be amplified through amplifiers. The latter are not considered in the model and actuators are directly powered via command input $\mathbf{u}_{\mathrm{m}}$.

permanent magnets. In a permanent magnet synchronous motor, the rotor is constituted of permanent magnets and is surrounded by three equally spaced fixed stator windings. When the motor is powered, the windings produce a rotating magnetic field which induces an attraction and repulsion mechanism with the rotor magnets. By controlling the current flowing in the three windings, a magnetic field of arbitrary direction and magnitude can be produced by the stator to control the electromagnetic torque. Various control strategies exist among which is the notable field oriented control. Simulation of the field oriented control applied to an AC permanent magnet synchronous motor is given in Appendix D. As the strategies to control permanent magnet synchronous motor are rather complex, brushless DC motor were integrated in the robot modelling.



(a) AC three phase servo motor (b) Model of a DC motor

Figure 3.23: Actuator modelling

Brushless DC and permanent magnet synchronous AC motors share the same architecture consisting of permanent magnets on the rotor and windings on the stator. The AC servo motors can thus be replaced by equivalent DC motors except that the control strategy is simpler. As a matter of fact, the computed command input $\mathbf{u}_m$ can directly be applied to the motor terminals i.e. there is no need to include an internal current loop (Figure 3.23b). In general, the torque generated by motor i can be expressed as

$$\tau_{m,i} = k_{t,i}\ i_{m,i}, \tag{3.94}$$

with $k_{t,i}$ the torque constant of motor i and $i_{m,i}$ the current flowing through its armature circuit. If a DC motor is assumed, the current associated with motor i derives from the second Kirchhoff's law as

$$u_{m,i} = L_{m,i}\ \frac{di}{dt} + R_{a,i}\ i_{m,i} + k_{v,i}\ \dot{q}_{m_i}, \tag{3.95}$$

with $R_{a,i}$ the armature resistance, $k_{v,i}$ the back electromotive constant and $L_{m,i}$ the armature inductance. The motor constants ($k_{t,i}$, $k_{v,i}$ and $R_{a,i}$) are assumed to be fixed, while in reality they may vary as a function of the angular velocity. Assuming that electrical phenomena take place at a faster regime than the mechanical motions, the inductance effects are neglected. Hence, current $i_{m,i}$ flowing in motor i is expressed as follows

$$i_{m,i} = \frac{u_{m,i} - k_{v,i}\dot{q}_{m_i}}{R_{a,i}}. \tag{3.96}$$

By substituting Eq. 3.96 in Eq. 3.94, motor torque $\tau_{\mathrm{m,i}}$ of motor i can be computed as a function of the command input $\mathbf{u}_{\mathrm{m}}$ minus the contribution of the back electromotive (Back EMF) force which provides damping to the motor (term $-k_{\mathrm{v,i}}\dot{q}_{\mathrm{m_i}}$ in Eq. 3.96).

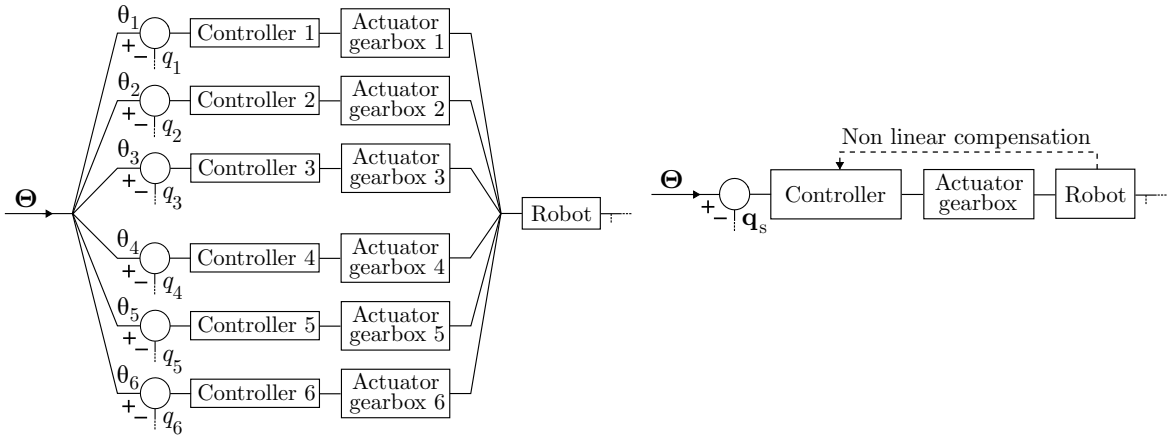
3.8 Motion control

Referring to the closed-loop robot model in Figure 3.22, only the controller block remains to be developed to complete the robot dynamic model. As mentioned earlier, the position controller generates command input $\mathbf{u}_{\mathrm{m}}$, which is the input to the actuators, based on the error between the actual positions of the rotors $\mathbf{q}_{\mathrm{s}}$ and the desired joint angles $\boldsymbol{\Theta}$ ($\dot{\boldsymbol{\Theta}}$ and $\ddot{\boldsymbol{\Theta}}$). In other words, the control problem can be summarised by determining which torques need to be developed by the actuators in order to achieve the desired motion. Here, it is assumed that the imposed trajectory of the TCP is known in terms of position $\mathbf{x}_{\mathrm{TCP,d}}$, but also at the velocity $\dot{\mathbf{x}}_{\mathrm{TCP,d}}$ and acceleration $\ddot{\mathbf{x}}_{\mathrm{TCP,d}}$ levels in order to supply the complete desired kinematics to the controller. This section is focussed on the *motion control* i.e. the controller does not react to force or torque applied at the TCP, as opposed to *force control*.

The task assigned to the TCP, in this case a milling trajectory, is defined in the operational space with respect to base frame O_{Base}. This fact naturally leads to the consideration of two types of general control schemes, namely the *joint space control* scheme and the *operational space control* scheme. In joint space control, the error between the desired and actual positions is computed at the joint level. It means that the desired motion in operational space is converted into joint variables outside the control loop as presented in Figure 3.22. The drawback is that the TCP position $\mathbf{x}_{\mathrm{TCP}}$ is only reflected to the controller via the encoders. There is no feedback on the TCP position which is thus controlled in an open-loop fashion i.e. unless a secondary encoder is mounted on the joint side, flexibility arising at the joints cannot be compensated from the robot controller. On the other hand, operational space control tackles the issue by directly including the inverse kinematics block inside the control loop. A feedback is therefore achieved on the TCP pose $\mathbf{x}_{\mathrm{TCP}}$ to compensate for uncertainties on the structure such as lack of calibration, gear backlash and joint or link elasticity. The price to pay is the measurement of the TCP pose, usually by expensive laser tracker. Without such equipment, only joint space control strategies are available. Joint space controller is therefore addressed in this work.

It is worth noticing that joint space controller can also be partitioned into two categories: the *decentralised* and *centralised* structures. In decentralised structure, each joint can be controlled independently from the others i.e. no coupling between the motors or the links is accounted. This type of structure is suitable for manipulators with high gear ratios. The high gear ratios tend to linearise the system dynamics by reducing the non-linearities such as the configuration dependent inertia and the effects related to Coriolis and centrifugal forces. Therefore, each joint can be controlled separately (Figure 3.24a). In addition, the decentralised structure is robust to parameter variations of the manipulator model

such as the mass and link inertias i.e. a rough knowledge of the manipulator dynamic parameters is enough to design a good controller.



(a) Decentralised structure (b) Centralised structure

Figure 3.24: Typical controller architectures

On the other hand, centralised structures are preferred in the case of low gear ratios and if high speed or acceleration are required. As non-linear effects and coupling become significant, inverse dynamics techniques are necessary to compensate them. Most of the time, a compensation of the main terms (diagonal elements) of the inertia matrix is sufficient. Further and if computational time allows it, a full compensation of all the terms of the inertia matrix, Coriolis effects and gravity forces can be attempted for better tracking performance. However, the centralised structure requires a very good knowledge of the robot dynamic parameters which can turn to be difficult to estimate accurately. Moreover, variations in the dynamic parameters can influence the performance of the centralised scheme unless adaptive features are integrated (Figure 3.24b) [3].

Without any information on the control scheme running on the Stäubli TX200 robot, one decentralised and one centralised controller were implemented: the independent joint control with decentralised feedforward action and the joint space inverse dynamics control. Note that their implementation was made in discrete time.

3.8.1 Independent joint control

As it belongs to the decentralised structures, the robot controller is formed by n_θ independent systems that control each joint axis in a *single-input/single-output* fashion. The control scheme that corrects the motion of axis i is presented in Figure 3.25. Thus, the same controller scheme is repeated for all the joints but may involve different gain values. The presented controller is an enhanced version of the one found in [3] as it includes a dynamic gravity compensation and damping via gain $K_{\mathrm{D,i}}$. Since each motor is controlled individually, all variables are scalar and seen from the motor side, hence the presence of gear ratio k_i. The control scheme is based on a PI controller $C(s) = K_{\mathrm{P,i}}\frac{1+sT_{\mathrm{P,i}}}{s}$ (with s the Laplace's variable): the proportional action, through gain $K_{\mathrm{P,i}}$, allows reducing the error between desired joint angle $k_\mathrm{i}\theta_\mathrm{i}$ and actual joint angle $q_{\mathrm{m_i}}$ while the integral term

$\frac{1}{s}$ cancels the effect of the gravitational component in steady state. The presence of the real zero at $s = \frac{-1}{T_{\mathrm{P,i}}}$ offers a stabilising action. Note that the transducer gains are set to unity.

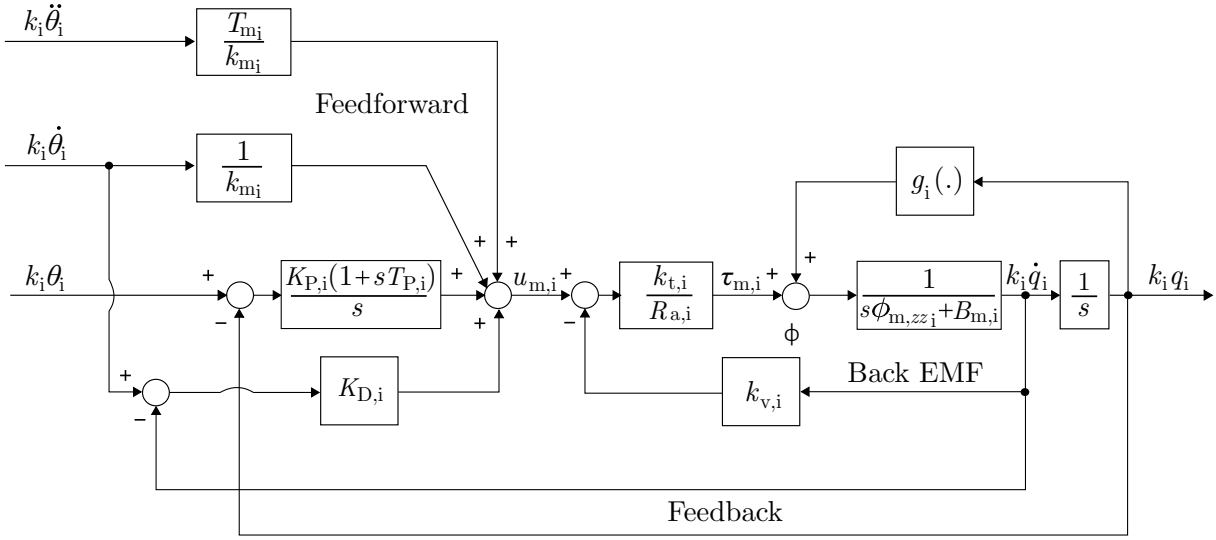


Figure 3.25: Independent joint control with feedforward action

Since a feedforward action is included to balance the motor inertia, the control scheme requires the knowledge of all the desired kinematics variables as inputs (θ, $\dot{\theta}$ and $\ddot{\theta}$) from the motor side. Feedforward gains are computed using the following relationships

$$k_{\mathrm{m_i}} = \frac{1}{k_{\mathrm{v,i}}} \qquad \text{and} \qquad T_{\mathrm{m_i}} = \frac{R_{\mathrm{a,i}}\phi_{\mathrm{m},zz_{\mathrm{i}}}}{k_{\mathrm{v,i}}k_{\mathrm{t,i}}}, \tag{3.97}$$

with $k_{\mathrm{m_i}}$ the inverse of the back electromotive constant and $T_{\mathrm{m_i}}$ the time constant of the motor. Feedback actions allow computing the errors at position and velocity levels from the motor side. Control actions are then summed together with the feedforward actions to generate command input $u_{\mathrm{m,i}}$ which can be considered as a voltage. After having removed the effects of the back electromotive forces ($k_{\mathrm{v,i}}\dot{q}_{\mathrm{i}}$) and accounted for resistance $R_{\mathrm{a,i}}$, the command input is converted into torque $\tau_{\mathrm{m,i}}$ via torque constant $k_{\mathrm{t,i}}$. Gravity compensation $g_{\mathrm{i}}(\cdot)$, which depends on the manipulator configuration, is finally applied before reaching the motor modelled with transfer function $\frac{1}{s\phi_{\mathrm{m},zz_{\mathrm{i}}}+B_{\mathrm{m,i}}}$, with B_{i} a mechanical damping coefficient (not included in this work). Torques for gravity compensation are computed using Eq. 3.40. Integration of the motor model allows computing instantaneous motor position $q_{\mathrm{m_i}}$ and velocity $\dot{q}_{\mathrm{m_i}}$ which are fed back to the controller.

If a root locus analysis is performed on the gain of the position loop which reads

$$K_{\mathrm{Position_i}} = \frac{k_{\mathrm{m_i}}\, K_{\mathrm{P,i}}\, T_{\mathrm{P,i}}}{T_{\mathrm{m_i}}}, \tag{3.98}$$

it can be observed that the real part of the poles becomes negative if $T_{\mathrm{P,i}} > T_{\mathrm{m_i}}$. Besides being stabilised, the system has a faster time response if $T_{\mathrm{P,i}} \gg T_{\mathrm{m_i}}$.

Since the implementation of the joint independent control was achieved in discrete time, PI controller $C(s)$ was discretised into C^k, with k the number of the discrete instant. Using the backward discretisation, C^k is computed as

$$C^k = C^{k-1} + K_{\mathrm{P,i}} \left(k_{\mathrm{i}} \tilde{q}_{\mathrm{i}}^{\,k} (T_{\mathrm{s}} + T_{\mathrm{P,i}}) - k_{\mathrm{i}} \tilde{q}_{\mathrm{i}}^{\,k-1} T_{\mathrm{s}} \right), \tag{3.99}$$

with $\tilde{q}$ the position error and T_{s} the sampling period.

3.8.2 Inverse dynamics control

The second implemented controller pertains to the centralised structure, namely the joint space inverse dynamics controller. Accounting for the non-linear and coupling effects between the joints, the robot controller is no longer formed by n_θ independent systems but it is a multivariable system with n_θ inputs (joint position errors) and n_θ outputs (vector of command input $\mathbf{u}$). The control scheme of the inverse dynamics controller is presented from the joint side in Figure 3.26 [3]. As it can be observed, all quantities were either turned into vector or matrix to take into account the coupling effects between variables. Unlike the computed torque controller, the non-linear compensation is computed on the basis of the actual rotor positions $\mathbf{q_s}$ and $\dot{\mathbf{q}}_\mathbf{s}$.

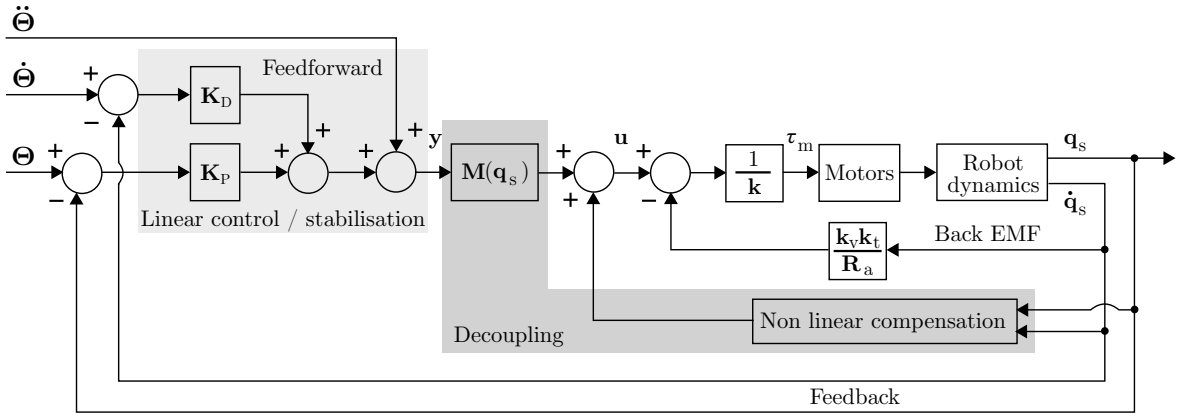


Figure 3.26: Joint space inverse dynamics control

The idea behind such a controller is to find a control vector $\mathbf{u}$ able to achieve an *exact linearisation* of the system dynamics thanks to a non-linear state feedback. In fact, the equations of motion of a n_θ-joint manipulator (Eq. 3.35) can be rewritten in a linear form with respect to command input $\mathbf{u}$, which has the dimension of a torque, such as

$$\mathbf{M}(\mathbf{q_s})\mathbf{y} + \mathbf{n}(\mathbf{q_s}, \dot{\mathbf{q}}_\mathbf{s}) = \mathbf{u}, \tag{3.100}$$

where $\mathbf{n}(\mathbf{q_s}, \dot{\mathbf{q}}_\mathbf{s})$ is the non-linear state feedback which is generally computed as

$$\mathbf{n}(\mathbf{q_s}, \dot{\mathbf{q}}_\mathbf{s}) = \mathbf{C}(\mathbf{q_s}, \dot{\mathbf{q}}_\mathbf{s})\dot{\mathbf{q}}_\mathbf{s} + \mathbf{F}_\mathrm{v}\dot{\mathbf{q}}_\mathbf{s} + \mathbf{F}_\mathrm{s}\mathrm{sgn}(\dot{\mathbf{q}}_\mathbf{s}) + \mathbf{g}(\mathbf{q_s}), \tag{3.101}$$

compensating for the Coriolis effects, friction and gravity. Excluding friction, the compensation can be computed thanks to equations 3.38 to 3.40. In Eq. 3.100, $\mathbf{y}$

represents a new input vector whose expression must be computed to achieve the exact linearisation ($\ddot{\mathbf{q}}_\mathrm{s}$=$\mathbf{y}$). In the same equation, it is recognised that the system under control is linear and decoupled with respect to $\mathbf{y}$.

By choosing a stabilising control $\mathbf{y}$ such as

$$\mathbf{y} = -\mathbf{K}_\mathrm{P}\mathbf{q}_\mathrm{s} - \mathbf{K}_\mathrm{D}\dot{\mathbf{q}}_\mathrm{s} + \mathbf{r}, \tag{3.102}$$

with $\mathbf{r}$ the vector of desired components ($\mathbf{r}$=$\ddot{\boldsymbol{\Theta}} + \mathbf{K}_\mathrm{D}\dot{\boldsymbol{\Theta}} + \mathbf{K}_\mathrm{P}\boldsymbol{\Theta}$), it leads to a position error dynamics ruled by the following second-order equation

$$\mathbf{0} = \ddot{\tilde{\mathbf{q}}}_\mathrm{s} + \mathbf{K}_\mathrm{D}\dot{\tilde{\mathbf{q}}}_\mathrm{s} + \mathbf{K}_\mathrm{P}\tilde{\mathbf{q}}_\mathrm{s}, \tag{3.103}$$

with $\tilde{\mathbf{q}}_\mathrm{s}$ the error vector at position level. Choosing $\mathbf{K}_\mathrm{P}$ and $\mathbf{K}_\mathrm{D}$ as diagonal matrices of the type

$$\mathbf{K}_\mathrm{P} = \mathrm{diag}\{\omega_1^2, \ldots, \omega_6^2\} \quad \text{and} \quad \mathbf{K}_\mathrm{D} = \mathrm{diag}\{2\zeta_1\omega_1, \ldots, 2\zeta_6\omega_6\}, \tag{3.104}$$

gives a decoupled system, with ω_{n_θ} and ζ_{n_θ} the selected natural frequency and damping ratio, respectively. The command input $\mathbf{u}$ is finally computed through

$$\mathbf{u} = \mathbf{M}(\mathbf{q}_\mathrm{s})\mathbf{y} + \mathbf{n}(\mathbf{q}_\mathrm{s}, \dot{\mathbf{q}}_\mathrm{s}), \tag{3.105}$$

which has the dimension of a torque, with $\mathbf{M}(\mathbf{q}_\mathrm{s})$ the inertia matrix of the manipulator only depending on the instantaneous degrees of freedom of the rotors from the joint side $\mathbf{q}_\mathrm{s}$ (encoders are only mounted on the motor side). Consequently, such a controller does not compensate for any flexibility originating from the joints nor the links. Vector of motor torques $\boldsymbol{\tau}_\mathrm{m}$ is applied to the motors after subtracting the back electromotive force and reducing the torque through the gear ratios. Notations $\mathbf{k}_\mathrm{v}$, $\mathbf{k}_\mathrm{t}$ and $\mathbf{R}_\mathrm{a}$ stand for diagonal matrices of the back electromotive constants, the torque constants and the armature resistances, respectively.

3.9 Discussion

The dynamic modelling of the Stäubli TX200 robot was covered in this chapter. The considered robot is installed at the Faculty of Engineering of the University of Mons and is dedicated to milling operations. Its kinematic structure encompasses six revolute joints with a spherical wrist which allows an analytical resolution of the inverse kinematics problem. Its equations of motion were progressively built by considering several modelling options. Using the multibody approach, the basic model represents the robot as a succession of rigid links connected by rigid joints. The joints are actuated via motors whose rotors are also treated as bodies. Several sources of flexibility were

then progressively appended to the modelling of the robot, since this work investigates the triggering of milling instabilities coming from the robot compliance. First, the typical flexibility of the transmissions is accounted around each of the motion axes. The joint axial flexibility is modelled by means of torsional springs and viscous dampers. Next, bearing flexibility is included by adding two more torsional springs and dampers orthogonal to each motion axis. Virtual flexibilities are also inserted between the robot base and the steel slab on which the robot is clamped. The flexibility of the links is modelled using the corotational formulation in which the elastic deformation is computed on the basis of an undeformed configuration. Only the contributions of the longest links are considered, i.e. the robot arm and forearm, and are handled using flexible beam elements. Overall, the flexible robot model comprises a total of 40 degrees of freedom of which 6 deal with the rotor positions, 21 account for the joint and ground flexibilities, 12 model the link deflections and the last one is related to the DVA. The gravity compensation mechanism that compensates the gravity effects experienced by the second motor is integrated in the modelling as a linear spring. With the developed mechanical model of the manipulator, actuators are described from their electrical equations in order to actuate the rotors. The robot modelling is eventually completed with the integration of a controller intended to correct the actual position of the TCP from the desired trajectory. Two joint space control schemes are developed, namely the independent joint control and the inverse dynamics control.

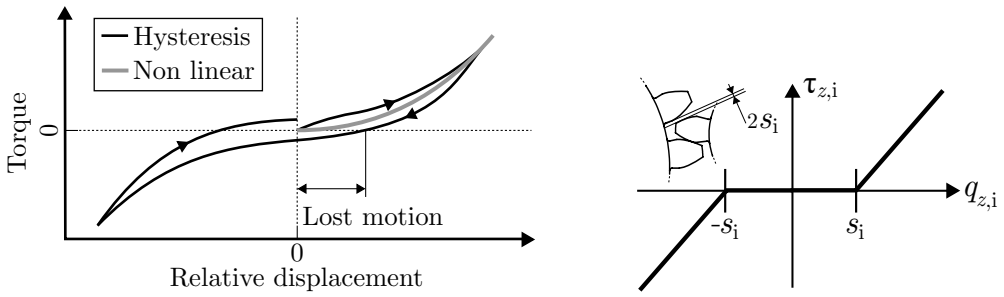
Some comments are worth to be noted regarding the joint model. In this work, joint stiffness and damping are modelled using torsional springs and viscous dampers with linear characteristics. It is obviously an idealisation of the behaviour of actual gearbox mounted on industrial robots.

Figure 3.27 shows an actual gearbox from an industrial robot. Typical robot gearbox comprises two reduction stages: a first reduction stage via spur gears and a second one with cycloidal gears. Reduction ratios over 100 are commonly achieved with such gearbox [131]. Note that they present a hollow centre in order to let the cables powering the motors pass inside the robot structure.

Figure 3.27: RV-E Nabtesco gearbox

The stiffness characteristic of an actual gearbox is not linear and can possibly exhibit a hysteretic behaviour (Figure 3.28a). Measurements of such characteristic are carried out by locking the motion of the gearbox from the motor side and by applying a torque from the load side. Starting from a null torque without any relative displacement, a positive torque is applied increasing the relative displacement. When releasing the torque, there is still a relative displacement for a null torque. This behaviour is termed as hysteresis lost motion i.e. a residual torsion without input/output torques [132]. It is also observed that as torque is increased, relative displacement tends to saturate due to the hardening

effect of the gear teeth in contact. The stiffness characteristic can be approximated by a non-linear curve only accounting for the hardening effect. The latter can be modelled using a polynomial function though several researchers suggest using piecewise linear function [133]. To also consider the hysteretic behaviour of the stiffness curve, some researchers rely on the extended Bouc-Wen differential model which is one of the most widely accepted phenomenological models of hysteresis in mechanics [134]. Note that the Bouc-Wen model was implemented on the first axis of the robot model but did not influence the results of milling simulations significantly. Another issue concerned the model parameters which are difficult to measure (realistic values were chosen). The Bouc-Wen model was therefore left behind.



(a) Stiffness characteristic [132] (b) Backlash modelling [62]

Figure 3.28: Perspectives for joint modelling

Due to the inherent constitution of gearbox with toothed wheels, contacts are intermittent due to some clearance between the teeth i.e. the backlash or lost motion. Its effects arise when the direction of motion is reversed as the clearance must be travelled back by the teeth without generating any motion of the load. Simple backlash model considers that no torque is transmitted within the clearance (Figure 3.28b). Hence, torque $\tau_{z,\mathrm{i}}$ transmitted through joint i, accounting for backlash and only originating from the stiffness around a motion axis (Eq. 3.44), takes the following form

$$\tau_{z,\mathrm{i}} = k_{z,\mathrm{i}} \begin{cases} (0 - (q_{z,\mathrm{i}} - s_\mathrm{i})) & : \text{if } q_{z,\mathrm{i}} \geq s_\mathrm{i} \\ (0 - (q_{z,\mathrm{i}} + s_\mathrm{i})) & : \text{if } q_{z,\mathrm{i}} \leq -s_\mathrm{i} \\ 0 & : \text{else,} \end{cases} \tag{3.106}$$

with $2s_\mathrm{i}$ the radial clearance between two successive teeth. Cycloidal gears are also known to exhibit low backlash. Gearbox catalogues provide information regarding the gap distance between teeth, about 1.0 arc.min. ($2.9\mathrm{e}^{-4}$ rad). Using the described backlash model, milling simulations were carried out without significant influence on the results. Results are presented in [135]. Note that during the simulated milling operations, joints mostly rotated in the same direction. Backlash effects might be prominent for trajectory involving reverse motions. Same comment applies to friction which remains unmodelled for the multibody model of the robot.

Regarding the flexibility included in the developed model, it is worth noticing that the compliances from the force sensor and the spindle were dismissed. Neglecting the

flexibility of the force sensor is a reasonable hypothesis since its referenced stiffnesses are quite high (at least 7.0e^7 N/m linearly and 3.3e^5 Nm/rad torsionally). However, neglecting the flexibility of the spindle fastening is less common. As in conventional machining, Cordes et al. [104] showed that chatter frequencies related to the spindle modes could arise in robotic milling operation. However, as it will be presented in Chapter 5, spindle modes were insignificant compared to the ones from the robot in the measured FRFs. Hence, they were neglected in the model of the Stäubli TX200 robot.

Some last comments are issued on the robot controller. Implemented controllers are also simplified versions of the actual one. Since controller is considered as the core know-how of robot manufacturers, only few information was obtained concerning the one running the Stäubli TX200 robot. Expanding the control scheme found in a Ph.D. thesis dating from 2007 [136] for the Stäubli RX90B robot, it seems that an independent joint controller is implemented (Figure 3.29). Inputs of the controller remain the desired joint angles (θ_i) and their time derivatives ($\dot{\theta}_i$ and $\ddot{\theta}_i$) computed from inverse kinematics. The desired quantities are filtered using a filter of type "Boxcar". A boxcar function is any function which is zero everywhere except for a single interval where its amplitude A is equal to a constant. It prevents the execution of fast motions that could damage the robot mechanics. The desired trajectory is then resampled at a higher frequency using a cubic interpolation. Informational frequency values are set in Figure 3.29 to provide orders of magnitude. Note that the velocity feedback loop reacts twice as fast. A typical decentralised controller corrects the positioning errors with the help of a feedforward action (FF block). Other feedforward actions (G_{ff} block) compensate velocity errors. The velocity loop is built up with three filters: an input filter G_{in}, a velocity feedback filter G_{fb} and an output filter G_{out}. The rotor positions q_{m_i} can be recorded via the motor encoders. Lastly, an Iterative Learning Control (ILC block) is set up to adapt the controller parameters to possible changes in the robot dynamic parameters.

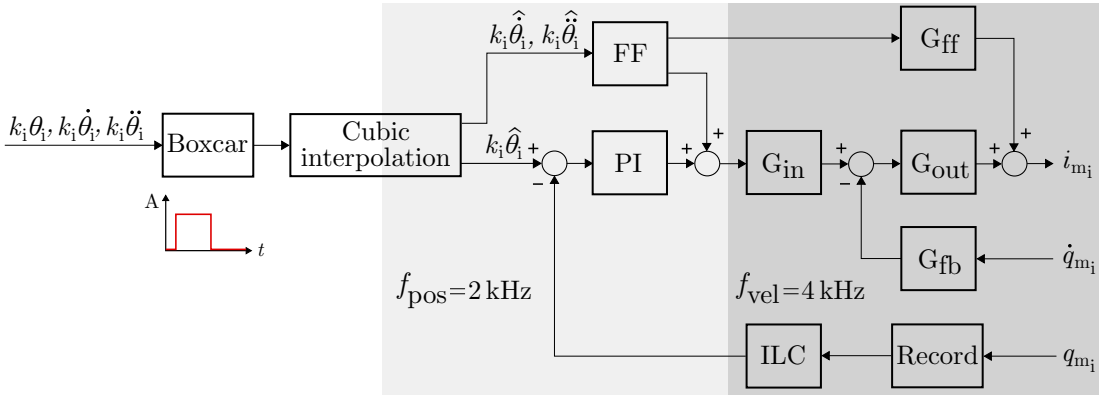


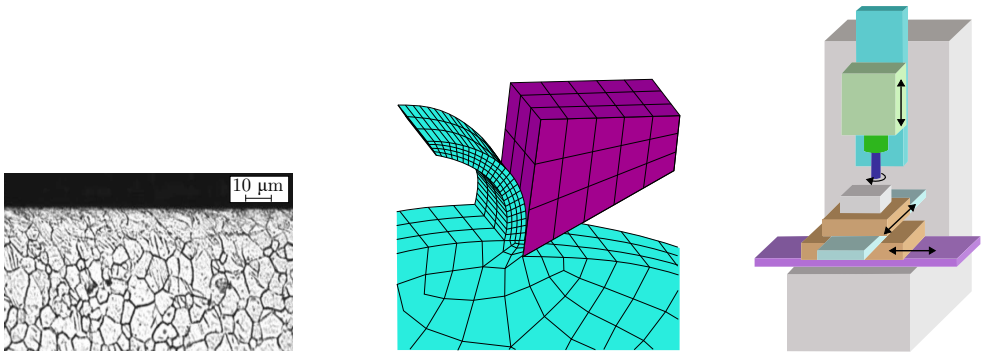
Figure 3.29: Real controller concept

In view of the milling applications, it is clear that a force controller would be more suitable for dealing with applied forces at the TCP. On the basis on a force feedback, an on-line chatter detection and avoidance could be achieved. In this work, only the motion controller operating in free space is considered without considering any interaction with the environment.

Chapter 4
Milling model

Extensive research has been carried out to model milling during the last decades. As a matter of fact, the final shapes of most mechanical parts are still obtained by machining operations. Usually, series of metal-removing operations are conducted after bulk processes such as forging or casting in order to achieve parts with desired shapes, dimensions and surface quality. However, one of the most common problems in machining originates from self-excited vibrations between the tool and the workpiece. Due to chatter vibrations or simply *chatter*, the tool jumps out of the cutting zone or breaks because of the exponentially growing forces between the tool and the workpiece [43]. The productivity of such operations is consequently limited by self-excited vibrations leading to shorter tool life, poor surface finish or damage to the milling system. In order to study and prevent chatter, but also to predict the workpiece-tool-machine interactions and the surface finish, simulation through milling models remains a key value for fast and cheap prototyping.

Cutting processes can be modelled according to different viewpoints depending on the spatial scale used to describe them [137]. Study of cutting forces can be carried out at three different scales: the microscopic, mesoscopic and macroscopic levels (Figure 4.1).



a) Microscopic b) Mesoscopic c) Macroscopic

Figure 4.1: Milling scales

a) Microscopic scale: this scale allows understanding the cutting phenomena by mainly considering the metallurgical characteristics of interactions between the active cutting edge and the machine material. Such modelling scale is concerned with the study of chip formation, phase change, grain dislocation and microstructure. In Figure 4.1a, the microstructure of AZ31 Mg alloy is analysed under optical microscope after a dry machining operation [138].

b) Mesoscopic scale: it is related to the study of interactions between the tool and the workpiece without any metallurgical consideration. Besides the chip formation,

thermal, thermo-mechanical and tribological aspects are considered through heat and material flows. Finite element methods are usually used to relate quantities difficult to measure in practice such as strain rate, deformation or temperature with the cutting forces (Figure 4.1b). Nevertheless, this approach relies on assumptions which are difficult to validate in practice, as for example an orthogonal or oblique cut.

c) Macroscopic scale: the workpiece-tool-machine interactions are globally accounted without any concern of phenomena located at the tool tip (Figure 4.1c). Only the cutting forces are considered as the cause of vibrations between the tool, the workpiece and the milling system. The macroscopic scale is therefore suitable for a global dynamic modelling of the cutting process, in particular to predict self-excited vibrations which cannot be detected at the microscopic and mesoscopic scales.

Since the robot model was carried out as a whole in Chapter 3, the macroscopic representation of the cutting forces was naturally selected. Hence, this chapter is interested in the cutting force modelling. Cutting forces will later be applied as external forces on the developed robot model to predict the stability limits under various milling conditions. Although many milling models were developed throughout the decades, only the most popular ones are considered in this work. After a brief presentation of virtual machining and chatter modelling, three cutting force models are covered, namely the zero-order, the semi-discretisation and the complete cutting force models. While the first one operates in the frequency domain, the last two ones predict milling stability in the time domain and therefore use different representations of the milling machine. Using the presented milling models, stability lobe diagrams will be generated and ultimately compared in Chapter 7 dedicated to stability analysis.

4.1 Virtual machining

The objective of virtual machining is the simulation of machining operations in order to optimise productivity and accuracy and prevent any issues related to the dynamic nature of the process. Virtual machining starts from CAD model and tool path planning to predict all dynamic effects that could happen well before the real operation. In order to perform simulations involving material removal by chip, three essential modules are needed, namely the workpiece/tool intersection module, the cutting force module and the machine dynamics module (Figure 4.2).

1. Workpiece/tool intersection module: machining simulation requires the knowledge of the tool and workpiece geometry. In milling, there are two categories of cutting tools, also called *end mills*: the solid end mills and the insert end mills. Solid end mills are generally small tools made of high-speed steel or carbide, as the one presented in Figure 4.2. Various geometries exist ranging from the cylindrical end mills to cutters with round corners to carve complex shapes. Parametric models allow defining their general shape through geometric parameters. Larger tools consist of

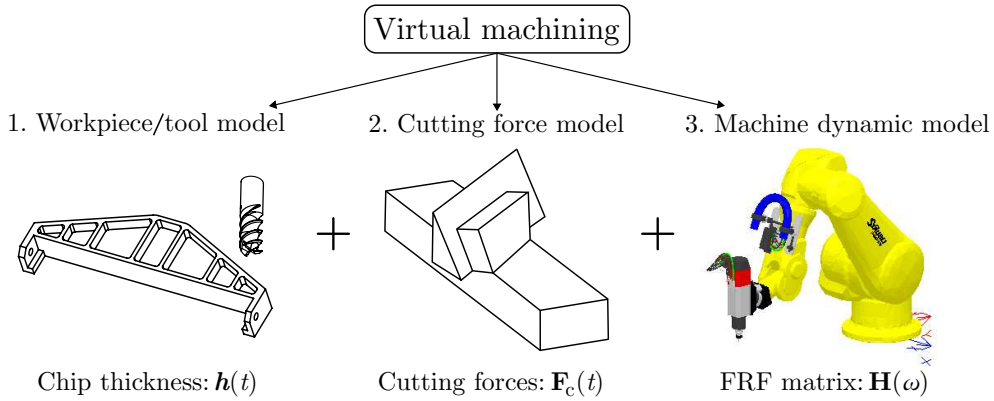


Figure 4.2: Three essential modules for virtual machining

a tool body on which small cutting tips, so-called *inserts*, are attached. Their modelling relies on homogeneous transformations to correctly position and orientate the inserts with respect to a fixed reference frame. On the other hand, modelling the machined surface is critical to estimate the chip thickness on the basis of which cutting forces are computed. Methods to reconstruct the surface from the tool motions exist as well as analytical, 2D and 3D models. The dynamic chip thickness $\boldsymbol{h}(t)$ is finally computed and results from the intersection between the modelled tool and workpiece.

2. Cutting force module: from a macroscopic viewpoint, it determines the cutting forces $\mathbf{F}_c(t)$ applied on the mechanical system from the dynamic chip thickness. The general equation deriving the cutting forces $\mathbf{F}_c(t)$ can be formulated as

$$\mathbf{F}_c(t) = \mathbf{K}_c a_p \boldsymbol{h}(t), \tag{4.1}$$

with $\mathbf{K}_c$ a matrix gathering the so-called *cutting force coefficients* for a dedicated machining operation, a_p the axial depth of cut (equivalent to the width of cut in turning) and $\boldsymbol{h}(t)$ the dynamic chip thickness. There are mainly two approaches to determine the coefficients of matrix $\mathbf{K}_c$ collecting the constant cutting force coefficients: the fundamental approach and the mechanistic approach. Both approaches are described below:

- The fundamental approach relies on a physical representation of the phenomena. The basic model derives from the theory of orthogonal cutting. In orthogonal cutting, the material is removed by a cutting edge that is perpendicular to the direction of relative tool–workpiece motion. In this model, cutting forces are derived by considering that the material is sheared and rubs on the tool faces. Assuming that the cutting forces are proportional to the dynamic chip thickness, the two quantities are related using the cutting force coefficients gathered in matrix $\mathbf{K}_c$. Using the fundamental approach, cutting force coefficients derive from relationships involving the shear stress, the shear angle and friction coefficients, which are hardly measurable in practice. Three-dimensional and geometrically complex cutting operations are ultimately handled using the orthogonal-oblique transformation [139].

- Orthogonal cutting, possibly requiring the oblique transformation, is however not directly applicable to many practical cutting tools with chip breakers or unconventional edge geometry. In addition, actual phenomena are difficult to characterise and measure. Hence, the mechanistic approach proposes a simplified modelling in which cutting force coefficients $\mathbf{K}_c$ are experimentally determined for the considered operation and are thus function of the tool-material couple. Although non-linear and exponential cutting force models exist, the linear model presented in Eq. 4.1 is the most frequently used. Again, cutting forces are expressed with a proportional relationship relative to the chip thickness. Following the same philosophy as the selected macroscopic approach, the mechanistic approach is naturally considered in this work.

Figure 4.3 presents a general view of a milling operation with a_e the radial depth of cut i.e. the radial engagement of the tool in the workpiece, Ω the spindle speed (usually expressed in rev/min) and the direction of the feed motion.

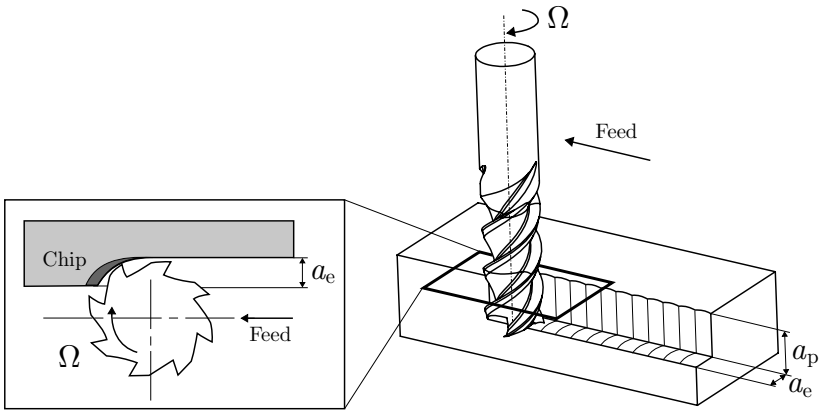


Figure 4.3: General view of a milling operation

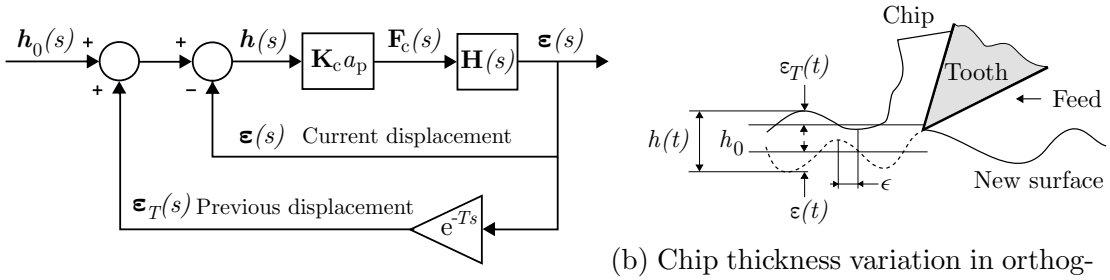
3. Machine dynamic module: the dynamic behaviour of the milling machine can be reduced to a transfer function matrix $\mathbf{H}(\omega)$ at the tool tip (i.e. TCP), often expressed in the receptance format. In particular, since cutting forces are applied at the tool tip, the transfer function matrix represents the dynamic behaviour of this particular point belonging to the milling machine subjected to external forces. In this work, the milling machine is the Stäubli TX200 robot.

4.2 Chatter mechanism in turning

Milling machines experience both forced and self-excited vibrations during machining operations. In milling, cutting forces are periodic and can exhibit significant harmonics up to four to five times the tooth or spindle passing frequencies. If any of the harmonics is close to one of the natural frequencies of the machine and/or workpiece structure, chatter vibrations are most likely to occur. Chatter vibrations result from a self-excitation mechanism related to the generation of the chip thickness. In milling, a wavy surface left by a previous tooth is removed during the following tooth period, which also

leaves a wavy surface thus leading to structural vibrations. Depending on the phase shift between the two successive waves in the chip thickness, vibrations may grow exponentially resulting in a poor and wavy surface finish or tool and machine damages. As discussed in Chapter 2, self-excited chatter vibrations can be caused by mode-coupling or regeneration of the chip thickness. To a lesser extent, dynamic instability can also originate from friction between the chip and the end mill, periodic separation of the chip or thermomechanical coupling [140, 141]. While regenerative chatter occurs earlier in machine tool operations, mode-coupling chatter was identified as the dominant source of instability in robotic milling [50]. Mode-coupling chatter occurs when vibrations interfere in at least two directions in the plane of cut. Even though milling models were developed with the prediction of regenerative chatter in mind, it must be pointed out that when the cross-coupling of vibration modes is considered, mode coupling chatter is inherently covered by the stability models.

As usual, chatter mechanisms are first described for turning operations since the feed motion lies within a plane as for orthogonal cutting. Regenerative chatter results from phase differences between the vibration waves left on both sides of the chip, phase shift ϵ. Figure 4.4 illustrates the chatter mechanism with s being the complex variable from the Laplace transform such as $s{=}i\omega$.



(a) General block diagram of chatter

(b) Chip thickness variation in orthogonal cutting

Figure 4.4: Regenerative chatter

If an orthogonal cutting is assumed, quantities presented in Figure 4.4a reduce to scalar values and Figure 4.4b depicts the orthogonal cutting process in turning. The edge of the cutting tool is considered flexible in a direction perpendicular to the feed motion. Initially, the surface of the workpiece is smooth but the tool starts leaving a wavy surface behind because of the vibrations of the tool. When the next pass occurs after one revolution of the workpiece, the surface of the chip has waves on both sides: inside the cut where the tool is cutting and on the outside surface of the cut after delay T. Hence, it results in a dynamic chip thickness which can be expressed as follows

$$h(t) = h_0 - [\varepsilon(t) - \varepsilon(t-T)]\,, \tag{4.2}$$

with h_0 the intended chip thickness, which is equal to the feed per revolution in turning, and $[\varepsilon(t) - \varepsilon(t-T)]$ the dynamic part of the chip thickness produced by vibrations at current time t and the waves left during the previous revolution [142]. Assuming that the

turning machine has one single degree of freedom ε perpendicular to the feed direction in Figure 4.4b and since the intended chip thickness does not contribute to the regeneration mechanism, the equation of motion of the system reads

$$m\ddot{\varepsilon} + c\dot{\varepsilon} + k\varepsilon = F_c(t) = K_c(t)b\left[\varepsilon(t) - \varepsilon(t-T)\right], \tag{4.3}$$

with m, c, k, the system mass, damping and stiffness, respectively, b the width of cut (in a direction normal to the cutting plane illustrated in Figure 4.4b), K_c a cutting coefficient and F_c the cutting force. It is observed that the forcing function depends on the present and past solutions of vibrations. Hence, chatter vibration expression is a delay differential equation.

However, the actual process is much more complex to model since the tool rubs against the wavy surface and can even jump out of the cut. Yet, it is still important to understand chatter stability using linear stability theory through which the explanation is clearer. For a one-degree-of-freedom system in orthogonal cutting, it can be demonstrated from Eq. 4.3 that the chatter governing equations in turning read [41, 45]

$$b_{\text{lim}} = \frac{-1}{2K\,\text{Re}[H(\omega)]} \quad \text{and} \quad \Omega = \frac{f_c 60}{\text{N}_c + \frac{\epsilon}{2\pi}} \text{ rev/min}, \tag{4.4}$$

with K a quantity similar to a cutting force coefficient, $H(\omega)$ one component of frequency response function matrix $\mathbf{H}(\omega)$ representing the dynamic behaviour of the system along the direction of the degree of freedom, b_{lim} the limit width of cut in turning, f_c the chatter frequency and N_c the largest integer such as $0 \leq \epsilon < 2\pi$. In other words, there is an integer number N_c of waves of vibration imprinted on the workpiece surface in one tooth period and $\frac{\epsilon}{2\pi}$ is any additional fraction of a wave. Using the chatter governing equations (Eq. 4.4), frequency response function $H(\omega)$ is the input to determine both the limiting width of the chip b_{lim} and phase shift ϵ leading to spindle speed Ω. Chatter frequency f_c is assumed to be known since $H(\omega)$ is swept for all $\omega{=}2\pi f_c$.

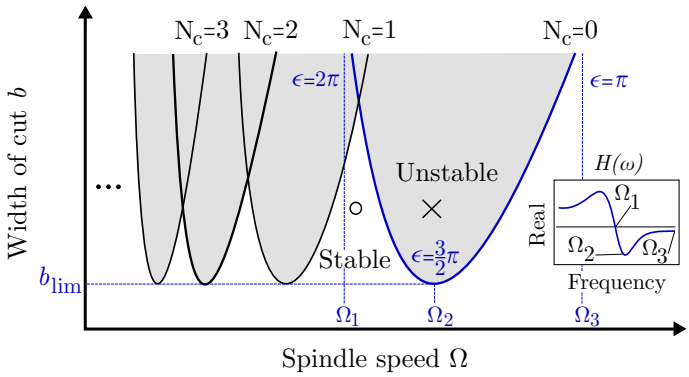


Figure 4.5: Example of stability lobe diagram in turning

An example of stability lobe diagram is provided in Figure 4.5. Stability lobe diagrams showcase a collection of curves, Ω versus b_{lim}, that separate the domain into two regions.

Any cutting condition (Ω, b_{lim}) that is located above the collective boundary exhibits unstable behaviour while any pair below the boundary is assumed to be stable. In fact, the union of the unstable regions determines the overall stability. It is naturally advised to avoid the unstable region in which cutting forces grow exponentially. In contrast, working with cutting condition (Ω, b_{lim}) with large width of cut but still below the boundary increases the productivity of the turning operation.

Stability lobe diagram in turning is built on the basis of Eq. 4.4 and each of the lobes corresponds to an integer number N_c of waves imprinted on the workpiece. In fact, width of cut b controls the stability of the turning operation and the real part of the frequency response function $\text{Re}[H(\omega)]$ defines the limiting width of the chip b_{lim}. In particular, the higher the system dynamics in terms of natural frequencies, the larger the width of cut without instability. Each individual stability curve is actually a mapping of the real part of the frequency response function. From Eq. 4.4, it is understood that the smallest width of cut b_{lim} is produced for the most negative value of $\text{Re}[H(\omega)]$. It corresponds to phase shift $\epsilon=\frac{3}{2}\pi$. On the other hand, the largest width of cut is obtained when phase shift $\epsilon=2\pi$, that is when $\text{Re}[H(\omega)]$ approaches zero. It also means that the chatter frequency f_c is in phase with f_n, the natural frequency of the system. Although Eq. 4.4 suggests an infinite size for the width of cut, it is still finite as lobe order $N_c=1$ cuts the one of order $N_c=0$. A similar situation occurs when phase shift is $\epsilon=\pi$, the out of phase case. The real part of the FRF also approaches zero but the size of the width of cut is still finite as a similar intersection between the lobes, i.e. lobes $N_c=0$ and $N_c=1$ in Figure 4.5, occur. Comparable stability curves are generated from Eq. 4.4 by considering a higher number of waves imprinted on the workpiece $N_c>0$. If lobes of multiple orders are superimposed on the same plot, similar patterns as the ones depicted in Figure 4.5 are obtained. It can be noticed that all lobes share the same minimum width of cut, also called *critical width of cut*, and that the allowable width of cut gets larger as the spindle speed is higher. Of course, using milling machines, more than one degree of freedom is considered, but still the derivation of such stability lobe diagram remains identical. In this case, all the modes of the system contribute to the overall shape of the stability limit. Hence, the matrix form of variables is written in Figure 4.4a depicting the chatter block diagram since modes interact in multiple directions as well. For milling applications, width of cut b is replaced by the axial depth of cut a_p. Fore more insights regarding chatter theory, interested readers are redirected to [142].

4.3 Zero-order method

Now that the basics of chatter theory have been introduced, the concept can be extended to model an actual milling operation. In a milling operation, the end mill usually encompasses several teeth carving the material periodically. As the tool rotates, the chip thickness therefore varies even without self-excited vibrations. Also, in the most general case, the end mill can move along the X, Y and Z directions of the space. Hence, this section discusses the so-called *zero-order* method to generate stability lobe diagrams in milling.

4.3.1 Stability limit in 2D milling

First developed for chatter prediction in machine tool, the machine dynamics is represented with an end mill owning two degrees of freedom: one along the feed direction, chosen as X direction and the other one, in the perpendicular direction, chosen as Y direction (Figure 4.6). The Z direction is not considered as machine tool usually exhibits higher rigidity along the tool rotation axis. The end mill is assumed to have N_z teeth with zero helix angle (straight flutes).

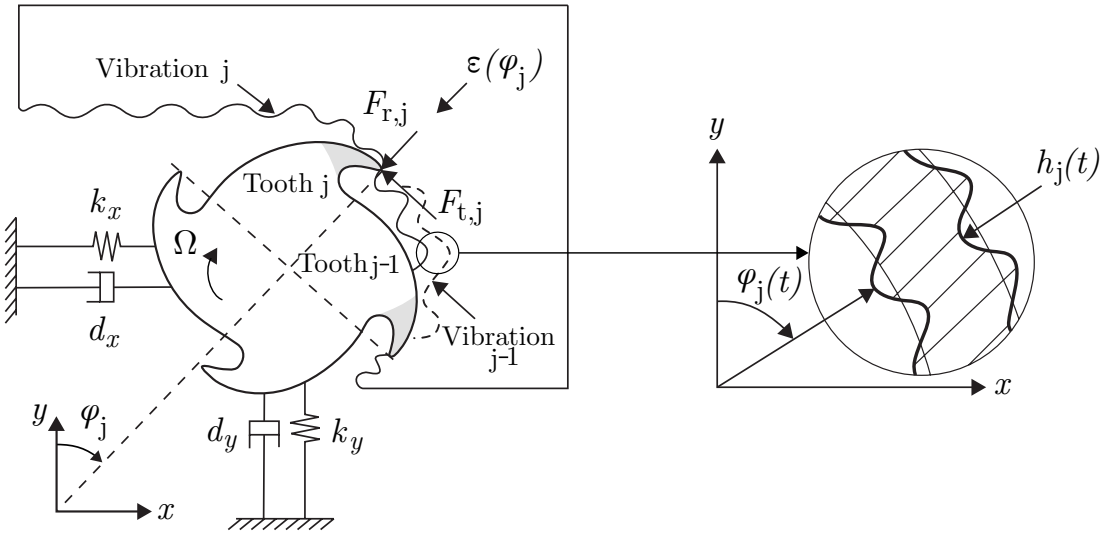


Figure 4.6: Chatter vibrations in 2D milling [142]

In order to express the chip thickness generated by each tooth along one single direction, its size is measured in a radial direction with coordinate transformation

$$\varepsilon_j = -x \sin \varphi_j - y \cos \varphi_j, \tag{4.5}$$

with φ_j the instantaneous angular immersion of tooth j (reference frame determining quantities x and y can be located anywhere since only the chip thickness variation between instants t and $(t-T)$ matters). Having expressed the chip thickness in one direction, the expression of the dynamic chip thickness can be formulated in the same way as the one obtained with the one-degree-of-freedom system (Eq. 4.2) such as

$$h(t) = h_0 \sin \varphi_j(t) - \Big[\underbrace{[-x(t) \sin \varphi_j(t) - y(t) \cos \varphi_j(t)]}_{\varepsilon(\varphi_j)} - \underbrace{[-x(t-T) \sin \varphi_j(t) - y(t-T) \cos \varphi_j(t)]}_{\varepsilon_T(\varphi_j)} \Big], \tag{4.6}$$

with a delay term $\varepsilon_T(\varphi_\mathrm{j})$ depending on angle φ_j which also depends on time t. In other words, the dynamic displacements of the cutter at the previous and present tooth periods can be expressed as

$$h(\varphi_\mathrm{j}) = \left[h_0 \sin\varphi_\mathrm{j} - (\varepsilon(\varphi_\mathrm{j}) - \varepsilon_T(\varphi_\mathrm{j}))\right] g(\varphi_\mathrm{j}), \tag{4.7}$$

with $g(\varphi_\mathrm{j})$ a unit step function that determines whether the tooth is in or out of cut defined as

$$\begin{cases} g(\varphi_\mathrm{j}) = 1 \;\leftarrow\; \varphi_\mathrm{st} < \varphi_\mathrm{j} < \varphi_\mathrm{ex}, \\ g(\varphi_\mathrm{j}) = 0 \;\leftarrow\; \varphi_\mathrm{j} < \varphi_\mathrm{st} \;\text{ or }\; \varphi_\mathrm{j} > \varphi_\mathrm{ex}, \end{cases} \tag{4.8}$$

with φ_st and φ_ex, the start and exit immersion angles, respectively. Similarly to the one-degree-of-freedom system, the intended chip thickness h_0 does not contribute to the regeneration mechanism and is therefore dropped. Hence, the dynamic chip thickness is expressed such as

$$\begin{aligned} h(\varphi_\mathrm{j}) &= \left[\varepsilon(\varphi_\mathrm{j}) - \varepsilon_T(\varphi_\mathrm{j})\right] g(\varphi_\mathrm{j}) \\ &= \left[\Delta x \sin\varphi_\mathrm{j} + \Delta y \cos\varphi_\mathrm{j}\right] g(\varphi_\mathrm{j}), \end{aligned} \tag{4.9}$$

with $\Delta x = x - x_T$ and $\Delta y = y - y_T$, the relative displacements along the X and Y axes, respectively. By projecting and summing the forces applied to all teeth, namely $F_{\mathrm{t,j}}$ and $F_{\mathrm{r,j}}$ representing the tangential and radial cutting forces acting on tooth j, respectively, it can be shown that the cutting forces computed in a fixed frame are proportional to the relative displacements, or *chip load*, such as

$$\underbrace{\begin{bmatrix} F_x \\ F_y \end{bmatrix}}_{\mathbf{F}_\mathrm{c}(t)} = \frac{1}{2} a_\mathrm{p} K_\mathrm{t} \underbrace{\begin{bmatrix} a_{xx} & a_{xy} \\ a_{yx} & a_{yy} \end{bmatrix}}_{\mathbf{A}(t)} \underbrace{\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}}_{\boldsymbol{\Delta}(t)}, \tag{4.10}$$

with a_p the axial depth of cut, $\mathbf{A}(t)$ a square matrix containing the time-varying *directional dynamic milling force coefficients* and K_t the tangential cutting coefficient. Note that in Eq. 4.10, cutting force vector $\mathbf{F}_\mathrm{c}(t)$ only includes the dynamic varying components of the forces since the intended chip thickness h_0 was dropped. From Eq. 4.10 providing the time dependent cutting forces, Altintas and Budak [45] developed the zero-order method to transform the dynamic milling equations into a time invariant, but radial immersion dependent system. The name of the method comes from the fact that the time varying coefficients gathered in matrix $\mathbf{A}(t)$, which depend on the angular orientation of the tool, are expanded into a Fourier series which is truncated to only include the average component i.e. the zero-order component. Recalling that matrix $\mathbf{A}(t)$ can be expanded into a Fourier series such as

$$\mathbf{A}(t) = \sum_{\mathrm{k}=-\infty}^{\infty} \mathbf{A}_\mathrm{k} e^{ir\omega t}, \qquad \mathbf{A}_\mathrm{k} = \frac{1}{T}\int_0^T \mathbf{A}(t) e^{-ir\omega t} dt, \tag{4.11}$$

with ω the tooth passing frequency or tooth period $T = \frac{2\pi}{\omega}$, and r the number of considered harmonics, only the zero-order component $\mathbf{A}_0$ is retained which leads to an approximate expression of the cutting forces such as

$$\mathbf{F}_{\mathrm{c}}(t) \approx \frac{1}{2} a_{\mathrm{p}} K_{\mathrm{t}} \mathbf{A}_0 \mathbf{\Delta}(t). \tag{4.12}$$

Finally, using transfer function matrix $\mathbf{H}(\omega)$ relating the forces to the displacements of the system, authors formulate the eigenvalue equation of the closed loop dynamic milling model such as

$$\mathbf{F}_{\mathrm{c}} e^{i\omega_{\mathrm{c}} t} = \frac{1}{2} a_{\mathrm{p}} K_{\mathrm{t}} \left[1 - e^{-i\omega_{\mathrm{c}} T}\right] \mathbf{A}_0 \mathbf{H}(i\omega_{\mathrm{c}}) \mathbf{F}_{\mathrm{c}} e^{i\omega_{\mathrm{c}} t}, \tag{4.13}$$

with ω_{c} the chatter frequency which is assumed to be known since $H(\omega)$ is swept for all ω_{c}=ω. Solving the eigenvalue problem in Eq. 4.13, the chatter free depth of cut $a_{\mathrm{p_{lim}}}$ can be expressed in a similar form as Eq. 4.4 from the one-degree-of-freedom system such as

$$a_{\mathrm{p_{lim}}} = \frac{-2\pi}{\mathrm{N}_{\mathrm{z}} K_{\mathrm{t}}} \left[\frac{\Lambda_{\mathrm{R}}(1 - \cos \omega_{\mathrm{c}} T) + \Lambda_{\mathrm{I}} \sin \omega_{\mathrm{c}} T}{1 - \cos \omega_{\mathrm{c}} T} + i \frac{\Lambda_{\mathrm{I}}(1 - \cos \omega_{\mathrm{c}} T) - \Lambda_{\mathrm{R}} \sin \omega_{\mathrm{c}} T}{1 - \cos \omega_{\mathrm{c}} T} \right], \tag{4.14}$$

with $\Lambda = \Lambda_{\mathrm{R}} + i\Lambda_{\mathrm{I}}$ the complex eigenvalue. Note that limit axial depth of cut $a_{\mathrm{p_{lim}}}$ is computed such that it is a real number. Complex eigenvalue Λ also leads to the computation of phase shift ϵ. In milling, if ϵ is the phase shift between inner and outer modulations and N_{c} the integer number of full vibration waves or lobes imprinted on the cutting arc, tooth passing periods T at chatter frequency ω_{c} and corresponding spindle speeds Ω are eventually computed as

$$T = \frac{1}{\omega_{\mathrm{c}}}(\epsilon + 2\mathrm{N}_{\mathrm{c}}\pi) \rightarrow \Omega = \frac{60}{\mathrm{N}_{\mathrm{z}} T} \text{ rev/min}. \tag{4.15}$$

Comparable stability lobe diagrams, providing the stability limit as a function of spindle speed Ω and axial depth of cut a_{p} as presented in Figure 4.5, are therefore obtained. In milling, the derived solution in Eq. 4.14 has the same form as the one obtained in turning in Eq. 4.4 with the generation of stability lobes and the appearance of phase shift ϵ. The complete derivation of the zero-order method is provided for 2D milling with and without cross transfer functions in $\mathbf{H}(\omega)$ in Appendix E. Taking into account the cross transfer functions i.e. components H_{ij} in $\mathbf{H}(\omega)$, is of prime importance in robotic milling since mode coupling chatter is the principal source of instability. As glimpsed in Eq. 4.13, the main advantage of the zero-order method is the possibility to directly use the measured frequency response function matrix $\mathbf{H}(\omega)$. The latter is usually measured through modal analysis technique as seen in Chapter 5 dedicated to the robot parameter identification. Therefore, it is needed to measure as many frequency response function matrices $\mathbf{H}(\omega)$ as there are robot configurations to study. Nevertheless, due to its direct analytical solution of critical axial depth of cut and spindle speed and its resolution in the frequency domain, the zero-order method is computationally efficient.

4.3.2 Extension to 3D case

Zero-order method was successfully extended to three-dimensional milling by Altintas [143]. In this situation, the axial component of the cutting forces $F_{\mathrm{a,j}}$ i.e. along the tool revolution axis, is no longer neglected. The purpose of the extension was to cover milling operations involving ball end, bull nose or inclined cutting edges. Besides, the transposition proves to be essential in robotic milling since the assumption that the axial direction is stiffer than the planar ones is no longer valid i.e. deflections at the TCP can appear in any direction (Figure 4.7).

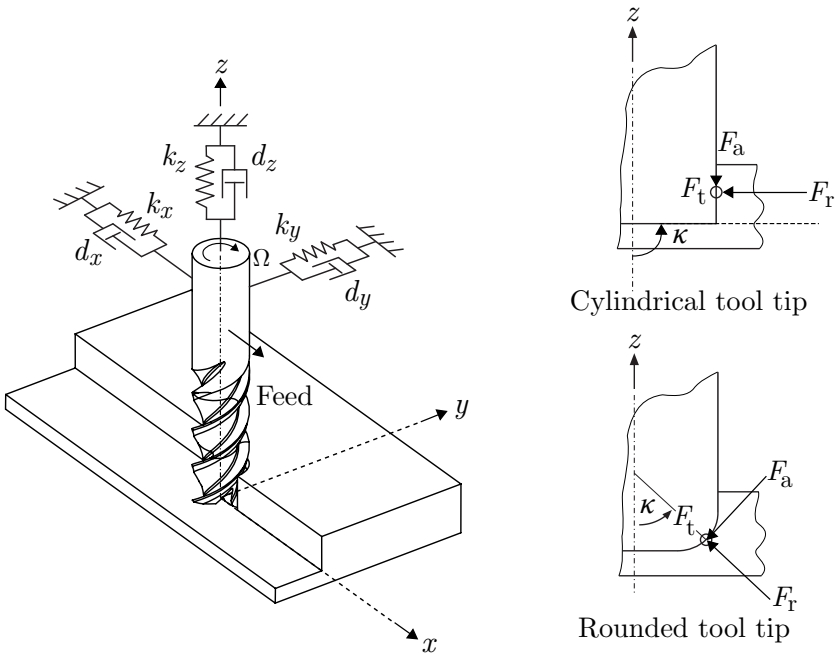


Figure 4.7: Three-dimensional milling

In case of 3D milling, the Cartesian cutting forces directed along the X, Y and Z directions read

$$\underbrace{\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}}_{\mathbf{F}_\mathrm{c}(\mathrm{t})} = \frac{1}{2} a_\mathrm{p} K_\mathrm{t} \underbrace{\begin{bmatrix} a_{xx} & a_{xy} & a_{xz} \\ a_{yx} & a_{yy} & a_{yz} \\ a_{zx} & a_{zy} & a_{zz} \end{bmatrix}}_{\mathbf{A}(\mathrm{t})} \underbrace{\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}}_{\boldsymbol{\Delta}(t)}, \tag{4.16}$$

with $\mathbf{A}(t)$ a (3×3) matrix formed by the directional dynamic milling force coefficients which only account for the translational motions along the X, Y and Z directions (coefficient $\frac{1}{2}$ is sometimes directly included in matrix $\mathbf{A}(t)$). Following the same idea to express matrix $\mathbf{A}(t)$ in Fourier series, only the zero-order component, which is (3×3) matrix $\mathbf{A}_0$, is kept. Components of matrix $\mathbf{A}_0$ are now expressed as a function of two cutting force coefficients, K_r and K_a which are the radial and axial cutting force coefficients, respectively. Furthermore, angle κ accounts for any rounded geometry at the tip of the tool; κ is called the *immersion/elevation angle*. In this work, as only cylindrical

end mills are used, κ=90°. Integrated components of matrix $\mathbf{A}(t)$ calculated from Eq. 4.11 and limited to the zero order are shown in Appendix E.3. Accounting for cutting forces in the three orthogonal directions implies that frequency transfer function matrix $\mathbf{H}(\omega)$ is measured along the X, Y and Z directions; $\mathbf{H}(\omega)$ also being a (3×3) matrix. Following the linear stability theory, it can be demonstrated that the critical axial depth of cut $a_{\mathrm{p_{lim}}}$ is still expressed in the same form as Eq. 4.14.

The presence of unstable poles in the characteristic equation can also be investigated using the Nyquist stability criterion in the frequency domain. Appendix E.4 explains how to derive the critical axial depth of cut $a_{\mathrm{p_{lim}}}$ by taking advantage of the Nyquist contour plot [144]. This alternative resolution is used to compute stability lobe diagrams in 3D milling while the analytical solution is computed in 2D and 3D milling. To sum up, the zero-order method, later called zero-order approximation (ZOA), will be used to generate stability lobe diagrams in three different cases:

1. ZOA in 2D milling with analytical resolution and cross-coupling of FRFs (2D ZOA analytical).
2. ZOA in 3D milling with analytical resolution and cross-coupling of FRFs (3D ZOA analytical).
3. ZOA in 3D milling using the Nyquist criterion and cross-coupling of FRFs (3D ZOA digital).

Resulting stability lobe diagrams will be compared and commented in Chapter 7 dedicated to the stability analysis in robotic milling.

4.4 Semi-discretisation method

The zero-order approximation is satisfactorily accurate in most milling processes, especially at the roughing stage as it involves large radial depth of cut and tools with many teeth. However in highly intermittent milling where the spindle speed is very high and the radial immersion of the tool is very low, additional stability lobes exist. It is the case at the finishing stage. As a matter of fact, two types of lobe are found in the literature: the Hopf and flip lobes [145]. Typical shapes of such lobes are depicted in Figure 4.8.

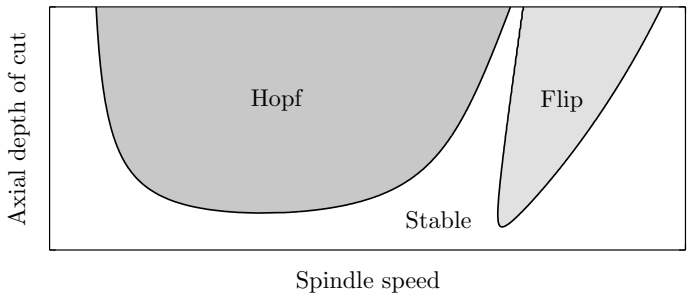


Figure 4.8: Typical shapes of Hopf and flip lobes

Zero-order method can only predict the Hopf lobes due to the truncation of the Fourier series. However, for low radial depth of cut i.e. less than 25 % of the diameter of the tool [41], so-called period-doubling chatter can appear leading to the flip lobes. Chatter frequencies related to the Hopf lobes occur at Hopf frequency f_{H} such as

$$f_{\mathrm{H}} = \{\pm\frac{\omega_{\mathrm{c}}}{2\pi} + \mathrm{j}\frac{\mathrm{N_z}\Omega}{60}\} \text{ Hz}, \quad \text{with } \mathrm{j} = \ldots, -1, 0, 1, \ldots, \tag{4.17}$$

whereas chatter frequencies related to the period-doubling effect f_{PD} arise at

$$f_{\mathrm{PD}} = \{\frac{\mathrm{N_z}\Omega}{30} + \mathrm{j}\frac{\mathrm{N_z}\Omega}{60}\} \text{ Hz}, \quad \text{with } \mathrm{j} = \ldots, -1, 0, 1, \ldots. \tag{4.18}$$

Either the frequency set f_{H} or f_{PD} shows up during chatter. In view of Figure 4.8, three cases are possible. In stable condition, the frequency content of the cutting force signals is constituted of harmonics corresponding to the tooth passing frequency. In period-doubling chatter, another signal exhibiting harmonics at half of the tooth passing frequency is superimposed to the original signal (Eq. 4.18). In contrast, Hopf chatter frequency appears as a modulation of the tooth passing frequency (Eq. 4.17). Physical explanation of the period-doubling effect can be inferred by considering that cutting forces in low radial immersion resemble a series of impacts. Between the impacts, the tool experiences free vibration and the new cutting force depends on the evolution of the decaying free vibration cycle [41].

Denomination of the two types of lobe comes from the stability condition of Floquet theory [146]. The general form of linear periodic ordinary differential equations (ODE) reads

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t), \qquad \mathbf{A}(t) = \mathbf{A}(t+P), \tag{4.19}$$

with $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{A}(t)$ a time periodic matrix at period P. For periodic ODEs, a stability condition is provided by the Floquet theory [147] and the solution of Eq. 4.19 for initial condition $\mathbf{x}(0)$ is given by $\mathbf{x}(t) = \mathbf{\Phi}(t)\mathbf{x}(0)$ where $\mathbf{\Phi}(t)$ is called the *transition matrix* or *monodromy matrix*. It connects the initial state to the state one period later such as

$$\mathbf{x}(P) = \mathbf{\Phi}(P)\mathbf{x}(0). \tag{4.20}$$

According to Floquet theory, transition matrix $\mathbf{\Phi}(t)$ can be written in the form

$$\mathbf{\Phi}(t) = \mathbf{P}(t)\mathrm{e}^{\mathbf{B}t}, \tag{4.21}$$

with $\mathbf{P}(t)$ a periodic matrix such as $\mathbf{P}(t) = \mathbf{P}(t+P)$ and $\mathbf{B}$ is a constant matrix.

The eigenvalues of $\mathbf{\Phi}(t)$ are the so-called *characteristic multipliers* or *Floquet multipliers* (μ_j, j=1, 2, ..., n) computed from

$$\det(\mu\mathbf{I} - \mathbf{\Phi}(t)) = 0. \tag{4.22}$$

On the other hand, the eigenvalues of matrix $\mathbf{B}$ are the *characteristic exponents* (λ_j, j=1, 2, ..., n) given by

$$\det(\lambda\mathbf{I} - \mathbf{B}) = 0. \tag{4.23}$$

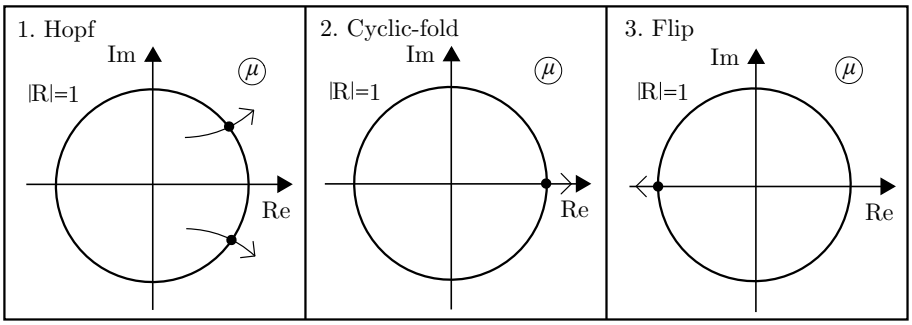


Figure 4.9: Instability of a periodic solution

If μ is a characteristic multiplier, then there are characteristic exponents λ such that $\mu = \exp(\lambda P)$, and vice versa. Beside the trivial solution of Eq. 4.19 for $\mathbf{x}(t)$=0, there are three typical cases for periodic systems (Figure 4.9):

1. The critical characteristic multipliers form a complex conjugate pair crossing the unit circle, i.e. $|\,\mu\,| = 1$ and $|\,\overline{\mu}\,| = 1$. This case is the equivalent of the Hopf bifurcation (quasi periodic chatter).

2. The critical characteristic multiplier is real and crosses the unit circle at +1. The bifurcation that arises is topologically equivalent to the *saddle-node* or *cyclic-fold* bifurcation.

3. The critical characteristic multiplier is real and crosses the unit circle at -1. This case is called period-doubling or *flip* bifurcation (period-doubling chatter).

As observed by Davies et al. in 1998 [148], only two of the three cases lead to milling instabilities: the Hopf and the flip bifurcations. They used the Poincaré sectioning technique (once-per-revolution sampling) displaying the tool vibrations to highlight the bifurcations. While stable case results in a small cluster of points, the Hopf bifurcation presents an elliptical distribution and the flip bifurcation is represented by two tightly grouped clusters of sampled points (Figure 4.10). Hence the name of the two types of lobe encountered in stability lobe diagrams.

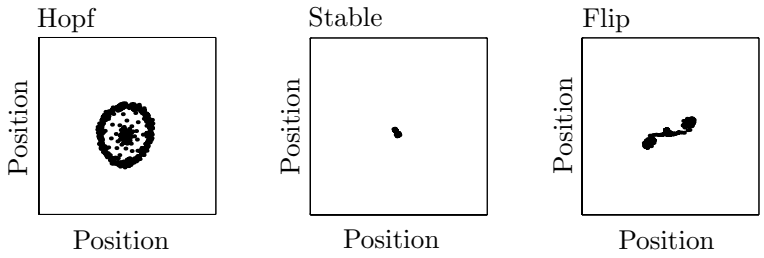


Figure 4.10: Poincaré sections [145]

From a modelling viewpoint, it is understood that in low radial immersion, the milling force waveforms are narrow and intermittent. Such waveforms have strong harmonic components in addition to the average value of $\mathbf{A}_0$ in the time-varying directional dynamic milling force coefficients $\mathbf{A}(t)$. Hence, the zero-order approximation was extended to take into account higher order terms in the Fourier series of the directional factor $\mathbf{A}(t)$ using multi-frequency models [149]. However, the computational load is significantly higher in the multi-frequency solution in comparison with the zero-order approximation. Hence, Insperger and Stépán presented an analytical solution of chatter stability in a discrete time domain [150]. The delay differential equation found in milling is discretised at discrete time intervals which allows linear and time domain simulation of forces and vibrations. The so-called semi-discretisation method (SDm) is able to compute stability lobe diagrams including the Hopf and flip lobes. The general idea of the semi-discretisation method is to transform the delay differential equation (DDE) into a series of autonomous ordinary differential equations by using the modal parameters of the system dynamics.

The n-dimensional delay differential equation reads

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{x}(t-T), \text{ with } \mathbf{A}(t+P) = \mathbf{A}(t), \text{ and } \mathbf{B}(t+P) = \mathbf{B}(t), \quad (4.24)$$

with P the time period chosen as an optional positive value. In milling, time period P is equal to the time delay such as T=P since the tooth passing period T is responsible of the regenerative chatter phenomenon.

Using the semi-discretisation method, time period T is discretised into time intervals such that $T = \mathrm{k}\Delta t$ in which k is an integer. The time interval is therefore composed of instants $[t_\mathrm{i}, t_{\mathrm{i}+1}]$. In [150], Insperger and Stépán approximate the delayed term as a weighted linear combination of the delayed discrete values taken by the milling system at time $t_{\mathrm{i\text{-}k}}$ and $t_{\mathrm{i\text{-}k+1}}$. Any mechanical system usually represented by second-order differential equations can be transformed into ordinary differential equations using the Cauchy's transformation. Consequently, following the notations found in [150], the equations of motion of the milling system can be expressed in their first order form as

$$\dot{\mathbf{x}}(t) = \mathbf{A}_\mathrm{i}\mathbf{x}(t) + \mathbf{B}_\mathrm{i}\mathbf{x}_{\mathrm{T,i}}, \quad (4.25)$$

where

$$\mathbf{A}_{\mathrm{i}} = \frac{1}{\Delta T}\int_{t_{\mathrm{i}}}^{t_{\mathrm{i+1}}} \mathbf{A}(t)dt, \quad \text{and} \quad \mathbf{B}_{\mathrm{i}} = \frac{1}{\Delta T}\int_{t_{\mathrm{i}}}^{t_{\mathrm{i+1}}} \mathbf{B}(t)dt, \tag{4.26}$$

with $\mathbf{x}_T$ the approximation of the delayed term chosen by the authors as

$$\mathbf{x}(t-T) \approx \mathbf{x}(t_{\mathrm{i}} + \frac{\Delta t}{2} - T) \approx w_a \mathbf{x}_{\mathrm{i\text{-}k+1}} + w_{\mathrm{b}} \mathbf{x}_{\mathrm{i\text{-}k}} = \mathbf{x}_{\mathrm{T,i}}. \tag{4.27}$$

The expressions of the weighting coefficients are as follows

$$w_a = \frac{T + \Delta t - \mathrm{k}\Delta t}{2\Delta t}, \quad \text{and} \quad w_b = \frac{\mathrm{k}\Delta t + \Delta t - T}{2\Delta t}, \tag{4.28}$$

which leads to $w_a = w_b = \frac{1}{2}$ in the milling case since the time delay is equal to the tooth time period T.

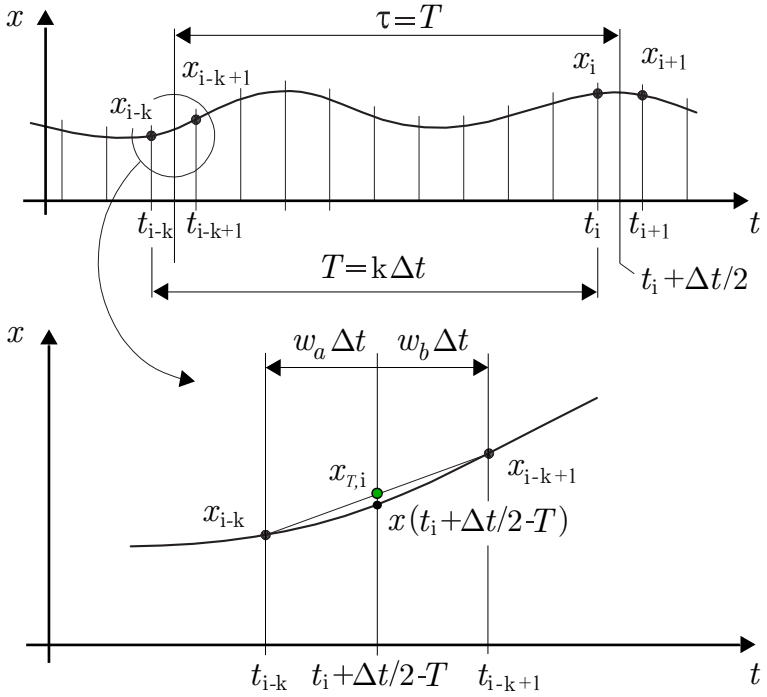


Figure 4.11: Approximation of delayed term in milling (adapted from [150])

As illustrated in Figure 4.11 for the milling case, the key of the method is to approximate the time delay as a weighted linear combination of the delayed discrete values $\mathbf{x}_{\mathrm{i}}$ and $\mathbf{x}_{\mathrm{i\text{-}k+1}}$. To initial condition $\mathbf{x}(t_{\mathrm{i}}) = \mathbf{x}_{\mathrm{i}}$, the solution of Eq. 4.25 reads

$$\mathbf{x}(t) = \exp(\mathbf{A}_{\mathrm{i}}(t - t_{\mathrm{i}}))(\mathbf{x}_{\mathrm{i}} + \mathbf{A}_{\mathrm{i}}^{-1}\mathbf{B}_{\mathrm{i}}\mathbf{x}_{\mathrm{T,i}}) - \mathbf{A}_{\mathrm{i}}^{-1}\mathbf{B}_{\mathrm{i}}\mathbf{x}_{\mathrm{T,i}}. \tag{4.29}$$

By substituting $t = t_{\mathrm{i+1}}$ in Eq. 4.29 and using Eq. 4.27, $\mathbf{x}_{\mathrm{i+1}} = \mathbf{x}(t_{\mathrm{i+1}})$ is defined as

$$\mathbf{x}_{\mathrm{i}+1} = \mathbf{P}_{\mathrm{i}}\mathbf{x}_{\mathrm{i}} + w_a\mathbf{R}_{\mathrm{i}}\mathbf{x}_{\mathrm{i\text{-}k}+1} + w_b\mathbf{R}\mathbf{x}_{\mathrm{i\text{-}k}}, \tag{4.30}$$

where

$$\mathbf{P}_{\mathrm{i}} = \exp(\mathbf{A}_{\mathrm{i}}\Delta t), \quad \text{and} \quad \mathbf{R}_{\mathrm{i}} = (\exp(\mathbf{A}_{\mathrm{i}}\Delta t) - \mathbf{I})\mathbf{A}_{\mathrm{i}}^{-1}\mathbf{B}_{\mathrm{i}}, \tag{4.31}$$

in which $\mathbf{I}$ denotes the identity matrix. According to Eq. 4.30, a discrete map between instants t_{i} and $t_{\mathrm{i}+1}$ can be defined as

$$\mathbf{y}_{\mathrm{i}+1} = \mathbf{C}_{\mathrm{i}}\mathbf{y}_{\mathrm{i}}, \tag{4.32}$$

where the $n(\mathrm{k}+1)$-dimensional vector is $\mathbf{y}_{\mathrm{i}} = [\mathbf{x}_{\mathrm{i}} \quad \mathbf{x}_{\mathrm{i\text{-}1}} \quad \dots \quad \mathbf{x}_{\mathrm{i\text{-}k}}]$ and coefficient matrix $\mathbf{C}_{\mathrm{i}}$ has the form

$$\mathbf{C}_{\mathrm{i}} = \begin{bmatrix} \mathbf{P}_{\mathrm{i}} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & w_a\mathbf{R}_{\mathrm{i}} & w_b\mathbf{R}_{\mathrm{i}} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{I} & \mathbf{0} \end{bmatrix}. \tag{4.33}$$

As explained with the Floquet theory, transition matrix $\mathbf{\Phi}$ connects the initial state to the state one period later $T = \mathrm{k}\Delta t$ such as

$$\mathbf{y}_{\mathrm{k}} = \mathbf{\Phi}\mathbf{y}_0. \tag{4.34}$$

Transition matrix $\mathbf{\Phi}$ is computed by coupling the solutions of the discretised intervals such as

$$\mathbf{\Phi} = \mathbf{C}_{\mathrm{k\text{-}1}}\mathbf{C}_{\mathrm{k\text{-}2}} \dots \mathbf{C}_1\mathbf{C}_0. \tag{4.35}$$

Finally, the stability of the system is reduced to an eigenvalue problem as mentioned in Eq. 4.22. Stability of the system is ensured if the eigenvalues of $\mathbf{\Phi}$ are in modulus less than one. As a result, by evaluating the system stability to the cutting forces generated by specific cutting conditions, spindle speed Ω and axial depth of cut a_{p}, stability lobe diagrams valid in low radial immersion can be obtained. Cutting forces are applied to the system using Fourier series introduced with ZOA in Eq. 4.11 except that they are now integrated over time for each discretised interval. Note as well that, in addition of being a time domain method, the determination of transition matrix $\mathbf{\Phi}$ whose size depends on the number of discretised intervals k, might be time consuming.

In [150], the semi-discretisation method is illustrated for a milling system comprising two orthogonal degrees of freedom similar to the one presented with the ZOA in Figure 4.6. Authors do not take into account the cross transfer functions in their 2D milling example (Appendix F). Even though Hajdu et al. [151] proposed a semi-discretisation method taking into account the mode coupling, original method presented in [150] was adapted to 3D milling without considering cross transfer functions instead. The extension is also able to account for several modes along each direction. The complete derivation of the 3D semi-discretisation method is developed in Appendix F.2. Method will be later called 3D SDm uncoupled and its resulting stability lobe diagrams will be compared and discussed in Chapter 7. Note that similarly to ZOA, it is needed to measure as many frequency response function matrices $\mathbf{H}(\omega)$ as there are robot configurations to study in order to assess milling stability.

4.5 Milling dynamic model

Zero-order and semi-discretisation methods are the two main approaches developed to assess the machining stability of CNC machine tools. The first one is computationally efficient since the cutting force model is truncated and the stability limits are determined analytically. The other one revamps the cutting force model and resorts to numerical integration in the time domain. The methods proved to be efficient for the prediction of stability lobe diagrams in machine tool [142, 152, 153]. As a matter of fact, since the machine tool dynamics can be considered as constant in its whole workspace, gathering its modal parameters in one single configuration is sufficient to assess its machining stability. It is however not the case in robotic machining because of the varying dynamics of the robot. The aforementioned methods would then be applied to a discrete number of robot configurations to cover the workspace of the milling task. This methodology was adopted in [109] leading to the planning of stable trajectories for their robot. However, the method is not convenient as it requires as many measurements through modal analysis techniques as there are robot configurations to study. In addition, dynamic effects such as the inertia forces due to the robot motion cannot be accounted for. Consequently, the chosen approach in this work takes advantage of the robot dynamic model developed in Chapter 3 on which the milling forces are applied in the time domain. Note that this approach is mentioned as one of the perspectives of Mousavi's Ph.D. thesis completed in 2016 [154].

The expression of the Cartesian milling forces $\mathbf{F}_{\mathrm{c}}(t)$ applied to the robot flexible multi-body model is sought in

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} + \mathbf{F}_{\mathrm{v}}\dot{\mathbf{q}} + \mathbf{F}_{\mathrm{s}}\mathrm{sgn}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \boldsymbol{\tau}_{\mathrm{ext}}, \tag{4.36}$$

where $\boldsymbol{\tau}_{\mathrm{ext}}$ is the combination of the actions of the gravity compensator $\boldsymbol{\tau}_{\mathrm{g}}$ and the milling forces such as

$$\boldsymbol{\tau}_{\mathrm{ext}} = \boldsymbol{\tau}_{\mathrm{g}} + [\mathbf{J}_{\mathrm{P}}^{T}(\mathbf{q})]_{\mathrm{Base}}\mathbf{F}_{\mathrm{c}}(t), \tag{4.37}$$

with $\mathbf{J}_{\mathrm{P}}(\mathbf{q})$ denoting the multibody model geometric Jacobian $\mathbf{J}(\mathbf{q})$ at the TCP limited to its first three rows such that $\mathbf{J}(\mathbf{q}) = [\mathbf{J}_{\mathrm{P}}^T(\mathbf{q}) \ \ \mathbf{J}_{\mathrm{O}}^T(\mathbf{q})]^T$ (translational motion). As discussed in Section 4.1 concerning the virtual machining principles, besides the machine dynamics module, the tool and workpiece interactions and a cutting force module are still required to consider the milling dynamics.

4.5.1 Modelling of the cutting tool

For solid end mills, their general shapes are defined according to a parametric model commonly used by CAD and CAM software: the automatically programmed tool (APT) standard. In 2001, Engin et al. [155, 156] proposed a generalised modelling of arbitrary end mill and inserted cutter geometry. In this work, only solid end mills are considered and their general envelope is defined by seven geometric parameters which are shown in Figure 4.12: D, $\mathrm{R_c}$, $\mathrm{R_r}$, $\mathrm{R_z}$, α, β and h.

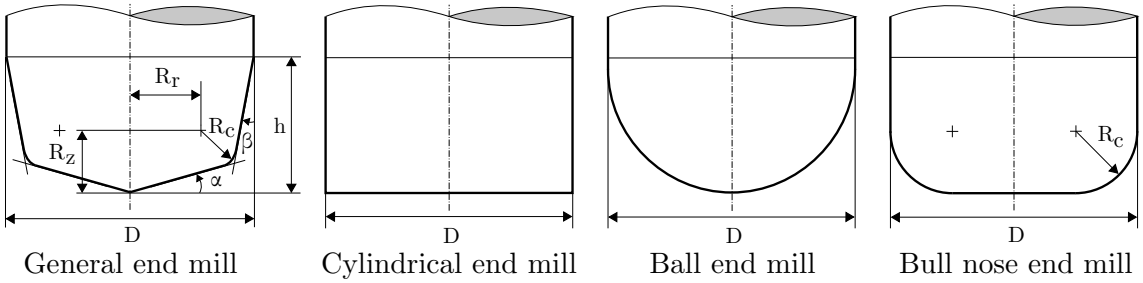


Figure 4.12: Generalised model of solid end mill

Also shown in the same figure, the envelope of typical end mills i.e. cylindrical, ball and bull nose end mills, can be obtained by adapting the values of the aforementioned parameters. Cylindrical and ball end mills are completely defined by their diameter D while bull nose end mills require one more parameter which is the inner radius of the cutting edge $\mathrm{R_c}$. Only cylindrical end mills are used in this work. Once the tool envelope is modelled, the positions of the cutting edges along the helical flute are mathematically formulated. A helical cutting edge is wrapped around the end mill envelope as illustrated in Figure 4.13. Its mathematical formulation allows determining any position along the cutting edge as a function of the Z coordinate. Since the milling process is digitalised, the end mill is spatially discretised into $\mathrm{N_s}$ superimposed and equally spaced slices of height dz along its revolution axis. Quantity dS represents the local length of the cutting edge. This approach is termed as a $2\mathrm{D}^{1/2}$ approach.

For cylindrical end mills with helix angle $\iota(z)$, the number of slices $\mathrm{N_s}$, which is the nearest positive non null integer, can optionally be derived by relying on the following relationship such as

$$\mathrm{N_s} = \frac{a_{\mathrm{p}} \tan \iota(z)}{2\pi \frac{\mathrm{D}}{2}}, \tag{4.38}$$

which depends on the axial depth of cut a_{p}. Although the relationship was used to obtain the results presented in later chapters, it is also possible to rely on a maximum

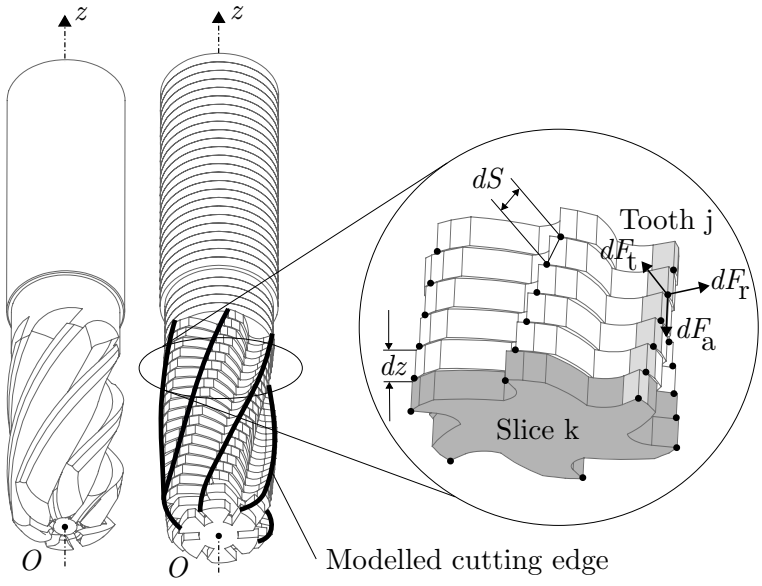


Figure 4.13: Axial discretisation of the end mill

admissible helix angle between each slice. Without any helix angle, only one slice is sufficient. At this point, it is therefore possible to compute the coordinates of any point along the cutting edge relatively to the tool revolution axis, the local helix angle $\iota(z)$, the elevation angle κ and the instantaneous angular immersion of tooth j (Figure 4.14). A variable pitch between the teeth can also be accounted for.

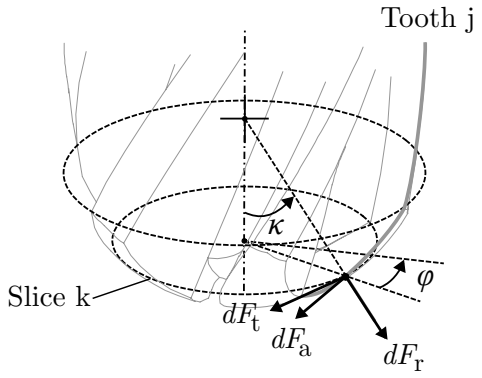


Figure 4.14: Local frame of cutting forces for a ball end mill

As illustrated in Figures 4.13 and 4.14, differential cutting forces dF_t (tangential), dF_r (radial) and dF_a (axial) are developed by the tool point pertaining to the bottom surface of slice k and tooth j. On the basis of a cutting force model, global cutting forces, for all slices and all teeth, are computed using the tool geometric model and the chip load.

4.5.2 Modelling of the workpiece

The computation of the chip load i.e. the chip thickness, requires a modelling of the workpiece. In addition, if the material removal process is simulated, an analysis of the virtually machined surface can be conducted and results can be compared to the actual

machined surface.

Even though techniques to model the workpiece in three dimensions exist, they were not retained in the robotic machining simulator for computational efficiency. Popular 3D modellings refer to B-rep, voxel or dexel methods. The dexel method was implemented but the intersection module with the end mill is still missing (Appendix G). Hence, similarly to the modelling of the tool, a $2D^{1/2}$ modelling of the workpiece was selected (Figure 4.15).

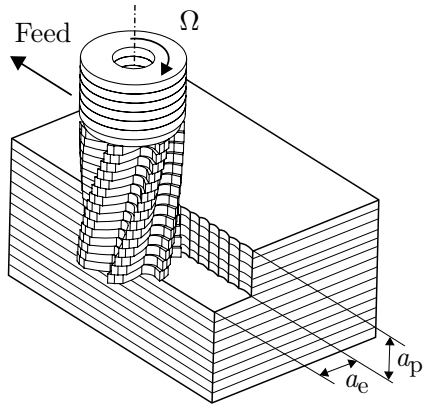


Figure 4.15: Discretised end mill and workpiece

This representation is acceptable in machine tool as many milling operations involve a tool motion which remains perpendicular to its revolution axis i.e. the tool centre point moves within a plane. In addition, the axial stiffness of machine tool is generally greater than the ones in the milling plane. However, in robotic milling, the assumption of a workpiece represented as a stack of slices is borderline. The rigidity at the tool centre point is of the same order of magnitude along the X, Y and Z directions. Axial cutting forces generated by the tool helix angle may deflect the tool from the programmed path. The assumption is still retained in this work as only shoulder milling operations are considered i.e. the main cutting forces are assumed to be developed in the milling plane, perpendicularly to the tool revolution axis. Beyond, the robotic machining simulator is theoretically able to handle contouring, slotting and face or pocket milling operations.

The contour of each slice is modelled with a list of points with coordinates along the X and Y directions. If the slice has a rectangular shape, it is defined with the coordinates of its corners (Figure 4.16). Points can be appended to the existing list to fit the motion of the teeth in the material. In this way, the geometry of the workpiece is updated.

4.5.3 Chip thickness computation

Using the presented models of end mill and workpiece, the chip thickness is eventually computed by following the method proposed by Peigné et al. [157]. The method is based on an "eraser of matter" and states that *the material area swept by each tooth is simply removed from the workpiece*. For each slice and each tooth, it is necessary to compute a

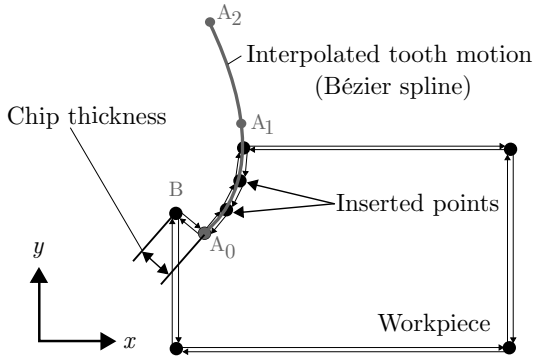


Figure 4.16: Workpiece modelling

chip thickness in a 2D plane. The following notations are used in Figure 4.17 to determine the chip thickness:

- Point O indicates the centre of the end mill in slice k.
- Points A_0, A_1 and A_2 designate the positions of tooth j at instants t_0, t_{-1} and t_{-2}.
- Point B is located at the intersection of segment OA_0 and the workpiece surface.
- Point C is positioned at the intersection between the interpolated tooth motion and the workpiece surface.

The length of segment $|BA_0|$ represents the differential chip thickness $dh_{j,k}$ for tooth j and slice k. Of course, the search of point B is managed for all teeth and all slices. Depending on the positioning of the tooth in the workpiece, three different cases can arise when the area of material representing the chip must be removed. Only positions A_0 and A_1 are necessary to detect in which case the considered tooth is. The three cases, illustrated in Figure 4.17, are described as follows:

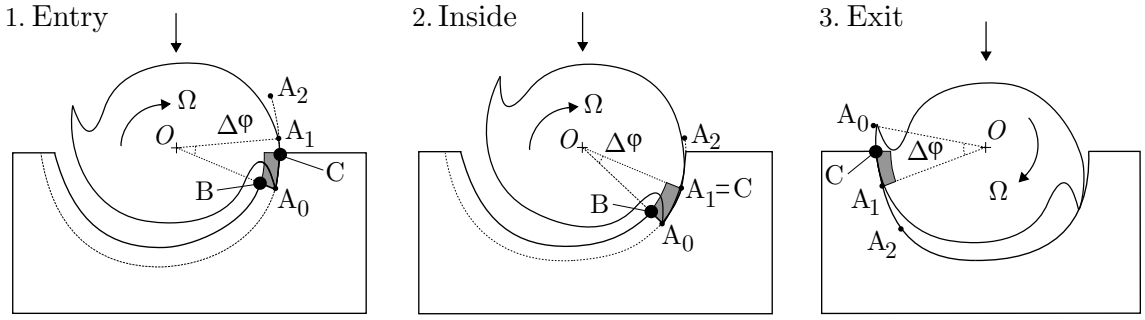


Figure 4.17: Surface updating

1. Tooth j enters into the workpiece: by definition, the entrance of one of the teeth into the material is characterised by the detection of an intersection with the surface. Point A_0 is inside the material whereas point A_1 is outside. It is then possible to define two other points in order to remove the corresponding chip: point B which represents the intersection between segment OA_0 and the surface and point

C which marks the intersection between the interpolation of the tooth positions and the surface. The path linking the successive positions of tooth j at instants t_0, t_{-1} and t_{-2} can be carried out using a circular or a Bézier interpolation. The chip bounded by points A_0, B and C is eventually removed from the workpiece geometry. It means that the list of points representing the workpiece is reorganised.

2. Tooth j is inside the workpiece: it is the most common case. Over one incrementation of angular position $\Delta\varphi_j$, tooth j remains into the material. In this case, both points A_0 and A_1 are inside the material and point B still represents the intersection between segment OA_0 and the surface. Point A_1 coincides with point C, thus reducing the computational load since the intersection search is time consuming. At the end of the operation, the chip is removed from the workpiece geometry as before.

3. Tooth j leaves the workpiece: the exit of the considered tooth from the workpiece is detected when point A_0 is outside the material while point A_1 is still inside. Point B is consequently undefined since there is no intersection between segment OA_0 and the surface. No force is therefore developed by the considered tooth. The update of the workpiece geometry consists in finding point C in order to remove the chip delimited by points A_1 and C.

The chip thickness is therefore computed as the length of segment $|BA_0|$ for each tooth j and each slice k and at each incrementation $\Delta\varphi_j$ of the tooth position by still taking into account the feed motion and the vibrations of the milling machine.

4.5.4 Cutting force model

The cutting force model is issued from the mechanistic approach in which parameters are experimentally determined for the considered operation. The linear model is selected and the cutting forces are evaluated through a proportional relationship in function of the undeformed chip thickness i.e. the chip load. Henceforth, the undeformed chip thickness for each tooth j and each slice k is denoted by $dh_{j,k}$. The chosen model includes the cutting forces developed in the primary and secondary deformation zones where the chip is formed. Furthermore, it is possible to add other linear terms to account for the ploughing action appearing when the tool edges rub against the tertiary deformation zone (Figure 4.18). The complete linear cutting force model reads

$$\begin{bmatrix} dF_{t_{j,k}} \\ dF_{r_{j,k}} \\ dF_{a_{j,k}} \end{bmatrix} = \begin{bmatrix} K_{tc} \\ K_{rc} \\ K_{ac} \end{bmatrix} dh_{j,k} db_{j,k} + \begin{bmatrix} K_{te} \\ K_{re} \\ K_{ae} \end{bmatrix} dS_{j,k}, \tag{4.39}$$

with K_{tc}, K_{rc} and K_{ac} the tangential (subscript t), radial (subscript r) and axial (subscript a) cutting force coefficients, respectively, modelling the cutting forces arising from the primary and secondary deformation zones. Coefficients K_{te}, K_{re} and K_{ae} are the so-called edge force coefficients (subscript e) and account for the rubbing forces at the

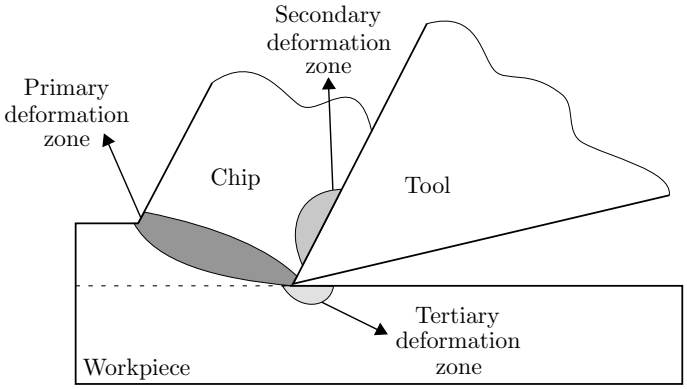


Figure 4.18: Deformation zones

tertiary deformation zone [142, 155]. The six aforementioned coefficients are the constant parameters of the model and must be identified through actual milling experiments. In this work, their values are identified in Chapter 5. Throughout the rest of the document, two models are distinguished: the complete force model and the one without the edge forces (standard model). To make the connection with the zero-order approach and the semi-discretisation method, their coefficients are computed as follows $K_{\mathrm{t}} = K_{\mathrm{tc}}$, $K_{\mathrm{r}} = \frac{K_{\mathrm{rc}}}{K_{\mathrm{t}}}$ and $K_{\mathrm{a}} = \frac{K_{\mathrm{ac}}}{K_{\mathrm{t}}}$ in this work. In Eq. 4.39, the undeformed chip area is evaluated as $dh_{\mathrm{j,k}} db_{\mathrm{j,k}}$ where $db_{\mathrm{j,k}} = \frac{dz}{\sin \kappa_{\mathrm{j,k}}}$. It represents the projection of slice of height dz on the corresponding local tangent of the cutting edge. Quantity $dS_{\mathrm{j,k}}$ is the local length of the cutting edge of tooth j and slice k. As a result, the cutting force model leads to the computation of the differential local cutting forces, $dF_{\mathrm{t_{j,k}}}$, $dF_{\mathrm{r_{j,k}}}$ and $dF_{\mathrm{a_{j,k}}}$. Having computed the undeformed chip thickness for each tooth and each slice, differential local cutting forces are summed together along cutting edge j discretised in $\mathrm{N_s}$ slices such as

$$\begin{cases} F_{\mathrm{t_j}} = \sum_{\mathrm{k}=1}^{\mathrm{N_s}} dF_{\mathrm{t_{j,k}}} \\ F_{\mathrm{r_j}} = \sum_{\mathrm{k}=1}^{\mathrm{N_s}} dF_{\mathrm{r_{j,k}}} \\ F_{\mathrm{a_j}} = \sum_{\mathrm{k}=1}^{\mathrm{N_s}} dF_{\mathrm{a_{j,k}}}. \end{cases} \tag{4.40}$$

Local forces are then projected in a Cartesian frame located at the tool tip and at the centre of the tool. The axes of the Cartesian frame are made parallel to the base frame presented in Chapter 3. The geometric transformation is expressed for each cutting edge j as it depends on the instantaneous angular immersion φ_{j} such as

$$\begin{bmatrix} F_{x_{\mathrm{j}}} \\ F_{y_{\mathrm{j}}} \\ F_{z_{\mathrm{j}}} \end{bmatrix} = \begin{bmatrix} -\cos\varphi_{\mathrm{j}} & -\sin\varphi_{\mathrm{j}}\sin\kappa & -\sin\varphi_{\mathrm{j}}\cos\kappa \\ \sin\varphi_{\mathrm{j}} & -\cos\varphi_{\mathrm{j}}\sin\kappa & -\cos\varphi_{\mathrm{j}}\cos\kappa \\ 0 & \cos\kappa & -\sin\kappa \end{bmatrix} \begin{bmatrix} F_{\mathrm{t_j}} \\ F_{\mathrm{r_j}} \\ F_{\mathrm{a_j}} \end{bmatrix}. \tag{4.41}$$

Finally, the Cartesian cutting forces are assembled depending on the number of cutting edges $\mathrm{N_z}$ such as

$$\begin{cases} F_x = \sum_{\mathrm{j}=1}^{\mathrm{N_z}} F_{x_\mathrm{j}} \\ F_y = \sum_{\mathrm{j}=1}^{\mathrm{N_z}} F_{y_\mathrm{j}} \\ F_z = \sum_{\mathrm{j}=1}^{\mathrm{N_z}} F_{z_\mathrm{j}}, \end{cases} \tag{4.42}$$

and the magnitude of the resultant cutting force acting on the milling cutter reads

$$F_{\mathrm{Rc}}(t) = \sqrt{F_x^2 + F_y^2 + F_z^2}. \tag{4.43}$$

The latter quantity is used when deriving the system stability in Chapter 7. Finally, the Cartesian milling forces to be applied at the robot TCP are eventually determined as $\mathbf{F}_\mathrm{c}(t) = [F_x(t) \;\; F_y(t) \;\; F_z(t)]^T$.

4.5.5 Stability criterion

Using the developed milling dynamic model, the limit axial depth of cut $a_{\mathrm{p_{lim}}}$ needs to be determined on the basis of a stability criterion. However, for time domain simulations, it is often difficult to distinguish between cases of vibrations due to instability and cases of excessive vibrations due to large periodic forces in order to calculate the stability limits. Altintas et al. [158] used the cutting force peak-to-peak difference between simulations of a rigid versus a flexible workpiece/cutter system. Their simulation provided good results for half and full immersion milling operations, but had difficulties to detect chatter at small radial immersions. Campomanes et al. [159] proposed an improved stability criterion based on the ratio between the undeformed rigid and dynamic chip thickness. The improved criterion reads

$$\varrho = \frac{dh_{\mathrm{j,k}}}{dh_{\mathrm{s_{j,k}}}}, \tag{4.44}$$

with $dh_{\mathrm{j,k}}$ the undeformed chip thickness during the time domain dynamic simulation for tooth j and at slice k and $dh_{\mathrm{s_{j,k}}}$ the equivalent quantity in which the workpiece and the milling machine are rigid. According to the authors, unstable chatter conditions are triggered when

$$\varrho > 1.25. \tag{4.45}$$

The selected threshold was experimentally validated by authors. In this work, the same stability criterion is used to distinguish between stable or unstable conditions for the considered milling operation. By repeating multiple milling operations in which the axial depth of cut a_p and the spindle speed Ω are varied, it leads to the construction of the stability lobe diagrams.

4.5.6 Roughness assessment

Thanks to the "eraser of matter" method, the kinematic roughness of the lateral face of the workpiece can be evaluated. Kinematic roughness is only concerned by the tooth passing disregarding possible material pullout, ridge or groove. Figure 4.19 illustrates, from a top view, the roughness evaluation of one slice that constitutes the workpiece. After the milling operation, the lateral profile exhibits a cycloidal pattern (red line). The roughness is computed along a portion of the profile so-called the *evaluation length* l_n.

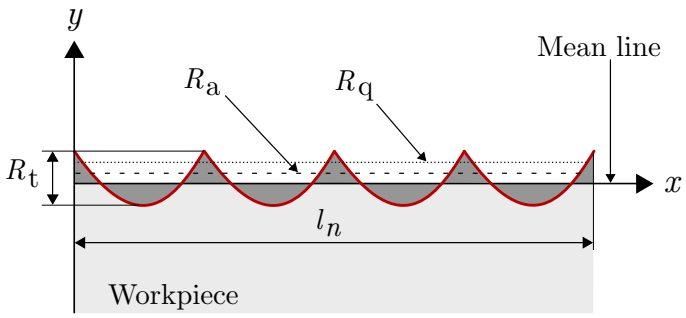


Figure 4.19: Resulting machined profile of slice k

Three types of roughness are computed:

- Total roughness R_t: the largest peak-to-peak distance of the profile within the evaluation length l_n computed as $R_t = \max(Y(x))$- $\min(Y(x))$.
- Arithmetic roughness R_a: the arithmetic mean of the departures of the roughness profile from the mean line within the evaluation length appraised as $R_a = \frac{1}{l_n} \int_0^{l_n} |Y(x)|$.
- Quadratic roughness R_q: it is the root-mean-square deviation of the profile such as $R_q = \sqrt{\frac{1}{l_n} \int_0^{l_n} Y^2(x)}$.

4.5.7 Limitations of the milling dynamic model

Related to the workpiece representation with a list of coordinate points distributed on a plane for each slice, some operations are not available. As for instance, a drilling operation cannot be performed since an internal contour representing the hole left by the drill bit would be needed. The creation of a new internal contour is presently not supported. Similarly, it is not possible to assess the roughness of the bottom face as the surfaces cannot be perforated. In addition, achieving a milling operation that would imply a tool axis parallel to the stack of slices cannot be performed. A small tilting angle of the mill axis relatively to the vertical axis is still tolerated (Figure 4.20).

Ultimately, it is worth noting that milling forces generated along the tool axis are only generated by the helix angle through cutting force coefficient K_{ac}. All these limitations could be solved using a triple-dexel representation of the workpiece with the corresponding intersection module of the cutting tool (Appendix G).

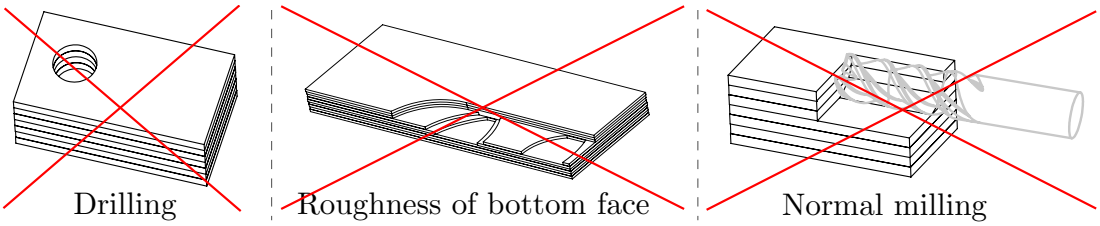


Figure 4.20: Limitations of the milling dynamic model

4.6 Discussion

Dynamics of milling are a well-studied subject. Hence, methods were developed during the last decades to predict the appearance of vibration instabilities, known as chatter, that hinder productivity. To avoid chatter, the common goal of the methods is the computation of stability lobe diagrams to determine the chatter-free stable combinations of axial depth of cut and spindle speed. Three methods were discussed in this chapter, namely the zero-order approximation (ZOA), the semi-discretisation method (SDm) and the complete milling dynamic model. While the first one is an approximation of the actual cutting process in the frequency domain which makes it computationally efficient, the computational cost of the last two models can be significantly heavier. Indeed, since they operate in the time domain, the accuracy of the cutting force prediction strongly depends on the selected digital integration interval, both spatially and temporally. Note that the milling dynamic model was developed during the Ph.D. thesis of Rivière-Lorphèvre [160]. It was later coupled with the multibody model of the robot in this work. The coupling is discussed in Chapter 6 dedicated to the validation of the robotic machining simulator.

The three presented methods are used in this work to compute stability lobe diagrams of milling operations. Using the ZOA, stability lobe diagrams are obtained according to three different implementations: 2D and 3D ZOA analytical and 3D ZOA digital. The different implementations account for the cutting forces in the milling plane (2D case) or append the axial force along the tool axis (3D case). The main advantage of the ZOA digital is the direct generation of the stability lobe diagram without the need of unifying the individual lobes to draw the global stability limit. The 3D SDm allows generating stability lobe diagrams but without the consideration of the robot mode cross-coupling. Finally, the milling dynamic model computes the cutting forces to apply on the robot multibody model. Despite some limitations related to the supported milling operations, stability lobe diagrams are generated in the time domain. In addition, thanks to the workpiece modelling and surface updating, it is possible to assess the roughness of the virtual machined part.

Table 4.1 summarises the features of the different models depending on whether the method accounts for the 2D or 3D forces, or includes the cross-coupling of robot FRFs at the TCP, and the type of implementation in the frequency or time domain.

Next chapter covers the identification of the parameters of the robot and the milling

	2D Forces	3D Forces	Cross-coupling	Frequency	Time
2D ZOA Analytical	×		×	×	
3D ZOA Analytical		×	×	×	
3D ZOA Digital		×	×	×	
3D SDm		×			×
Complete dyn. model		×	×		×
Without edge forces		×	×		×

Table 4.1: Summary of milling models for stability lobe diagram generation

models. Concerning the milling models, ZOA and SDm require the identification of the cutting force coefficients i.e. K_{tc}, K_{rc} and K_{ac} and the FRFs at the robot TCP. The complete milling dynamic model necessitates the identification of three more parameters, namely the edge force coefficients K_{te}, K_{re} and K_{ae}.

Chapter 5
Parameter identification

Following the development of the robot and the milling models, their respective parameters need to be identified. Regarding the robot model with rigid links, the inertial parameters are first identified through CAD modelling technique and rigid body identification method. The latter allows an experimental identification of the link inertial parameters without dismantling the robot. Identified parameters are further compared on the basis of the actual inertial parameters. The identification of the joint elastic parameters, namely the stiffness and damping characteristics of the torsional springs and dampers, requires measurements through modal analysis techniques. Therefore, robot modal parameters are measured in different postures to capture its varying dynamics over the workspace. A new straightforward method suitable for the identification of elastic parameters of any multibody model is presented. It is a three-step method based on the combination of genetic and deterministic algorithms. The proposed method is applied to the identification of elastic parameters pertaining to the robot with joint axial and triaxial flexibilities. The elastic parameters of the flexible beam representing the arm and forearm are later determined via finite element modelling. Beside the robot mechanical structure, the actuator parameters are collected through information related to the actual servo motors and the parameters of both controllers are chosen on the basis of the modal measurements. The cutting coefficients of the milling models are eventually identified through milling experiments with the robot.

5.1 Robot inertial parameters

The inertial parameters constitute a set of coefficients defining the link mass m_i and their inertia tensor at the centre of mass $\mathbf{\Phi}_{\mathrm{G,i}}$ in the robot model with rigid links. In this work, they are collected via three different ways, namely from the robot manufacturer, from CAD modelling and from a rigid body identification method. The latter is first explained before displaying the fitting results with the experimental measurements. Identified parameters are ultimately compared with the inertial parameters provided by the robot manufacturer.

5.1.1 Actual inertial parameters

Under a non-disclosure agreement with the robot manufacturer, the actual inertial parameters of the Stäubli TX200 robot were obtained. The parameters are lumped at the centre of mass of each body, thus the location of the centre of mass of each link $\mathbf{C}_\mathrm{i}$ is provided by the robot manufacturer. The supplied central inertia tensor $\mathbf{\Phi}_{\mathrm{G,i}}$ is complete for each link, namely six different values per link i are given: ϕ_{xx_i}, ϕ_{yy_i}, ϕ_{zz_i}, ϕ_{xy_i}, ϕ_{xz_i}

and ϕ_{yz_i}. For the link mass m_i, it aggregates in one single point the contributions of the link material, the motor if included in the considered link and the associated gearbox and bearing (and the gravity compensator mechanism for the robot arm). From this representation, it is understood that the bodies representing the rotors in the multibody model are massless but have their own inertia tensor $\mathbf{\Phi}_{G,m_i}$, which is also provided by the robot manufacturer. It is not known how the values are collected from the robot manufacturer, either by 3D modelling or measurements. Note that even if the values of the inertial parameters are confidential, the robot manufacturer adopts a modelling convention similar to the one presented in Subsection 3.3.1, i.e. the distances between each joint position are given as well as the centre of mass position relative to the joint location. In addition, the characteristics of the gravity compensator mechanism are known, i.e. the stiffness of the spring, its preload and its anchor points. The robot model that uses the actual inertial data is later referenced as the *Stäubli TX200 model*.

5.1.2 CAD modelling technique

Although it is not a recommended technique, it is the simplest way to gather all the needed inertial parameters: the full inertia tensor $\mathbf{\Phi}_{G,i}$ and the link mass m_i. Such technique is inaccurate because of the simplification typically introduced by geometric modelling [3]. Nevertheless, the method was still retained to highlight that comparable results can be obtained if the actual inertial parameters are not available. As illustrated in Figure 5.1, the original 3D designs provided by the robot manufacturer are made of solid shapes without the representation of the interior of the parts whereas the actual ones are hollowed.

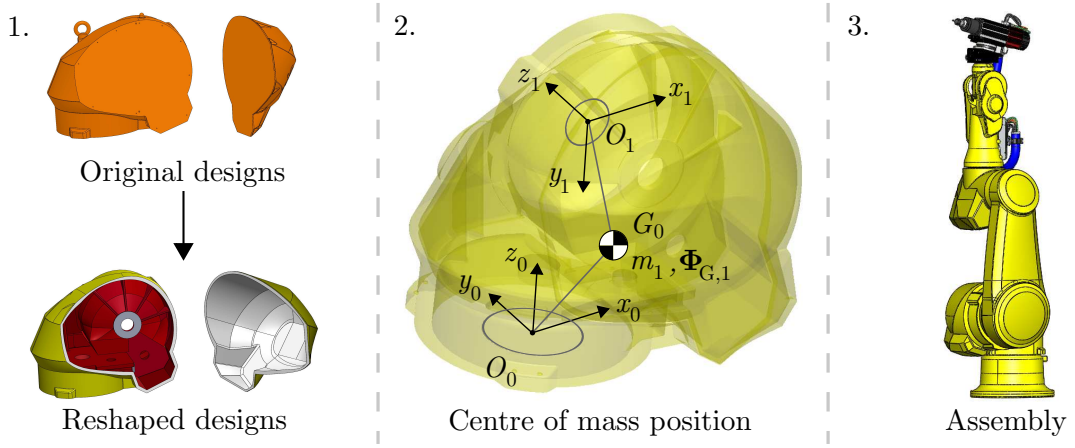


Figure 5.1: UMons TX200 model obtained from reshaped CAD designs

In order to identify the mass and inertial properties, and to locate the centre of mass of each link, it was decided to redesign each part on the basis of photographs of the robot interior. The mass of each component is calculated after the assignment of the material (aluminium or steel if magnetisable). Knowing that the robot total weight is 1000 kg, the 3D representation was also enriched with the inclusion of servo motor and gearbox masses based on their actual dimensions in order to get a realistic distribution of the robot mass. An estimation of the mass of each servo motor can be obtained by size

comparison with the ones specified in the catalogue of the motor brand (Wittenstein motors for the Stäubli TX200 robot [161]). Even if the actual motors are customised by the robot manufacturer, equivalent ones in terms of size can be found in the catalogue.

As for the robot model with the actual inertial parameters, they are lumped at the centre of mass of each link. The servo motors are again modelled as massless bodies but with an inertia tensor. Hence, the modelling convention explained in Subsection 3.3.1 is applied. The coordinates of each joint frame with respect to the preceding one, identified on the reshaped 3D models, are gathered in Table 5.1. For instance, distance O_0O_1 illustrated in Figure 5.1 is projected into the preceding joint frame, i.e. frame O_0.

	$O_{\text{Base}}O_0$	O_0O_1	O_1O_2	O_2O_3	O_3O_4	O_4O_5	O_5O_6	O_6O_{EE}	$O_{\text{EE}}O_{\text{TCP}}$
x [m]	0	0.25	0.95	0	0	0	0	0	0.277
y [m]	0	0.235	0	-0.179	0	-0.177	0	0	0
z [m]	0.344	0.298	0.044	-0.279	0.621	0	0	0.017	0.326

Table 5.1: Joint frame positions obtained from reshaped CAD models

Still projected in the same frame, the coordinates of the centre of mass of each link are enclosed in Table 5.2.

	$O_{\text{Base}}G_{\text{B}}$	O_0G_0	O_1G_1	O_2G_2	O_3G_3	O_4G_4	O_5G_5	$O_{\text{EE}}G_{\text{Spindle}}$	$O_3G_{\text{Acquisition}}$
x [m]	0.007	0.150	0.401	0.039	0	0	0	-0.048	-0.141
y [m]	0.001	-0.008	0.007	0.043	0.004	-0.067	0	-0.001	-0.054
z [m]	0.119	0.209	0.139	-0.219	0.266	0	0.005	0.173	0.285

Table 5.2: Centre of mass positions obtained from reshaped CAD models

Finally, the inertial parameters of each link, projected in the coordinate system of the preceding joint, are delivered in Table 5.3 and Table 5.4 provides the inertia tensor of each rotor j based on the motor catalogue, knowing that the z axis is the motion axis.

Body i	0	1	2	3	4	5	6	7	8
m_{i} [kg]	226.2	364.0	256.1	98.7	40.8	13.6	0.6	61.4	5.4
$\phi_{\text{G},xx_{\text{i}}}$ [kg.m^2]	8.303	12.248	6.574	0.569	1.230	0.026	5e^{-4}	1.342	0.078
$\phi_{\text{G},yy_{\text{i}}}$ [kg.m^2]	7.979	17.833	36.584	0.462	1.172	0.016	5e^{-4}	1.622	0.024
$\phi_{\text{G},zz_{\text{i}}}$ [kg.m^2]	11.869	16.950	39.916	0.512	0.299	0.026	0.001	0.548	0.058
$\phi_{\text{G},xy_{\text{i}}}$ [kg.m^2]	0	-0.204	0.178	0.073	0.001	0	0	0.002	0.002
$\phi_{\text{G},xz_{\text{i}}}$ [kg.m^2]	-0.012	1.552	0.741	-0.018	0	0	0	0.016	0
$\phi_{\text{G},yz_{\text{i}}}$ [kg.m^2]	0	0.250	0.054	-0.025	0.062	0	0	0.018	0.005

Table 5.3: Link inertia properties obtained from reshaped CAD models

Note that the rotor inertia properties are not used in this work. If one desires to check the correctness of its assembled robot model, a global centre of mass position at coordinates (0.457, 0.092, 0.593) [m] should be found with respect to the global reference frame if the robot is in the posture presented in Figure 3.11a ($q_2 = \frac{\pi}{2}$ and $q_3 = \frac{-\pi}{2}$ if the encoder pose

Rotor j	1	2	3	4	5	6
$\phi_{\mathrm{m},xx_\mathrm{j}}$ [kg.m^2]	0.3185	0.3185	0.186	0.01	0.02	0.01
$\phi_{\mathrm{m},yy_\mathrm{j}}$ [kg.m^2]	0.3185	0.3185	0.186	0.01	0.02	0.01
$\phi_{\mathrm{m},zz_\mathrm{j}}$ [kg.m^2]	0.013	0.013	0.009	5e^{-4}	5e^{-4}	5e^{-4}

Table 5.4: Rotor inertia properties obtained from the servo motor catalogue [161]

is the reference pose). The total mass of the assembled robot with its payloads composed of the spindle and acquisition device is 1066.8 kg. The robot model that employs the inertial data derived from the reshaped CAD models is later designated as the *UMons TX200 model*.

Introducing the so-called *milling pose* in Figure 5.2 in which the vertical axis of the spindle is parallel to z_{Base} and above the vice, the mass matrices of the Stäubli and UMons TX200 models are compared. Joint positions of the milling pose referenced in the encoder pose described in Subsection 3.2.1 are presented in Table 5.5.

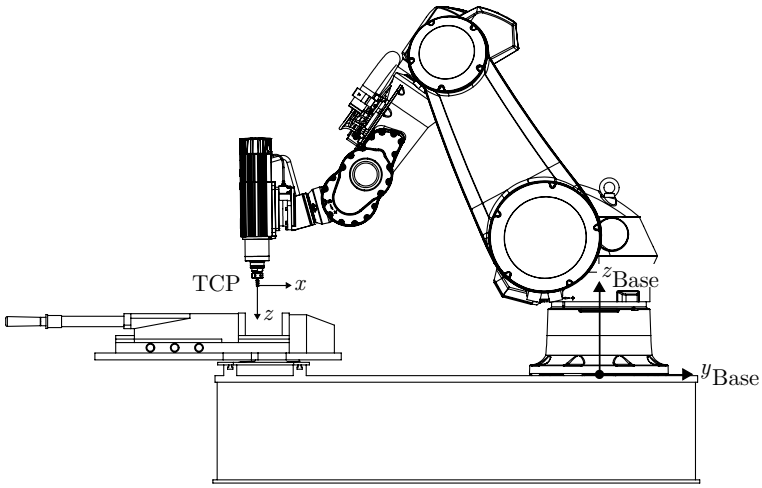


Figure 5.2: Milling pose of the Stäubli TX200 robot

q_1 [°]	q_2 [°]	q_3 [°]	q_4 [°]	q_5 [°]	q_6 [°]
-89.34	26.26	117.20	1.02	-38.46	-0.63

Table 5.5: Joint positions of the milling pose in the encoder reference

Mass matrices result from the application of the Lagrange's theorem on the rigid models of the robot i.e. without any flexibility. In the milling pose, the values of the (6×6) mass matrix using the actual inertial parameters from the robot manufacturer reads

$$\mathbf{M}_{\text{Stäubli}} = \begin{bmatrix} 392.78 & -32.25 & 0.42 & -28.02 & 0.47 & -4.14 \\ -32.25 & 333.07 & 78.99 & -0.20 & 31.53 & 0.07 \\ 0.42 & 78.99 & 115.19 & -0.50 & 24.53 & 0.08 \\ -28.02 & -0.20 & -0.50 & 6.10 & 0.01 & 1.63 \\ 0.47 & 31.53 & 24.53 & 0.01 & 10.54 & 0.02 \\ -4.14 & 0.07 & 0.08 & 1.63 & 0.02 & 1.49 \end{bmatrix}, \tag{5.1}$$

while the equivalent one derived from the CAD models is

$$\mathbf{M}_{\text{UMons}} = \begin{bmatrix} 364.51 & -39.67 & -0.24 & -27.05 & 0.45 & -4.62 \\ -39.67 & 318.83 & 67.60 & 0.13 & 30.54 & 0.05 \\ -0.24 & 67.60 & 96.49 & 0.34 & 23.43 & 0.07 \\ -27.05 & 0.13 & 0.34 & 5.72 & 2\text{e}^{-3} & 1.69 \\ 0.45 & 30.54 & 23.43 & 2\text{e}^{-3} & 10.14 & 0.01 \\ -4.62 & 0.05 & 0.07 & 1.69 & 0.01 & 1.44 \end{bmatrix}. \tag{5.2}$$

Computing the error in percentage between each component of both matrices, it leads to the following error matrix

$$\frac{|\ \mathbf{M}_{\text{Stäubli}} - \mathbf{M}_{\text{UMons}}\ |}{|\ \mathbf{M}_{\text{Stäubli}}\ |}100 = \begin{bmatrix} 7.20 & 23.03 & 157.78 & 3.46 & 4.19 & 11.54 \\ 23.03 & 4.28 & 14.42 & 166.72 & 3.12 & 27.99 \\ 157.78 & 14.42 & 16.23 & 167.73 & 4.50 & 16.40 \\ 3.46 & 166.72 & 167.73 & 6.26 & 74.20 & 3.26 \\ 4.19 & 3.12 & 4.50 & 74.20 & 3.77 & 43.36 \\ 11.54 & 27.99 & 16.40 & 3.26 & 43.36 & 3.15 \end{bmatrix} \%. \tag{5.3}$$

It can be observed that the diagonal components are quite well estimated with a relative error less than 20 %. Smallest error is 3.12 % for elements $M_{2,5}$ and $M_{5,2}$ and largest error regards terms $M_{3,4}$ and $M_{4,3}$ with an error of 167.73 % (very small term in Eq. 5.2). Similar conclusions were derived in other tested robot configurations. Note that in the UMons TX200 model, no gravity compensator mechanism is included.

5.1.3 Rigid body identification method

In order to find accurate estimates of the inertial parameters, it is worth resorting to identification techniques which conveniently exploit a property of the equations of motion. In Chapter 3 was introduced the general form of the equations of motion for a robotic manipulator without any flexibility, recalled as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_{\text{v}}\dot{\mathbf{q}} + \mathbf{F}_{\text{s}}\text{sgn}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \boldsymbol{\tau}_{\text{ext}}. \tag{5.4}$$

The fact that the equations of motion of a mechanical system are non-linear is well established. Indeed, inertial, Coriolis, centrifugal and gravitational forces introduce a

non-linear coupling and also vary during manipulator motion. Nonetheless, a very useful property characterises manipulators with open kinematic chain. The property refers to the linearity of the *dynamic model* with respect to the *dynamic parameters* that describe the links and rotors. In other words, equations of motion (Eq. 5.4) can be written in a linear form with respect to parameters gathering information about the inertial properties of the manipulator. Note that it does not mean that high-order terms are neglected as it would be in a linearisation procedure. The parameters, allowing putting the equations of motion in a linear form, are computed in such a way that they constitute a minimum set defining the equations of motion of the manipulator uniquely. They are called the *base parameters* of the manipulator. Each of them is composed of linear combinations of elementary inertial parameters (m_{i}, $\mathbf{\Phi}_{\mathrm{G,i}}$) relative to the links and rotors. This interesting property is a valuable asset for the identification of inertial parameters of a robot.

Equation 5.4 is linear with respect to the base parameters of the manipulator and can therefore be reformulated in the following linear representation such that

$$\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\pi} = \mathbf{\Gamma}, \tag{5.5}$$

with $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \in \mathbb{R}^{n \times n_{\mathrm{p}}}$ the so-called *regressor* matrix which only depends on the kinematics of the manipulator with n the number of considered robot joints in the manipulator model, $\boldsymbol{\pi} \in \mathbb{R}^{n_{\mathrm{p}} \times 1}$ the vector of base parameters with n_{p} the number of base parameters and $\mathbf{\Gamma}$ the summation of the actuator $\boldsymbol{\tau}$ and external $\boldsymbol{\tau}_{\mathrm{ext}}$ torques.

In order to determine the expressions of the base parameters, the dynamic model of the manipulator must be represented following the so-called *augmented link* model. Such model directly derives from the representation given when defining the standard DH parameters (Figure 3.8a). In the augmented link model, joints and their associated rotors are both located on joint frames O_0 to O_5. Furthermore, the central tensor of inertia of each link $\mathbf{\Phi}_{\mathrm{G,i}}$ is also brought back to the joint frame. An illustration of the augmented link model is depicted in Figure 5.3. Mass m_{i} of each link i and mass $m_{\mathrm{m_i}}$ of each motor are assumed to be non-zero, and vector $\mathbf{C}_{\mathrm{i}} = [C_{x_{\mathrm{i}}}\ C_{y_{\mathrm{i}}}\ C_{z_{\mathrm{i}}}]^T$ locates the centre of mass of link i in the corresponding local frame. To put it differently, the mass properties of motor i and link i are localised in joint frame $O_{\text{i-1}}$.

By virtue of the Steiner's theorem, the inertia tensor of augmented link i in frame $O_{\text{i-1}}$ is

$$\begin{aligned}
\mathbf{\Phi}_{\mathrm{i}} &= \begin{bmatrix} \phi_{xx_{\mathrm{i}}} & \phi_{xy_{\mathrm{i}}} & \phi_{xz_{\mathrm{i}}} \\ \phi_{yx_{\mathrm{i}}} & \phi_{yy_{\mathrm{i}}} & \phi_{yz_{\mathrm{i}}} \\ \phi_{zx_{\mathrm{i}}} & \phi_{zy_{\mathrm{i}}} & \phi_{zz_{\mathrm{i}}} \end{bmatrix} \\
&= \begin{bmatrix} \phi_{\mathrm{G},xx_{\mathrm{i}}} + m_{\mathrm{i}}\left(C_{y_{\mathrm{i}}}^2 + C_{z_{\mathrm{i}}}^2\right) & \phi_{\mathrm{G},xy_{\mathrm{i}}} + m_{\mathrm{i}} C_{x_{\mathrm{i}}} C_{y_{\mathrm{i}}} & \phi_{\mathrm{G},xz_{\mathrm{i}}} + m_{\mathrm{i}} C_{x_{\mathrm{i}}} C_{z_{\mathrm{i}}} \\ \phi_{\mathrm{G},yx_{\mathrm{i}}} + m_{\mathrm{i}} C_{y_{\mathrm{i}}} C_{x_{\mathrm{i}}} & \phi_{\mathrm{G},yy_{\mathrm{i}}} + m_{\mathrm{i}}\left(C_{x_{\mathrm{i}}}^2 + C_{z_{\mathrm{i}}}^2\right) & \phi_{\mathrm{G},yz_{\mathrm{i}}} + m_{\mathrm{i}} C_{y_{\mathrm{i}}} C_{z_{\mathrm{i}}} \\ \phi_{\mathrm{G},zx_{\mathrm{i}}} + m_{\mathrm{i}} C_{z_{\mathrm{i}}} C_{x_{\mathrm{i}}} & \phi_{\mathrm{G},zy_{\mathrm{i}}} + m_{\mathrm{i}} C_{z_{\mathrm{i}}} C_{y_{\mathrm{i}}} & \phi_{\mathrm{G},zz_{\mathrm{i}}} + m_{\mathrm{i}}\left(C_{x_{\mathrm{i}}}^2 + C_{y_{\mathrm{i}}}^2\right) \end{bmatrix}.
\end{aligned} \tag{5.6}$$

Hence, each link involves ten elementary inertial parameters including m_{i}, $m_{\mathrm{i}}C_{x_{\mathrm{i}}}$, $m_{\mathrm{i}}C_{y_{\mathrm{i}}}$, $m_{\mathrm{i}}C_{z_{\mathrm{i}}}$, $\phi_{xx_{\mathrm{i}}}$, $\phi_{yy_{\mathrm{i}}}$, $\phi_{zz_{\mathrm{i}}}$, $\phi_{xy_{\mathrm{i}}}$, $\phi_{xz_{\mathrm{i}}}$ and $\phi_{yz_{\mathrm{i}}}$. Each motor also brings three additional

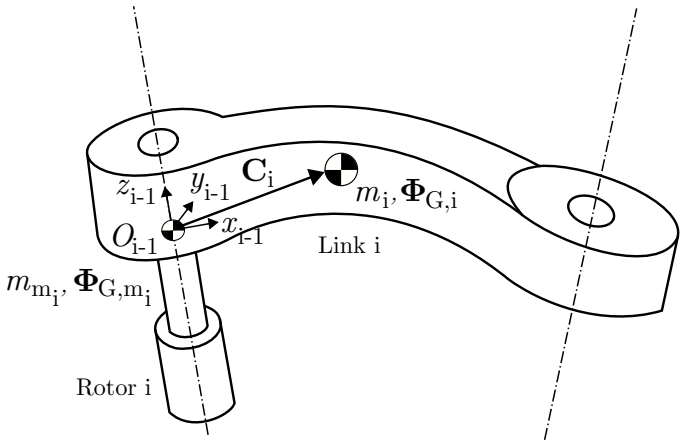


Figure 5.3: Augmented link model

elementary inertial parameters $\phi_{\text{m},xx_\text{i}}$, $\phi_{\text{m},yy_\text{i}}$ and $\phi_{\text{m},zz_\text{i}}$ and its mass. The vector of base parameters $\boldsymbol{\pi}$ is therefore constituted of combinations of those elementary inertial parameters. Following Mayeda et al. [162] and excluding the rotors, the expressions of the base parameters relative to the links can be determined from the symbolic construction of the equations of motion. The derivation of base parameters for the Stäubli TX200 robot limited to its first three joints is presented in Appendix H. The inertial properties of the motors are then appended to the determined set of base parameters. As viscous F_{v_i} and static F_{s_i} friction coefficients directly appear in a linear form in the equations of motion (Eq. 5.4), they can also be adjoined to the list of base parameters. It is necessary to work symbolically since the symbolic expression of regressor matrix $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is desired in order to identify the values of the base parameters $\boldsymbol{\pi}$ from torque measurements $\boldsymbol{\Gamma}$ using Eq. 5.5 recursively.

As mentioned earlier, the Stäubli TX200 robot encompasses a gravity compensation system. The latter generates an external torque $\boldsymbol{\tau}_\text{ext}$ reducing the torque needed for actuation $\boldsymbol{\tau}$. For the Stäubli TX200 robot, the external torque is applied on the motor of the second joint. The external torque developed by the gravity compensation system can be approximated using finite Fourier series expansion truncated to cosine terms [163,164]. The expression of the external torque $\tau_{\text{gravity,i}}$ from gravity compensator i reducing the load of motor i is

$$\tau_{\text{gravity,i}} = \sum_{\text{f}=1}^{n_\text{f}} B_\text{f} \cos(\text{f}q_\text{i}), \tag{5.7}$$

with n_f the number of terms included in the harmonic series (often limited to n_f=3 which is sufficient). The Stäubli TX200 robot only gathers one gravity compensator system reducing the load of the second motor. Therefore, only $\tau_{\text{gravity,2}}$ is defined leading to three more parameters (B_1, B_2 and B_3) to append to the set of base parameters. Of course, during the identification procedure, τ_ext is solely constituted of gravity compensation torque $\tau_{\text{gravity,2}}$ without any milling force.

Limiting the study to the first three joints n=3 of the Stäubli TX200 robot as they mainly lead its dynamics, the expressions of the determined base parameters are provided in Table 5.6. Note that the considered robot has a common structure, shared with many other industrial robots, which makes general the usage of the obtained expressions. It is conveniently considered that bodies after the third joint form one bigger link (m_3, $\mathbf{\Phi}_3$). In the table, variables a_i still refer to the link length as defined in the standard DH convention. A total of 27 parameters are determined: parameters 1 to 15 are mainly related to the links, parameters 16 to 18 refer to the motors, parameters 19 to 21 are involved in the Fourier series of the gravity compensator system and the last parameters represent the static and viscous friction coefficients from the joint side. It can be observed that rotor inertias of motors 1 and 2 around the Z-axis (ϕ_{m,zz_1} and ϕ_{m,zz_2}) appear in parameters 1 and 2. As a result, they cannot be separately identified as for parameter 16 (ϕ_{m,zz_3}). Another remark can be issued concerning the linear combinations of elementary inertial parameters. From the derived augmented link model, it is unfortunately not possible to extract each elementary inertial parameter. First, because some of them do not appear in the set of base parameters (e.g. ϕ_{xx_1} and ϕ_{yy_1}) and secondly, because the elementary inertial parameters are redundant in the equations of motion i.e. different values of elementary inertial parameters might deliver the same dynamic equations. Part of the answer lies in the fact that the links are only driven by the actuators around the axes of motion. No actuation is settled perpendicular to them.

With the symbolic expressions of the base parameters provided in Table 5.6, the corresponding regressor matrix $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ can be computed from the symbolic expression of the equations of motion (Eq. 5.4). The symbolic expression of the resulting regressor matrix is given in Appendix I for the Stäubli TX200 robot limited to its first three joints.

Once the expressions of the equations of motion (Eq. 5.5) are established into a linear form, a Multi-Input Multi-Output (MIMO) Least Squares Estimation (LSE) is used to identify the base parameters gathered in vector $\boldsymbol{\pi}$. The input signals to the identification are the position, velocity, and acceleration signals of the first three joints of the manipulator denoted by vector $\mathbf{q} = [q_1 \ q_2 \ q_3]^T$ in this section. The output signals are the corresponding joint torque signals enclosed in vector $\boldsymbol{\tau} = [\tau_1 \ \tau_2 \ \tau_3]^T$. These signals are obtained by moving the end effector on optimised trajectories within the workspace of the robot and by recording the joint encoder and motor torque signals at a set of discrete time instants, $t_1, t_2, ..., t_f$ (t_f being the *final* instant). The base parameters $\boldsymbol{\pi}$ are then obtained using the LSE on a set of equations obtained by repeating Eq. 5.5 at every time instant such as

$$\mathbf{A}\boldsymbol{\pi} = \mathbf{T}, \tag{5.8}$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{Y}(\mathbf{q}(t_1),\ \dot{\mathbf{q}}(t_1),\ \ddot{\mathbf{q}}(t_1)) \\ \mathbf{Y}(\mathbf{q}(t_2),\ \dot{\mathbf{q}}(t_2),\ \ddot{\mathbf{q}}(t_2)) \\ \vdots \\ \mathbf{Y}(\mathbf{q}(t_f),\ \dot{\mathbf{q}}(t_f),\ \ddot{\mathbf{q}}(t_f)) \end{bmatrix}, \text{ and } \mathbf{T} = \begin{bmatrix} \boldsymbol{\tau}(t_1) \\ \boldsymbol{\tau}(t_2) \\ \vdots \\ \boldsymbol{\tau}(t_f) \end{bmatrix}. \tag{5.9}$$

π_{i}	Physical meaning of parameter i
1	$\phi_{\mathrm{m},yy_2} + \phi_{\mathrm{m},yy_3} + \phi_{yy_2} + \phi_{yy_3} + \phi_{zz_1} + {a_1}^2\,(m_2 + m_3) + m_3\,{a_2}^2 + m_{\mathrm{m}2}\,{a_1}^2 + m_{\mathrm{m}3}\,{a_1}^2 + m_{\mathrm{m}3}\,{a_2}^2 + \phi_{\mathrm{m},zz_1}\,{k_1}^2$
2	$\phi_{zz_2} + m_3\,{a_2}^2 + m_{\mathrm{m}3}\,{a_2}^2 + \phi_{\mathrm{m},zz_2}\,{k_2}^2$
3	$m_3\,a_2 + m_{\mathrm{m}3}\,a_2 + C_{x2}\,m_2$
4	$C_{y_2}\,m_2$
5	$\phi_{xx2} - \phi_{yy_2} - m_3\,{a_2}^2 - m_{\mathrm{m}3}\,{a_2}^2$
6	$\phi_{xz2} + C_{z3}\,m_3\,a_2$
7	ϕ_{xy_2}
8	ϕ_{yz_2}
9	ϕ_{zz3}
10	$C_{x3}\,m_3$
11	$C_{y_3}\,m_3$
12	$\phi_{xx3} - \phi_{yy3}$
13	ϕ_{xz3}
14	ϕ_{xy3}
15	ϕ_{yz_3}
16	ϕ_{m,zz_3}
17	$\phi_{\mathrm{m},xx_2} - \phi_{\mathrm{m},yy_2}$
18	$\phi_{\mathrm{m},xx_3} - \phi_{\mathrm{m},yy_3}$
19	B_1
20	B_2
21	B_3
22	F_{v_1}
23	F_{s_1}
24	F_{v_2}
25	F_{s_2}
26	F_{v_3}
27	F_{s_3}

Table 5.6: Expressions of the base parameters as well as the gravity compensation and friction coefficients for the Stäubli TX200 robot limited to its first three joints

In order to ensure that the obtained torque and joint displacement data are informative [165], persistently excitation trajectories are designed to minimise the condition number of observation matrix $\mathbf{A}$ in Eq. 5.8. The design of such optimum trajectories in joint space was proposed by Swevers et al. [164] in which the joint position, velocity and acceleration along the trajectory are made of a finite sum of harmonic sine and cosine functions. Finite Fourier series expansions therefore define the time evolution of the manipulator joints and their time derivatives as follows

$$q_{\mathrm{i}}(t) = \sum_{l=1}^{\mathrm{N_f}} \frac{a_{\mathrm{i},l}}{\omega_{\mathrm{F}} l} \sin(\omega_{\mathrm{F}}\, l\, t) - \frac{b_{\mathrm{i},l}}{\omega_{\mathrm{F}} l} \cos(\omega_{\mathrm{F}}\, l\, t) + q_{\mathrm{i},0}, \tag{5.10}$$

$$\dot{q}_{\mathrm{i}}(t) = \sum_{l=1}^{\mathrm{N_f}} a_{\mathrm{i},l} \cos(\omega_{\mathrm{F}}\, l\, t) + b_{\mathrm{i},l} \sin(\omega_{\mathrm{F}}\, l\, t), \tag{5.11}$$

$$\ddot{q}_{\mathrm{i}}(t) = \sum_{l=1}^{\mathrm{N_f}} -a_{\mathrm{i},l}\, \omega_{\mathrm{F}}\, l\, \sin(\omega_{\mathrm{F}}\, l\, t) + b_{\mathrm{i},l}\, \omega_{\mathrm{F}}\, l\, \cos(\omega_{\mathrm{F}}\, l\, t), \tag{5.12}$$

with ω_{F} the assumed fundamental angular frequency of the Fourier series and $\mathrm{N_f}$ the number of harmonics in the Fourier series. The value of ω_{F} is chosen such that the flexibilities of the manipulator are not excited during the motion, i.e. a frequency smaller than the first natural frequency of the manipulator is often selected. Constant coefficients $a_{\mathrm{i},l}$ and $b_{\mathrm{i},l}$ constitute the optimisation design parameters and $q_{\mathrm{i},0}$ is the robot configuration around which the robot excitation occurs. To simplify the notations, let $\boldsymbol{\delta}$ be a function of $a_{\mathrm{i},l}$, $b_{\mathrm{i},l}$ and $q_{\mathrm{i},0}$ and $\widehat{\boldsymbol{\delta}}$ is determined by minimising the condition number of observation matrix $\mathbf{A}$. The constrained optimisation problem is summarised such that[1]

$$\widehat{\boldsymbol{\delta}} = \arg\, \min_{\boldsymbol{\delta}}\, \mathrm{cond}(\mathbf{A}(\boldsymbol{\delta}, \omega_{\mathrm{F}})), \tag{5.13}$$

subjected to

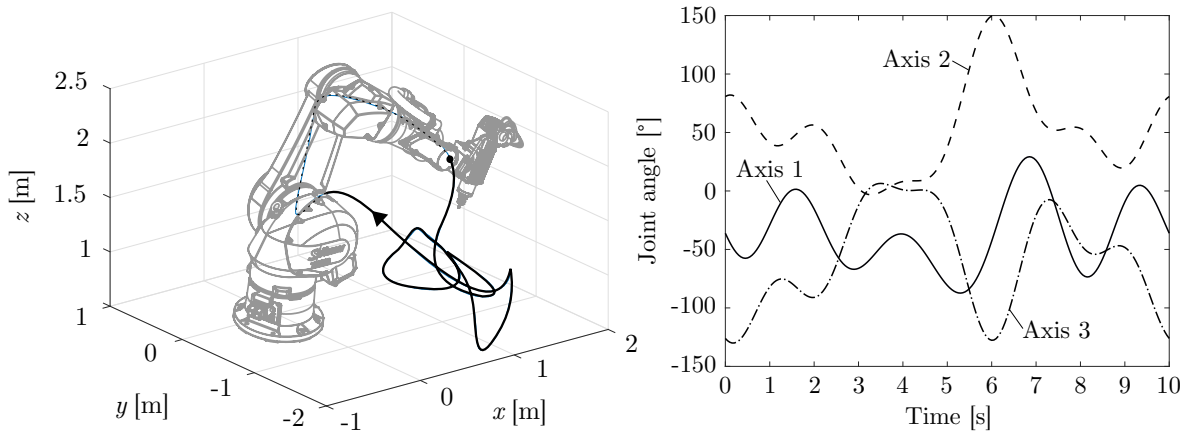
$$\begin{cases} \mathbf{q}_{\min} \leqslant \mathbf{q}(pT_s, \boldsymbol{\delta}) \leqslant \mathbf{q}_{\max} \\ \dot{\mathbf{q}}_{\min} \leqslant \dot{\mathbf{q}}(pT_s, \boldsymbol{\delta}) \leqslant \dot{\mathbf{q}}_{\max} \\ \ddot{\mathbf{q}}_{\min} \leqslant \ddot{\mathbf{q}}(pT_s, \boldsymbol{\delta}) \leqslant \ddot{\mathbf{q}}_{\max} \\ \{\mathbf{s}\,(\mathbf{q}(pT_s, \boldsymbol{\delta}))\} \subset \mathbf{S} \end{cases} \quad \text{for } 0 \leqslant p \leqslant \frac{T_{\mathrm{F}}}{T_s}, \tag{5.14}$$

with T_s being the sampling period and $T_{\mathrm{F}}{=}\frac{2\pi}{\omega_{\mathrm{F}}}$. The set of equations shown in Eq. 5.14 determines the constraints of the optimisation problem. These constraints are governed by the position, velocity and acceleration limitations of the joints, and the workspace of the robot. Vectors $\mathbf{q}_{\min}$ and $\mathbf{q}_{\max}$ determine the allowable range of axis displacement, vectors $\dot{\mathbf{q}}_{\min}$ and $\dot{\mathbf{q}}_{\max}$ determine the allowable axis velocity range and vectors $\ddot{\mathbf{q}}_{\min}$ and $\ddot{\mathbf{q}}_{\max}$ determine the allowable axis acceleration range. Also, $\mathbf{S}$ stands for the available workspace while $\mathbf{s}(\mathbf{q})$ is a subset of positions reachable by the end effector of the manipulator according to the optimised trajectories. Figure 5.4a shows a sample of optimised end effector trajectory in the Cartesian workspace and Figure 5.4b plots the corresponding joint trajectory obtained for the Stäubli TX200 robot.

5.1.4 Experimental results

The identification procedure described in Section 5.1.3 is applied to determine the base parameters of the rigid body model of the Stäubli TX200 robot. As a reminder, the architecture of this arm is common to most industrial robots with a_1=0.25 m and a_2=0.95

[1] *arg min* stands for the points of the domain at which the function values are minimised.



(a) TX200 end effector path

(b) TX200 corresponding joint trajectory

Figure 5.4: Example of excitation trajectory for the Stäubli TX200 robot

m being the length of the first two moving links in Figure 3.8a (Denavit-Hartenberg parameters). Optimum excitation trajectories were determined by solving the optimisation problem presented in Eqs. 5.13 and 5.14. Five harmonic terms N_f=5 were considered in the Fourier series expansion of the trajectory of each joint in Eqs. 5.10 to 5.12. The fundamental frequency of the excitation trajectory, $\frac{\omega_F}{2\pi}$, was set to 0.1 Hz; a sufficiently low frequency to not excite the joint flexibility. The joint constraints restricting their position and velocity were provided in Section 3.2.1. Since no joint acceleration limit was provided by the robot manufacturer, high values were set ($\ddot{q}_{i,min}$=-8 rad/s^2 and $\ddot{q}_{i,max}$=8 rad/s^2).

Workspace limits **S** are presented in Table 5.7.

x_{min} [m]	x_{max} [m]	y_{min} [m]	y_{max} [m]	z_{min} [m]	z_{max} [m]
-1	2	-2	1	0.5	4

Table 5.7: Workspace limits **S** for the Stäubli TX200 robot

From Eq. 5.13, one may ask if matrix **A** is computed on the basis of the original regressor matrix derived from the first 15 base parameters related to the links or from the full set of parameters presented in Table 5.6. The 15 base parameters related to the links (exposed in Table H.1 in Appendix H) are obtained by applying the method from Mayeda et al. [162] and leads to a regressor matrix $\mathbf{Y} \in \mathbb{R}^{(3\times15)}$, or simply $\mathbf{Y}^{(3\times15)}$. On the other hand, if the full set of parameters is used, another regressor matrix $\mathbf{Y} \in \mathbb{R}^{(3\times27)}$ is constructed; it is later denoted by $\mathbf{Y}^{(3\times27)}$. Without knowing the influence of the base parameters on the condition number of observation matrix **A**, both are used to generate excitation trajectories. As a result, six trajectories are established using $\mathbf{Y}^{(3\times15)}$ and six others are generated using $\mathbf{Y}^{(3\times27)}$. The generation of the twelve excitation trajectories is conducted by starting from various initial values for the optimisation parameter $\boldsymbol{\delta}$. The constrained non-linear optimisation problem is eventually solved using the `fmincon` function of Matlab. Using $\mathbf{Y}^{(3\times15)}$, the average condition number among the six solutions

is 9.5 which can be considered as acceptable relatively to the large size of the workspace. In contrast, when generating the excitation trajectories with $\mathbf{Y}^{(3\times27)}$, the average condition number gets very high and reaches a value of 2528.4. In view of the results presented in the next section, it seems that the design of excitation trajectories has few influence on the identified base parameters, provided that the trajectory is sufficiently fast. This effect was also observed by Waiboer [136] in its Ph.D. stating that a random trajectory with an adequate number of harmonics is already sufficiently exciting. The Stäubli TX200 robot was programmed using the `MoveJ` command without blending between the provided via points in joint space in order to produce smooth motions. Using the *recorder addon* of the Stäubli programming language (VAL3), motor torques and encoder positions were recorded using a sampling rate of 250 Hz on a Personal Computer, and the joint velocity and acceleration signals were obtained by numerical differentiation of the encoder data using the FFT filtering method [166]. Prior the numerical differentiation, a low-pass filter was applied to prevent noise magnification with a tapering from unity to zero from 3 to 4 Hz. Application example of the low-pass filter for one of the excitation trajectories is shown in Figure 5.5: raw torque and position measurements are compared to their filtered version and numerical differentiation leads to the corresponding velocity and acceleration signals.

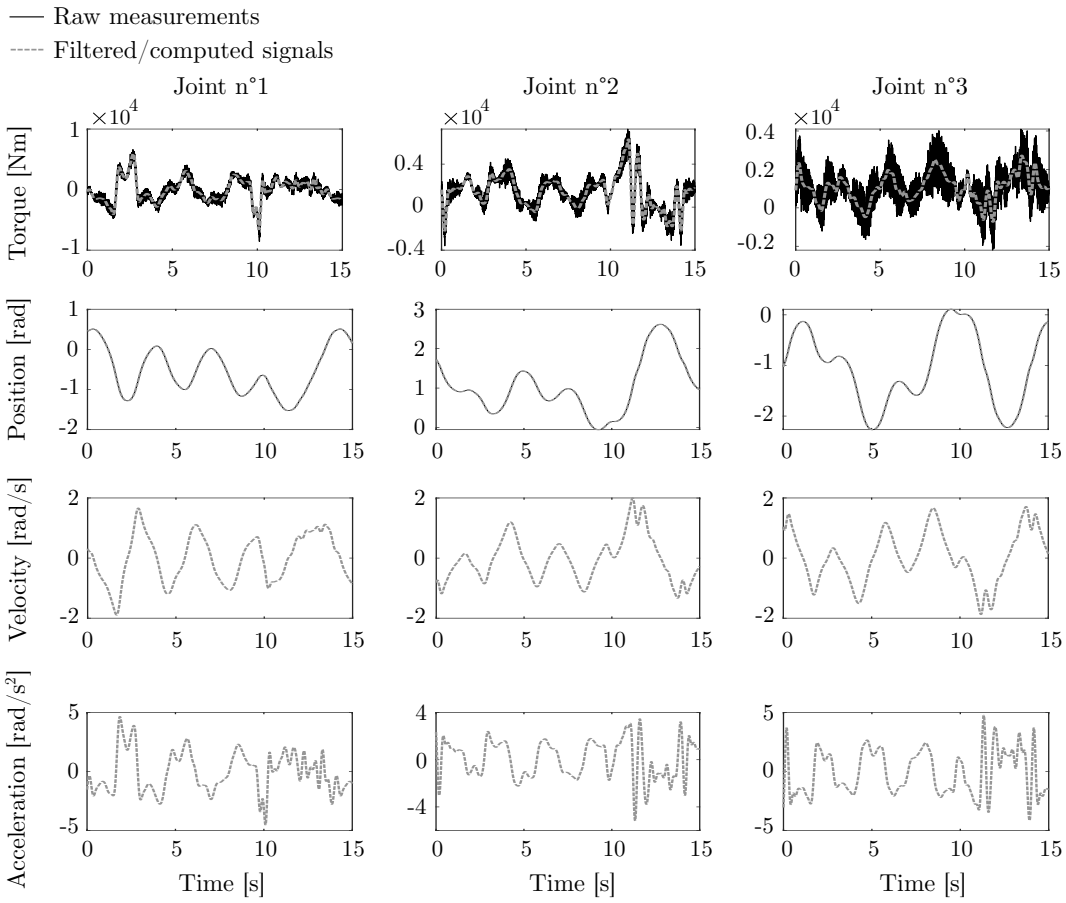


Figure 5.5: Example of raw measurements and filtered/computed signals obtained with the Stäubli TX200 robot

Applying the low-pass filter did not change the position signals which might be due

to the internal filtering of the Stäubli software. Note as well that the Stäubli software directly allows the measurements of the rotor positions thus leading to the reduction ratios. For the Stäubli TX200 robot, reduction ratios of the first three joints are as follows: k_1=k_2=180.0344 and k_3=160 (k_4=150.4, k_5=65.08 and k_6=93.0).

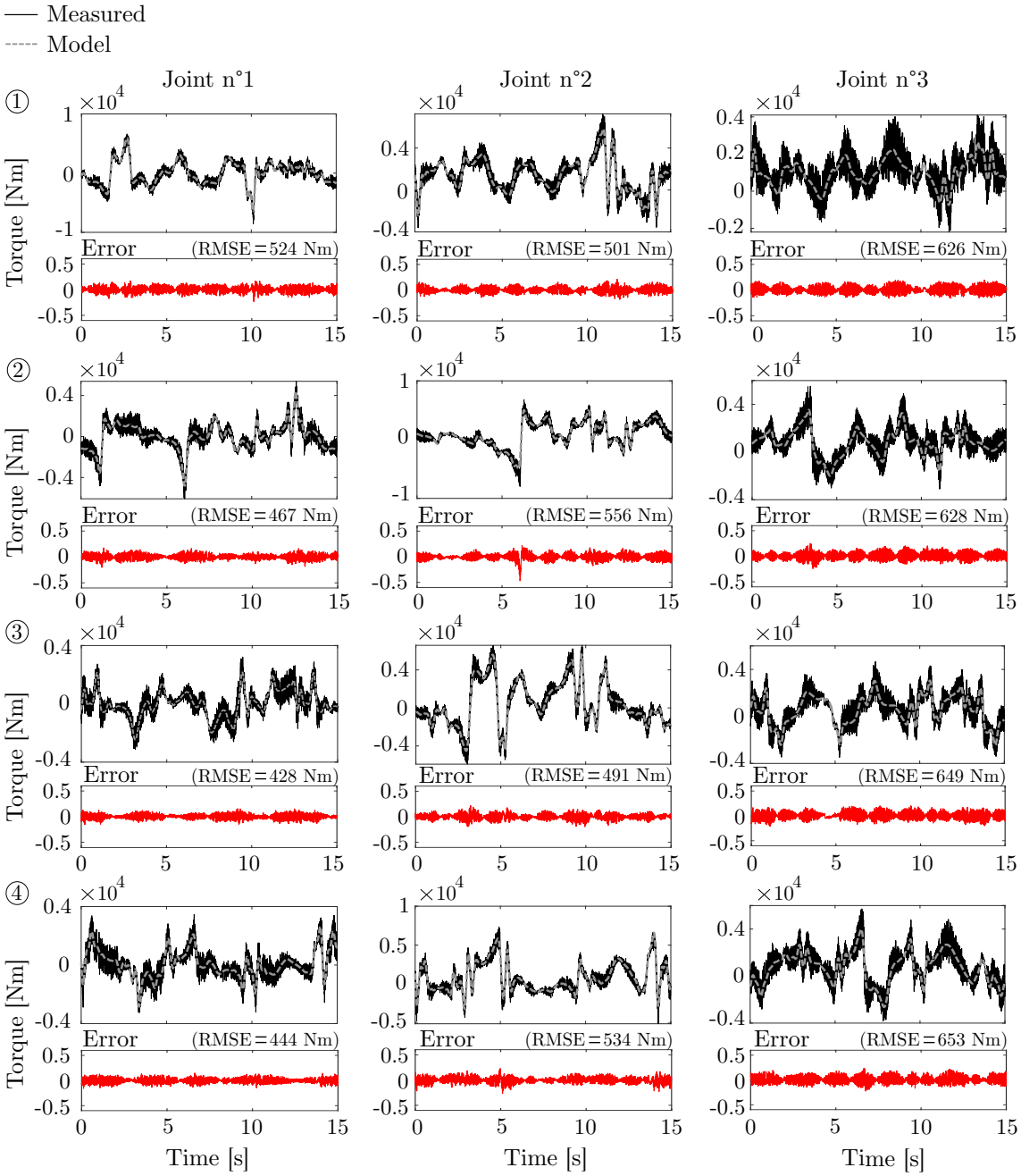


Figure 5.6: Fitting of the measured torques over four excitation trajectories generated with $\mathbf{Y}^{(3\times15)}$ for the Stäubli TX200 robot

Either using $\mathbf{Y}^{(3\times15)}$ or $\mathbf{Y}^{(3\times27)}$ for the design of the excitation trajectory, the 27 parameters, comprising the 18 base parameters along with the friction and gravity compensation coefficients, were estimated by providing the joint and torque measurements obtained from four of the six generated trajectories to the LSE problem defined in Eq. 5.8. Two

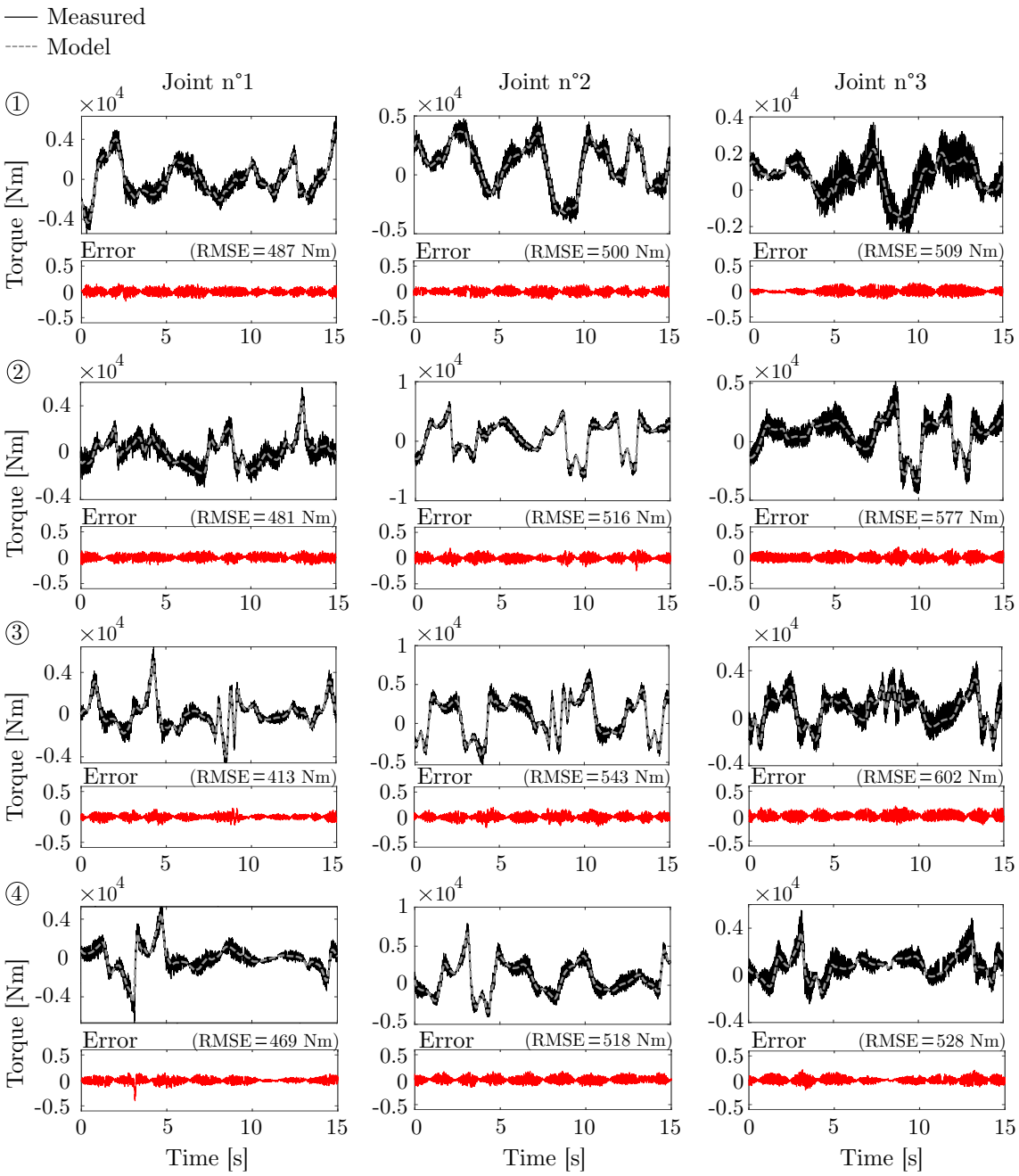


Figure 5.7: Fitting of the measured torques over four excitation trajectories generated with $\mathbf{Y}^{(3\times27)}$ for the Stäubli TX200 robot

sets of 27 parameters are therefore identified. The raw measured and estimated torque signals corresponding to the four trajectories in each case ($\mathbf{Y}^{(3\times15)}$ or $\mathbf{Y}^{(3\times27)}$) are shown in Figures 5.6 and 5.7, as well as the time evolution of their error in Nm. The root mean square error (RMSE) is indicated and has an average value of 527 Nm. Note that even if it is known that the LSE is a natural low-pass filter, it was decided to provide the low-pass filtered torques to the identification procedure for better adjustment of the cut-off frequency (tapering from unity to zero from 3 to 4 Hz). The close agreement of the estimated and measured torque signals confirms the accuracy of the identification of the multibody model of the robot with rigid joints described with the linear form of the

equations of motion in Eq. 5.5.

The 27 identified parameters $\boldsymbol{\pi}$ along with the joint positions measured during the last two trajectories in each case are used in Eq. 5.8 to predict the corresponding joint torque signals. The predicted and measured torque signals are shown in Figures 5.8 and 5.9 for $\mathbf{Y}^{(3\times15)}$ and $\mathbf{Y}^{(3\times27)}$, respectively. The root mean square error has an average value of 531 Nm. As shown in these figures, the identified model accurately estimates the joint torques for arbitrary trajectories.

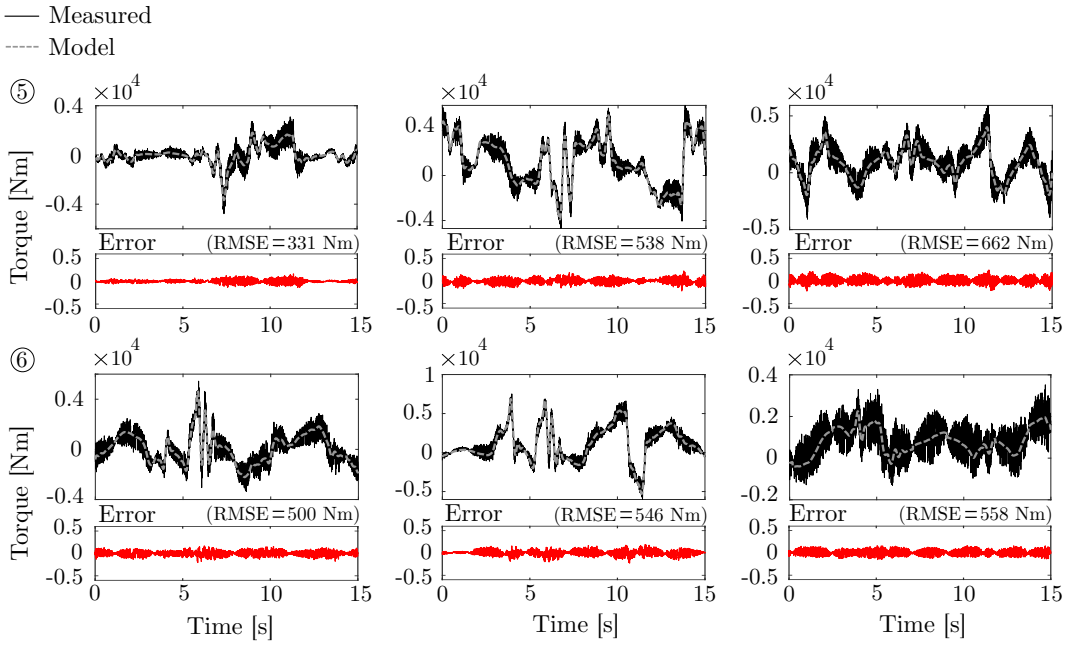


Figure 5.8: Torque prediction over two other measured trajectories generated with $\mathbf{Y}^{(3\times15)}$ for the Stäubli TX200 robot

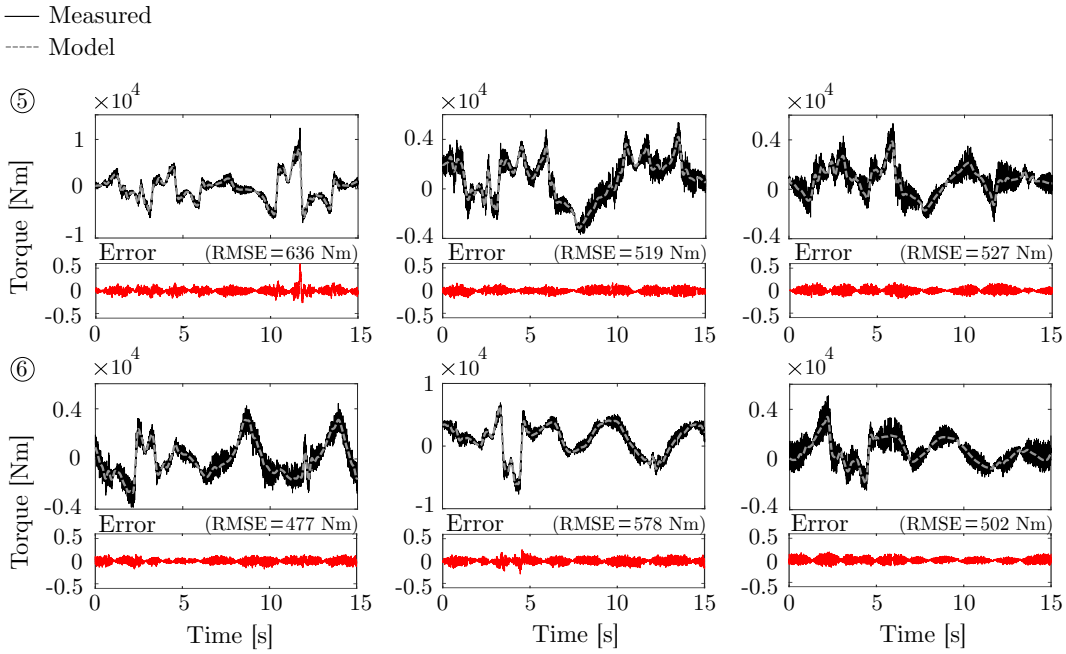


Figure 5.9: Torque prediction over two other measured trajectories generated with $\mathbf{Y}^{(3\times27)}$ for the Stäubli TX200 robot

5.1.5 Validation of the identified inertial parameters

Using the Steiner's theorem on the elementary inertial parameters provided by the robot manufacturer, it is possible to compute the actual values of the base parameters of the Stäubli TX200 robot. Each actual base parameter is then compared with its corresponding estimated parameter. Figure 5.10 proposes an overview of the estimation error in percentage of the first 15 base parameters originating from Table 5.6. Similar error trends are observed for both sets of estimated base parameters $\boldsymbol{\pi}$ either from $\mathbf{Y}^{(3\times15)}$ or $\mathbf{Y}^{(3\times27)}$ used in the trajectory generation. Large errors are noted for parameter number 4 ($C_{y_2}m_2$), 7 (ϕ_{xy_2}), 8 (ϕ_{yz_2}), 13 (ϕ_{xz_3}) and 15 (ϕ_{yz_3}). Errors mainly concern the inertia products which might be less excited than the other base parameters. It is not critical knowing that base parameters leading the dynamics are π_1 and π_2 because of the presence of ϕ_{zz_i} and $\phi_{\mathrm{m},zz_\mathrm{i}}k_\mathrm{i}^2$ in the same parameter. In this case, the errors on π_1 and π_2 are less than 5 %. Figure 5.10 also shows that taking into account the full regressor matrix $\mathbf{Y}^{(3\times27)}$ in the trajectory generation leads to less error in the parameter estimation.

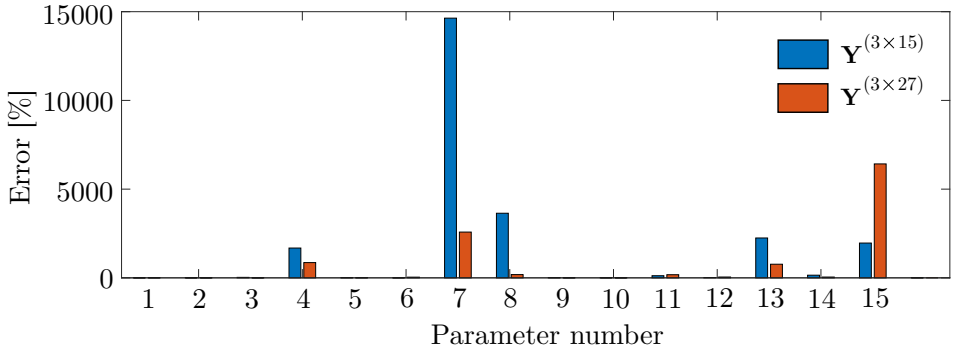


Figure 5.10: Parameter error for the Stäubli TX200 robot

The identified parameters when using $\mathbf{Y}^{(3\times15)}$ and $\mathbf{Y}^{(3\times27)}$ as well as the base parameters computed based on the robot manufacturer's data are shown in Table 5.8. The comparison of the parameters is not provided beyond π_{15} for confidentiality reasons. The friction coefficients $F_{\mathrm{v_i}}$ and $F_{\mathrm{s_i}}$ are given from the joint side. Using a full regressor matrix accounting for the gravity and friction effects seems to improve the accuracy of the parameter estimation. For instance, parameters π_1, π_2, π_3, π_5, π_9 and π_{10} which are not related to inertia products but exhibiting large values are better estimated.

The identification of the robot inertial parameters allows determining the joint elastic parameters on the basis of modal measurements. Indeed, the estimated parameters define the mass matrix of the rigid robot model limited to its first three axes. Parameters obtained with $\mathbf{Y}^{(3\times27)}$ for the trajectory generation are naturally used. Torsional spring and viscous damper are later appended around the motion axis of each of the first three joints. As a result, the elastic parameters of the robot model with joint axial flexibility can be determined using modal measurements. Therefore, next section is concerned with the presentation of modal measurements carried out on the Stäubli TX200 robot. It is followed by a section regarding the fitting of the robot elastic parameters. Lastly, note

		$\mathbf{Y}^{(3\times15)}_{\text{Trajectory}}$		$\mathbf{Y}^{(3\times27)}_{\text{Trajectory}}$	
π_{i}	Stäubli	Identified	Error [%]	Identified	Error [%]
1	914.72	920.76	0.66	952.96	4.18
2	699.49	708.96	1.35	703.06	0.51
3	315.12	227.51	27.80	310.00	1.62
4	-0.55	-9.72	1679.97	-5.24	859.98
5	-283.94	-318.39	12.13	-311.05	9.55
6	-36.82	-38.28	3.98	-51.42	39.64
7	-0.25	-36.84	14636.67	6.20	2580.00
8	0.21	-7.44	3641.41	0.60	183.64
9	126.50	141.21	11.63	131.18	3.70
10	121.50	117.32	3.44	119.75	1.44
11	1.41	3.11	120.57	-1.09	177.02
12	-121.10	-144.59	19.39	-177.94	46.94
13	-0.58	-13.61	2246.86	-5.03	768.08
14	3.78	9.46	150.33	5.50	45.43
15	0.17	-3.20	1961.43	11.22	6421.12
16		0.00927		0.00979	
17		-0.01		0.00	
18		0.00		0.00	
19		-1751.01		-2524.76	
20		73.17		18.90	
21		-231.94		-208.44	
22		841.28		814.35	
23		25.71		10.10	
24		968.77		822.62	
25		14.24		38.82	
26		555.45		461.62	
27		94.71		137.15	

Table 5.8: Comparison of the actual and identified base parameters for the Stäubli TX200 robot

that the same procedure was applied to the identification of the base parameters of the KUKA KR90 R3100 HA robotic arm thanks to a collaboration with the University of Victoria in Canada. Results are presented in Appendix J.

5.2 Experimental modal analysis

Modal analysis techniques are commonly used to identify the dynamic behaviour of mechanical structures. In this particular situation, experimental modal analysis is adopted in order to identify the natural frequencies, dampings and mode shapes of the robot structure. The method does not require any particular set-up and is therefore easy to implement. In contrast, operational modal analysis allows determining the robot dynamic

behaviour when it is moving but requires a post-processing of the measured signals and shaker tests bring complexity to the set-up. On the basis of FRF measurements, the aforementioned characteristics are determined and later used into a fitting procedure to infer the values of the torsional spring and damping constants representing the joint flexibility. The so-called *roving hammer* technique is applied to the Stäubli TX200 robot in several static configurations. Namely, the robot is excited through hammer impacts distributed all over its structure. The impact force and the resulting vibrations are measured to construct FRFs. In this section, the set-up and measuring conditions are first described before presenting the results gathered in the milling pose (Figure 5.2), treated as the reference pose. Afterwards, other results obtained in other typical poses are exposed. The latter are later exploited to address the validation of the posture dependency of the robot model.

5.2.1 Experimental set-up

Modal properties of the Stäubli TX200 robot were identified by conducting a set of experimental modal analysis tests using the roving hammer technique. The set-up is pictured in Figure 5.11. It is composed of an acquisition hardware to gather the signals generated by the force sensor of the hammer and the accelerometer. The signals are measured simultaneously. The acquisition hardware is connected to a personal computer for data post-processing, e.g. FRFs computations and mode shape visualisation.

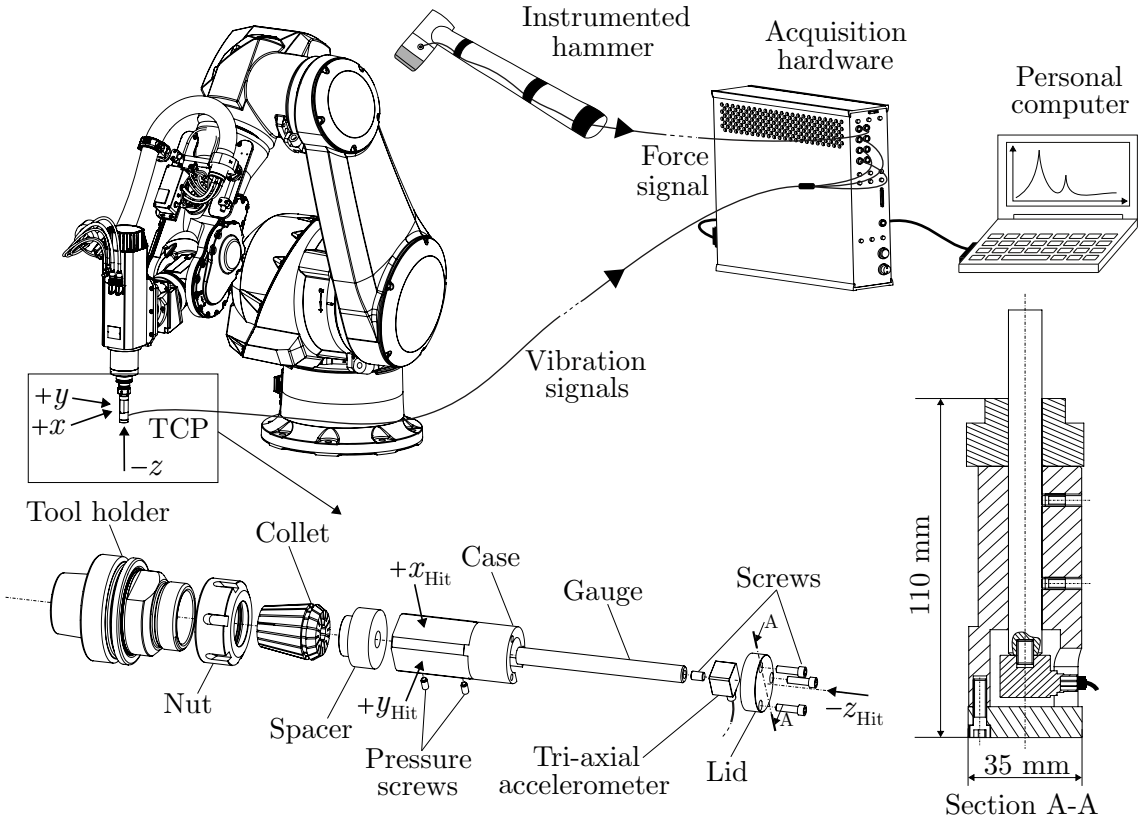


Figure 5.11: Acquisition chain for hammer test on the Stäubli TX200 robot

The hammer that is utilised is in fact a modified sledgehammer whose weight of about 1

kg is sufficient to excite the robot modes with an impact. The sledgehammer is modified with the inclusion of a force sensor between the tip and the handle. Force sensor is the Dytran 1061V2 with a sensitivity of 890.47 N/V. The selected hammer tip is a soft impact tip made of rubber to excite the lower frequencies (<200 Hz) of the robot. A hard plastic tip was also used to excite a wider range of frequencies including the spindle modes. Regarding the accelerometer located at the TCP, it is the tri-axial accelerometer Dytran 3093B1 (serial number 1345) with a sensitivity of 100 mV/g. As depicted in Figure 5.11 in the split view, the tri-axial accelerometer is mounted at the TCP through a cylinder gauge, with a diameter similar to the end mill, which is tightened in the tool holder. The tri-axial accelerometer is screwed to the gauge. In order to not directly hit the accelerometer when the FRFs at the TCP are assessed, which would cause improper measurements due to channel overloading, an aluminium case surrounds the sensor. The sectional view of Figure 5.11 shows that the case is rigidly attached to the cylinder gauge via two pressure screws. Two flat areas are machined on the cylinder case to properly hit the TCP in the $+x$ and $+y$ directions, corresponding to the local axes of the tri-accelerometer. Analogously, the lid of the case allows a clean hit along the $-z$ direction of the tri-axial accelerometer. Note that during the tests, local X-direction of the tri-axial accelerometer is always made parallel to the robot plane (formed by the arm and forearm links). Force and vibration signals are acquired using data acquisition hardware SCADAS SC305 commercialised by Siemens. It is a modular system in which different cards are nested to meet the user needs. Card SP90 includes an anti-aliasing filter and a 16-bit A/D data conversion. After digitisation, the measured force and acceleration signals are transferred to a personal computer (PC). Frequency response functions are eventually computed in LMS Test.Lab Rev. 8A on the PC. Modal parameter identification (natural frequency, damping ratio and mode shape) is also carried out in LMS software in which the time domain Least Squares Complex Exponential (LSCE) method is implemented. Succinctly, the LSCE method allows estimating the values of modal frequency and damping for several modes simultaneously. The structure is excited at one single point and responses are measured at several locations. It is therefore a Single-Input Multi-Output (SIMO) method leading to global estimates of modal parameters [167].

5.2.2 Measuring conditions

Regarding the acquisition parameters used to record vibration and force signals, preliminary hammer impacts are given to the robot structure to determine them. First the bandwidth is chosen to suit the needs of modal identification; here, the robot modes or the spindle modes. Hence, there are two categories of hammer tests. To measure the robot modes, a bandwidth of 200 Hz with a frequency resolution of Δf=0.39 Hz is selected knowing that it is reported that the main flexible modes of industrial robots are under 30 Hz. It is not advised to lower the frequency resolution since it would lead to low signal to noise ratio and therefore bad coherence in measurements. In contrast, to assess the spindle modes, a bandwidth of 4096 Hz is chosen with a frequency resolution of Δf=1 Hz. Having two categories of hammer tests permits the selection of suitable frequency resolution maximising the signal to noise ratio. The acquisition duration is computed from the chosen frequency resolution such as $T_{\text{Acquisition}}=\frac{1}{\Delta f}$. Next, acquisition software

allows defining the windowing of the measured signals. As it is a common practice with impacts tests, a force window is defined for the measured force signals and an exponential window is selected for the vibration measurements. The force window is a rectangular window spanning over 5.3 % of the acquisition duration. The exponential window multiplies the measured vibration signals by a decaying exponential to reduce the leakage effect. A decay of 37.6 % provided satisfactory results during the preliminary tests. Note as well that when performing the roving hammer, four measurements are carried out at each point to average the computed FRFs. Acquisition parameters are summarised in Table 5.9 for the two categories of hammer tests.

	Low-frequency modes	High-frequency modes
Hammer tip	Rubber	Hard plastic
Bandwidth [Hz]	200	4096
Sampling frequency [Hz]	400	8192
Frequency resolution Δf [Hz]	0.39	1
Acquisition duration [s]	2.56	1
Force window [%]	5.3	5.3
Exponential window [%]	37.6	37.6
Number of measurements per point	4	4

Table 5.9: Acquisition parameters for the experimental modal analysis

In order to visualise the robot mode shapes with a sufficient accuracy, it was decided to spread 77 locations of impact points all over the manipulator structure. The map of impact points is presented in Figure 5.12.

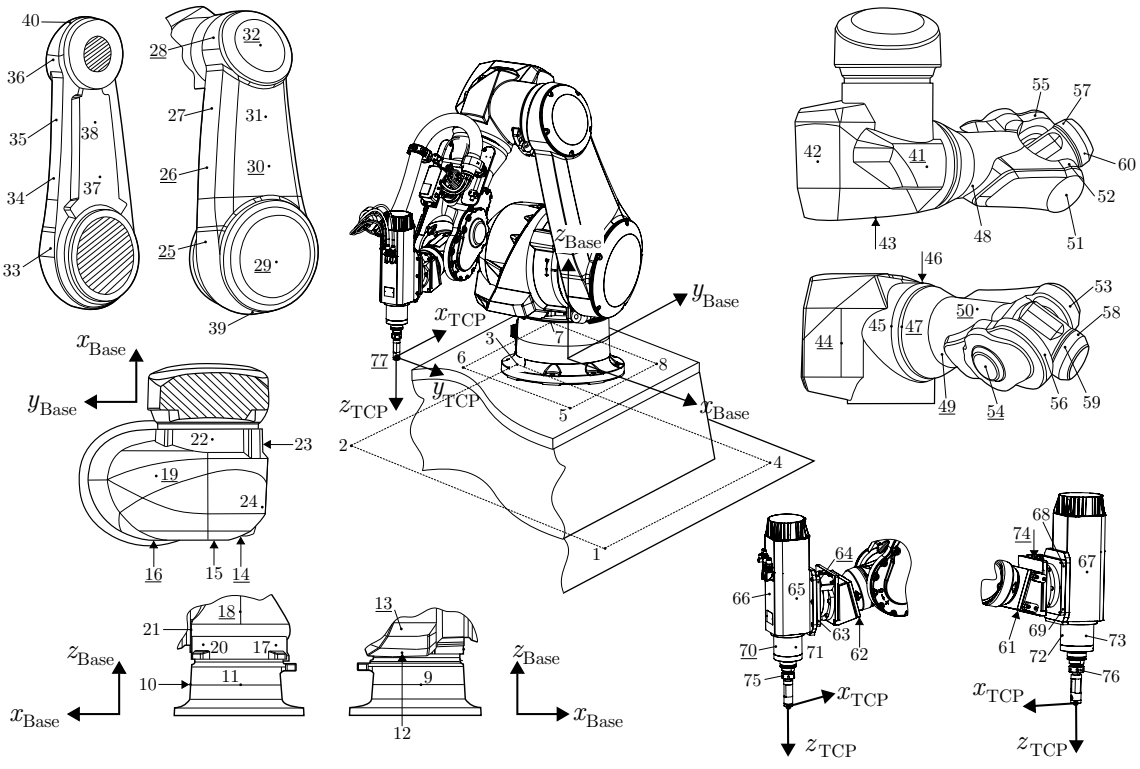


Figure 5.12: Impact point map of the Stäubli TX200 robot

As observed, measurement points are distributed all over the robot and spindle surfaces to gather a spatial discretised view of the robot structure. Four measurement points are also located on the steel slab to assess the flexibility with the robot base (points 5, 6, 7 and 8 form a square on the top surface) and four others are placed on the ground (points 1, 2, 3 and 4). Such a high number of impact points allows a global visualisation of the mode shapes in LMS Test.Lab. However, assuming that the links are rigid below 200 Hz, correlations between the measured and simulated mode shapes will be carried out on 21 points leading to a compact geometric representation of the robot (underlined points in Figure 5.12). Of course, a measurement point is located at the TCP, at the location of the tri-axial accelerometer. Each impact point is positioned with respect to the base frame and defines the origin of a local frame whose Z axis is made perpendicular to the robot surface.

As such, all hammer impacts are given perpendicularly to the surfaces of the robot. As depicted in Figure 5.13, if an impact point is situated on a tilted surface, the direction of hammer impact is oriented such that it corresponds to the Z axis of the local impact frame. This precaution is taken in order to excite the robot with a maximum energy transmitted to the structure.

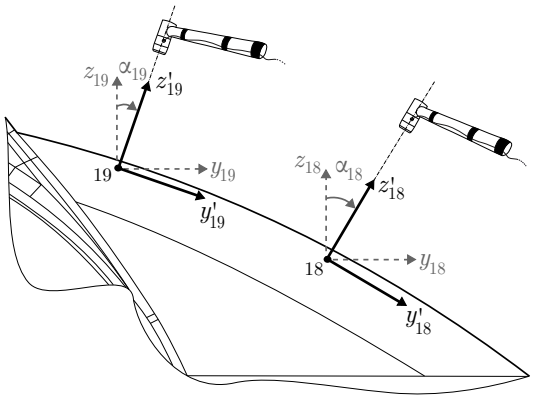


Figure 5.13: Hammer impacts perpendicular to the local surface

At the TCP, beside the impact along the local Z direction, two more impacts are given along the X and Y directions, respectively. The frequency response function matrix $\mathbf{H}(\omega)$ is therefore constructed on the basis of the measured direct and cross FRFs at the TCP. As a result, for each posture in which the robot modes are assessed, a total of 237 FRFs are measured. The same map of impact points is used throughout all the postures. Note that if the spindle modes are appraised, only the FRF matrix $\mathbf{H}(\omega)$ at the TCP is measured.

During the measurement campaign of each posture, coherence and amplitude of each measured frequency response function were checked to ensure the data quality. Coherence function is an indicator varying from 0 to 1 and appraises the quality of measurement [167]. Unless otherwise specified, impacts exciting the robot structure were given while the brakes of the actuator were disengaged and the robot controller was in action.

5.2.3 Analysis in the milling posture

The reference posture for the experimental modal analysis tests is the milling posture which is illustrated in Figures 5.2 and 5.12. The robot is set in a configuration for typical

milling operations with a spindle axis aligned with vertical axis z_{Base} and the TCP is at about 100 mm above the bench vice. Figure 5.14 depicts the actual Stäubli TX200 robot in the milling posture along with its geometric model in LMS Test.Lab. Each node of the geometric model represents one impact point location. The spatial discretisation of the robot is used to animate the mode shapes on the basis of the modal analysis results.

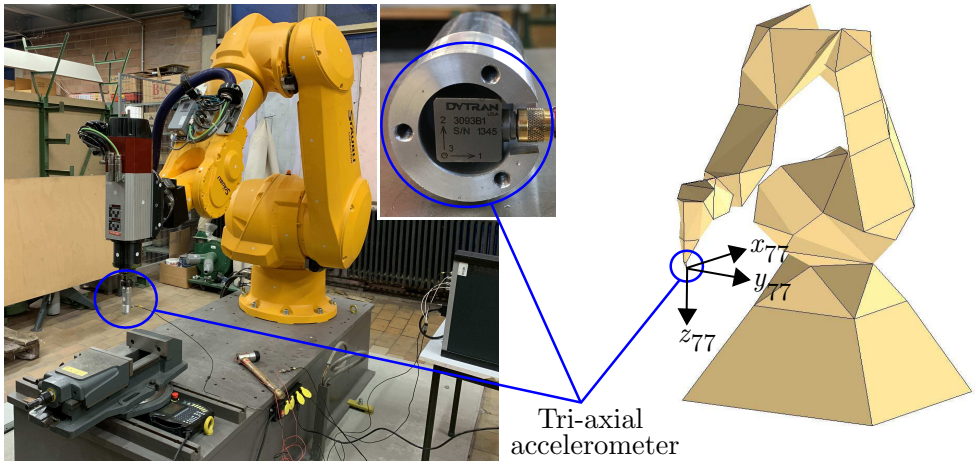


Figure 5.14: Spatial discretisation of the Stäubli TX200 robot in the milling posture

Implementing the roving hammer technique, impulses were applied to the 79 points distributed on the robot surfaces, including the three X, Y, and Z directions at the TCP, points x_{77}, y_{77} and z_{77}. It means that for one complete experimental modal analysis, 79 $\times$ 3 FRFs (n_{FRF}) are measured. Frequency response function $H(\omega)$ is defined as the ratio of the Fourier transforms of the response denoted by $X(\omega)$ divided by the exciting input $F(\omega)$ such as

$$H(\omega) = \frac{X(\omega)}{F(\omega)}. \tag{5.15}$$

For machining applications, FRFs are typically expressed in the receptance format (μm/N) to assess the deviation of the TCP under a force. As it is the case in this work, they are measured in the inertance format ((m/s^2)/N) using accelerometers and then converted to the receptance format. Although measurements at the TCP are sufficient in machine tool, complete experimental modal analyses must be carried out in robotic applications to account for the posture dependency. In other words, the changes in frequency, damping and mode shape with the robot configuration must be quantitatively evaluated. As a result, four cases are examined in the milling posture to appraise various effects:

1. EMA with controller: a complete experimental modal analysis is carried out with the controller in action and brakes are disengaged. FRFs are measured over a bandwidth of 200 Hz. Results are extensively commented as it is the reference case to which effects are compared.

2. EMA with the motor brakes: a complete experimental modal analysis is achieved but the power of the robot is shut down and brakes are engaged. FRFs are also measured over a bandwidth of 200 Hz.

3. Spindle modes: FRFs are only measured at the TCP using the extended bandwidth of 4096 Hz.

4. Mode evolution: FRFs are measured over a bandwidth of 200 Hz at the TCP in two other configurations near the milling posture.

The four cases are discussed and compared hereafter.

1. EMA with controller

All raw measured FRFs in the inertance format[2] (g/N) are superimposed in Figure 5.15. It is observed that the majority of the robot modes are situated under 100 Hz. Only two modes are found between 100 and 200 Hz. Irrespectively of the amplitudes, clusters of peaks are easily detectable.

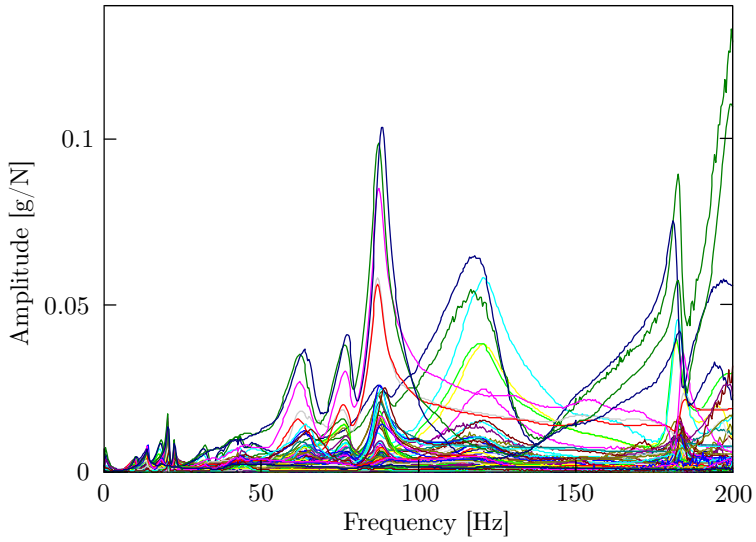


Figure 5.15: Superimposition of all measured FRFs in the controlled milling posture of the Stäubli TX200 robot

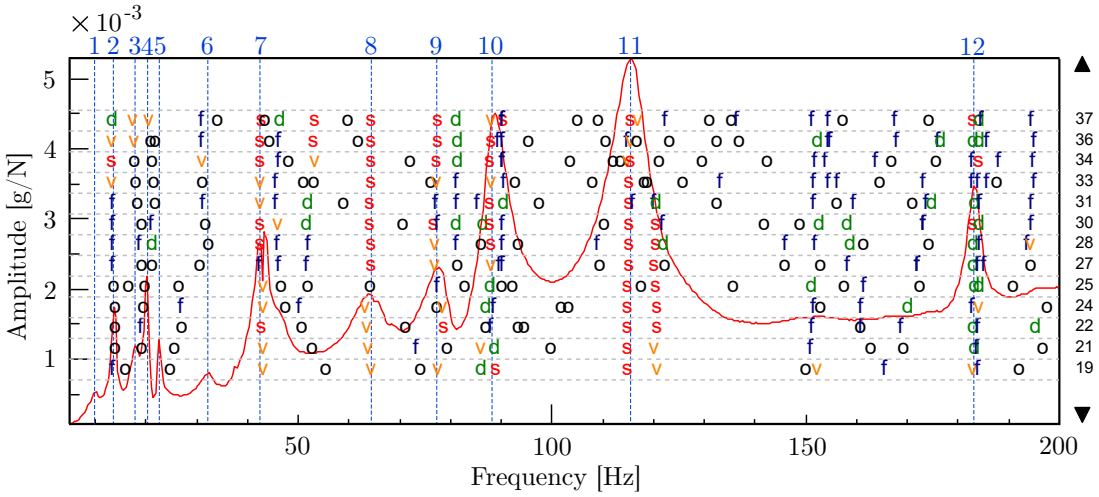
Using the summation function indicator on all measured FRFs defined as

$$I_{\mathrm{SUM}} = \frac{1}{n_{\mathrm{FRF}}} \sum_{\mathrm{i}}^{n_{\mathrm{FRF}}} \left| H_{\mathrm{i}}(\omega) \right|, \tag{5.16}$$

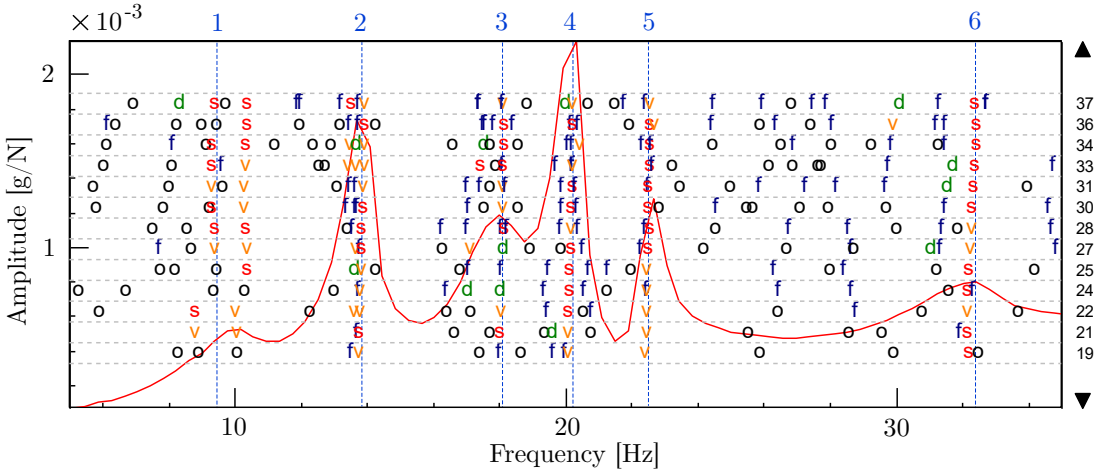
the time domain least squares complex exponential method is applied to identify the modal parameters. The summation indicator allows easily identifying the natural frequencies of the system through the prominent peaks of its curve. It leads to a stabilisation diagram in which I_{SUM} is displayed along with the stability analysis of each mode (Figure 5.16a).

[2]Knowing that 1 g = 9.81 m/s^2.

A streak of "s" characters indicates that the identification algorithm converges to a stable mode at the corresponding frequency. The meaning of the other symbols is presented as follows: "o" pole not stable, "f" pole is stable in frequency, "d" pole is stable in frequency and damping, "v" pole vector is stable, "s" pole is completely stable. If the whole bandwidth of 200 Hz is considered, few streaks of stable poles are observable around the frequency peaks (Figure 5.16a). Indeed, it is advised to sweep the whole diagram by limiting the analysis to frequency windows of a few dozen of Hz for a better modal identification. Figure 5.16b presents the modal identification for a frequency window between 5 and 35 Hz. Six streaks of stable poles are uncovered leading to the identification of the frequency, damping ratio and mode shape of the first six robot modes. If two streaks of stable poles are related to one peak, streak with the lowest damping ratio is retained. The procedure is repeated for the higher frequencies.



(a) Bandwidth of 200 Hz



(b) Bandwidth of 35 Hz

Figure 5.16: Stabilisation diagrams in the controlled milling posture of the Stäubli TX200 robot

A total of 12 modes are identified under 200 Hz. However, since the main modal content is under 100 Hz, it is decided to leave out the last two modes. The identified modal parameters corresponding to the first 10 robot modes are presented in Table 5.10.

Mode	1	2	3	4	5	6	7	8	9	10
Frequency [Hz]	9.4	13.7	17.7	20.2	22.5	32.4	42.5	64.2	78.2	88.7
Damping ratio [%]	7.9	2.2	4.5	1.0	0.8	2.7	1.1	4.7	2.8	2.3

Table 5.10: Measured natural frequencies and damping ratios in the controlled milling posture of the Stäubli TX200 robot

Using the first 10 modes, it is possible to estimate the accuracy of the modal parameter identification by comparing the measured and synthesized FRFs. The latter are obtained by using the identified modal parameters. Only the 9 FRFs at the TCP are compared in the receptance format from Figures 5.17 to 5.19 since they constitute the frequency response function matrix $\mathbf{H}(\omega)$ to which the milling forces are applied.

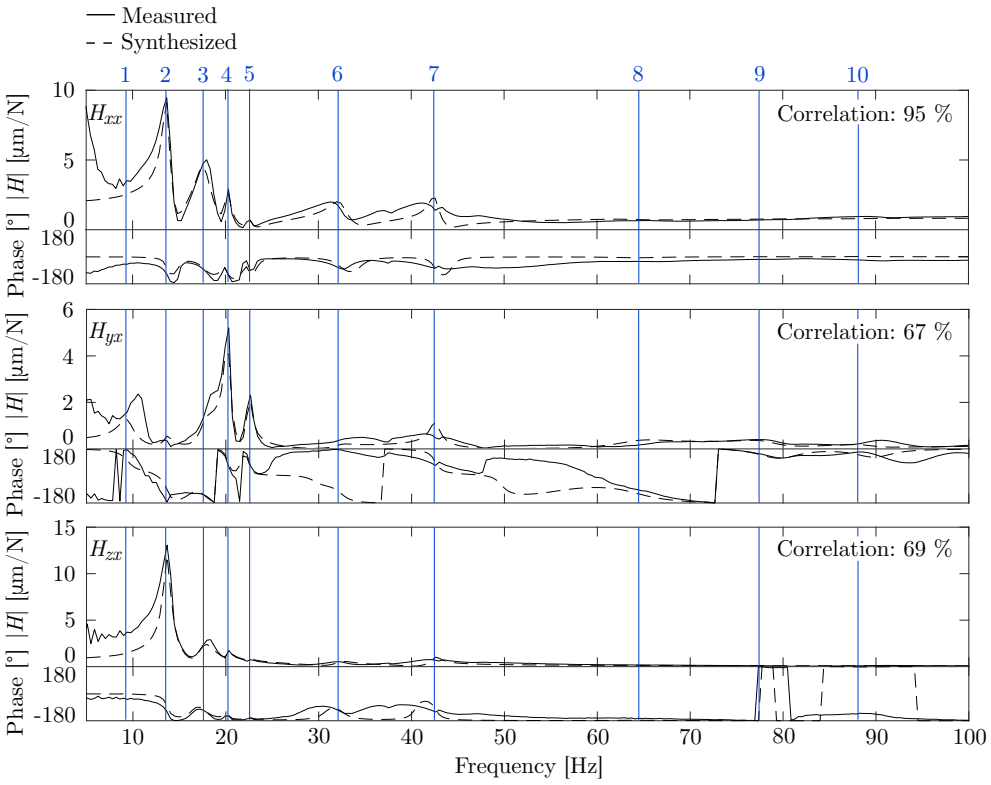


Figure 5.17: Comparison of measured and synthesized FRFs for an impact at the TCP along the X direction in the controlled milling posture of the Stäubli TX200 robot

The average correlation percentage of the measured and synthesized FRFs at the TCP is 77 %, which is acceptable considering a bandwidth of 100 Hz. Note that if the two detected modes between 100 and 200 Hz are involved, the average correlation percentage decreases to 55 %. Highest correlation percentage (97 %) is obtained for direct FRF H_{yy}. The Y direction is perpendicular to the robot plane (Figure 5.14) and exhibits large deflections on the basis of the FRF which can explain the high correlation value. The largest deflections are measured along the Z direction for H_{zz}. It confirms the relevance of the transposition of the stability analysis in milling to the 3D case (Chapter 4). On the other hand, the lowest correlation values are observed for cross-FRFs H_{zy} (41 %) and H_{yx} (67 %) as they present small deflections. It can also be noticed that, in addition

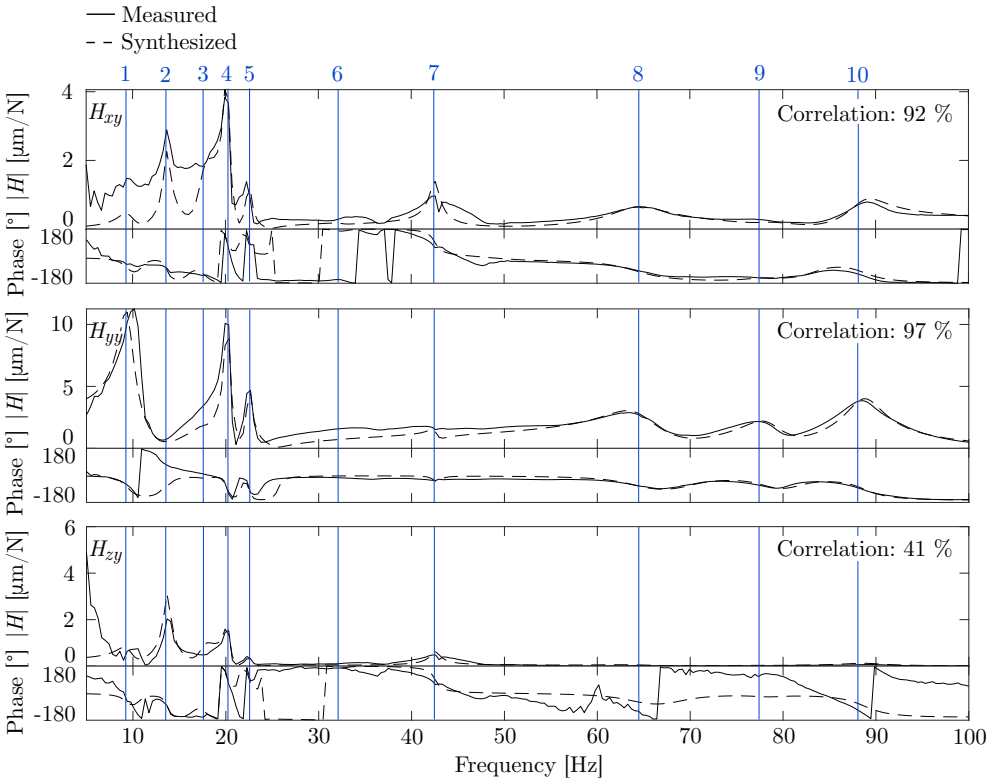


Figure 5.18: Comparison of measured and synthesized FRFs for an impact at the TCP along the Y direction in the controlled milling posture of the Stäubli TX200 robot

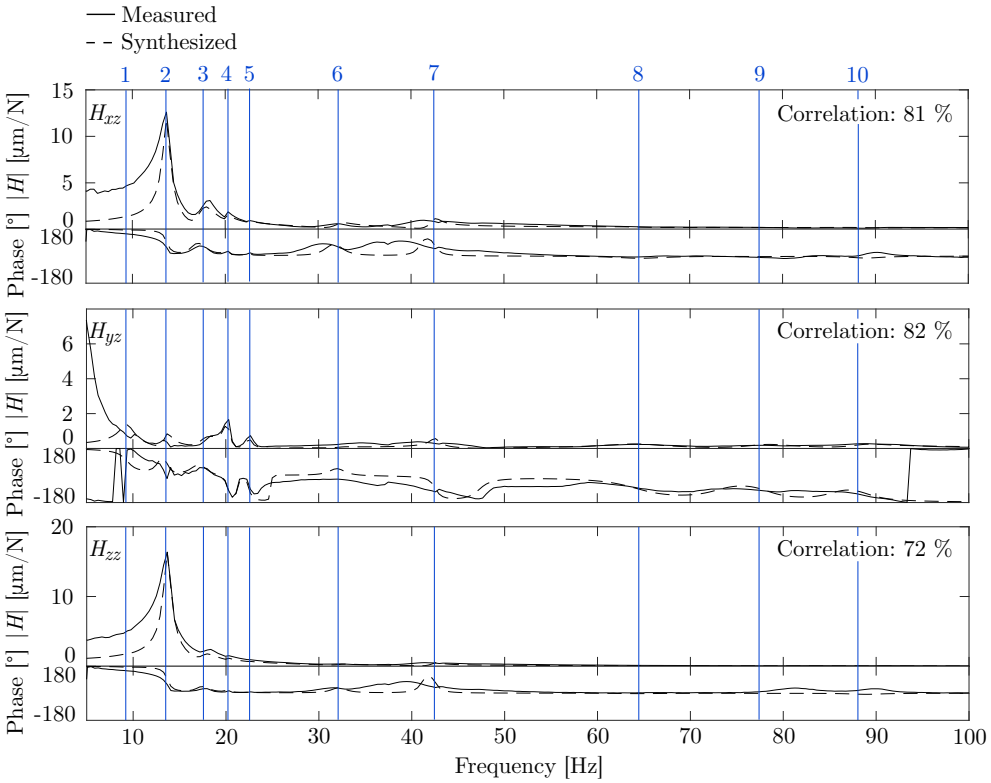


Figure 5.19: Comparison of measured and synthesized FRFs for an impact at the TCP along the Z direction in the controlled milling posture of the Stäubli TX200 robot

to being a few times more rigid than the direct-FRFs, cross-FRFs show a considerable level of asymmetry (e.g. $H_{xy} \neq H_{yx}$) which confirms that robots are structurally highly non-linear. In other words, the impact force slightly affects the level of FRFs. Inaccuracy in the impact direction during the hammer test might also amplify the phenomenon. The non-symmetry of the FRF matrix in articulated industrial robots has been reported in other research as well [104, 168] and is a subject for further studies.

Overall, measured FRFs at the TCP reveal that the modal content of interest for a milling application is situated below 35 Hz. Indeed, in the receptance format, it appears that dominant frequency peaks are mostly concentrated in this region. The shape, frequency and modal damping ratio of each of the ten identified modes in the controlled milling posture are shown in Figure 5.20.

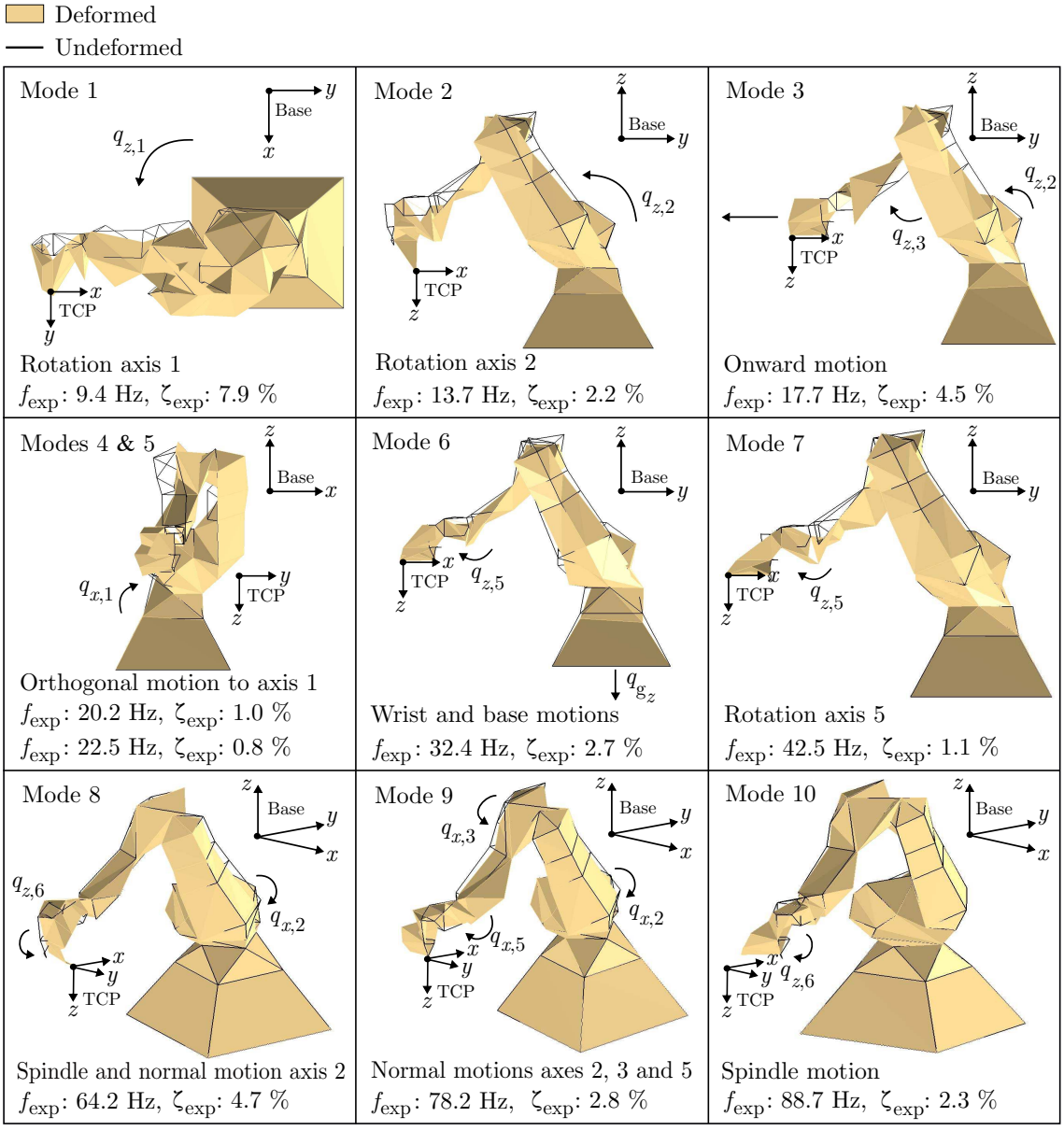


Figure 5.20: Identified mode shapes in the controlled milling posture of the Stäubli TX200 robot using the animations generated by the LMS Test.Lab software

According to the modal analysis results demonstrated in Figure 5.20, all the identified mode shapes originate from the deflections at the joints and the links appear as rigid. In the figure, base frame is moved to the upper right corner of the frame of each mode and TCP frame is also shown to make the connection between the mode shape motion and the corresponding peaks in the measured FRFs. From the animations generated by the LMS Test.Lab software, the first ten identified mode shapes can be described as follows:

1: Mode one is greatly influenced by the stiffness of joint one around its rotation axis ($q_{z,1}$). Hence, mode shape associated to mode one showcases a rotation of the whole robot structure around the first motion axis. TCP frame therefore moves along y_{TCP}. In measured FRF H_{yy} in Figure 5.18, it corresponds to the first peak at 9.4 Hz. First mode is highly damped with a damping ratio of 7.9 %.

2: Mode two is formed by the deflection of the second joint around its rotation axis ($q_{z,2}$). The resulting motion of the robot is a downward movement of the TCP frame (z_{TCP} direction). Main peak associated to the second mode is observed in FRF H_{zz} at 13.7 Hz in Figure 5.19. The mode also has a component along x_{TCP} as FRF H_{xx} tells in Figure 5.17. Damping ratio is comparable to the other remaining modes at 2.2 %.

3: Mode three involves the simultaneous rotations of the second and third joints around their respective rotation axis ($q_{z,2}$ and $q_{z,3}$). The produced robot motion is an onward motion along x_{TCP} as observed with the third peak at 17.7 Hz in H_{xx} in Figure 5.17.

4: Mode four results from the deflection of the first joint around an axis perpendicular to its motion axis ($q_{x,1}$) combined with a small motion of the first joint around its rotation axis. The resulting motion is a falling robot towards direction x_{Base} but with a spindle moving in the opposite direction along direction y_{TCP}. This very flexible mode is therefore highlighted in FRF H_{yy} at 20.2 Hz in Figure 5.18. In the context of a milling operation taking place in the robot plane (along y_{Base}), it can be understood that the lateral milling forces (along y_{TCP}) can easily excite this mode exhibiting a low damping ratio ($\zeta_{\text{exp},4}$=1.0 %).

5: Mode five is a weakened version of mode four thus exhibiting a similar mode shape. The auto-MAC matrix of the identified mode shapes, comparing the similarities between mode shape i with mode shape j, is computed with

$$\text{MAC}_{\text{ij}} = \frac{\left(\boldsymbol{\psi}_{\text{i}}^T \boldsymbol{\psi}_{\text{j}}^*\right)^2}{\|\boldsymbol{\psi}_{\text{i}}\|^2 \|\boldsymbol{\psi}_{\text{j}}\|^2}, \tag{5.17}$$

with $\boldsymbol{\psi}_{\text{i}}$ the mode shape vector i $\in \mathbb{R}^{79\times 1}$ and $\boldsymbol{\psi}_{\text{j}}^*$ the conjugate of mode shape vector $\boldsymbol{\psi}_{\text{j}}$ [167]. Figure 5.21 presents the computed auto-MAC matrix of the controlled robot in the milling posture. It appears that modes four and five are very similar with a correlation value of 96 % i.e. mode shapes of modes four and five are almost the same. However, their frequencies are slightly different and peak of mode five is

encountered at 22.5 Hz still in FRF H_{yy} in Figure 5.18. Damping ratio is similar at 0.8 %. A closer look at H_{yy} in Figure 5.18 in the region around 21 Hz may suggest that a former and larger peak around 21 Hz was divided into two smaller peaks. This hypothesis brings the unverified existence of a dynamic vibration absorber (DVA introduced in Section 3.5) damping out modes originating from a deflection perpendicular to the motion axis of the first joint ($q_{x,1}$). Although unverified with the actual Stäubli TX200 robot, the implementation of the DVA in the multibody model allows reproducing the splitting of the former peak into two smaller peaks.

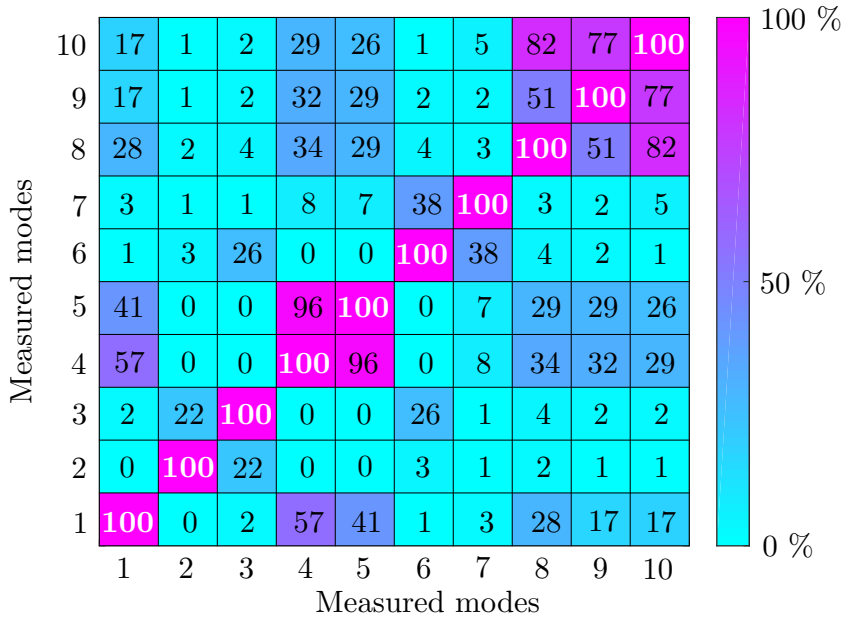


Figure 5.21: Auto-MAC matrix of the measured modes of the Stäubli TX200 robot over a bandwidth of 100 Hz

6: Mode six combines the motions of the fifth joint around its motion axis ($q_{z,5}$) and the lifting of the robot base along direction z_{Base} (q_{g_z}). Main peak corresponding to the sixth mode is observed in FRF H_{xx} at 32.4 Hz in Figure 5.17.

7: Mode seven is somewhat similar to mode six except that there is no motion from the robot base. Only a rotation around the motion axis of the fifth joint ($q_{z,5}$) is visible in the animation. Consequently, main peak corresponding to the seventh mode is also observed in FRF H_{xx} but at 42.5 Hz in Figure 5.17.

8: Mode eight also combines the motion of two rotational motions: one around the robot flange ($q_{z,6}$) and a perpendicular deflection to the motion axis of the second joint ($q_{x,2}$). As the sixth axis rotates, the main component of the motion is captured along direction y_{TCP} as shown in H_{yy} with the peak at 64.2 Hz in Figure 5.18.

9: Mode nine seems to only involve perpendicular deflections to motion axes 2 ($q_{x,2}$), 3 ($q_{x,3}$) and 5 ($q_{x,5}$). The complex motion reduces to a peak at 78.2 Hz in H_{yy} in Figure 5.18. Note that animations of modes eight and nine are particularly difficult to describe as they suggest perpendicular deflections to motion axes different from the first joint.

10: Mode ten only exhibits a deflection around the motion axis of the sixth joint ($q_{z,6}$). Main component of the motion is thus observed along direction y_{TCP} as shown in H_{yy} with the peak at 88.7 Hz in Figure 5.18.

2. EMA with the motor brakes

The same experimental modal analysis is carried out on the Stäubli TX200 robot except that the robot is completely turned off and motor brakes are engaged. During the hammer tests, it is noticed an improvement of the coherence function for frequencies over 100 Hz. Indeed, when the robot is turned on, the fan of the controller generates noise which affects vibration measurements. Consequently, identification of modal properties is handled easily using stabilisation diagrams as before where longer streaks of stable poles are observed.

Although impacts are given on all the measurement points distributed on the robot, only direct FRFs H_{xx}, H_{yy} and H_{zz} at the TCP are reported in Figure 5.22 and compared to the same FRFs measured when the controller was in action. As observed, frequency peaks occur at the same locations for the considered frequency bandwidth. However, their amplitude is damped out when the controller is in action. The effect of controller damping is particularly pronounced for frequencies beyond 40 Hz as H_{xx} tells. An explanation can be found in the fact that hammer tests along x_{TCP} are made in the direction of rotations of joints two, three and five which are regulated by the controller. In contrast, for H_{yy}, impact along y_{TCP} mostly affects a direction perpendicular to the motion axes which are not under control. It might explain the shape of FRF H_{yy}. FRF H_{zz}, only depicting one major frequency peak around 14 Hz is greatly affected by the controller considerably reducing its amplitude.

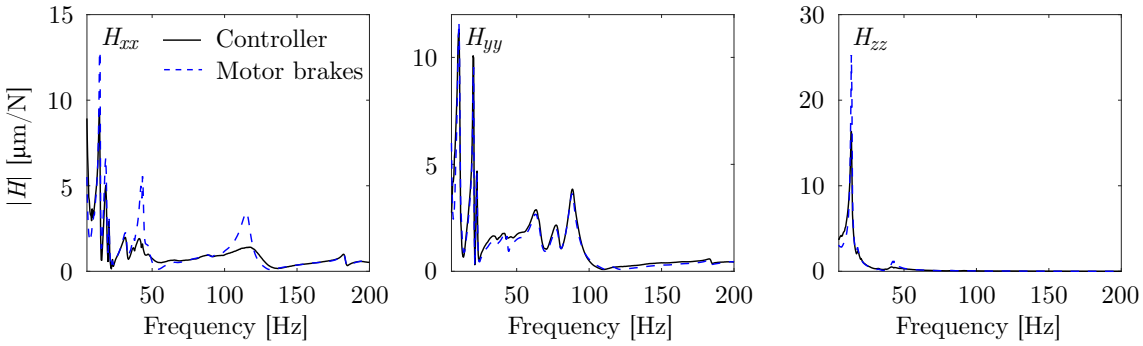


Figure 5.22: Effect of the controller on the direct FRFs of the Stäubli TX200 robot

Regarding the resulting mode shapes, animations provided by LMS Test.Lab software illustrate the same robot motions as presented in Figure 5.20. Overall, even if controller has a slight effect on the modal characteristics, author still advises to achieve experimental modal analysis of manipulator with the controller in action, as it would be during a milling operation. Performing an experimental modal analysis when the manipulator is turned off allows clearly identifying the mode shapes and natural frequencies of the system.

3. Spindle modes

Hammer impacts are only given at the TCP when the controller is in action and the frequency bandwidth is set to 4096 Hz. The purpose is to evaluate the flexibility of the spindle modes over the modes originating from the robot joints. As the main frequency peaks are observed in direct FRFs, Figure 5.23 shows the raw measurements of H_{xx}, H_{yy} and H_{zz} in the inertance format (g/N). Four major frequency peaks are observed at 729, 826, 1129 and 1575 Hz in FRFs H_{xx} and H_{yy}. Frequency peaks in H_{zz} are more rigid and thus disregarded.

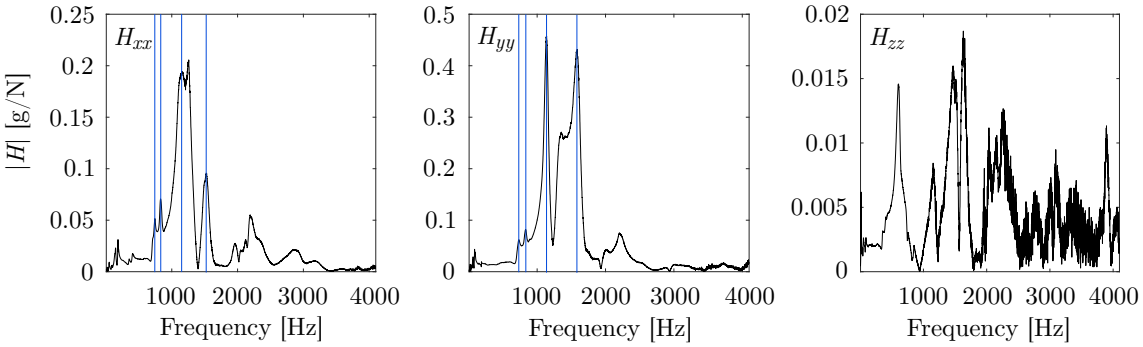


Figure 5.23: Spindle modes in inertance format

If the same direct FRFs are converted into the receptance format (μm/N) as depicted in Figure 5.24, it appears that the most flexible modes still occur below 200 Hz (bandwidth is limited to 2000 Hz in the figure since the inertance format exhibits lower frequency peaks after 2000 Hz), i.e. the modes originating from the robot joints. In receptance format, the compliance of spindle modes is about 0.1 μm/N which means hundred times more rigid than the robot modes. It is then unlikely that the spindle modes can contribute to milling instability using this particular assembly of spindle and robot. Spindle modes can thus be ignored in the robot modelling.

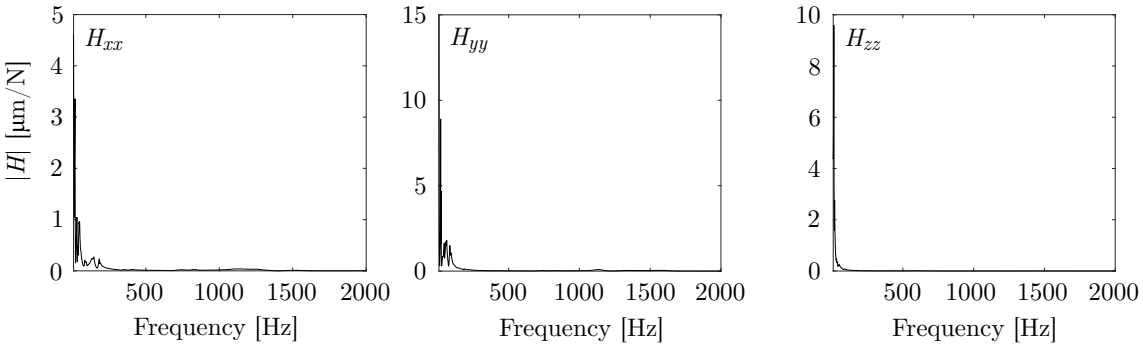


Figure 5.24: Spindle modes in receptance format

It should be imperatively pointed out that this assumption might not be valid for all milling robots. As can be seen in Cordes et al. [104], they reported spindle modes as flexible as robot modes (Figure 5.25). As a result, they found chatter frequencies related to the spindle modes while performing milling tests in aluminium, i.e. at high spindle

speeds (Ω >15,000 rev/min). Figure 5.25 also allows comparing the amplitudes of the robot mode flexibility for the Stäubli TX200 and the ABB IRB 6660-205/1.9 robots (FRF labels were transposed to the TCP frame defined in this work in Figure 5.25). For the ABB IRB 6660 robot, largest frequency peak reaches about 22 μm/N (H_{yy} in Figure 5.25) whereas it is about 16 μm/N for the Stäubli TX200 robot (H_{zz} in Figure 5.19).

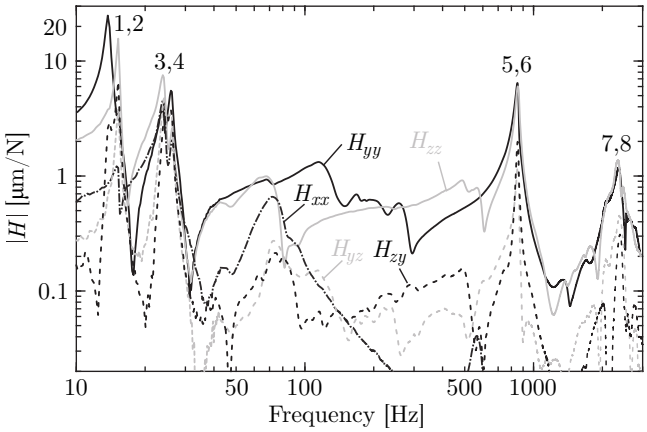


Figure 5.25: Measured direct-FRFs and two cross-FRFs for the ABB IRB 6660-205/1.9 robot in [104]

4. Mode evolution

The last effect analysed related to the milling pose is the evolution of the natural frequencies if the posture of the robot slightly changes around the reference pose. FRFs at the TCP are measured in three distinct poses spaced 500 mm apart. To keep the same orientation for the spindle and the same alignment for the robot axis, Pose 1 in Figure 5.26 defines the nearest TCP location with respect to the base frame with the same elevation as in the milling posture; joint angles are provided in Table 5.11. Poses 2 and 3 are then found on a straight line along y_{Base} at the same altitude. Pose 2 is almost similar to the milling posture.

q_1 [°]	q_2 [°]	q_3 [°]	q_4 [°]	q_5 [°]	q_6 [°]
-87.69	10.95	141.18	4.37	-47.19	-2.10

Table 5.11: Joint positions of Pose 1 in the encoder reference

Again, only direct FRFs measured at the TCP are shown in Figure 5.27 for the three studied poses. As might be expected, Pose 1 exhibits the higher natural frequencies for modes 1 and 2 which are related to the motions of joints 1 and 2 around their motion axis in the mode shapes. As the robot arm extends, natural frequencies of these modes decrease. For mode three related to the onward motion of the robot, the natural frequency increases as robot arm and forearm are almost parallel. The same trend and justification apply to modes 4, 5 and 7. The natural frequency of mode 6 slightly decreases as it is

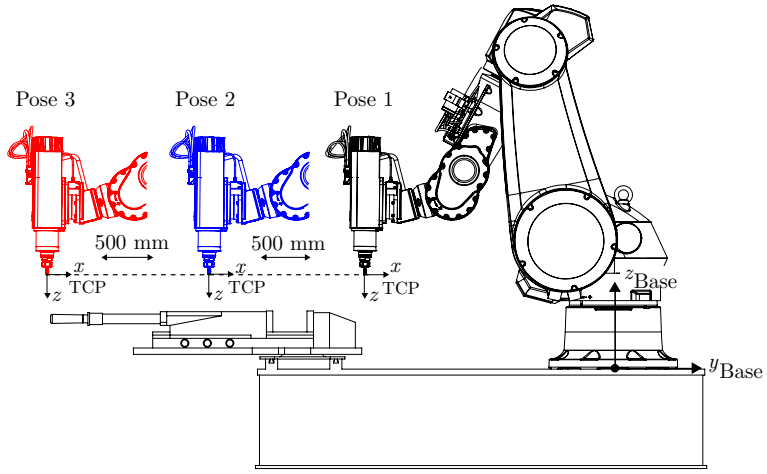


Figure 5.26: Studied poses on a linear path for the Stäubli TX200 robot

also affected by the lifting motion of the robot base, i.e. higher flexibility at the anchor point between the robot and steel slab when the arm is outstretched.

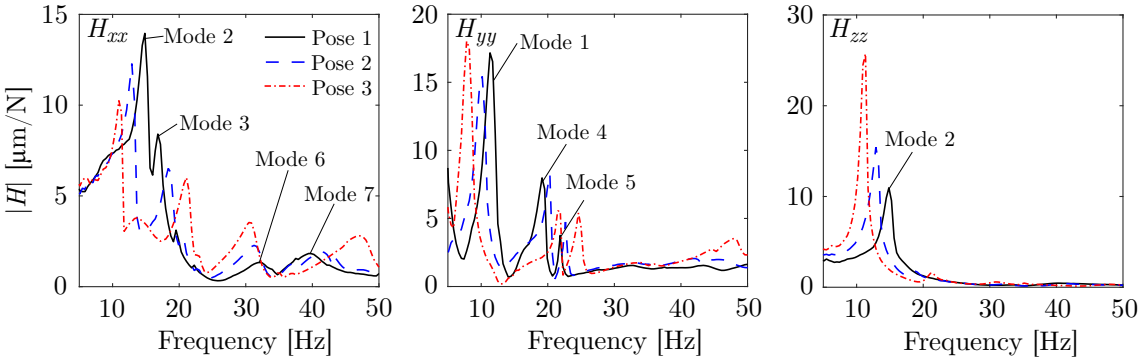
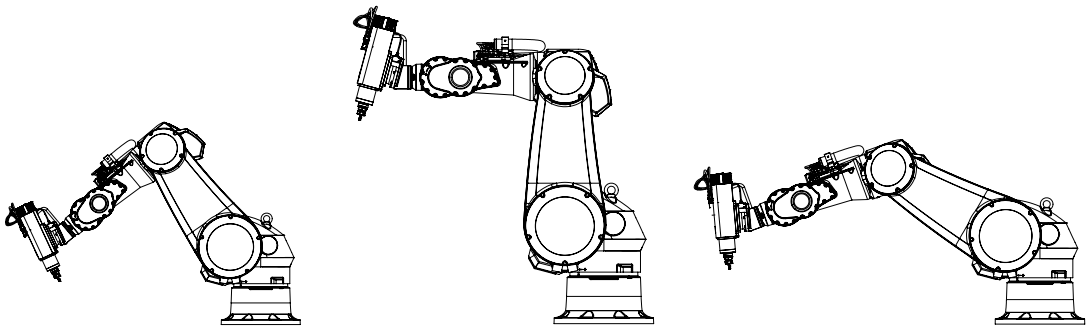


Figure 5.27: Mode evolution on a linear path for the Stäubli TX200 robot

5.2.4 Results in three other postures

Three other robot postures are investigated through complete experimental modal analyses in order to further validate the posture dependency on the robot modes of the multibody model. Controller is in action and FRFs are measured over a bandwidth of 200 Hz during the measurements of all impact points.

The three studied poses are depicted in Figure 5.28 and are successively named *milling straight wrist* pose, *square* pose and *flexible* pose. As noticed, all poses showcase a straight wrist i.e. axes of joints 4 and 6 are made parallel. The reason for this particular choice is that the robot multibody model comprising a flexibility only around the motion axes was identified by considering the first three joints and a straight wrist (Subsection 5.1.3). In this way, joint elastic parameters are later fitted to the EMA results involving a robot posture with a straight wrist. Joint angular positions are reported in Table 5.12 for the three postures with a straight wrist. Even if extensive analyses of the three poses were carried out as for the milling pose, only the visualisation of the mode shapes and



(a) Milling straight wrist pose (b) Square pose (c) Flexible pose

Figure 5.28: EMA in three arbitrary postures of the Stäubli TX200 robot with a straight wrist

auto-MAC matrices are shown in this subsection. Note that corresponding stabilisation diagrams were similar to the one shown in Figure 5.16.

	q_1 [°]	q_2 [°]	q_3 [°]	q_4 [°]	q_5 [°]	q_6 [°]
Milling straight wrist pose	-90	30	95	0	0	0
Square pose	-90	0	90	0	0	0
Flexible pose	-90	60	50	0	0	0

Table 5.12: Joint positions of the milling straight wrist, square and flexible poses in the encoder reference

Milling straight wrist pose

As mentioned during the analysis of the FRFs shown from Figures 5.17 to 5.19 for the milling posture, the modal content of interest for a robotic milling operation is situated below 35 Hz i.e. the most flexible modes. Hence, Figure 5.29 only illustrates identified mode shapes below 35 Hz with their natural frequencies and damping ratios for the milling straight wrist pose. As demonstrated in Figure 5.29, identified mode shapes in this posture are the same as the ones reported in the milling posture exposed in Figure 5.20 for the first six modes. Indeed, modes under 35 Hz mostly depend on the arrangement of the first three joints which did not change much. Natural frequencies of modes 1, 2 and 6 slightly decrease but are overall comparable.

The computed auto-MAC matrix in the milling posture with a straight wrist is presented in Figure 5.30. It is naturally very similar to the one reported for the milling pose in Figure 5.21. Again, very high correlation values (98 %) are remarked for modes 4 and 5 exhibiting a motion perpendicular to the rotation axis of the first joint.

Square and flexible poses

Identified mode shapes in the square and flexible poses are illustrated in Figures 5.31 and 5.33. With different natural frequencies and damping ratios, animations of the mode

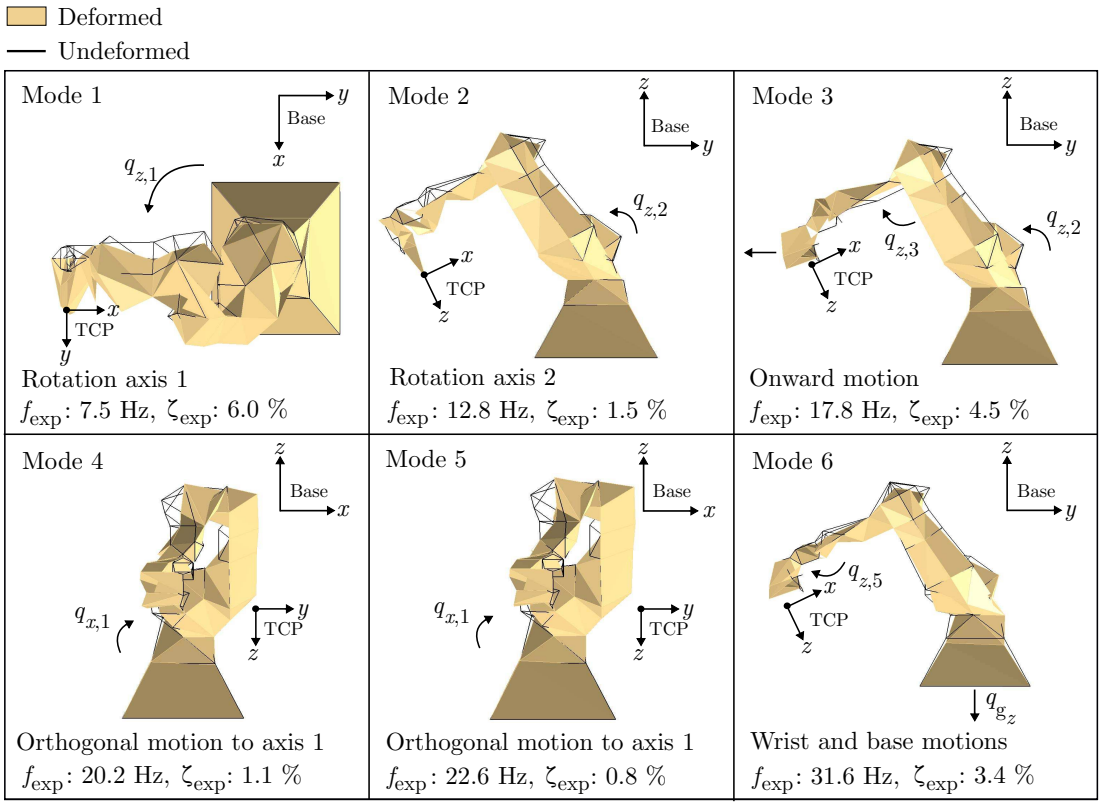


Figure 5.29: Mode shapes in the controlled milling posture with straight wrist of the Stäubli TX200 robot

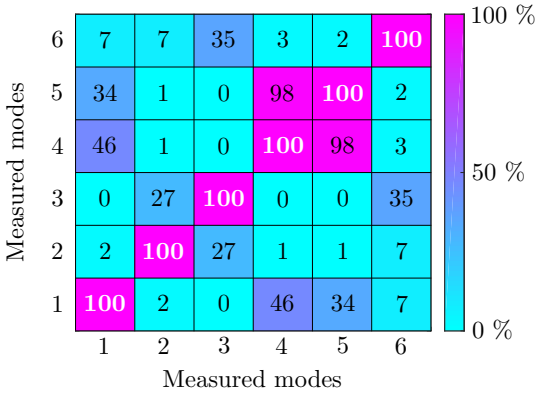


Figure 5.30: Auto-MAC matrix in the controlled milling posture with straight wrist of the Stäubli TX200 robot

shapes are again similar to the ones reported for the milling posture in Figure 5.20. Mode shapes are also in the same order of appearance. Note that even if the shapes of mode do not change, other complete experimental modal analyses revealed that modes can switch places depending on the robot configuration. In Appendix K, complete experimental modal analyses are carried out on three uncommon robot postures (vertical, upright and rigid poses). It is observed that as the robot arm reaches a vertical configuration, modes 1 (rotation around the first joint) and 2 (rotation around the second joint) are swapped (mode 2 becomes mode 1). When the arm extends towards a horizontal configuration, mode three (onward motion) can be located between modes 4 and 5 (perpendicular

motion to the first joint) exhibiting high correlation values when comparing their mode shapes. These are the main trends but sometimes mode shapes are still difficult to interpret even for low frequencies in uncommon postures. Fortunately, few milling operations are requiring these extreme postures.

Corresponding auto-MAC matrices for the square and the flexible poses are displayed in Figure 5.32. They are also similar to the ones reported for the milling pose in Figure 5.21.

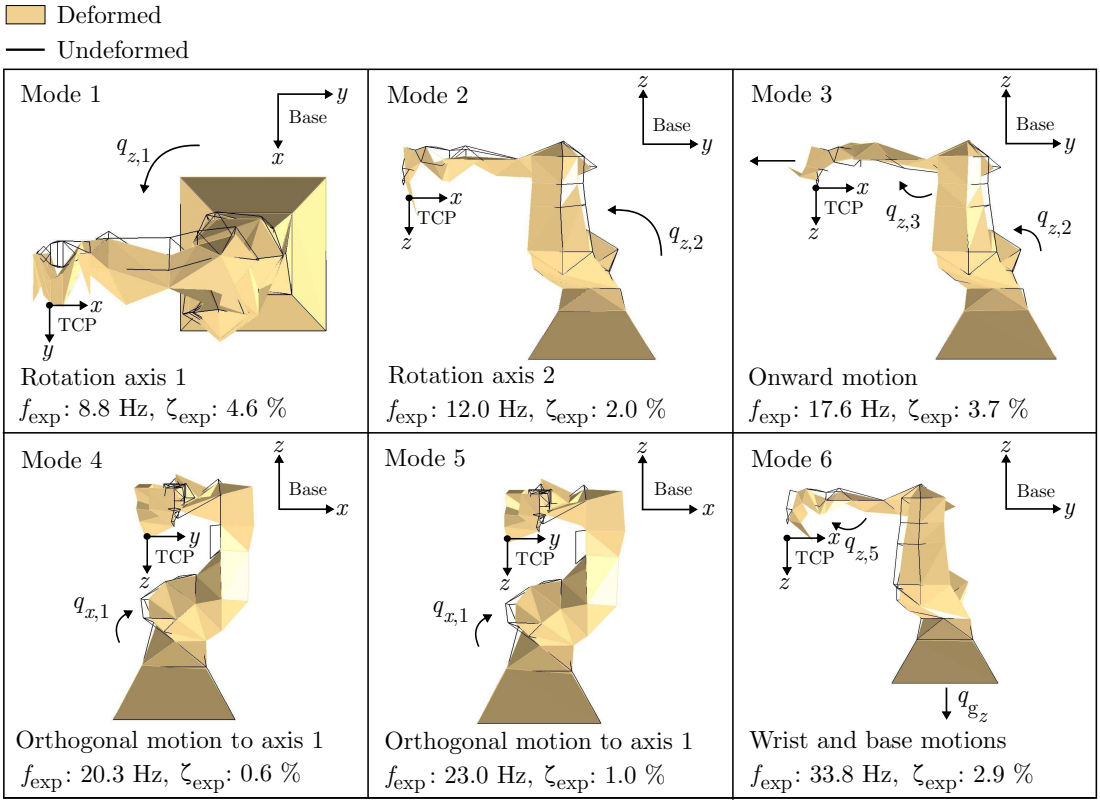


Figure 5.31: Mode shapes in the controlled square posture of the Stäubli TX200 robot

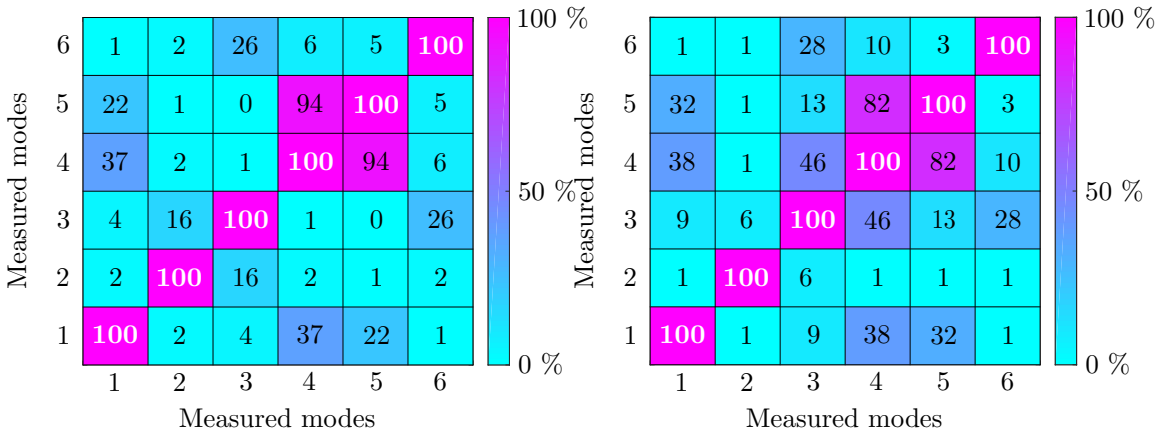


Figure 5.32: Auto-MAC matrices of the measured modes of the Stäubli TX200 robot in two other postures

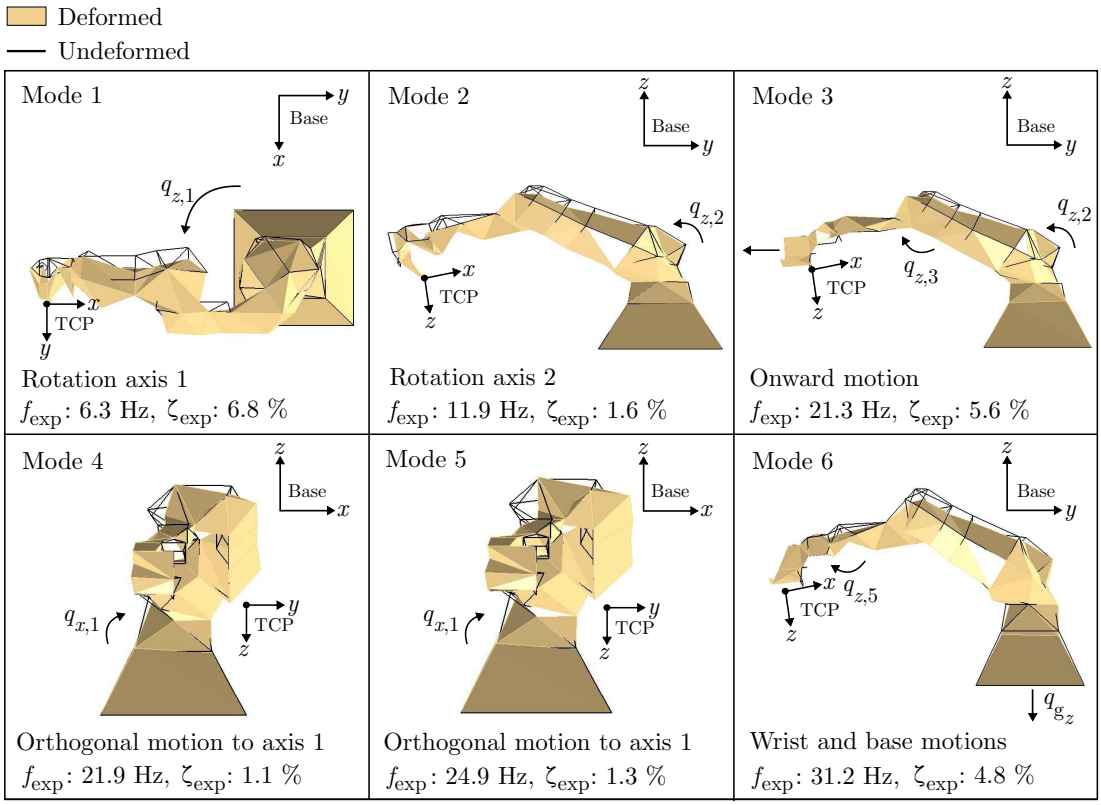


Figure 5.33: Mode shapes in the controlled flexible posture of the Stäubli TX200 robot

5.3 Joint elastic parameters

The elastic parameters of the robot model are determined on the basis of the modal parameters identified via experimental modal analysis. Elastic parameters are composed of joint stiffness $\mathbf{k}_{x|y|z}$, ground stiffness $\mathbf{k}_{\text{ground}}$, joint damping $\mathbf{d}_{x|y|z}$ and ground damping $\mathbf{d}_{\text{ground}}$ as defined in Chapter 3. All the 42 elastic parameters are gathered in vector $\mathbf{p}_{\text{Elastic}}$. The goal is to find appropriate values for the components of $\mathbf{p}_{\text{Elastic}}$ such that measured FRFs, and more particularly the FRF matrix at the TCP, are optimally reproduced by the robot multibody model. From a general point of view, FRF matrix $\mathbf{H}(\omega)$ can be constructed from the constitutive matrices of the system such as

$$\mathbf{H}(\omega) = \left(-\mathbf{M}\omega^2 + \mathbf{C}i\omega + \mathbf{K}\right)^{-1}. \tag{5.18}$$

This section sheds light on a straightforward identification method of elastic parameters for multibody models in general and determines a suitable set of joint stiffness and damping to fit the measured FRFs at the TCP of the Stäubli TX200 robot (Figures 5.17 to 5.19).

5.3.1 Flexible multibody model identification method

The proposed identification method is appropriate to achieve a model updating of any multibody system comprising flexible joints. The idea of the method is to perform a curve fitting of the measured FRF matrix $\mathbf{H}(\omega)$ in the sensor frame. In other words, measured

FRFs are directly used in the algorithm without the need of frame transposition. The method is said to be straightforward as the algorithm is supposed to deliver the optimum values for the sought parameters in one run. Global minimum of the error function between the modelled and measured FRFs is ensured by means of a genetic algorithm. Each iteration of the genetic algorithm can be quickly completed since the modelled FRFs are computed in the frequency domain and in the joint space of the manipulator before being transposed to the sensor frame.

The fitting algorithm can be decomposed into three steps. For simple systems with few degrees of freedom and thus few elastic parameters, only performing the first step is usually sufficient. However, for systems with lots of elastic parameters as for the multibody model of the Stäubli TX200 robot with three elastic elements per joint, the completion of the three steps provides better results. The three steps are described as follows:

1. Find the elastic parameters $\mathbf{p}_{\text{Elastic}}$ such that they minimise the error between the **measured and modelled FRFs** in the sensor frame using a genetic algorithm. The latter allows identifying the region of global minimum through a curve fitting process. It is classically followed by a deterministic algorithm to refine the parameters found by the genetic algorithm.

2. Larger systems with many elastic parameters often lead to an underdetermined problem i.e. there are more elastic parameter couples (joint stiffness and damping) to fit than measured FRF peaks. In this case, after having identified the region of global minimum with the genetic algorithm, another deterministic algorithm is run to minimise the error between the values of the **measured and modelled natural frequencies and damping ratios**. It allows ensuring that the model optimally matches the measured modal parameters. However, it cannot guarantee the fitting of the mode shapes but they are supposed to correspond at best since the genetic algorithm previously found optimal elastic parameter values which are slightly changed using the deterministic algorithm of the second step.

3. Since the second step can sometimes lead to unacceptable error for high frequency modes, a last and optional step can be achieved. A sensitivity analysis can manually be performed on the effect of the tuning of each joint stiffness and damping value on the resulting simulated modal parameters. Under the appraisal of the user, a trade-off can be found on the values of the elastic parameters with respect to the measured modal parameters.

Each step is further detailed below and a fitting example is provided at the end of the subsection.

Step 1:

The idea is to simultaneously fit all the measured FRFs at the TCP with simulated FRFs using a genetic algorithm. The general flowchart of a genetic algorithm is presented in Figure 5.34. From an arbitrary set of values of elastic parameters $\mathbf{p}_{\text{Elastic}}$, the algorithm

starts by deriving an initial population i.e. multiple sets of elastic parameters. Each set is evaluated by submitting the elastic parameters to the model which simulates the corresponding FRFs at the TCP. Experimental and simulated FRFs are compared and the value of the cost function J is computed as the summation of all errors. From then, the genetic algorithm selects individuals (some sets of $\mathbf{p}_{\text{Elastic}}$) that perform well and crosses and mutates them. Throughout the generations, the new populations are then evaluated again and again until the value of the cost function J becomes lower than a certain tolerance δ. In this work, criteria on the number of generations and on the difference with the previous population are used.

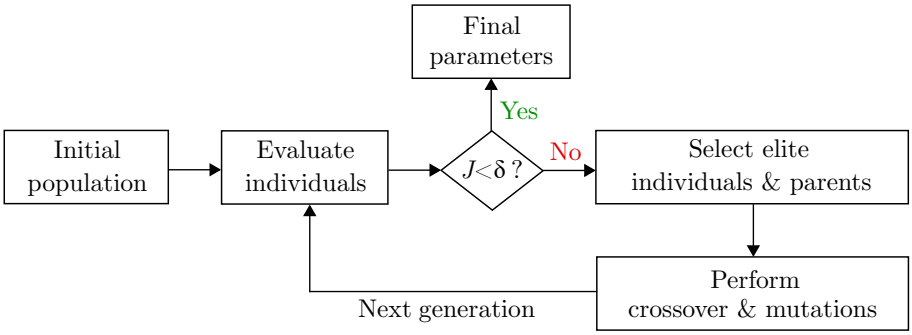


Figure 5.34: Flowchart of a genetic algorithm

In the context of model updating, the cost function J that minimises the error between the measured $H_{\text{ij}}(\omega)$ and simulated $H_{\text{s,ij}}(\omega)$ FRFs can be defined as

$$J = \min_{\mathbf{p}_{\text{Elastic}}} \sum_{\text{i}=x|y|z} \sum_{\text{j}=x|y|z} \sum_{\text{k}=1}^{\text{N}_\omega} \log\left[H_{\text{s,ij}}(\omega_{\text{k}}, \mathbf{p}_{\text{Elastic}}) - H_{\text{ij}}(\omega_{\text{k}})\right]^* \; W(\omega_k) \\ \log\left[H_{\text{s,ij}}(\omega_{\text{k}}, \mathbf{p}_{\text{Elastic}}) - H_{\text{ij}}(\omega_{\text{k}})\right], \tag{5.19}$$

with N_ω the number of discretised intervals in the measured frequency response function, $W(\omega_k)$ a weighting function that magnifies values around peaks in the measured FRFs and superscript * denotes the conjugate transpose vector. A similar cost function is used by Neubauer et al. in [124] for the fitting of FRFs in the joint space of a manipulator.

A typical iteration of genetic algorithm is hereafter presented. Considering the assumed small deflection of the robot joints, the non-linear equations of motion, introduced in Eq. 3.21, can be linearised around an operational joint configuration, $\mathbf{q}^0$ ($\dot{\mathbf{q}}^0$=$\mathbf{0}$ and $\ddot{\mathbf{q}}^0$=$\mathbf{0}$), resulting in the following linear equation describing the oscillations of the system ($\Delta\mathbf{q}$, $\Delta\dot{\mathbf{q}}$ and $\Delta\ddot{\mathbf{q}}$) in joint space such as [169]

$$\mathbf{M}^0\Delta\ddot{\mathbf{q}} + \mathbf{C}^0\Delta\dot{\mathbf{q}} + \mathbf{K}^0\Delta\mathbf{q} = \mathbf{0} \quad \equiv \quad \mathbf{f}(\Delta\mathbf{q}, \Delta\dot{\mathbf{q}}, \Delta\ddot{\mathbf{q}}) = \mathbf{0}, \tag{5.20}$$

with $\mathbf{M}^0$ the mass matrix in the chosen configuration (in which all terms related to rotor inertia are removed since it is assumed that rotors are perfectly regularised at zero velocity), $\mathbf{C}^0$ the linearised damping matrix (combining the effects of the Christoffel

$\mathbf{C}_{\text{Christoffel}}$ and virtual damping $\mathbf{D}$ matrices), $\mathbf{K}^0$ the linearised stiffness matrix (combining the effects of the gravity $\mathbf{g}(\mathbf{q})$ and stiffness $\mathbf{K}$ matrices) and $\mathbf{f}$ the so-called residual form of the equations of motion. The mass matrix and the so-called tangent or linearised damping and stiffness matrices can be computed as

$$\mathbf{M}^0 = \mathbf{M}(\mathbf{q}^0), \;\; \mathbf{C}^0 = \frac{\partial \mathbf{f}}{\partial \dot{\mathbf{q}}}\Big|_0, \;\; \mathbf{K}^0 = \frac{\partial \mathbf{f}}{\partial \mathbf{q}}\Big|_{\mathbf{q}^0}. \tag{5.21}$$

Once the constitutive system matrices are computed in a chosen reference configuration ($\mathbf{q}^0$) for the manipulator, simulated FRFs at the TCP can be derived by transposing them from the manipulator joint space to the sensor frame (usually a Cartesian frame). Homogeneous transformation matrix locating and orienting the sensor frame (tri-axial accelerometer in this case) is expressed in the base frame using the direct kinematics such as $\mathbf{T}_{\text{Base,Sensor}}(\mathbf{q}^0)$, thus defining its translational position with vector $\mathbf{p}_{\text{Sensor}}$. A virtual experimental modal analysis is then performed by applying force F in the direction of excitation j, representing one of the directions X, Y or Z, such as

$$\{\mathbf{F}_{\text{j}}\}_{\text{Base}} = F \;\; \mathbf{R}_{\text{Base,Sensor}}\mathbf{u}_{\text{j}}, \tag{5.22}$$

with $\mathbf{u}_{\text{j}}$ a unit vector e.g. $\mathbf{u}_x$=$[1\ 0\ 0]^T$.

Applied force is projected in the base frame and then converted into vector of joint torques $\boldsymbol{\tau}_{\text{j}}$ using geometric Jacobian matrix $\mathbf{J}_{\text{Sensor}}$ for the sensor and projected in the base frame such as

$$\{\boldsymbol{\tau}_{\text{j}}\}_{\text{Base}} = [\mathbf{J}_{\text{Sensor}}]^T_{\text{Base}}\{\mathbf{F}_{\text{j}}\}_{\text{Base}}. \tag{5.23}$$

The resulting variations in the degrees of freedom $d\mathbf{q}$, which are complex values, corresponding to joint torque $\{\boldsymbol{\tau}_{\text{j}}\}_{\text{Base}}$ are computed at each frequency ω_{k} using the linearised matrices as follows

$$d\mathbf{q}(\omega_{\text{k}})_{\text{j}} = \left(-\mathbf{M}^0\omega_{\text{k}}^2 + \mathbf{C}^0(\mathbf{p}_{\text{Elastic}})i\omega_{\text{k}} + \mathbf{K}^0(\mathbf{p}_{\text{Elastic}})\right)^{-1}\{\boldsymbol{\tau}_{\text{j}}\}_{\text{Base}}. \tag{5.24}$$

Linearised damping $\mathbf{C}^0(\mathbf{p}_{\text{Elastic}})$ and stiffness $\mathbf{K}^0(\mathbf{p}_{\text{Elastic}})$ matrices depend on the elastic parameters which are iterated throughout the fitting process. Variations in the pose of the sensor are computed as the difference between the pose considering the applied force and the reference sensor pose which are transposed back in the sensor frame such as

$$\{\text{Re}\,(\Delta\mathbf{p}_{\text{j}})\}_{\text{Sensor}} = \{\mathbf{p}\left(\mathbf{q}^0 + \text{Re}\,(d\mathbf{q}(\omega_{\text{k}})_{\text{j}})\right)_{\text{Sensor}}\}_{\text{Sensor}} - \{\mathbf{p}(\mathbf{q}^0)_{\text{Sensor}}\}_{\text{Sensor}}, \tag{5.25}$$

$$\{\text{Im}\,(\Delta\mathbf{p}_{\text{j}})\}_{\text{Sensor}} = \{\mathbf{p}\left(\mathbf{q}^0 + \text{Im}\,(d\mathbf{q}(\omega_{\text{k}})_{\text{j}})\right)_{\text{Sensor}}\}_{\text{Sensor}} - \{\mathbf{p}(\mathbf{q}^0)_{\text{Sensor}}\}_{\text{Sensor}}. \tag{5.26}$$

Simulated FRFs are constructed by using the translational components contained in $\mathrm{Re}\,(\Delta\mathbf{p}_\mathrm{j})$ and $\mathrm{Im}\,(\Delta\mathbf{p}_\mathrm{j})$ projected in the sensor frame for each frequency (ω_k) such as

$$\begin{bmatrix} H_{\mathrm{s},x\mathrm{j}}(\omega_\mathrm{k}) \\ H_{\mathrm{s},y\mathrm{j}}(\omega_\mathrm{k}) \\ H_{\mathrm{s},z\mathrm{j}}(\omega_\mathrm{k}) \end{bmatrix} = \frac{1}{F}\left(\{\mathrm{Re}\,(\Delta\mathbf{p}_\mathrm{j})\}_\mathrm{Sensor} + i\{\mathrm{Im}\,(\Delta\mathbf{p}_\mathrm{j})\}_\mathrm{Sensor}\right). \tag{5.27}$$

The nine simulated FRFs at the TCP, forming FRF matrix $\mathbf{H}_\mathrm{s}(\omega)$, are derived by successively applying force F along the X, Y and Z directions in Eq. 5.22. Cost function J is eventually evaluated by using the measured FRFs at the TCP $\mathbf{H}(\omega)$ in Eq. 5.19. One evaluation of the genetic algorithm is thus completed.

Once the genetic algorithm is terminated, either because the cost function reaches a value smaller than the set tolerance or either because the maximum number of iterations is reached, a deterministic algorithm is deployed using the determined values of elastic parameters. Deterministic algorithm uses the same cost function as the genetic algorithm (Eq. 5.19). It allows efficiently and rapidly refining the values of elastic parameters to further minimise the value of the cost function. An implementation of the genetic algorithm can be found in Matlab under toolbox *GA* as well as the deterministic algorithm with function *fminsearch*(). The latter implements the Nelder-Mead simplex direct algorithm [170].

Step 2:

Another form of deterministic algorithm can be implemented to match experimental modal parameters such as natural frequencies f_n and modal damping ratios ζ_n. By introducing state vector $\Delta\mathbf{q}_\mathrm{u} = \left[\Delta\mathbf{q}^T \;\; \Delta\dot{\mathbf{q}}^T\right]^T$, the linearised equations of motion (Eq. 5.20) are expressed in their first order form as

$$\widehat{\mathbf{A}}\Delta\dot{\mathbf{q}}_\mathrm{u} + \widehat{\mathbf{B}}\Delta\mathbf{q}_\mathrm{u} = \mathbf{0}, \tag{5.28}$$

with

$$\widehat{\mathbf{A}} = \begin{bmatrix} \mathbf{C}^0 & \mathbf{M}^0 \\ \mathbf{M}^0 & \mathbf{0} \end{bmatrix} \;\text{ and }\; \widehat{\mathbf{B}} = \begin{bmatrix} \mathbf{K}^0 & \mathbf{0} \\ \mathbf{0} & -\mathbf{M}^0 \end{bmatrix}. \tag{5.29}$$

The simulated natural frequencies of the robot structure $f_{\mathrm{s},n}$, and associated simulated damping ratios $\zeta_{\mathrm{s},n}$, are obtained from the generalised eigenvalues $\lambda_{\mathrm{s},n}$ of matrices $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{B}}$ such as

$$f_{\mathrm{s},n} = \frac{\mathrm{Im}(\lambda_{\mathrm{s},n})}{2\pi}\;[\mathrm{Hz}],\;\text{ and }\; \zeta_{\mathrm{s},n} = \frac{-\mathrm{Re}(\lambda_{\mathrm{s},n})}{\sqrt{\mathrm{Re}(\lambda_{\mathrm{s},n})^2 + \mathrm{Im}(\lambda_{\mathrm{s},n})^2}}, \tag{5.30}$$

The cost function of the second deterministic algorithm is built on the basis of the error between the simulated ($f_{\mathrm{s},n}$ and $\zeta_{\mathrm{s},n}$) and measured (f_n and ζ_n) frequencies and damping ratios. The quadratic-like cost function is defined as

$$J = \sum_{n=1}^{n_{\mathrm{Mode}}} \left(W_f \left| \frac{f_{\mathrm{s},n,\mathrm{k}} - f_n}{f_n} \right|^2 + W_\zeta \left| \frac{\zeta_{\mathrm{s},n,\mathrm{k}} - \zeta_n}{\zeta_n} \right|^2 \right), \tag{5.31}$$

where subscript n is the mode number and k is the k^{th} iteration in the deterministic algorithm. Weighting coefficients on the frequency W_f and on the damping ratio W_ζ emphasise the quantities to fit in priority, the frequencies in this context (usually W_f=100 and W_ζ=1). Normalisation factors, f_n and ζ_n are applied in order to include both the targeted experimental frequency f_n and damping ratio ζ_n expressed in Hz and in %, respectively. Again, implementation of the second deterministic algorithm can be carried out by function *fminsearch*() from Matlab.

Step 3:

The last step can optionally be executed if unacceptable errors subsist between the simulated and measured modal parameters regarding the natural frequencies and damping ratios. It may happen for higher frequency modes. In this case, it is possible to perform a sensitivity analysis assessing the influence of modifying one joint stiffness or one joint damping on the modal parameters. As a result, a trade-off between the fitting of modal parameters for low frequency and high frequency modes is found by the user. In practice, the impact of changing one joint stiffness or one joint damping is noted for each modal parameter. Then, knowing the influence of each joint parameter, it is possible to tune them accordingly to the need and tolerance of the user.

Example of model updating for manipulators:

An example of model updating is provided for an arbitrary manipulator comprising three revolute joints $n = 3$. Following the Craig's convention defined in Appendix H, the manipulator is geometrically characterised with the length of its first two links such as $a_1 = 0.35$ m and $a_2 = 1.35$ m. Joint flexibility is only considered around the three motion axes and set joint stiffness and damping are presented in Table 5.13. From random initial joint stiffness and damping values, the proposed identification has to recover the sought joint parameter set in Table 5.13. It leads to the following modal parameters: f_1=10.17 Hz, ζ_1=1.89 %, f_2=11.24 Hz, ζ_2=1.8 %, f_3=22.6 Hz and ζ_3=1.66 %.

As the manipulator only comprises three joints, only implementing the first step is sufficient to retrieve the exact sought joint parameters. As such, the genetic algorithm followed by its deterministic algorithm is implemented. Parameters of the genetic algorithm are chosen as follows with a number of generations of 15, a population size of 80 (number of joint stiffness and damping sets evaluated during each generation), a function tolerance of δ=1e^{-12} and a stalling generation limit of 50 (genetic algorithm stops if the fitness

Joint stiffness [Nm/rad]	Joint damping [Nm.s/rad]
$k_{z,1}$	$d_{z,1}$
2.0 e^6	1.2 e^3
$k_{z,2}$	$d_{z,2}$
3.0 e^6	1.545 e^3
$k_{z,3}$	$d_{z,3}$
1.55 e^6	0.34 e^3

Table 5.13: Set joint stiffness and damping for an arbitrary manipulator

function reaches the exact same values 50 times in a row). Number of generations and population size must naturally be increased if more elastic parameters are fitted. For the deterministic algorithm, the function tolerance is set to δ=1e^{-15} and the maximum number of iterations is fixed at 5000. Once the genetic algorithm initialised, iterations are completed by following Eqs. 5.22 to 5.27. The homogeneous transformation matrix providing the pose of the sensor can be computed by using relationships 5.32 to 5.35 and the dynamics of the manipulator correspond to the values of inertial parameters proposed in Table 5.14. For the sake of simplicity, note that the manipulator do not comprise any motor and degree of freedom $q_{z,\mathrm{i}}$ represents the joint deflection around the motion axis.

	Body 1	Body 2	Body 3
m_i [kg]	511.726	249.53	222.659
C_{x_i} [m]	-0.018	0.532	0.423
C_{y_i} [m]	0.002	-0.007	0.018
C_{z_i} [m]	-0.21	0.248	-0.018
ϕ_{G,xx_i} [kg.m^2]	19.45	5.014	5.882
ϕ_{G,yy_i} [kg.m^2]	33.893	57.621	49.927
ϕ_{G,zz_i} [kg.m^2]	30.242	56.853	48.9
ϕ_{G,xy_i} [kg.m^2]	2.274	-1.058	-1.372
ϕ_{G,xz_i} [kg.m^2]	0.938	5.498	0.074
ϕ_{G,yz_i} [kg.m^2]	0.373	-0.069	-1.202

Table 5.14: Inertial data for the model updating of an arbitrary manipulator

$$\mathbf{T}_{\text{Body 1}} = \mathbf{T}_{\text{rotz}}(q_{z,1}) \cdot \mathbf{T}_{\text{disp}}(C_{x_1}, C_{y_1}, C_{z_1}). \tag{5.32}$$

$$\begin{aligned}\mathbf{T}_{\text{Body 2}} = \mathbf{T}_{\text{Body 1}} \cdot \mathbf{T}_{\text{disp}}(a_1 - C_{x_1}, -C_{y_1}, -C_{z_1}) \cdot \\ \mathbf{T}_{\text{rotx}}(\frac{\pi}{2}) \cdot \mathbf{T}_{\text{rotz}}(q_{z,2}) \cdot \mathbf{T}_{\text{disp}}(C_{x_2}, C_{y_2}, C_{z_2}).\end{aligned} \tag{5.33}$$

$$\mathbf{T}_{\text{Body 3}} = \mathbf{T}_{\text{Body 2}} \cdot \mathbf{T}_{\text{disp}}(a_2 - C_{x_2}, -C_{y_2}, -C_{z_2}) \cdot \mathbf{T}_{\text{rotz}}(q_{z,3}) \cdot \mathbf{T}_{\text{disp}}(C_{x_3}, C_{y_3}, C_{z_3}). \tag{5.34}$$

$$\mathbf{T}_{\text{TCP}} = \mathbf{T}_{\text{Body 3}} \cdot \mathbf{T}_{\text{disp}}(1.566 - C_{x_3}, 0.042 - C_{y_3}, 0.005 - C_{z_3}) \cdot \mathbf{T}_{\text{rotx}}(\frac{-\pi}{2}). \tag{5.35}$$

Elastic parameters are fitted in the following manipulator configuration: $q_{z,1}$= 0°, $q_{z,2}$= 70° and $q_{z,3}$= -100°. In this configuration and with the set of elastic parameters proposed in Table 5.13, values of linearised system matrices read

$$\mathbf{M}^0 = \begin{bmatrix} 482.1 & -30.5 & 0.3 \\ -30.5 & 588.8 & 72.1 \\ 0.3 & 72.1 & 88.7 \end{bmatrix}, \qquad \mathbf{C}^0 = \begin{bmatrix} 1200 & -0.01 & 0 \\ 0.35 & 1545 & 0.13 \\ -0.07 & -0.13 & 340 \end{bmatrix}, \tag{5.36}$$

$$\mathbf{K}^0 = \begin{bmatrix} 2\text{e}^6 & 0 & 0 \\ 0 & 3\text{e}^6 & 427.816 \\ 0 & 427.816 & 1.55\text{e}^6 \end{bmatrix}. \tag{5.37}$$

Figure 5.35 shows the frequency response function matrix $\mathbf{H}(\omega)$ in the sensor frame computed with the set joint parameters. Chosen frequency resolution and bandwidth are Δf=0.2 Hz and 30 Hz, respectively. Note that the inertance format is deliberately chosen as it was observed that high frequency peaks are fitted more quickly as they clearly appear. Still in the same figure, red cross markers emphasize the region around peaks in which weighting function $W(\omega_\text{k})$ takes higher values; a value of 50 is selected around peaks and 0.5 otherwise.

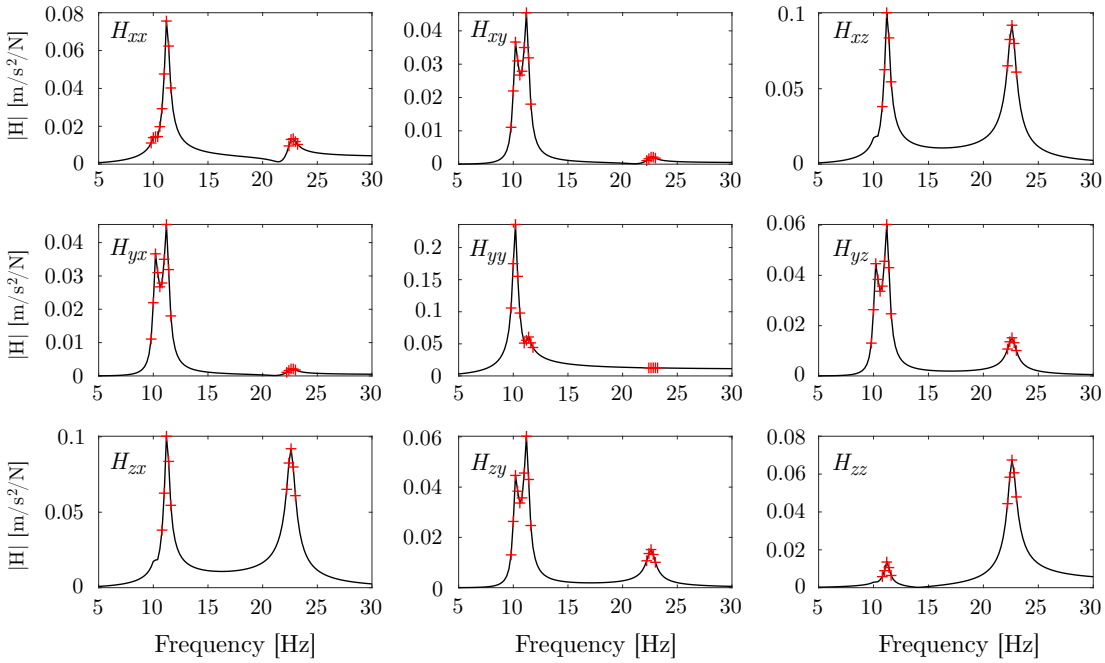


Figure 5.35: Weighting of peaks of interest in the FRF matrix in inertance format

After the completion of the first step of the proposed identification method, simulated and pseudo-measured FRF curves in the sensor frame are almost not discernible. Sought joint stiffness and damping parameters are recovered with a negligible error (<1e^{-5} %). Following the set 1200 iterations of the genetic algorithm, it takes 1114 iterations for the deterministic algorithm to converge. The evolution of the corresponding cost function is depicted in Figure 5.36. As witnessed, process is indeed convergent since the value of

the cost function starting around $2e^5$ m/s^2/N stabilises around $1e^3$ m/s^2/N thanks to the genetic algorithm. The deterministic algorithm eventually drops the cost function value near zero. Proposed method is fast as convergence is reached in about 4 minutes using an Intel i7-8750H processor on the model updating example.

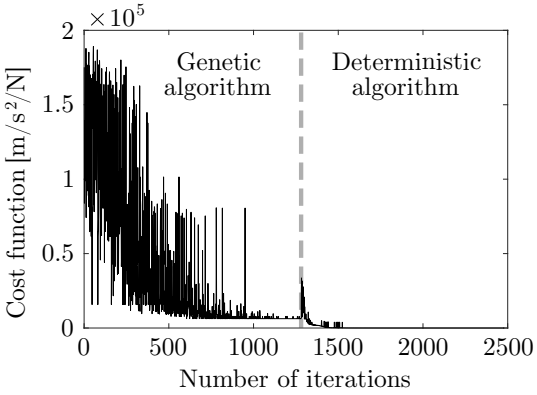


Figure 5.36: Evolution of the cost function for the FRF fitting applied to the manipulator comprising three revolute joints

5.3.2 Updating of the joint axial flexibility model

Using the model of the Stäubli TX200 robot limited to its first three joints and whose base parameters were identified on the basis of excitation trajectories in Subsection 5.1.3, it is updated on the basis of the modal analysis results. Of course, one torsional spring and one torsional viscous damper are appended to each of the three joints around their motion axis. This modelling was introduced in Subsection 3.3.3 and named the joint axial flexibility model. The considered model still possesses three degrees of freedom ($q_{z,\mathrm{i}}$) representing the deflections around the motion axes because of the assumption that the rotors are perfectly regularised at zero velocity. In a static pose, controller or motor brakes have the same locking effect on the rotors such that their inertia does not intervene in the mass matrix. All components related to rotor inertia are thus removed in mass matrix $\mathbf{M}(\mathbf{q})$. In other words, rotor inertia terms must be removed from base parameter vector $\boldsymbol{\pi}$ as well as the related effects such as friction and the gravity compensation coefficients mainly contributing at the motor level. Out of the three $\phi_{\mathrm{m},zz_{\mathrm{i}}}$ terms, only ϕ_{m,zz_3} can be explicitly identified in Subsection 5.1.3 as π_{16}, and ϕ_{m,zz_1} and ϕ_{m,zz_2} are only implicitly identified as part of π_1 and π_2 as seen in Table 5.6. Since these values cannot be identified individually, assuming a reasonable identification of ϕ_{m,zz_3}, values for rotor inertias ϕ_{m,zz_1} and ϕ_{m,zz_2} can be derived by considering a proportionality between rotor inertia and the gear ratios i.e. $\phi_{\mathrm{m},zz_1} \approx \frac{k_1}{k_3}\phi_{\mathrm{m},zz_3}$. Without the effects related to the rotors, regression matrix and base parameter vector are denoted by $\widetilde{\mathbf{Y}}$ and $\widetilde{\boldsymbol{\pi}}$, respectively. Hence, linearised constitutive matrices of the system are obtained from the following relationships such as

$$\mathbf{M}^0 = \left.\frac{\partial \widetilde{\mathbf{Y}}\widetilde{\boldsymbol{\pi}}}{\partial \ddot{\mathbf{q}}}\right|_0, \quad \mathbf{C}^0 = \left.\frac{\partial \widetilde{\mathbf{Y}}\widetilde{\boldsymbol{\pi}}}{\partial \dot{\mathbf{q}}}\right|_0 + \mathbf{D}, \quad \mathbf{K}^0 = \left.\frac{\partial \widetilde{\mathbf{Y}}\widetilde{\boldsymbol{\pi}}}{\partial \mathbf{q}}\right|_{\mathbf{q}^0} + \mathbf{K}. \tag{5.38}$$

with **K** the stiffness matrix and **D** the virtual damping matrix. All parameters of the mass, stiffness and damping matrices are obtained from the rigid body identification, except the joint stiffness and damping parameters gathered in the **K** and **D** matrices.

In order to identify the joint stiffness and damping parameters, modal analysis results presented in Subsection 5.2.4, showcasing robot configurations with a straight wrist, are used. The elastic parameters are determined in the milling posture with a straight wrist and then retained to predict the natural frequencies, damping ratios and mode shapes in the square and flexible postures. Since the robot model only comprises the deflections around the first three motion axes, only three modes can be created with this model, but much more modes were measured during the experimental modal analysis tests. It is clear that mode shapes involving a deflection perpendicular to a motion axis cannot be simulated using the presented model. As discussed in Subsection 5.2.4, modes one, two and three in the milling posture with a straight wrist are associated with dominant deflections of the first three joints around their motion axis. The stiffness and damping coefficients of each joint can therefore be adjusted to obtain the measured natural frequencies and modal damping ratios for the first three modes. The resulting joint parameters are shown in Table 5.15. They are identified by following the proposed identification method in Subsection 5.3.1. In particular, for the second step, convergence in cost function J is reached after 279 iterations dropping its value from $1e^4$ to 20 (Eq. 5.31). Stopping criterion is triggered when the value of the cost function does not change in a tolerance interval of 0.001.

	Joint 1	Joint 2	Joint 3
Joint stiffness [Nm/rad]	1.15e6	3.76e6	1.29e6
Joint damping [Nm.s/rad]	2.82e3	0.61e3	1.87e3

Table 5.15: Identified joint stiffness and damping parameters with the joint axial flexibility model for the Stäubli TX200 robot

The identified joint parameters are used in Eqs. 5.28 to 5.30 to compute the simulated natural frequencies, damping ratios and mode shapes of the three modes, in the three tested postures: milling with straight wrist, square and flexible. The simulated mode shapes and associated natural frequencies and damping ratios in the milling posture with straight wrist are shown in Figure 5.37. Compared to the measured mode shapes and modal parameters in the same posture, shown in Figure 5.29, the identified model predicts similar mode shapes. Animations of simulated mode shapes are provided by EasyAnim, a free and in-house visualisation tool for multibody systems. In terms of simulated natural frequencies f_s and damping ratios ζ_s, while the values for the first two modes are acceptable, the simulated natural frequency of the third mode (21.8 Hz) is somewhat far from the measured value (17.8 Hz).

In view of the frequency response function matrix $\mathbf{H}(\omega)$ comparing the magnitudes of the measured and simulated FRFs at the TCP in Figure 5.38, the third mode (along

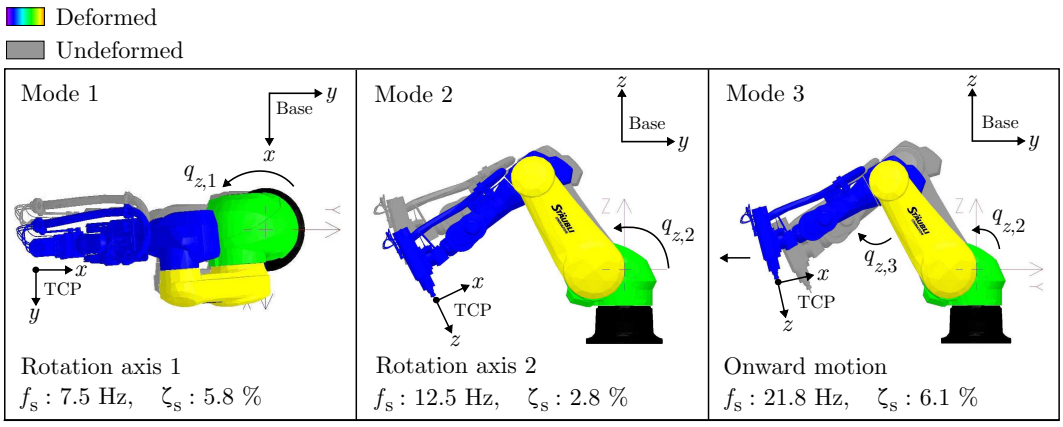


Figure 5.37: Simulated mode shapes with the joint axial flexibility model for the milling posture with straight wrist

H_{xx}) fortunately exhibits small deflections compared to the first two modes. Depending on the robot geometry and identified base parameters, the model updating of such robot model can lead to satisfactory results for some of the modes or all the modes. In Section J.2 in Appendix J, it is shown that all the first three modes of the KUKA KR90 robot, associated with deflections around motion axes, can be fitted with the proposed method. As depicted in Figure 5.38, besides the third mode (H_{xx}), a reasonable agreement is observed between the predicted and measured FRFs for the direct FRFs. Note as well that in Figure 5.38, cross-FRFs H_{yz} and H_{zy} exhibit a strong asymmetry which again demonstrate the non-linearity of robot manipulators.

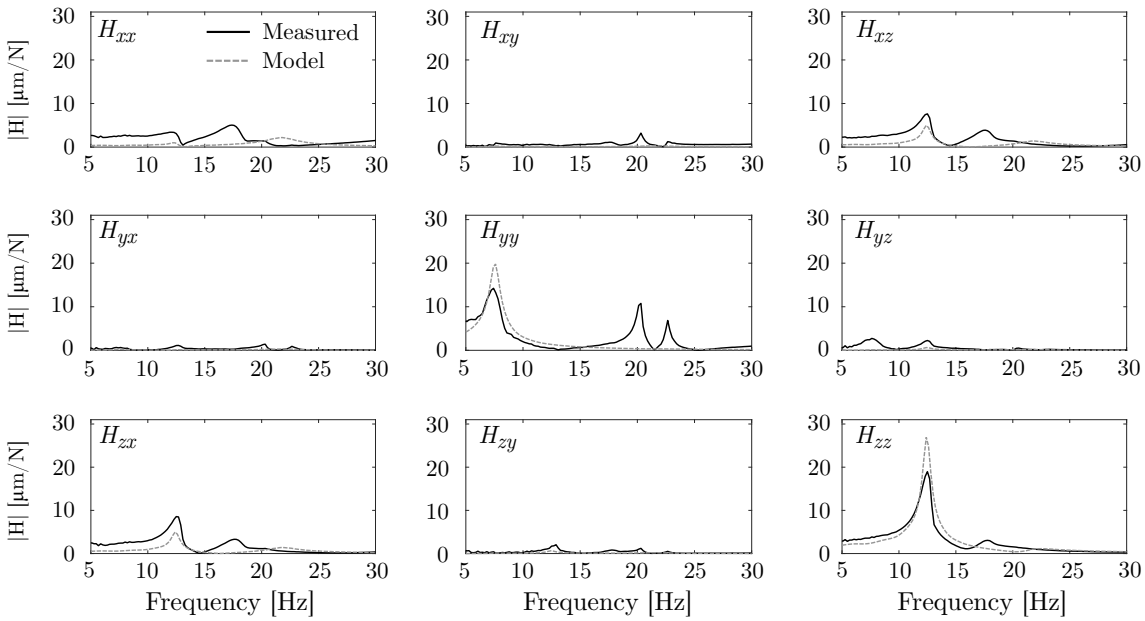


Figure 5.38: Amplitude of measured and fitted frequency response function matrix $\mathbf{H}(\omega)$ at the TCP obtained with the joint axial flexibility model for the milling posture with straight wrist

The MAC matrix presented in Figure 5.39 shows the correlations between the measured

and identified mode shapes in the milling posture with a straight wrist. Considered mode shapes in the computation of the MAC matrix only comprise the 21 impact points underlined in Figure 5.12. High correlations are observed for the first three mode shapes. Other measured modes below 35 Hz are not correlated since they either include a perpendicular deflection to a motion axis or a wrist motion.

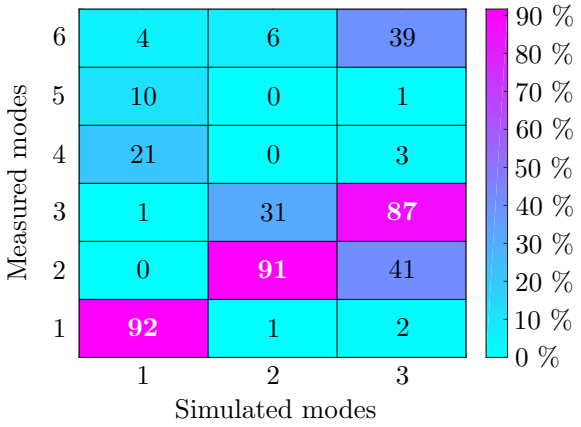
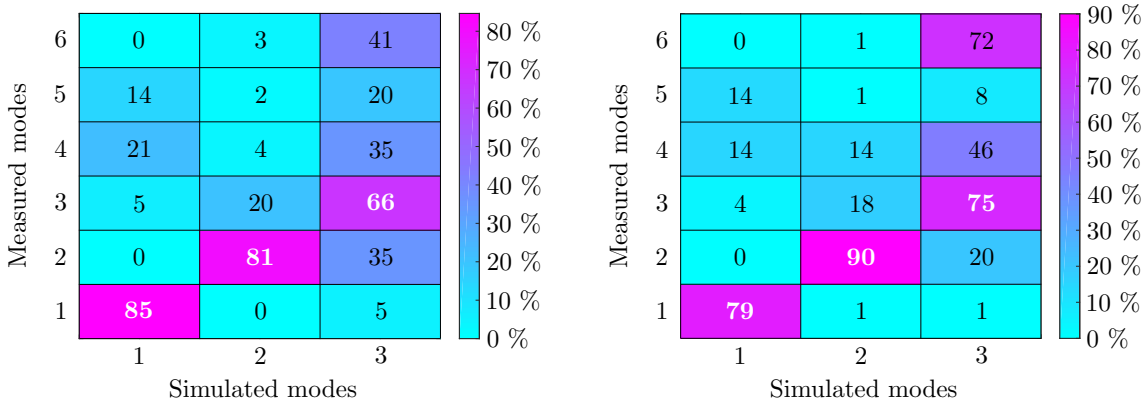


Figure 5.39: MAC matrix obtained with the joint axial flexibility model for the milling posture with straight wrist

In order to study the accuracy of the identified model in predicting the structural vibrations of the robot in arbitrary postures, it is used to compute the mode shapes and modal parameters in the square and flexible postures. The MAC tables between the predicted and measured mode shapes in the square and flexible postures are shown in Figure 5.40a and Figure 5.40b, respectively. Excluding modes 4, 5 and 6, high levels of correlations are again observed for the first three modes. In both postures to predict, the third mode shows the lowest level of correlation. Indeed, for the Stäubli TX200 robot, a strong dependency is observed between the reached simulated natural frequencies and the set joint stiffness, i.e. a change in one joint stiffness, either for joint two or three, results in the alteration of both simulated natural frequencies for modes 2 and 3.



(a) MAC in square pose (b) MAC in flexible pose

Figure 5.40: Assessment of the posture dependency prediction through the MAC matrices with joint axial flexibility model

Also the comparisons between the predicted and measured modal frequencies and damping ratios in the milling posture with straight wrist and square and flexible poses are shown in Table 5.16. The last row of the table indicates the average error in percentage for the natural frequencies and damping ratios. Overall, natural frequencies are better estimated than the damping ratios in all the postures. As mentioned, the largest error in terms of natural frequencies is observed for the third mode. The natural frequencies of modes one and two are well correlated. Largest errors in natural frequencies and damping ratios are remarked for the flexible pose.

	Milling straight wrist		Square		Flexible	
Mode	f_n [Hz]	ζ_n [%]	f_n [Hz]	ζ_n [%]	f_n [Hz]	ζ_n [%]
1	7.5 (7.5)	5.8 (6.0)	9.3 (8.8)	7.2 (4.6)	5.8 (6.3)	4.5 (6.8)
2	12.5 (12.8)	2.8 (1.5)	12.2 (12.0)	2.7 (2.0)	10.5 (11.9)	2.0 (1.6)
3	21.8 (17.8)	6.1 (4.5)	22.4 (17.6)	6.5 (3.7)	27.2 (21.3)	8.6 (5.6)
$\text{Error}_{\text{Average}}$ [%]	6.9	25.4	9.5	35.0	14.5	35.3

Table 5.16: Prediction of the natural frequencies and damping ratios in other postures for the identified model (experimental frequencies and damping ratios in brackets)

5.3.3 Updating of the joint tri-axial flexibility model

As presented in the previous subsection, the developed multibody model including a flexibility only around the motion axes can be identified using rigid body identification methods and EMA. However, it is unable to simulate the mode shapes involving joint deflections around an axis other than the rotation axis. In order to capture the missing modes involving such motion and also the wrist deflections, the multibody model with three elastic elements for all the six joints ($q_{z|x|y}$) and with the ground deflections ($q_{g_{z|x|y}}$) is considered. Nevertheless, the inertia parameters of the model with three elastic elements per joint cannot be identified using rigid body identification methods because only one out of the three DOFs per joint is actuated. Therefore, the Stäubli (Subsection 5.1.1) and UMons (Subsection 5.1.2) multibody models of the Stäubli TX200 robot are used to fit the 42 elastic parameters by following the three steps of the proposed identification method (Subsection 5.3.1). With such multibody model, the joint elastic parameters in the milling posture are determined on the basis of the modal analysis results in the milling posture exposed in Subsection 5.2.3. The resulting joint parameters for both multibody models are shown in Tables 5.17 and 5.18.

Note that during the first step of the proposed fitting method, the FRF curve fitting is achieved on the synthesised FRFs in order to increase the frequency resolution. As such, a frequency step of Δf=0.2 Hz is chosen for the synthesized FRFs. In addition, since the use of FRFs implies that the system is linear, synthesised cross-FRFs H_{xy}, H_{xz} and H_{yz} are duplicated in place of H_{yx}, H_{zx} and H_{zy}. The joint elastic parameters are updated over a bandwidth of 100 Hz. During the fitting, values that are manually tuned for the dynamic vibration absorber are retained. For both models, the inertia values of the body representing the DVA are set as $\phi_{\text{DVA},xx}$=$\phi_{\text{DVA},yy}$=3.7 kg.m^2 and $\phi_{\text{DVA},zz}$=0 kg.m^2. The

Joint stiffness [Nm/rad]			Joint damping [Nm.s/rad]		
$k_{z,1}$	$k_{x,1}$	$k_{y,1}$	$d_{z,1}$	$d_{x,1}$	$d_{y,1}$
2.3 e^6	13.1 e^6	17.0 e^6	4.6 e^3	0.1 e^3	0.2 e^3
$k_{z,2}$	$k_{x,2}$	$k_{y,2}$	$d_{z,2}$	$d_{x,2}$	$d_{y,2}$
8.6 e^6	4.1 e^6	19.9 e^6	5.0 e^3	1.0 e^3	2.2 e^3
$k_{z,3}$	$k_{x,3}$	$k_{y,3}$	$d_{z,3}$	$d_{x,3}$	$d_{y,3}$
2.8 e^6	18.8 e^6	2.9 e^6	2.5 e^3	2.6 e^3	0.1 e^3
$k_{z,4}$	$k_{x,4}$	$k_{y,4}$	$d_{z,4}$	$d_{x,4}$	$d_{y,4}$
3.0 e^6	2.3 e^6	3.1 e^6	0.3 e^3	1.0 e^3	0.01 e^3
$k_{z,5}$	$k_{x,5}$	$k_{y,5}$	$d_{z,5}$	$d_{x,5}$	$d_{y,5}$
0.2 e^6	2.1 e^6	0.9 e^6	0.01 e^3	0.02 e^3	0.02 e^3
$k_{z,6}$	$k_{x,6}$	$k_{y,6}$	$d_{z,6}$	$d_{x,6}$	$d_{y,6}$
0.9 e^6	14.8 e^6	8.3 e^6	0.01 e^3	0.01 e^3	0.01 e^3
k_{g_z}	k_{g_x}	k_{g_y}	d_{g_z}	d_{g_x}	d_{g_y}
20.3 e^6	48.1 e^6	32.1 e^6	5.0 e^3	5.4 e^3	5.0 e^3

Table 5.17: Identified joint stiffness and damping for the joint tri-axial flexibility model for the Stäubli TX200 model

Joint stiffness [Nm/rad]			Joint damping [Nm.s/rad]		
$k_{z,1}$	$k_{x,1}$	$k_{y,1}$	$d_{z,1}$	$d_{x,1}$	$d_{y,1}$
1.9 e^6	27.9 e^6	22.4 e^6	2.6 e^3	0.07 e^3	3.4 e^3
$k_{z,2}$	$k_{x,2}$	$k_{y,2}$	$d_{z,2}$	$d_{x,2}$	$d_{y,2}$
10.2 e^6	17.0 e^6	20.5 e^6	0.6 e^3	0.3 e^3	0.3 e^3
$k_{z,3}$	$k_{x,3}$	$k_{y,3}$	$d_{z,3}$	$d_{x,3}$	$d_{y,3}$
2.1 e^6	1.6 e^6	3.7 e^6	0.2 e^3	0.3 e^3	0.6 e^3
$k_{z,4}$	$k_{x,4}$	$k_{y,4}$	$d_{z,4}$	$d_{x,4}$	$d_{y,4}$
3.0 e^6	4.0 e^6	1.2 e^6	0.01 e^3	0.1 e^3	2.2 e^3
$k_{z,5}$	$k_{x,5}$	$k_{y,5}$	$d_{z,5}$	$d_{x,5}$	$d_{y,5}$
0.2 e^6	1.7 e^6	1.1 e^6	0.001 e^3	0.01 e^3	0.001 e^3
$k_{z,6}$	$k_{x,6}$	$k_{y,6}$	$d_{z,6}$	$d_{x,6}$	$d_{y,6}$
1.3 e^6	14.0 e^6	14.1 e^6	0.4 e^3	1.7 e^3	0.001 e^3
k_{g_z}	k_{g_x}	k_{g_y}	d_{g_z}	d_{g_x}	d_{g_y}
27.4 e^6	43.5 e^6	33.7 e^6	1.4 e^3	1.0 e^3	8.8 e^3

Table 5.18: Identified joint stiffness and damping for the joint tri-axial flexibility model for the UMons TX200 model

stiffness and damping of the torsional elements composing the DVA are chosen so that its anti-resonance falls between modes 4 and 5 in the measured FRFs of the milling posture presented in Subsection 5.2.3. DVA stiffness and damping are eventually tuned as k_{DVA}=6.7e^4 Nm/rad and d_{DVA}=10.0 Nm.s/rad. The evolution of the corresponding cost function is presented in Figure 5.41. A convergence of the genetic algorithm applied to the fitting of the 42 elastic parameters is observed after about 11,000 iterations. Genetic algorithm is further stopped once all set generations are accomplished (it does

not stop from the stall criterion since the error tolerance was too tight). Less iterations could have been ordered before the beginning of the deterministic algorithm but the convergence evolution was unknown in the first instance. Using the set parameters, it took about 2.5 days to complete the genetic and deterministic algorithms for the fitting of 42 elastic parameters.

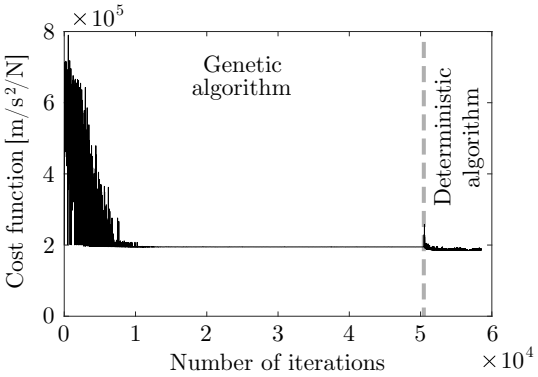


Figure 5.41: Evolution of the cost function for the FRF fitting applied to the Stäubli TX200 model with three elastic elements per joint

Regarding the stiffness values around the motion axes, values seem in accordance with the ones reported in the literature usually in the range of 0.1e^6 to 7.0e^6 Nm/rad [53]. On the other hand, stiffness values normal to the motion axes are unsurprisingly higher than the ones around the motion axes. Determined joint damping values are more difficult to interpret but provide coherent damping ratios for the targeted modes.

The identified joint parameters are again used in Eqs. 5.28 to 5.30 to compute the simulated natural frequencies, damping ratios and mode shapes in the milling posture. The simulated mode shapes and associated modal frequencies and damping ratios in the milling posture are illustrated in Figure 5.42. Compared with the measured mode shapes and the modal parameters in the same posture, shown in Figure 5.20, it appears that similar mode shapes, natural frequencies and damping ratios are obtained. The first seven mode shapes clearly correspond to the animations provided by the LMS Test.Lab software while the last three mode shapes are more difficult to interpret. Therefore, the first simulated mode shape correctly reproduces the rotation around the motion axis of the first joint. Although the simulated frequency ($f_{s,1}$=9.8 Hz) is close to the measured one (f_1=9.4 Hz), the simulated damping ratio ($\zeta_{s,1}$=3.9 %) is almost halved compared to the measured one (ζ_1=7.9 %). Same conclusions are derived for the second mode, which is the rotation around the motion axis of the second joint, and the third mode, which is the onward motion. Thanks to the introduction of the DVA, former mode generating a deflection perpendicular to the motion axis of the first joint ($q_{x,1}$) is well split into two modes (modes four and five) with the same mode shapes and close in terms of frequency at $f_{s,4}$=21 and $f_{s,5}$=22.8 Hz, respectively. The simulated modal damping ratios of those two modes ($\zeta_{s,4}$=$\zeta_{s,5}$=0.9 % for both modes) are close to the measured ones (ζ_4=1.0 and ζ_5=0.8 %). Modes six and seven present mode shapes similar to the ones that were

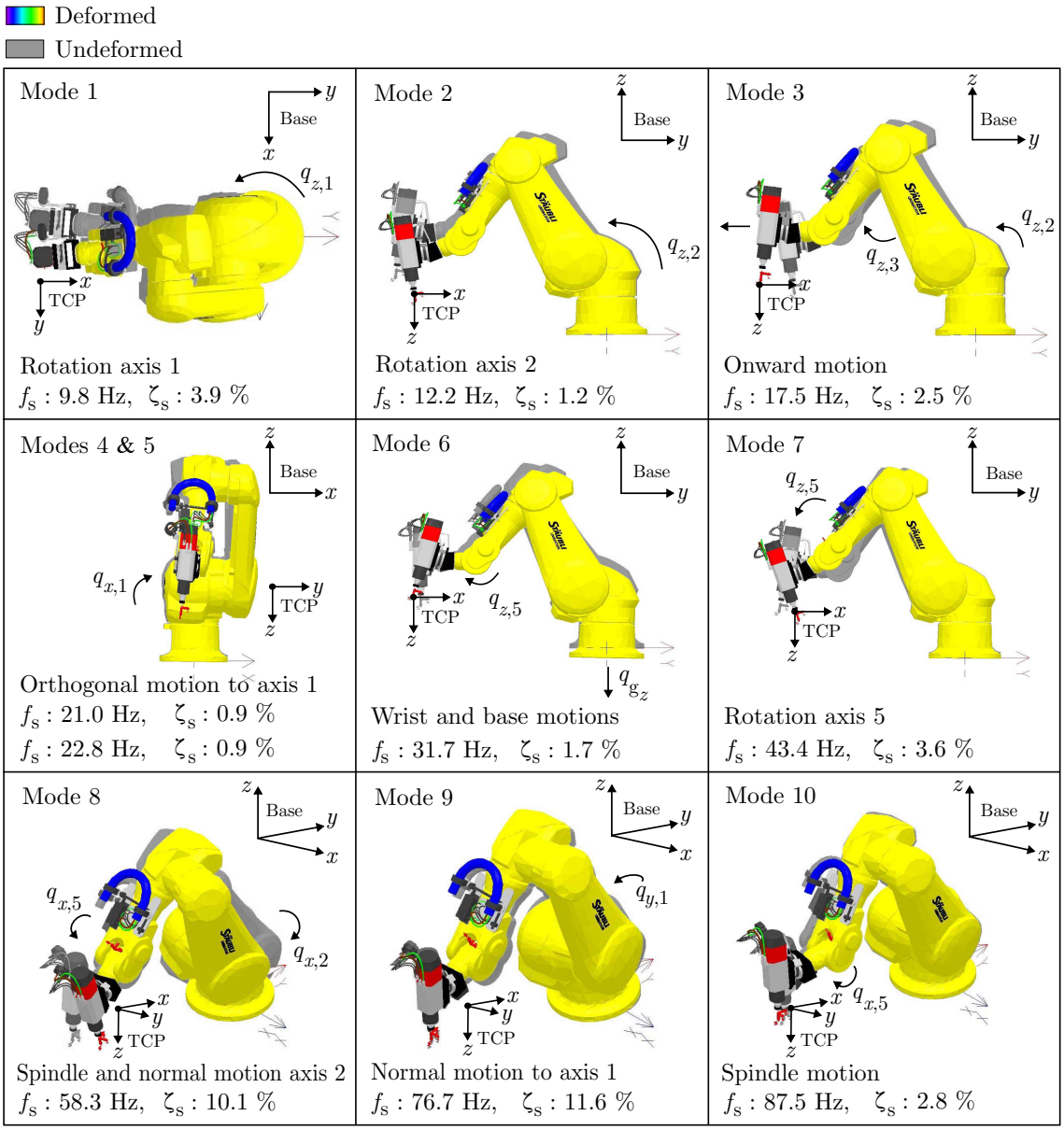


Figure 5.42: Simulated mode shapes in the milling posture of the Stäubli TX200 robot using joint tri-axial flexibility model and the manufacturer's data for the link properties

measured. Simulated mode shape of mode eight involves motions at same areas as the measured mode shape. However, it seems that the motion of the simulated mode shape arises from a perpendicular deflection to the fifth axis instead of a rotation of the sixth joint as suggested in the measured mode shape. Simulated mode shape of mode nine is the least comparable because it suggests a deflection perpendicular to the motion axis of the first joint but around the y local axis ($q_{y,1}$) instead of motions around $q_{x,2}$, $q_{x,3}$ and $q_{x,5}$. Note that the simulated damping ratios of modes eight ($\zeta_{s,8}$=10.1 %) and nine ($\zeta_{s,9}$=11.6 %) are overestimated compared with the experimental ones (ζ_8=4.7 % and ζ_9=2.8 %). For the last simulated mode shape, while the simulated frequency and damping ratio ($f_{s,10}$=87.5 Hz and $\zeta_{s,10}$=2.8 %) are close to the measured ones (f_{10}=88.7 Hz and ζ_{10}=2.3 %), the simulated motion seems to come from an axis perpendicular to joint five while the measured mode shape suggests a rotation of the last joint around

its motion axis. Nevertheless, since impact points in the LMS Test.Lab software move according to the directions of impact (translational motion), it is somewhat difficult to assert the correlation with simulated rotational motions. The comments relative to the mode shapes are applicable to the Stäubli and UMons models.

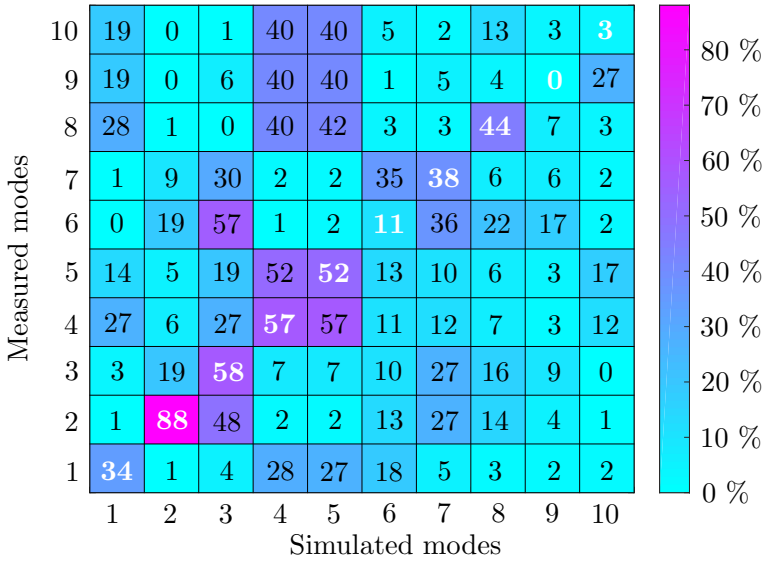


Figure 5.43: MAC matrix in the milling posture of the Stäubli TX200 model using the first 10 modes

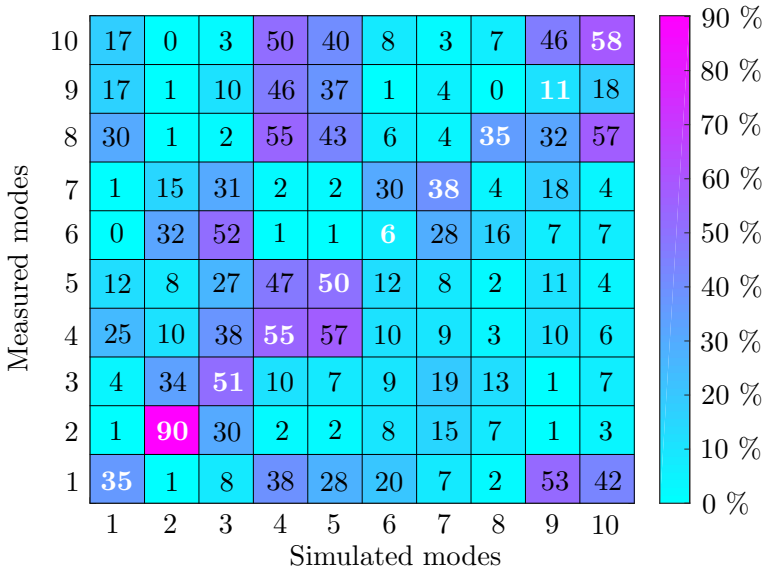
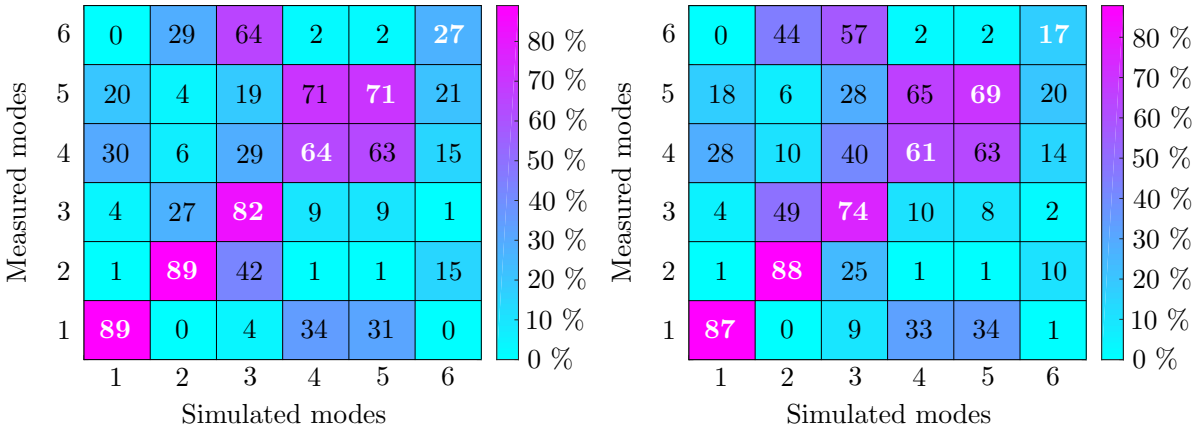


Figure 5.44: MAC matrix in the milling posture of the UMons TX200 model using the first 10 modes

Regarding the corresponding MAC matrices of the simulated mode shapes, they are presented in Figures 5.43 and 5.44. In these figures, all the ten modes below 100 Hz are considered and hence dispersal in the modal identification provided by LMS Test.Lab may happen. However, highest correlations are found for the first five modes for both

models. In order to reduce the dispersal in the modal identification provided by LMS Test.Lab and focus on the modes hindering milling operations below 35 Hz, modes are identified over a bandwidth of 35 Hz for the milling posture through the stabilisation diagram. Resulting MAC matrices comparing the new measured mode shapes over a bandwidth of 35 Hz are again compared to the same simulated mode shapes limited to the first six modes in Figures 5.45a and 5.45b for the Stäubli and UMons models. While similar results are derived for both models, higher correlations are now observed if the bandwidth is limited to the first six modes. High correlations over 60 % are noticed for the first five mode shapes with modes one and two being sharply correlated. Although the simulated mode shapes of the sixth mode seemed to agree well with the animations of LMS Test.Lab, their correlation does not reach 30 %.



(a) Stäubli model (b) UMons model

Figure 5.45: MAC matrices in the milling posture using the first 6 modes

The identified modal parameters of both models are used to construct the FRFs between the three DOFs at the TCP, x_{77}, y_{77} and z_{77} in the milling posture. The magnitudes of the frequency response function matrix $\mathbf{H}(\omega)$ are presented in the receptance format in Figure 5.46 over a bandwidth of 100 Hz. Overall, simulated frequency peaks, which are the most flexible, are concentrated in the low frequency region as for the measured FRFs. Discrepancies occur for higher frequency peaks beyond 35 Hz.

A close-up of Figure 5.46 over the frequency bandwidth below 35 Hz is proposed in Figure 5.47. Even if simulated frequency peaks do not strictly match the measured ones, all the first six peaks are located along the correct impact directions e.g. three simulated peaks observed in H_{yy} (9.4, 20.2 and 22.5 Hz) as in the measured FRFs. It should be pointed out that the identification of the 42 modal parameters is a highly undetermined problem. Since measured mode shapes presented deflection, even small from all the directions of the joints, it was decided to keep all three elastic elements for all the joints during the fitting of modal parameters. Effect of DVA is noticeable around 21 Hz in H_{yy} with the splitting into two smaller peaks.

Figures 5.46 and 5.47 presented results if the whole proposed identification procedure is followed. However, if synthesized FRFs are fitted over a bandwidth of 35 Hz and only the

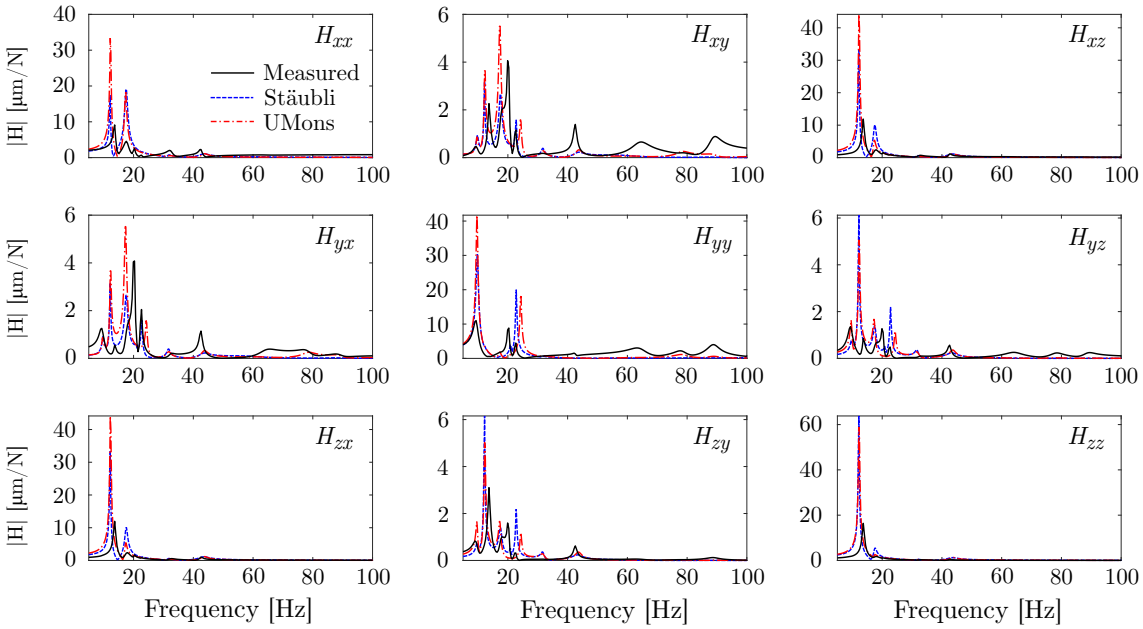


Figure 5.46: Amplitude of measured and fitted frequency response function matrix $\mathbf{H}(\omega)$ at the TCP obtained with the joint tri-axial flexibility model for the milling posture over a bandwidth of 100 Hz

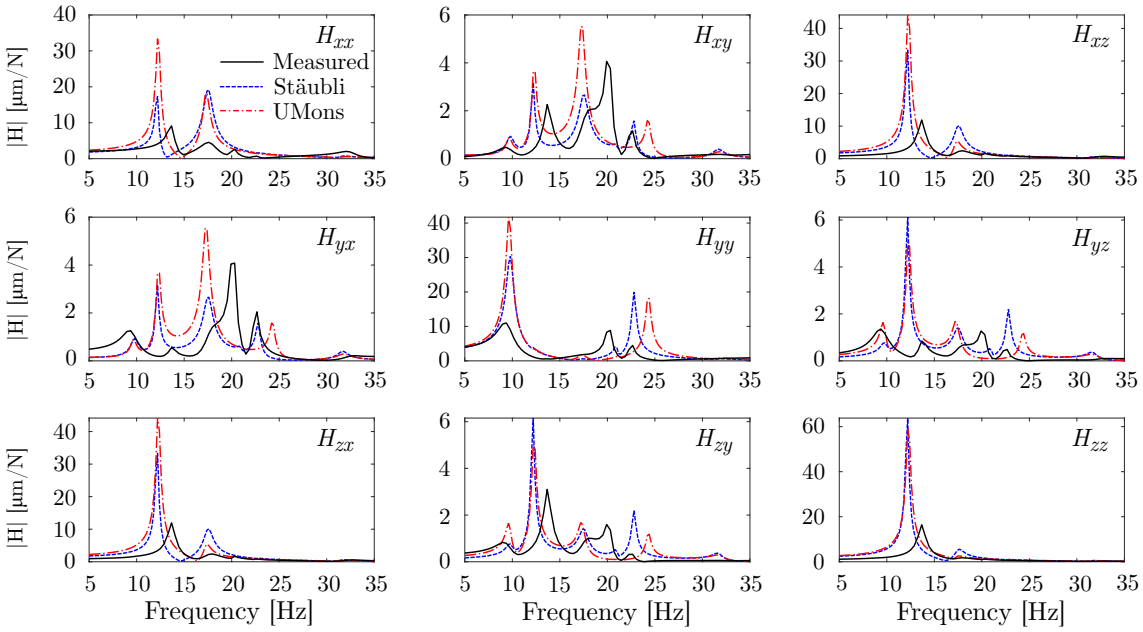


Figure 5.47: Amplitude of measured and fitted frequency response function matrix $\mathbf{H}(\omega)$ at the TCP obtained with the joint tri-axial flexibility model for the milling posture over a bandwidth of 35 Hz

first step of the identification procedure is completed, frequency response function matrix at the TCP presented in Figure 5.48 is obtained. Overall, FRFs seem to fit quite well the synthesized FRFs, for the Stäubli model, but a closer look reveals that the damping of some modes is too high. Hence, resulting stability lobe diagrams are not expected to correlate experimental results since the fitting of the dampings is as important as the one

for the natural frequency for the simulation of milling operations.

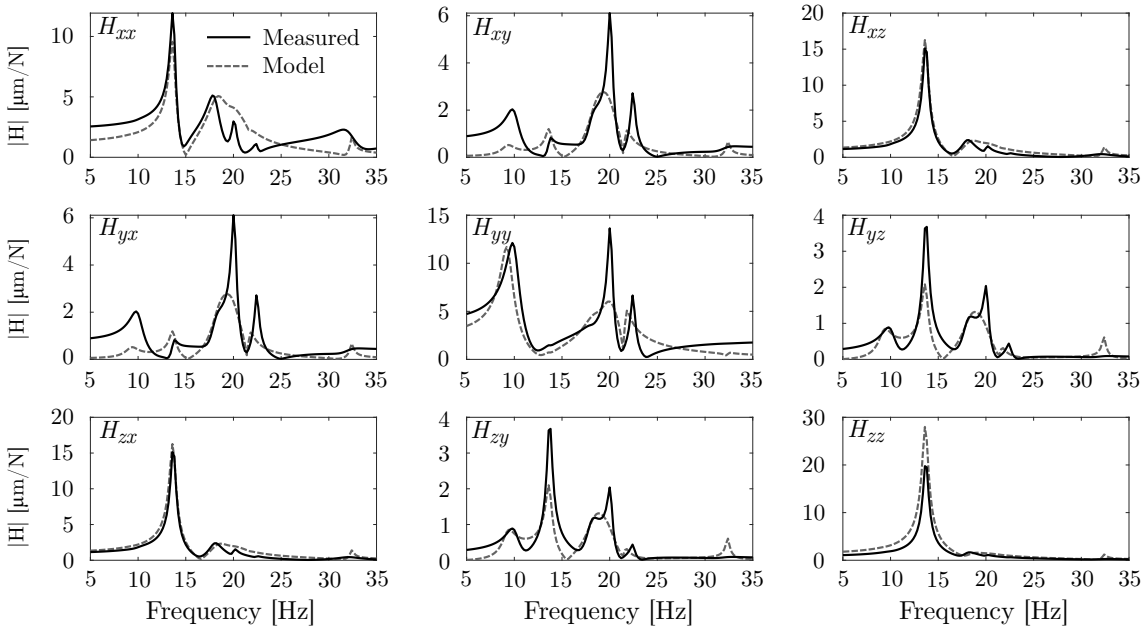


Figure 5.48: Amplitude of measured and fitted frequency response function matrix $\mathbf{H}(\omega)$ at the TCP obtained with the joint tri-axial flexibility model for the milling posture over a bandwidth of 35 Hz only using the genetic algorithm and the Stäubli model

The comparison between the fitted and measured modal frequencies and damping ratios in the milling posture is presented in Table 5.19 for the Stäubli and UMons models. The last row shows the average errors on the natural frequencies and damping ratios relatively to the measured ones. Again, it is observed that the errors on the natural frequencies are lower than those on the damping ratios. For this robot posture, it seems that the UMons model provides a better fit for the modal damping ratios.

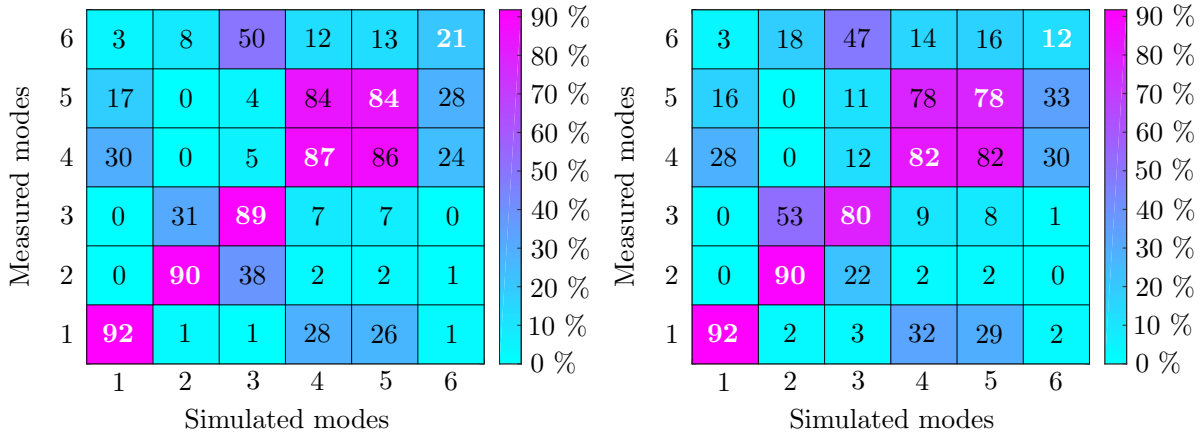
	Measured		Stäubli model		UMons model	
Mode	f_n [Hz]	ζ_n [%]	f_n [Hz]	ζ_n [%]	f_n [Hz]	ζ_n [%]
1	9.4	7.9	9.8	3.9	9.7	3.1
2	13.7	2.2	12.2	1.2	12.3	1.7
3	17.7	4.5	17.5	2.5	17.3	2.0
4	20.2	1.0	21.0	0.9	21.1	1.0
5	22.5	0.8	22.8	0.9	24.3	1.0
6	32.4	2.7	31.7	1.7	31.8	2.3
7	42.5	1.1	43.4	3.6	43.6	2.3
8	64.2	4.7	58.3	10.1	63.2	6.4
9	78.2	2.8	76.7	11.6	78.3	2.8
10	88.7	2.3	87.5	2.8	88.8	2.3
$\text{Error}_{\text{Average}}$ [%]	/	/	3.8	87.8	3.4	32.4

Table 5.19: Fitting of the natural frequencies and damping ratios in the milling posture using the joint tri-axial flexibility model

5.3.4 Validation of the posture dependency

Identified joint elastic parameters with the joint tri-axial flexibility model (Stäubli and UMons) are used to predict the mode shapes and modal parameters in the milling posture with straight wrist, and in the square and flexible postures. The MAC matrices between the simulated and measured mode shapes in all three postures with straight wrist are shown from Figures 5.49 to 5.51.

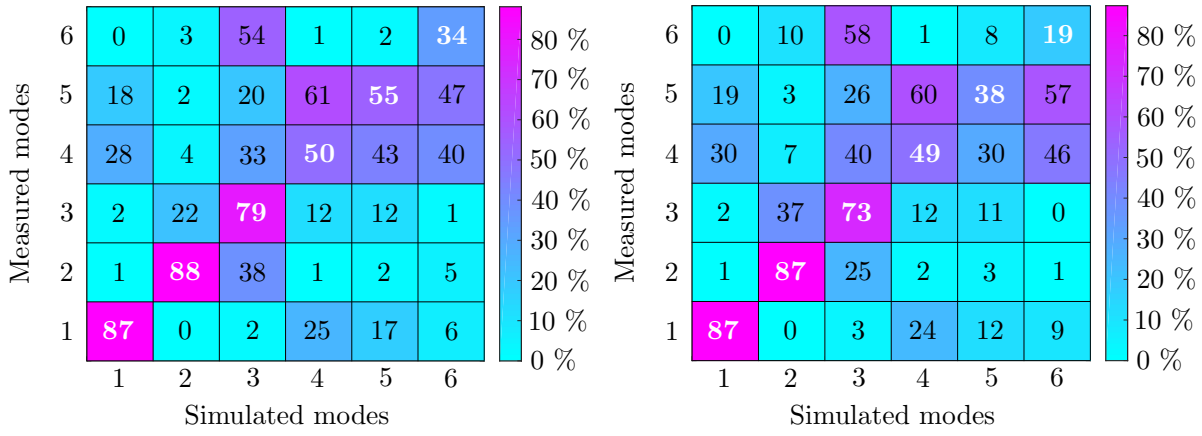
For the milling posture with straight wrist (Subsection 5.2.4), similar results are obtained in terms of mode shape correlation relatively to the milling posture (Subsection 5.2.3), on which modal parameters were fitted. It is not a surprise since the first modes are not affected by wrist motions.



(a) Stäubli model (b) UMons model

Figure 5.49: MAC matrices in the milling posture with straight wrist using the first 6 modes

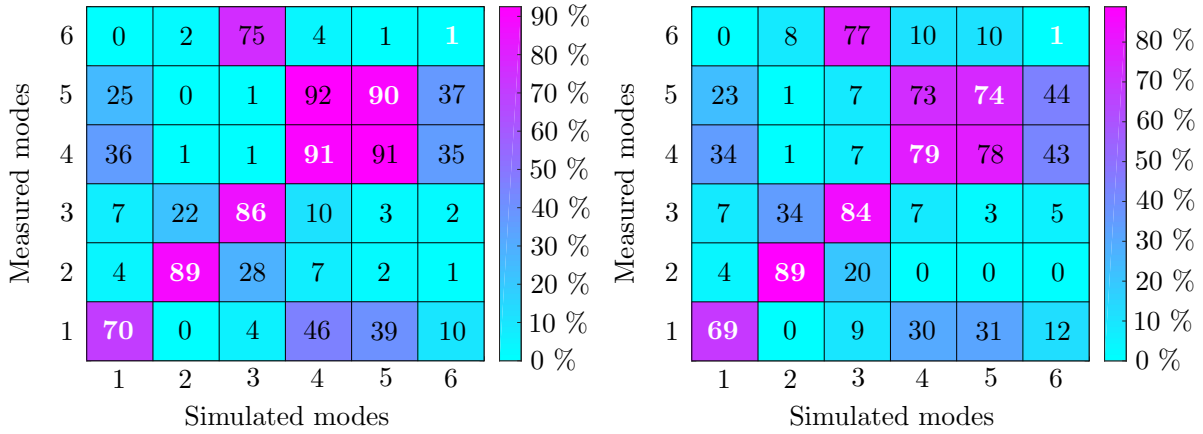
In the square posture, while the first three modes are still highly correlated, there is a drop of correlation for modes four and five. Animations of the simulated mode shapes seem still close to the ones generated by the LMS Test.Lab software.



(a) Stäubli model (b) UMons model

Figure 5.50: MAC matrices in the square posture using the first 6 modes

In the flexible posture, the reverse trend happens as there is a very slight drop in the correlation of the first three modes but an increase in mode correspondence for modes four and five. Animations of the simulated mode shapes are also close to the ones generated by the LMS Test.Lab software.



(a) Stäubli model (b) UMons model

Figure 5.51: MAC matrices in the flexible posture using the first 6 modes

Finally, the comparisons between the predicted and measured modal frequencies and damping ratios are shown for all the three postures in Table 5.20. Only the simulated modal frequencies and damping ratios obtained with the Stäubli TX200 model are displayed even though both models output similar modal parameters. Again, the prediction of the natural frequencies is better managed than for the damping ratios. Smallest average error is evidently reached for the milling posture with a straight wrist. Then, the average error is less than 10 % for the square and flexible postures in terms of natural frequencies. The average errors on the prediction of the damping ratios is still acceptable since it is known to be elusive for such complex mechanical systems.

	Milling straight wrist		Square		Flexible	
Mode	f_n [Hz]	ζ_n [%]	f_n [Hz]	ζ_n [%]	f_n [Hz]	ζ_n [%]
1	8.9 (7.5)	3.6 (6.0)	9.9 (8.8)	2.8(4.6)	7.5 (6.3)	3.6 (6.8)
2	11.3 (12.8)	1.3 (1.5)	10.9 (12.0)	1.3 (2.0)	9.9 (11.9)	1.2 (1.6)
3	17.0 (17.8)	2.3 (4.5)	18.2 (17.6)	2.6 (3.7)	20.2 (21.3)	2.4 (5.6)
4	21.0 (20.2)	0.9 (1.1)	21.0 (20.3)	1.0 (0.6)	21.1 (21.9)	1.0 (1.1)
5	22.7 (22.6)	1.0 (0.8)	24.6 (23.0)	0.7 (1.0)	24.7 (24.9)	1.4 (1.3)
6	32.1 (31.6)	1.7 (3.4)	30.3 (33.8)	1.5 (2.9)	34.6 (31.2)	1.8 (4.8)
$\text{Error}_{\text{Average}}$ [%]	6.6	53.3	7.6	56.1	9.3	73.2

Table 5.20: Prediction of the natural frequencies and damping ratios in the milling posture with straight wrist, square posture and flexible posture using the joint tri-axial flexibility model with the manufacturer's data for the link inertia

Still using the same identified modal parameters from the milling posture, prediction of the mode shapes, natural frequencies and damping ratios in three uncommon postures

are shown in Section K.2 in Appendix K. The conclusion is that only some modes can be captured in the presented extreme postures. Shown results suggest a finer modelling of the joint to better handle the posture dependency of the mode shapes. Nevertheless, as presented in this section, the proposed joint model with three rotational elastic elements per joint might be sufficient for large milling workspace if the robot keeps the configuration in which it was identified.

5.4 Flexible body identification

The characteristics of the equivalent flexible beams representing the robot arm and robot forearm are identified by relying on finite element models. Flexible beam elements are only appended to the Stäubli TX200 model i.e. the UMons model is only limited to the joint flexibilities. The CAD models of the arm and forearm are imported in Abaqus which can derive the mass, damping and stiffness matrices of the flexible element as well as its modal parameters. Note that since the exact geometries of the links are not provided by the robot manufacturer and that redesigned CAD models found in Subsection 5.1.2 are too complex to be meshed, simpler CAD models respecting the overall dimensions and shapes had to be created to generate a healthy finite element mesh. Both links are meshed with tetrahedron elements (C3D10: 10-node quadratic tetrahedron). After assigning the material properties (steel for the arm and aluminium for the forearm), usually defining the Young's modulus and Poisson's ratio, the boundary conditions are set free and a virtual modal analysis of the flexible beams is carried out.

The only subtlety included in the finite element model is that a so-called *kinematic coupling* is specified at the connecting points of the links. In other words, a constraint is applied to the interface nodes (usually on the surface representing the joint) to merge them into one node. As a result, the surfaces at the joint interface are considered as rigid. Figure 5.52 illustrates the kinematic coupling of the fifth joint connecting the wrist. Kinematic coupling is also applied to the extremities of the robot arm.

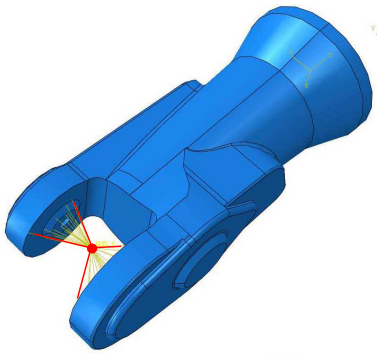


Figure 5.52: Kinematic coupling at joint

Density ρ:	1770 kg/m^3	Length L:	0.95 m
Young's modulus E:	15.78 e^9 N/m^2	Width b:	0.425 m
Poisson's ratio ν:	0.3	Height h:	0.255 m

Table 5.21: Beam characteristics for the representation of the flexible robot arm

Equivalent flexible beam properties for the arm and forearm are presented in Tables 5.21

and 5.22. The lengths L of the beams are set with respect to the geometric dimensions of the Stäubli TX200 robot while the widths b and heights h are defined relatively to the overall shape of the links. The density ρ is tuned relatively to the mass of the links. Finally, the Young's modulus E of each beam is manually fitted on the basis of the virtual modal analysis in Abaqus. Poisson's ratio is usually taken as ν=0.3. Damping is also added to flexible beams using the Rayleigh's proportional damping coefficients such as $\alpha_{\text{Damping}}=\beta_{\text{Damping}}=1\text{e}^{-7}$. They are chosen so that computed modal damping ratios of modes correspond to typical values for steel and aluminium beams (around 0.02 %).

Density ρ:	705 kg/m^3	Length L:	0.65 m
Young's modulus E:	3.33 e^9 N/m^2	Width b:	0.3 m
Poisson's ratio ν:	0.3	Height h:	0.3 m

Table 5.22: Beam characteristics for the representation of the flexible robot forearm

Comparisons of the mode shapes, natural frequencies and modal damping ratios for the robot arm between the finite element model and the flexible beam model are depicted in Figure 5.53. Only the first three modes are considered.

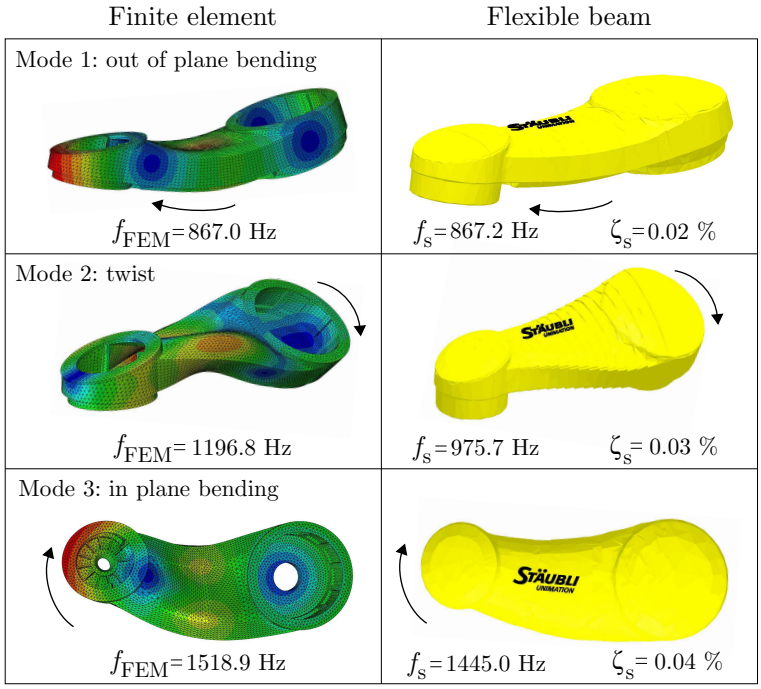


Figure 5.53: Finite element modelling of the robot arm and corresponding representation with flexible beam elements

The tuning of the Young's modulus allows updating the first natural frequency of the simulated flexible beam. Mode shapes are still coherent with the animations generated with the finite element model in Abaqus. First mode is a bending of the arm all along the length of the flexible beam at 867 Hz. Second mode is represented by a twist motion at 1196.8 Hz and the third mode is again a bending but in the lateral direction of the beam at 1518.9 Hz. Similar results are obtained for the robot forearm. The first three modes are as follows: $f_{\text{FEM},1}$=1042.4 Hz, $f_{\text{FEM},2}$=1087.2 Hz

and $f_{\text{FEM},3}$=1236.6 Hz. Such model updating is sufficient for robotic milling operations since the excited frequencies are far below the ones accounting for the link flexibility.

Finally, Table 5.23 proposes a summary of all simulated modal parameters when the joint tri-axial flexibility model of the robot includes the compliances of the arm and forearm for the Stäubli TX200 model. The model therefore includes n=34 degrees of freedom since it is virtually identified in a static pose. It is observed that the addition of the flexible beams slightly lowers the natural frequencies relative to the joint modes. After assembling the flexible beams to the model, first mode shape involving a deflection of the link is found at 270.1 Hz with an out of plane bending of the arm.

Mode	f_n [Hz]	ζ_n [%]
1	9.62	3.55
2	11.78	1.19
3	16.88	2.43
4	20.77	0.84
5	21.88	1.01
6	31.23	1.58
7	43.91	3.66
8	60.52	10.6
9	74.77	10.9
10	83.76	4.05
11	91.95	2.62
12	140.9	6.6
13	179	4.09
14	207.6	46.1
15	210.9	2.85
16	219.4	7.19
17	270.1	6.53
18	303.1	7.55
19	346.7	6.75
20	520.7	12
21	523.5	1.78
22	590.9	2.94
23	672	9.63
24	729.7	2.82
25	831	7.12
26	855.5	1.52
27	921.9	18.9
28	1163	9.26
29	1195	10.2
30	1220	0.644
31	1387	60.7
32	1994	1.14
33	2282	23.1
34	3439	27.1

Table 5.23: Modal properties of the Stäubli robot model using the manufacturer's data for the link inertia and including the flexible beams representing the arm and forearm

5.5 Actuator parameters

Actuators are only included in the Stäubli TX200 model while the UMons TX200 model is limited to its flexible joints i.e. perfect regularisation of motors is assumed for the UMons TX200 model. The rotor inertias are known from manufacturer's data. From the selected model of actuator introduced in Chapter 3, two additional parameters must be known to model the behaviour of a DC motor: the torque constant $k_{\mathrm{t,i}}$ (numerically equal to the back electromotive constant $k_{\mathrm{v,i}}$) and $R_{\mathrm{a,i}}$ the armature resistance of the motor associated to joint i.

Unfortunately, no information is provided from the robot manufacturer regarding the two additional parameters. Hence, the missing parameters were derived from the data written on the motor plates of the robot drive, namely the stall torque M_0 and the no load speed N_0. These characteristics are supplied at fixed voltage U_{DC}=560 V. The torque and back electromotive constants of each DC motor are then computed using the following relationships

$$k_{\mathrm{t}} = \frac{U_{\mathrm{DC}}}{N_0}, \qquad \text{and} \qquad R_{\mathrm{a}} = \frac{k_{\mathrm{t}} U_{\mathrm{DC}}}{M_0}. \tag{5.39}$$

Derived motor parameters for each joint i are presented in Table 5.24.

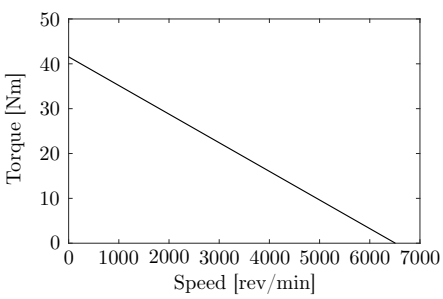
Axis i	$R_{\mathrm{a,i}}$ [Ω]	$k_{\mathrm{t,i}}$ [Nm/A] = $k_{\mathrm{v,i}}$ [V/(rad/s)]
1	11.1	0.8
2	11.1	0.8
3	11.1	0.8
4	51.6	0.6
5	60.0	1.3
6	51.6	0.6

Table 5.24: Parameters of the modelled DC motors

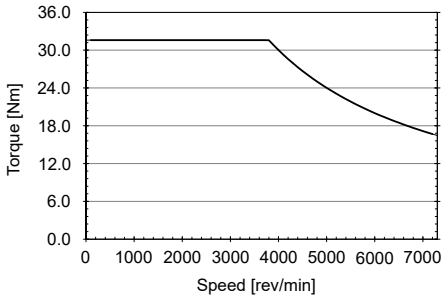
An example of torque-speed characteristic curve is presented in Figure 5.54a for the second motor of the Stäubli TX200 model for a stall torque $M_0 = 41.55$ Nm and a no load speed N_0=681.8 rad/s (6,511 rev/min). Nevertheless, for an actual servo-motor, due to current limitation, the torque-speed characteristic curve is limited to a constant torque for a large portion of speed. A typical curve for an equivalent motor found in the servo motor catalogue [161] is illustrated in Figure 5.54b.

5.6 Controller parameters

The gains of the two selected controllers introduced at the end of Chapter 3 are tuned on the basis of the measured FRFs presented in Subsection 5.2.3 in the milling posture. As observed in Figure 5.22 showing the comparison of FRFs either measured with the controller in action or with the motor brakes, robot natural frequencies do not change



(a) Simulated curve of DC motor



(b) Actual curve of servo motor [161]

Figure 5.54: Torque-speed characteristic of motor for the second joint

significantly as the modes under 200 Hz are mainly driven by the flexibility of the joints. Main controller effect on those modes is reflected by a damping increase. Anticipating that measured chatter frequencies are mostly below 90 Hz (as shown in Chapter 7) and that the last modelled robot mode is at 87.5 Hz, gains must be set such that the first controller frequency is beyond those frequencies to not affect modes related to the joints. In other words, the dynamics of the controller must be faster than the robot structural dynamics.

Considering this constraint, a simple tuning of the inverse dynamics controller consists in selecting gains $K_{\mathrm{P,i}}$ and $K_{\mathrm{D,i}}$ such that the eigenvalues of the second-order equations giving the error position dynamics (Eq. 3.103) lead to the desired controller frequencies. The first controller frequency is chosen at 90.4 Hz (beyond, the controller requires such high gains that the time step of numerical integration must be significantly reduced during motion simulations). For the sake of simplicity all axes are tuned at the same frequency with the same damping (50 %). It leads to the values of gains $K_{\mathrm{P,i}}$ and $K_{\mathrm{D,i}}$, feeding the diagonal terms of matrices $\mathbf{K}_{\mathrm{P}}$ and $\mathbf{K}_{\mathrm{D}}$, presented in Table 5.25.

Axis i	$K_{\mathrm{P,i}}$	$K_{\mathrm{D,i}}$
1 - 6	4.3 e^5	$\sqrt{K_{\mathrm{P,i}}}$

Table 5.25: Gains of the inverse dynamics controller

Since the inverse dynamics controller provides a full compensation of non-linear terms in the equations of motion, it was retained for the simulation of robotic milling operations. Note that the developed independent joint control can be tuned analogously. For a first controller frequency chosen at 91.7 Hz and a damping ratio of 35 %, the gains must be set as follows: $K_{\mathrm{P,1\text{-}6}}$=500, $T_{\mathrm{P,1\text{-}6}}$=250, $K_{\mathrm{D,1\text{-}3}}$=150 and $K_{\mathrm{D,4\text{-}6}}$=50.

5.7 Cutting coefficients

Simulations of milling operations require the identification of the cutting coefficients introduced in Chapter 4 when presenting the three methods assessing their stability. As a reminder, the complete linear cutting force model reads

$$\begin{bmatrix} dF_{\mathrm{t_{j,k}}} \\ dF_{\mathrm{r_{j,k}}} \\ dF_{\mathrm{a_{j,k}}} \end{bmatrix} = \begin{bmatrix} K_{\mathrm{tc}} \\ K_{\mathrm{rc}} \\ K_{\mathrm{ac}} \end{bmatrix} dh_{\mathrm{j,k}} db_{\mathrm{j,k}} + \begin{bmatrix} K_{\mathrm{te}} \\ K_{\mathrm{re}} \\ K_{\mathrm{ae}} \end{bmatrix} dS_{\mathrm{j,k}}, \tag{5.40}$$

with K_{tc}, K_{rc} and K_{ac} the tangential (subscript t), radial (subscript r) and axial (subscript a) cutting force coefficients, and coefficients K_{te}, K_{re} and K_{ae} the edge force coefficients. The identified coefficients can directly be used in the milling dynamic model relying on Eq. 5.40 or converted as follows $K_{\mathrm{t}} = K_{\mathrm{tc}}$, $K_{\mathrm{r}} = \frac{K_{\mathrm{rc}}}{K_{\mathrm{t}}}$ and $K_{\mathrm{a}} = \frac{K_{\mathrm{ac}}}{K_{\mathrm{t}}}$ to be applied in the zero-order or semi-discretisation method. All the six aforementioned coefficients are identified from milling experiments with the Stäubli TX200 robot and are valid for one tool/material couple. Coefficients are determined by relying on an inverse method using the measured cutting forces.

5.7.1 Experiments in robotic milling

Since the identification method requires the measurements of cutting forces, a dedicated force sensor is localised under the workpiece to machine. Raw signals are measured using force sensor Kistler 9257B and are then amplified and acquired using charge amplifier Kistler 5070A11100 SN4178679 and acquisition card Kistler 5697A SN3173295. Used cutting force sensor is able to measure the cutting forces along three orthogonal directions. By convention and throughout the rest of the document, measured forces are projected in a frame such that F_x designates a force along the feed direction, F_y indicates a force perpendicular to F_x and F_z is oriented along the revolution axis of the tool. Measured cutting force signals will also be used to determine the stability of milling operations. To not only rely on the cutting force signals for the stability analysis, a tri-axial accelerometer is also appended to the set-up depicted in Figure 5.55. Another reason of adding the same tri-axial accelerometer that was used for modal analysis tests is that its acquisition card (NI9233 inserted in the NI-cDAQ9184 chassis) comprises an anti-aliasing filter. The measuring chain is therefore aimed at capturing both cutting force and vibration signals from the workpiece side during milling operations. All the measured signals are then collected on a Personal Computer (HP630 Core I3 4GB RAM) by using Dynoware software for the cutting forces and the data acquisition toolbox implemented in Matlab for the vibrations. Sampling rates for the cutting force and vibration signals are chosen at 20,000 Hz and 25,000 Hz, respectively.

A close-up of the actual set-up, focussing on both sensors is shown in Figure 5.56. The workpiece is clamped on the top surface of the force sensor which is mounted on a support. The bench vice squeezes the force sensor support between its jaws. The tri-axial accelerometer is fastened on the force sensor support. Overall, the force sensor and the accelerometer are reasonably rigidly attached to the steel slab via the bench vice.

Two materials are considered during the milling tests, namely aluminium (6082) and steel (St52-3). Consequently, two different tools, whose characteristics are presented in Table 5.26, are utilised for all the considered milling operations. State of the tool is

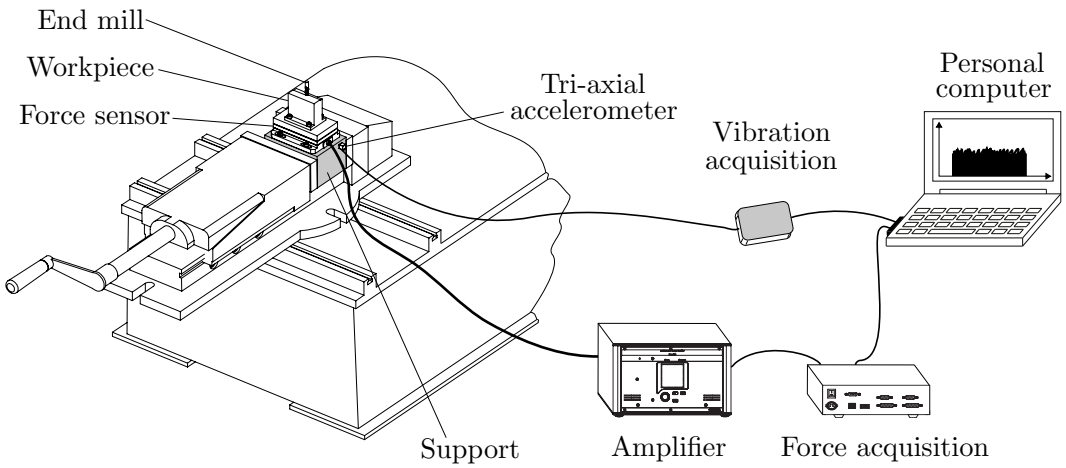


Figure 5.55: Measuring chain for milling tests with a force sensor and a tri-axial accelerometer

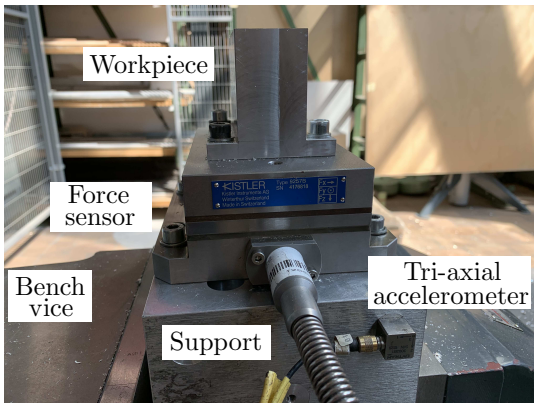


Figure 5.56: Actual set-up of the workpiece bolted on the force sensor

checked after each milling operation and replaced if necessary. Conditions of replacement regard a tool breakage (which did not happen), severely worn or cracked tooth. As mentioned earlier, both tools are cylindrical and modelled using the parametric model exposed in Subsection 4.5.1 by including their helix angle and tooth variable pitch.

Dimensions of the workpiece utilised for cutting coefficient identification and stability analysis are showcased in Figure 5.57a. The same geometry is machined for milling operations in aluminium and steel. Layers of material are removed from the top surface of the block by face milling via the zig-zag and the one way strategies with a constant radial depth of cut a_e=4 mm. Axial depths of cut and spindle speeds are varied in order to assess stability on a large domain of milling conditions. Ranges of axial depth of cut are determined on site on the basis of the stability of the operation. The axial depth of cut is increased until unstable cutting conditions occur. If severe instability occurs, the produced sound is different and well recognisable from a stable case. Chatter marks are also left on the workpiece. Spindle speed range in aluminium is set from 2,500 rev/min to 22,500 rev/min with a step of 2,500 rev/min whereas in steel, it is defined from 2,000 rev/min to 10,000 rev/min with a step of 500 rev/min as presented in the stability lobe diagrams shown in Chapter 7. The full height of the block is machined. Note that

End mill for aluminium milling	
Type	Cylindrical
Tooth count N_z	2
Variable pitch	170°-190°
Helix angle	30°
Feed per tooth f_z	0.13 $\frac{mm}{tooth}$
Manufacturer	Seco
Serial number	JS412100D2SZ2.0

(a) Aluminium milling

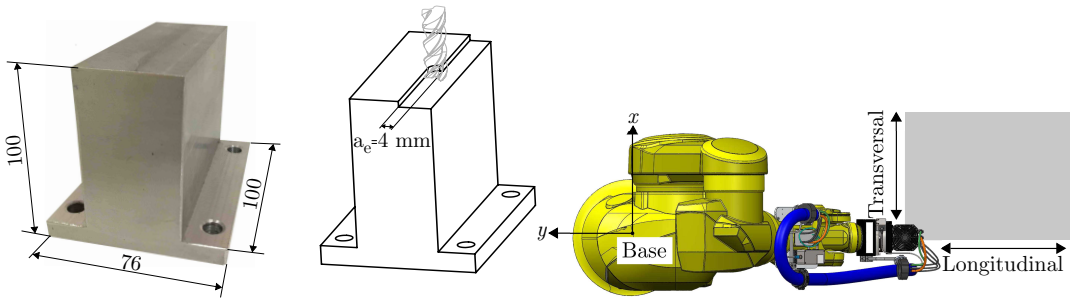
End mill for steel milling	
Type	Cylindrical
Tooth count N_z	3
Variable pitch	115°-135°-110°
Helix angle	48°
Feed per tooth f_z	0.09 $\frac{mm}{tooth}$
Manufacturer	Seco
Serial number	553100Z3.0-SIRON-A

(b) Steel milling

Table 5.26: Characteristics of the used end mills

some thin plates (100 × 90 × 10 mm) of the same materials are also machined. Each plate is machined for one specific combination of axial depth of cut and spindle speed, while keeping the same radial depth of cut (a_e= 4 mm). The purpose is to analyse the resulting surfaces with a coordinate measuring machine (CMM WENZEL LH 54) and a roughmeter (Diavite DH-6).

Beside assessing various cutting conditions, the influence of the robot configuration on the milling stability is also studied. Henceforth, milling operations taking place within the plane formed by its arm and forearm are termed as *longitudinal* (parallel to y_{Base}) and perpendicular motions (parallel to x_{Base}) are so-called *transversal*, as illustrated in Figure 5.57b. The distance between the workpiece and the robot base is about 1.5 m along y_{Base}.



(a) Workpiece (b) Robot motions

Figure 5.57: Definition of the workpiece and the robot motions for cutting force coefficient identification and stability analysis (dimensions in mm)

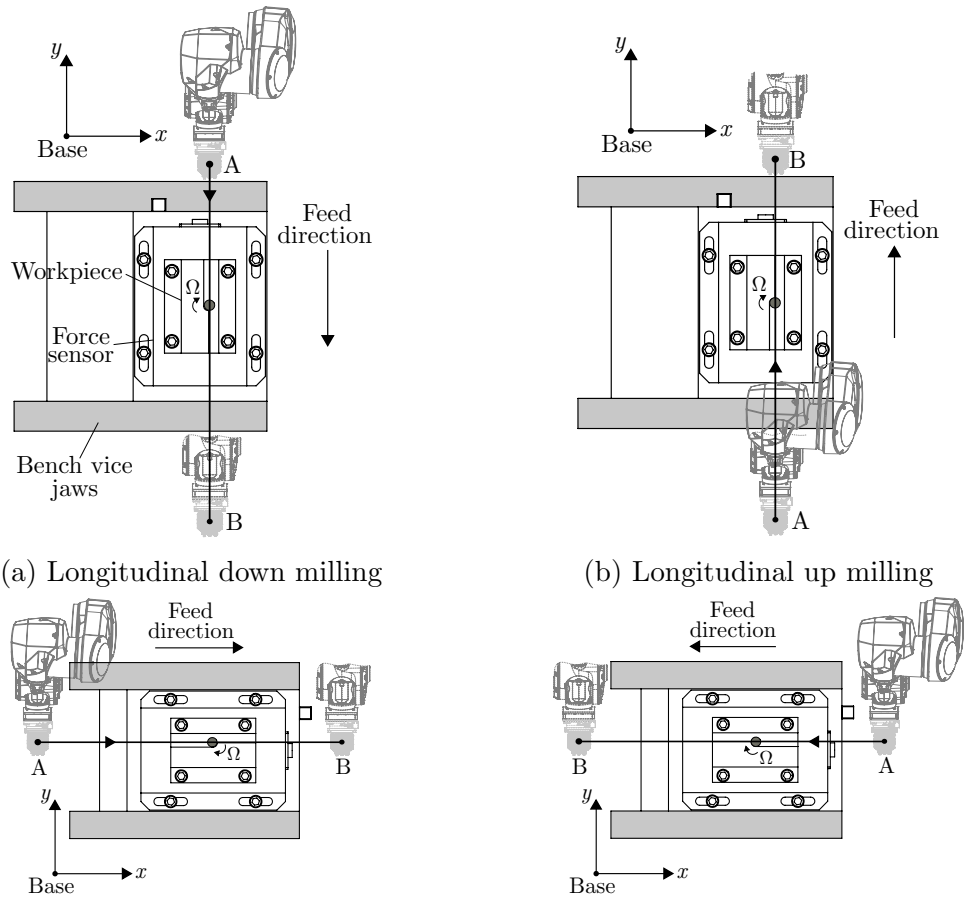
As a result, cutting coefficient identification and stability analysis are carried out in four different situations depicted in Figure 5.58:

1. Longitudinal down milling (Figure 5.58a): TCP moves from point A to B by following a longitudinal motion. Since the end mill always rotates in the clockwise direction, moving away from the robot base results in a down milling pass

2. Longitudinal up milling (Figure 5.58b): TCP moves from point A to B by following a longitudinal motion. In this case, TCP moves towards the robot base and an up milling pass is generated.

3. Transversal down milling (Figure 5.58c): TCP moves from point A to B by following a transversal motion. Material is removed in a down milling fashion as the tool still rotates in a clockwise direction.

4. Transversal up milling (Figure 5.58d): TCP moves from point A to B by following a transversal motion and material is cut out following the up milling direction.



(a) Longitudinal down milling

(b) Longitudinal up milling

(c) Transversal down milling

(d) Transversal up milling

Figure 5.58: The four milling situations

The two considered materials and four milling situations naturally lead to the determination of eight experimental stability lobe diagrams that will be presented in Chapter 7.

5.7.2 Inverse identification method

For each of the eight different milling situations, cutting coefficients are identified by relying on an inverse identification method. The latter was developed by Rivière-Lorphèvre et al. in [171] and performs a least square fitting on the basis of the measured cutting forces to retrieve the optimal cutting coefficients. It is an off-line identification method based on the matrix form of Eq. 5.40 that allows simultaneously determining all the six cutting coefficients (K_{tc}, K_{rc}, K_{ac}, K_{te}, K_{re} and K_{ae}). Equation 5.40 is written in a matrix format such as

$$\begin{bmatrix} dF_{\mathrm{t_{j,k}}} \\ dF_{\mathrm{r_{j,k}}} \\ dF_{\mathrm{a_{j,k}}} \end{bmatrix} = \underbrace{\begin{bmatrix} dh_{\mathrm{j,k}}db_{\mathrm{j,k}} & 0 & 0 & dS_{\mathrm{j,k}} & 0 & 0 \\ 0 & dh_{\mathrm{j,k}}db_{\mathrm{j,k}} & 0 & 0 & dS_{\mathrm{j,k}} & 0 \\ 0 & 0 & dh_{\mathrm{j,k}}db_{\mathrm{j,k}} & 0 & 0 & dS_{\mathrm{j,k}} \end{bmatrix}}_{\mathbf{A}_\mathrm{c}} \underbrace{\begin{bmatrix} K_{\mathrm{tc}} \\ K_{\mathrm{rc}} \\ K_{\mathrm{ac}} \\ K_{\mathrm{te}} \\ K_{\mathrm{re}} \\ K_{\mathrm{ae}} \end{bmatrix}}_{\mathbf{K}_\mathrm{c}}. \tag{5.41}$$

As developed in Chapter 4, Cartesian cutting forces are computed by summing all local forces on the $\mathrm{N_z}$ cutting edges and on the $\mathrm{N_s}$ slices axially discretising the end mill such that

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \underbrace{\left(\sum_{\mathrm{j}=1}^{\mathrm{N_z}} \sum_{\mathrm{k}=1}^{\mathrm{N_s}} \mathbf{B}_\mathrm{c}\mathbf{A}_\mathrm{c} \right)}_{\mathbf{C}_\mathrm{c}} \mathbf{K}_\mathrm{c}, \tag{5.42}$$

with $\mathbf{B}_\mathrm{c}$ being the projection matrix of the local forces to the Cartesian space which was defined as

$$\mathbf{B}_\mathrm{c} = \begin{bmatrix} -\cos\varphi_\mathrm{j} & -\sin\varphi_\mathrm{j}\sin\kappa & -\sin\varphi_\mathrm{j}\cos\kappa \\ \sin\varphi_\mathrm{j} & -\cos\varphi_\mathrm{j}\sin\kappa & -\cos\varphi_\mathrm{j}\cos\kappa \\ 0 & \cos\kappa & -\sin\kappa \end{bmatrix}. \tag{5.43}$$

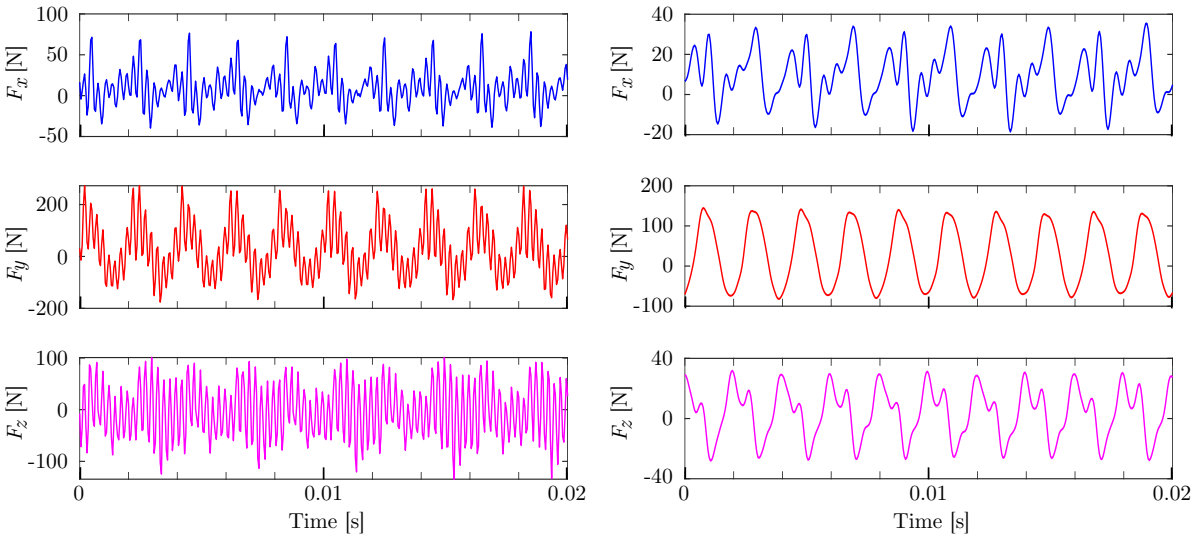
Equation 5.42 can be written for each sampling instant i for which cutting force F_x, F_y and F_z are measured. By assembling all the instants, a linear relationship can be established in which the matrix of cutting coefficients $\mathbf{K}_\mathrm{c}$ is unknown. The linear relationship including all the sampling instants ii reads

$$\underbrace{\begin{bmatrix} F_{x,1} \\ F_{y,1} \\ F_{z,1} \\ \hline F_{x,2} \\ F_{y,2} \\ F_{z,2} \\ \hline \vdots \\ \hline F_{x,ii} \\ F_{y,ii} \\ F_{z,ii} \end{bmatrix}}_{\mathbf{F}_\mathrm{c}} = \underbrace{\begin{bmatrix} \mathbf{C}_{\mathrm{c},1} \\ \hline \mathbf{C}_{\mathrm{c},2} \\ \hline \vdots \\ \hline \mathbf{C}_{\mathrm{c},ii} \end{bmatrix}}_{\mathbf{D}_\mathrm{c}} \mathbf{K}_\mathrm{c}. \tag{5.44}$$

All the six cutting coefficients, comprised in matrix $\mathbf{K}_\mathrm{c}$, are eventually determined by using the LSE method on the above matrices such as

$$\mathbf{K}_\mathrm{c} = (\mathbf{D}_\mathrm{c}^T\mathbf{D}_\mathrm{c})^{-1}(\mathbf{D}_\mathrm{c}^T\mathbf{F}_\mathrm{c}). \tag{5.45}$$

The inputs of the inverse identification method are the measured cutting forces. The latter being noisy, they are first filtered using a low pass filter whose cut-off frequency is set at 1,000 Hz below the first natural frequency of the force sensor. Filtering effect is demonstrated in Figure 5.59. It compares the raw measured cutting forces with the filtered counterparts for a robotic milling operation in aluminium at 15,000 rev/min for an axial depth of cut of 1 mm.



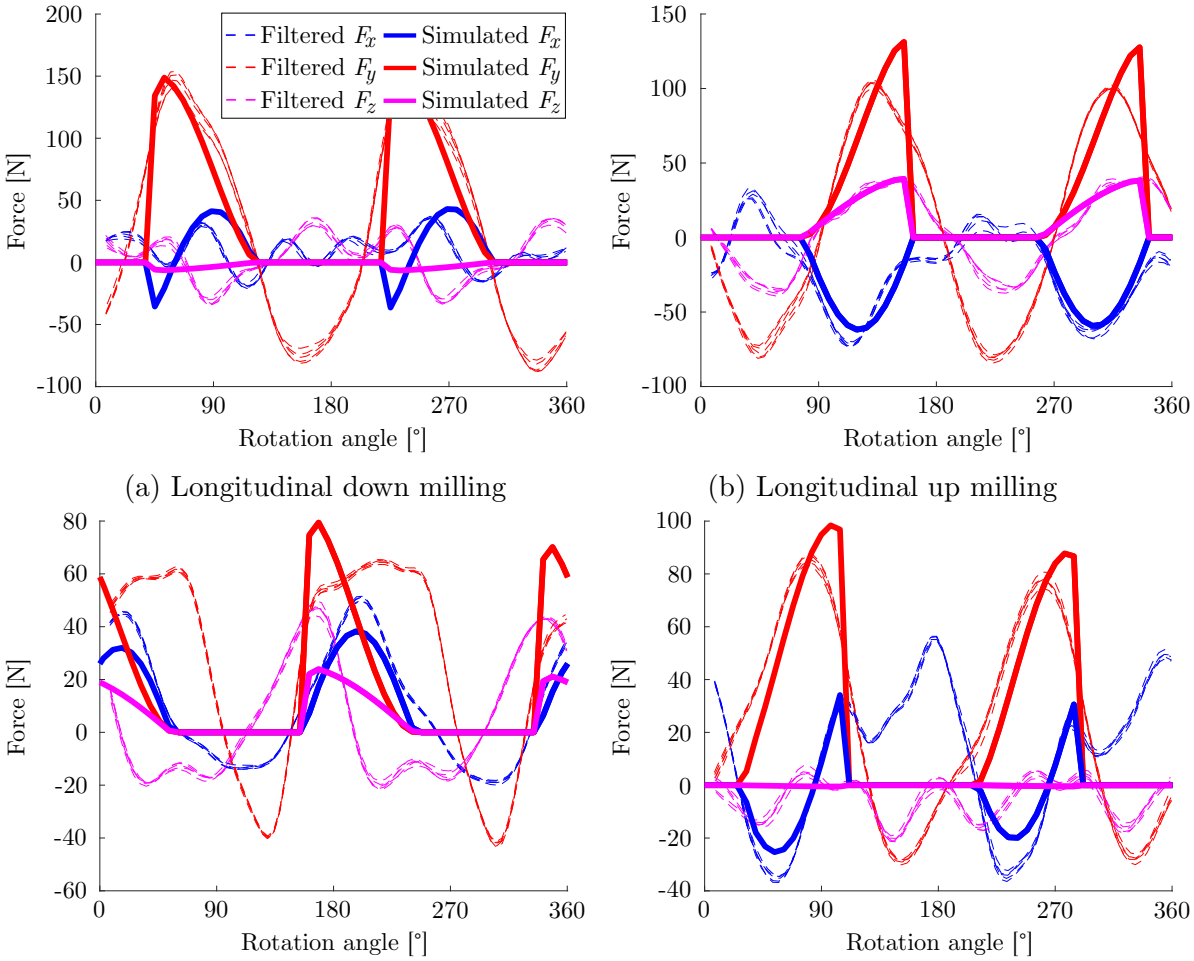
(a) Unfiltered (b) Filtered

Figure 5.59: Comparison of the raw and filtered cutting force signals in aluminium with an axial depth of cut of 1 mm at 15,000 rev/min and a feed rate of 0.13 mm/tooth for the longitudinal pass in down milling

For one milling situation (e.g. longitudinal case in down milling), filtered cutting force signals obtained in stable conditions and at a fixed axial depth of cut are provided to the inverse identification method. As for example in aluminium, Figure 5.60 shows the fitting of the filtered cutting forces with the simulated ones using the identified coefficients for all the milling situations, at 15,000 rev/min and for an axial depth of cut a_{p}= 1 mm. Several tool turns of filtered cutting forces are superimposed in Figure 5.60 under the form of dashed curves. For each milling situation at a constant spindle speed and axial depth of cut, one set of cutting coefficients $\mathbf{K}_{\mathrm{c}}$ is determined. The procedure is repeated for the tested spindle speeds by keeping the same axial depth of cut and for all the milling cases.

Figure 5.61 shows the average cutting coefficients K_{tc}, K_{rc} and K_{ac} obtained for the transversal pass in down milling and with an axial depth of cut of 1 mm in aluminium. It is also observed that the identified cutting coefficients are quite constant over the range of spindle speeds. Spindle speeds beyond 15,000 rev/min are excluded because it was observed that the cutting force level was influenced by static deviation of the robot.

Repeating and averaging the mean coefficients of each situation and for both materials, application of the inverse identification method yields the identified cutting coefficients



(a) Longitudinal down milling

(b) Longitudinal up milling

(c) Transversal down milling

(d) Transversal up milling

Figure 5.60: Cutting force fitting in aluminium with an axial depth of cut of a_p= 1 mm (a_e= 4 mm), a spindle speed of 15,000 rev/min and a feed rate of 0.13 mm/tooth in the four milling situations

enclosed in Table 5.27. The whole procedure is carried out twice in order to determine the cutting coefficients for the standard model (without edge force) and for the cutting force model with edge forces. Cutting force coefficients reported in Table 5.27a for aluminium and in Table 5.27b for steel are coherent with the values commonly found in the literature.

5.8 Discussion

In this chapter, all the values of parameters involved in the robot multibody models and in the milling models are determined through various identification methods. The identification of the inertial parameters of the Stäubli TX200 robot is first examined. Although the actual values of the link and rotor inertias are provided by the robot manufacturer under a non-disclosure agreement (Stäubli TX200 model), a public model of the robot was created on the basis of reshaped CAD models (UMons TX200 model). By comparing the mass matrix of both models in a milling configuration, it is observed that they are quite close. It means that a sufficiently accurate robot model can be set up

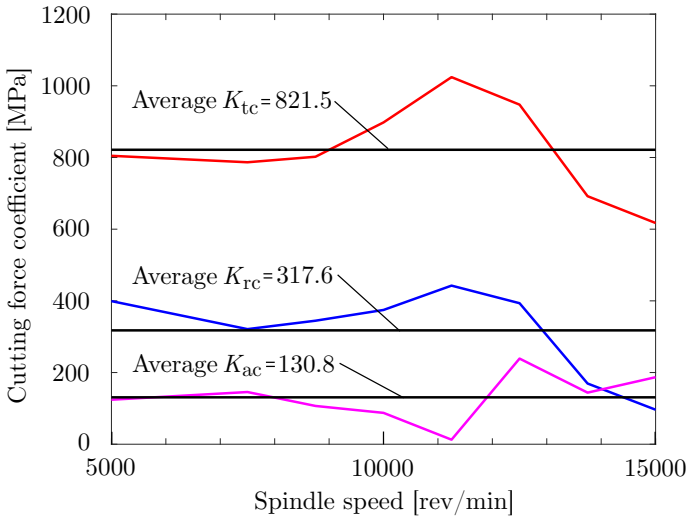


Figure 5.61: Identification of the cutting force coefficients for the transversal pass in down milling with an axial depth of cut of 1 mm in aluminium

Standard model	
K_{tc} [MPa]	733.5
K_{rc} [MPa]	346.6
K_{ac} [MPa]	127.9

Edge force model			
K_{tc} [MPa]	492.5	K_{te} [N/mm]	27.7
K_{rc} [MPa]	-90.1	K_{re} [N/mm]	32.7
K_{ac} [MPa]	-191.0	K_{ae} [N/mm]	7.2

(a) Aluminium milling

Standard model	
K_{tc} [MPa]	1930.4
K_{rc} [MPa]	1159.6
K_{ac} [MPa]	200.6

Edge force model			
K_{tc} [MPa]	1452.3	K_{te} [N/mm]	30.7
K_{rc} [MPa]	6.7	K_{re} [N/mm]	46.9
K_{ac} [MPa]	-194.4	K_{ae} [N/mm]	20.8

(b) Steel milling

Table 5.27: Identified cutting force coefficients

based on redesigned CAD models. A rigid body identification method, which relies on torque measurements along excitation trajectories, is also implemented and leads to the determination of combinations of inertial parameters, so-called base parameters. A strong correlation is noticed between the identified base parameters and their counterparts computed on the basis of the actual inertial parameters. The inconvenience is that such identification does not allow the identification of elementary inertial parameters, but only combinations of them. Having determined the inertial properties of the Stäubli TX200 robot, experimental modal analyses are conducted to identify its natural frequencies, modal damping ratios and mode shapes in different postures. EMA results are eventually used to update the three models of the Stäubli TX200 robot (Stäubli, UMons TX200 models and the one relying on the base parameters). Robot model identified via the excitation trajectories shows limitations as it is unable to capture mode shapes involving deflections perpendicular to the motion axis of the joints. A novel and straightforward identification method is developed to tune the joint stiffness and damping on the basis of directly measured FRFs at the tooltip. As a result, Stäubli and UMons TX200 models, comprising three elastic elements per joint, are updated and are both able

to replicate the first ten measured modes of the robot. They yield similar results in terms of prediction of natural frequencies, modal damping ratios and mode shapes when the robot is moved in arbitrary configurations. The mode prediction is satisfying for robot configurations similar to the one in which elastic parameters are determined. However, discrepancies arise in extreme postures e.g. outstretched arm, mainly reflected by inconsistent mode shapes. Next, the elastic properties of equivalent flexible beams representing the robot arm and forearm are tuned with respect to their finite element model. The actuator parameters are set from the information of the motor plates of the actual robot. Regarding the controller gains, they are defined on the basis of EMA results highlighting the effect of the controller on the joint modes. It is observed that controller does not modify significantly the natural frequencies of the robot under 100 Hz and only adds slight damping. Cutting coefficients are finally identified through milling experiments in aluminium and in steel.

Discrepancies between the simulated and measured FRFs at the robot tooltip can partly result from the simplistic representation of joint flexibility by using three orthogonal springs and dampers. A finer model and understanding of joint mechanics seem to be a necessity in order to faithfully reproduce the modal behaviour of the robot in its whole workspace. Adopting non-linear stiffness characteristics might improve the prediction of modal parameters in arbitrary configurations. On the other hand, modal identification techniques relying on mode shape sensitivity are worth to be attempted to determine the joint elastic parameters. Finally, FRF measurements at the TCP also revealed that robot structure is highly non-linear as non symmetry was observed in the cross-FRFs. It would be interesting to exactly determine the sources of non-linearity resulting in this phenomenon.

Chapter 6
Simulator validation

By coupling the developed dynamic multibody and milling models, robotic milling operations are simulated on the basis of the parameters identified in the previous chapter. The milling process is simulated in the time domain by integrating the equations of motion with an implicit solver. The numerical integration algorithm is completely detailed and coupled simulations are first carried out by replicating results originating from literature examples in conventional machine tool. As actual milling tests are conducted with the Stäubli TX200 robot, simulation results are compared to measurements in terms of cutting force, vibration and metrological feature. Comparisons are provided for some of the milling test cases in aluminium and steel described at the end of the previous chapter. Influence of the consideration of the link flexibility and/or the controller on the simulation results is investigated as well.

6.1 Coupled multibody and milling simulation

In this section, the numerical integration algorithm, coupling the multibody and milling models, is described and validated on the basis of machine tool examples. Although applicable to the robot model which was identified through excitation trajectories, it is only applied to the Stäubli and UMons TX200 models. Indeed, the latter are able to sufficiently reproduce the first ten vibration modes of the robot. No milling operation is simulated with the robot model relying on base parameters.

The block diagram of the implicit numerical integration algorithm is illustrated in Figure 6.1. The inputs of the simulation are related to parameters describing the robot multibody and milling models. Regarding the considered robot multibody model presented in Chapter 3, the kinematics of the robot (DH parameters) must be known and values of inertial parameters (link and rotor masses m, m_{m} and link and rotor central tensors of inertia $\mathbf{\Phi}_{\mathrm{G}}$, $\mathbf{\Phi}_{\mathrm{G,m}}$), elastic properties of the joints and the links (joint stiffness $k_{z|x|y}$ and damping $d_{z|x|y}$ and flexible beam parameters, Young's modulus E and Poisson's ratio ν), motor coefficients (gear ratio matrix $\mathbf{k}$, armature resistance R_{a} and torque constant k_{t}) and controller gains ($\mathbf{K}_{\mathrm{P}}$ and $\mathbf{K}_{\mathrm{D}}$) were determined in Chapter 5. The dynamic milling model requires the identification of the cutting coefficients (K_{tc}, K_{rc}, K_{ac}, K_{te}, K_{re} and K_{ae}), the tool and the workpiece description. The cutting conditions must of course be provided and are composed of the axial depth of cut a_{p}, the radial depth of cut a_{e}, the feed per tooth $\mathrm{f_z}$ and the spindle speed Ω. The time domain simulation also demands the specification of the time step Δt and the total duration of the simulation t_{End}. The tool path along with the robot configuration are eventually supplied to virtually conduct the milling operation.

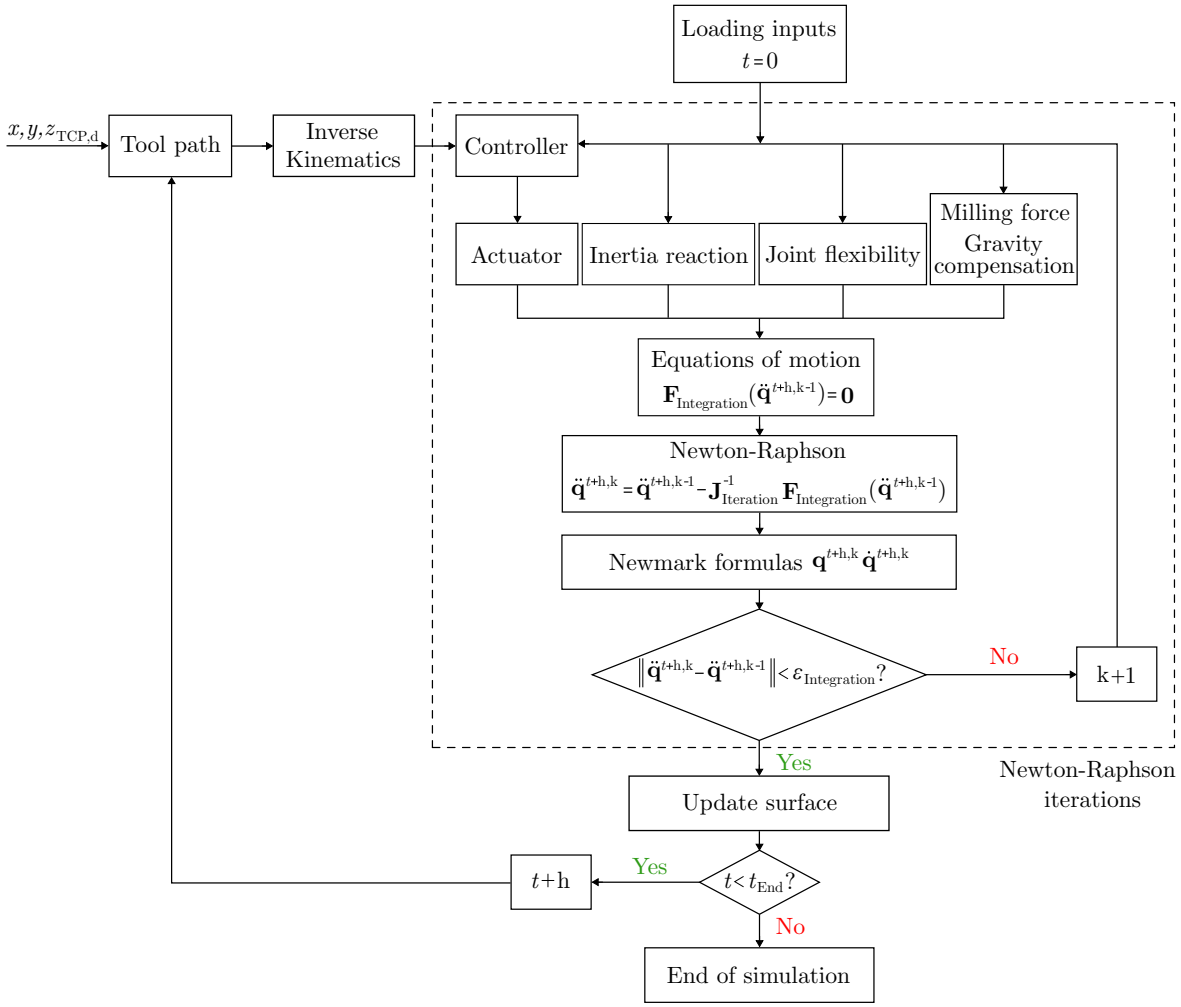


Figure 6.1: Implicit numerical integration algorithm

Once the input parameters are loaded, the numerical integration process can be started in t=0 s (t_0). From the desired TCP trajectory, a first Cartesian point $x, y, z_{\mathrm{TCP,d}}$ must be reached by the tool tip. The inverse kinematics handles the conversion into desired angular set points ($\boldsymbol{\Theta}$) for the robot controller. The latter generates the appropriate control inputs ($\boldsymbol{u}_{\mathrm{m}}$) (Eq. 3.93) to the motors which deliver the motor torques $\boldsymbol{\tau}_{\mathrm{m}}$ (Eq. 3.105). Motor torques are the inputs of the robot equations of motion in which they are considered as applied moments ($\boldsymbol{\mathcal{M}}_{\mathrm{Rotor_i}}$). Equations of motion are built in their residual form ($\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$=$\mathbf{0}$) at each time step using the d'Alembert's principle. Both expressions of the d'Alembert's principle for the rigid bodies (Eq. 3.31) and for the flexible bodies (Eq. 3.65) are combined. They include the reactions of inertia ($\boldsymbol{\mathcal{R}}(-m\mathrm{a})$ and $\boldsymbol{\mathcal{M}}(-m\mathrm{a})$) of the bodies and the gravity effects ($\boldsymbol{\mathcal{R}}(\mathbf{g})$ and $\boldsymbol{\mathcal{M}}(\mathbf{g})$). The joint tri-axial flexibility model develops deflection force and torque computed with Eqs. 3.52 and 3.53, and are treated as applied force and moment in the expression of the d'Alembert's principle. Finally, gravity compensation torque (Eq. 3.92) and milling forces (Eq. 4.42) are also added as applied forces to the equations of motion. Partial contributions $\mathbf{d}$ and $\mathbf{w}$, involved in the computation of the residual form of the equations of motion, results from the forward kinematics using the integrated degrees of freedom $\mathbf{q}^{t+\mathrm{h}}$. In view of Figure 6.2,

the integration time step, or simply integration step, is h. Time step Δt is in fact divided into multiple intervals. Equations of motion are integrated at each integration step thus deriving the integrated degrees of freedom $\mathbf{q}^{t+\mathrm{h}}$. Smaller integration steps usually conduct to more accurate results at the cost of larger simulation duration (integration step can even be non constant to control the integration error and to manage convergence issues). Time step Δt can be seen as the instant when the simulation results are saved on disk. For the sake of simplicity, choice h=Δt is retained for the robotic milling simulations. Assigned time step value is discussed in Section 6.3 related to the machine tool examples.

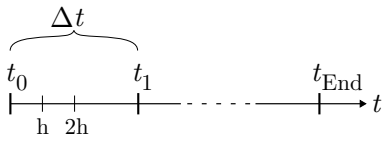


Figure 6.2: Time step Δt and integration step h

Numerical integration is a step by step procedure which consists in determining the configuration at time t+h ($\mathbf{q}^{t+\mathrm{h}}$, $\dot{\mathbf{q}}^{t+\mathrm{h}}$, $\ddot{\mathbf{q}}^{t+\mathrm{h}}$) from a given number of configurations at and before time t ($\mathbf{q}^{\leq t}$, $\dot{\mathbf{q}}^{\leq t}$, $\ddot{\mathbf{q}}^{\leq t}$). If the multibody system gathers n degrees of freedom, there are then $(3 \times n)$ unknowns which are enclosed in vectors $\mathbf{q}^{t+\mathrm{h}}$, $\dot{\mathbf{q}}^{t+\mathrm{h}}$ and $\ddot{\mathbf{q}}^{t+\mathrm{h}}$. While $\ddot{\mathbf{q}}^{t+\mathrm{h}}$ is usually computed during the numerical integration, vectors $\mathbf{q}^{t+\mathrm{h}}$ and $\dot{\mathbf{q}}^{t+\mathrm{h}}$ are derived from the so-called *integration formulas* which are denoted by $\check{\Lambda}$ and Λ such that

$$\begin{aligned} \dot{\mathbf{q}}^{t+\mathrm{h}} &= \check{\Lambda}(\dot{\mathbf{q}}^{\leq t}, \ddot{\mathbf{q}}^{\leq t}, \ddot{\mathbf{q}}^{t+\mathrm{h}}), \\ \mathbf{q}^{t+\mathrm{h}} &= \Lambda(\mathbf{q}^{\leq t}, \dot{\mathbf{q}}^{\leq t}, \ddot{\mathbf{q}}^{\leq t}, \ddot{\mathbf{q}}^{t+\mathrm{h}}). \end{aligned} \tag{6.1}$$

Implicit solver refers to the fact that integration formulas involve the accelerations at time t+h. In that case, equations of motion ($\mathbf{f}(\mathbf{q}^{t+\mathrm{h}}, \dot{\mathbf{q}}^{t+\mathrm{h}}, \ddot{\mathbf{q}}^{t+\mathrm{h}}, t+\mathrm{h})=\mathbf{0}$) can still be integrated if the positions and velocities are replaced by their integration formulas (Eq. 6.1) as follows

$$\begin{aligned} \mathbf{f}(\mathbf{q}^{t+\mathrm{h}}, \dot{\mathbf{q}}^{t+\mathrm{h}}, \ddot{\mathbf{q}}^{t+\mathrm{h}}, t+\mathrm{h}) &= \mathbf{f}(\Lambda(\mathbf{q}^{\leq t}, \dot{\mathbf{q}}^{\leq t}, \ddot{\mathbf{q}}^{\leq t}, \ddot{\mathbf{q}}^{t+\mathrm{h}}), \check{\Lambda}(\dot{\mathbf{q}}^{\leq t}, \ddot{\mathbf{q}}^{\leq t}, \ddot{\mathbf{q}}^{t+\mathrm{h}}), \ddot{\mathbf{q}}^{t+\mathrm{h}}, t+\mathrm{h}) \\ &= \mathbf{F}_{\text{Integration}}(\ddot{\mathbf{q}}^{t+\mathrm{h}}) = \mathbf{0}. \end{aligned} \tag{6.2}$$

Non-linear equations $\mathbf{F}_{\text{Integration}}$ are solved at each integration step h in order to determine the only remaining unknown $\ddot{\mathbf{q}}^{t+\mathrm{h}}$. Computing $\ddot{\mathbf{q}}^{t+\mathrm{h}}$ is usually managed by means of the Newton-Raphson's iterative procedure (inner loop in Figure 6.1) which reads

$$\ddot{\mathbf{q}}^{\,t+\mathrm{h,k}} = \ddot{\mathbf{q}}^{\,t+\mathrm{h,k\text{-}1}} - \mathbf{J}^{-1}_{\text{Iteration}} \mathbf{F}_{\text{Integration}}(\ddot{\mathbf{q}}^{\,t+\mathrm{h,k\text{-}1}}), \tag{6.3}$$

with k the number of iterations and $\mathbf{J}_{\text{Iteration}}$ the so-called iteration matrix. The Newmark's integration formulas are often used to solve second order differential equations such as the equations of motion. They read

$$\begin{aligned} \dot{\mathbf{q}}^{t+\mathrm{h}} &= \dot{\mathbf{q}}^{t} + (1-\gamma_{\text{Newmark}})\mathrm{h}\ddot{\mathbf{q}}^{\,t} + \gamma_{\text{Newmark}}\mathrm{h}\ddot{\mathbf{q}}^{\,t+\mathrm{h}}, \\ \mathbf{q}^{t+\mathrm{h}} &= \mathbf{q}^{t} + \mathrm{h}\dot{\mathbf{q}}^{t} + (0.5-\beta_{\text{Newmark}})\mathrm{h}^2\ddot{\mathbf{q}}^{\,t} + \beta_{\text{Newmark}}\mathrm{h}^2\ddot{\mathbf{q}}^{\,t+\mathrm{h}}, \end{aligned} \tag{6.4}$$

where β_{Newmark} and γ_{Newmark} are the Newmark parameters. The Newmark-$\frac{1}{4}$ scheme is used leading to β_{Newmark}=0.25 and γ_{Newmark}=0.5. It ensures unconditional stability of the numerical scheme. In the case of the Newmark scheme, the iteration matrix is given by

$$\mathbf{J}_{\text{Iteration}} = \mathbf{M}^0 + \mathbf{C}^0\gamma_{\text{Newmark}}\text{h} + \mathbf{K}^0\beta_{\text{Newmark}}\text{h}^2, \tag{6.5}$$

where $\mathbf{M}^0$, $\mathbf{C}^0$, $\mathbf{K}^0$ are the linearised system matrices introduced in Eq 5.20.

In practice, the iterations are performed until the correction on accelerations satisfies a given tolerance $\varepsilon_{\text{Integration}}$ such as

$$\|\ddot{\mathbf{q}}^{t+\text{h,k}} - \ddot{\mathbf{q}}^{t+\text{h,k-1}}\| \leq \varepsilon_{\text{Integration}}. \tag{6.6}$$

The convergence of the inner loop is usually attained in about 3 or 4 iterations. The integration tolerance is currently set at $\varepsilon_{\text{Integration}}$=1e^{-7}. Note that it is not necessary to recompute iteration matrix $\mathbf{J}_{\text{Iteration}}$ at each Newton-Raphson iteration. As its inversion may be time consuming, it is kept constant as long as possible and is usually updated every four iterations.

It is worth noting that during the iteration process of Newton-Raphson, cutting forces are kept constant. Cutting forces are not iterated because it would first burden the computations and the simulation duration and, most importantly, because they result from a discrete procedure when exploring for an intersection between the tool and the workpiece. The computation of the cutting forces is carried out by computing the current TCP position from the integrated degrees of freedom $\mathbf{q}^{t+\text{h}}$ using the forward kinematics. From the current TCP position, and the knowledge of the previous ones, the chip thickness can be computed thus leading to the cutting forces.

Once the convergence is reached and vectors $\mathbf{q}^{t+\text{h}}$, $\dot{\mathbf{q}}^{t+\text{h}}$ and $\ddot{\mathbf{q}}^{t+\text{h}}$ are determined and the surface of the machined workpiece is updated i.e. the quantity of material swept by the teeth is removed from the virtual workpiece. During the next time step, the cutting tool will therefore intersect with the updated surface ensuring a coherent computation of the cutting forces all along the robotic milling process. At instant t+h, the next set point of the tool path is provided to the controller and another integration step is performed. Finally, simulation ends when instant t_{End}, set by the user, is attained.

From the point of view of implementation, the codes used to complete coupled multibody and milling simulations are written in `C++` ensuring fast computations. The robot multibody model is developed using the so-called framework `EasyDyn` [172] which is an in-house multibody library. From the kinematics of the mechanical system and applied forces provided by the user, the equations of motion are automatically built using the d'Alembert's principle (Chapter 3) and solved in `EasyDyn`. Even if there exist other free tools permitting multibody simulations (`MBDyn` [173], `Hotint` [174], ...), the `EasyDyn`

framework is retained for its scalability and its proven efficiency. It is indeed essential that all functionalities of the multibody library are understood to couple it with the milling routine implementing the milling dynamic model presented in Section 4.5. A public release of the milling routines can also be found under the framework name `Dystamill` [175]. Both `EasyDyn` and `Dystamill` are frameworks for which their open-source codes are freely available on the Internet.

6.2 Tool trajectory

In view of Figure 6.1 depicting the numerical integration algorithm of the coupled multibody and milling simulations, the only block that is not yet described is the "tool path" block. More precisely, as term tool path refers to a set of coordinate points that the TCP must follow, it is more appropriate to qualify it as a tool trajectory which depends on time. In order to reproduce the tool trajectory of the milling tests presented in Subsection 5.7.1, the straight line motion is commanded to the TCP of the robot multibody model. The commanded motion follows a line segment from point A to point B, at the same elevation with respect to the base frame (z_{Base}), as illustrated in Figure 6.3 for the longitudinal pass in down milling. Straight line motion is either parallel to x_{Base} (transversal pass) or y_{Base} (longitudinal pass). Of course, all the milling situations have been reproduced as they were conducted in the actual milling tests i.e. as depicted in Figure 5.58, the robot TCP moves from point A to point B while keeping the clockwise direction for the tool rotation. It is just a matter of changing the starting robot configuration. Besides the straight line motion at the same elevation, note that the robotic machining simulator also allows controlling both the orientation and position of the TCP by relying on quaternion interpolation for the rotation part [176].

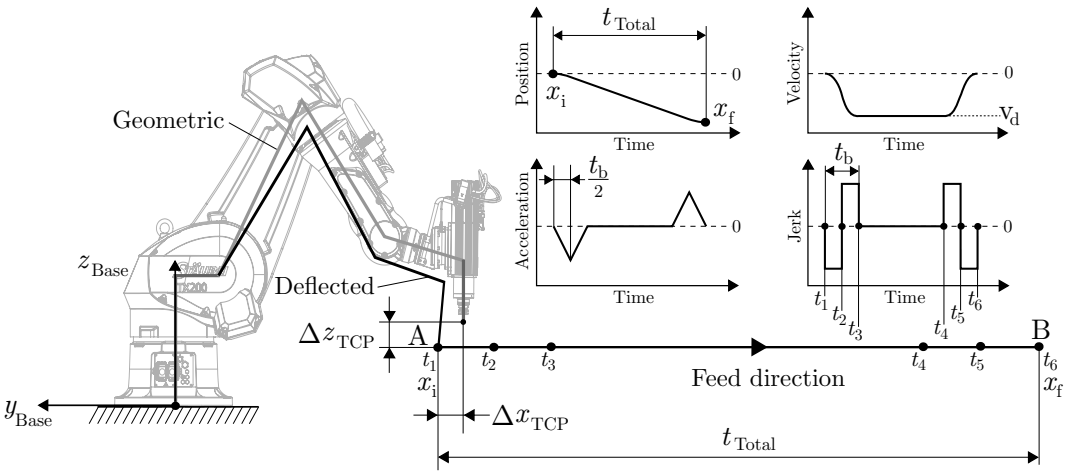


Figure 6.3: Constant jerk trajectory for milling operations

6.2.1 Jerk trajectory

Figure 6.3 illustrates the starting posture of the Stäubli TX200 robot before initiating the straight line motion. The geometric posture of the robot i.e. if the joints and links were infinitely rigid, is shown in light grey whereas the deflected configuration is drawn with a plain black line. Under the gravity effects, the position of the TCP is slightly altered due to the cumulative effects of all flexibilities. Actual deflections were not measured since it necessitates the use of a laser tracker or similar costly equipment. From the identified joint stiffness and the robot multibody model, an order of magnitude of deflections can be estimated such as $\Delta x_{\mathrm{TCP}} \approx 1$ mm and $\Delta z_{\mathrm{TCP}} \approx 2$ mm. Deflections along y_{TCP}, which are about some dozens of μm, are discussed when dealing with simulations of milling operations. Milling operations are usually conducted at constant feed rate i.e. the velocity of the TCP must be kept constant. In order to ensure smooth transitions between the null and the desired velocities, it is selected to maintain the jerk (derivative of the acceleration) at a constant value (different from zero) during the acceleration (interval $[t_1\text{-}t_3]$) and deceleration phases (interval $[t_4\text{-}t_6]$). The typical and resulting profiles of position, velocity and acceleration are also presented in Figure 6.3. The velocity profile exhibits a second-order polynomial curve ensuring a smooth transition to the desired feed rate (desired velocity $\mathrm{v_d}$). The described motion is later called the *jerk trajectory*.

The jerk trajectory is divided in five phases between the initial position x_{i} and the final position x_{f} of the TCP during the total duration of the virtual milling operation t_{Total}. The transition between null and constant velocities, and vice versa, lasts duration t_{b} (subscript b designates a blending motion). First phase lasts $\frac{t_{\mathrm{b}}}{2}$ during time interval $[t_1\text{-}t_2]$ and is an acceleration phase. Second phase also lasts $\frac{t_{\mathrm{b}}}{2}$ during time interval $[t_2\text{-}t_3]$ and is an acceleration phase for which jerk changes sign. Third phase during time interval $[t_3\text{-}t_4]$ is concerned with the desired feed rate. The last two phases represent deceleration motions for which jerk changes sign to switch from interval $[t_4\text{-}t_5]$ to $[t_5\text{-}t_6]$, each lasting $\frac{t_{\mathrm{b}}}{2}$. A symmetric jerk trajectory is chosen for the sake of simplicity. Hence, the jerk trajectory is entirely defined by four quantities: $\mathrm{v_d}$, x_{i}, x_{f} and t_{Total}. Time duration of the blending motion is computed such as

$$|t_{\mathrm{b}}| = \left| \frac{(x_{\mathrm{i}} - x_{\mathrm{f}} + \mathrm{v_d} t_{\mathrm{Total}})}{\mathrm{v_d}} \right|. \tag{6.7}$$

Prior the determination of duration t_{b}, desired velocity must be ensured not too large ($|\mathrm{v_d}| > \frac{2|x_{\mathrm{f}} - x_{\mathrm{i}}|}{t_{\mathrm{Total}}}$) nor not too small ($|\mathrm{v_d}| < \frac{|x_{\mathrm{f}} - x_{\mathrm{i}}|}{t_{\mathrm{Total}}}$). The jerk trajectory is described by the following general equations

$$
\begin{aligned}
j(t) =& j_{0,\text{Phase}_\text{p}} \\
a(t) =& a_{0,\text{Phase}_\text{p}} + j_{0,\text{Phase}_\text{p}} t_{\text{Phase}}(t) \\
v(t) =& v_{0,\text{Phase}_\text{p}} + a_{0,\text{Phase}_\text{p}} t_{\text{Phase}}(t) + \frac{1}{2} j_{0,\text{Phase}_\text{p}} t_{\text{Phase}}(t)^2 \\
x(t) =& x_{0,\text{Phase}_\text{p}} + v_{0,\text{Phase}_\text{p}} t_{\text{Phase}}(t) + \frac{1}{2} a_{0,\text{Phase}_\text{p}} t_{\text{Phase}}(t)^2 + \frac{1}{6} j_{0,\text{Phase}_\text{p}} t_{\text{Phase}}(t)^3,
\end{aligned}
\tag{6.8}
$$

with $j(t)$ the jerk, $a(t)$ the acceleration, $v(t)$ the velocity and $x(t)$ the position of the TCP at current time t. The quantities either represent a motion along x_{Base} (transversal pass) or y_{Base} (longitudinal pass). For each of the five phases, jerk ($j_{0,\text{Phase}_\text{p}}$), acceleration ($a_{0,\text{Phase}_\text{p}}$), velocity ($v_{0,\text{Phase}_\text{p}}$) and position ($x_{0,\text{Phase}_\text{p}}$) constants must be defined. Time $t_{\text{Phase}}(t)$ is the current time of phase p. Subscript p refers to the index of the phase varying from 1 to 5. The constants are defined phase by phase in Appendix L.

All the constants defined from Eq. L.3 to Eq. L.7 are successively used in the general equations of the jerk trajectory presented in Eq. 6.8. It leads to the determination of the jerk $j(t)$, acceleration $a(t)$, velocity $v(t)$ and position $x(t)$ over time. Computed position $x(t)$ is substituted in place of $p_{\text{TCP},x}$ (longitudinal pass) or $p_{\text{TCP},y}$ (transversal pass) in the Cartesian pose of the TCP defined in Eq. 3.2. Using the inverse kinematics, the TCP pose is converted in joint angles which constitute the set points ($\boldsymbol{\Theta}$) of the controller. Since the velocity $v(t)$ and the acceleration $a(t)$ are also derived during the computation of the jerk trajectory, inverse relationships of the differential kinematics presented in Subsection 3.2.4 lead to the joint velocities and accelerations ($\dot{\boldsymbol{\Theta}}$, $\ddot{\boldsymbol{\Theta}}$) feeding the feedforward action of the chosen controller.

The jerk trajectory is implemented in the robotic machining simulator. Note that the trajectory is first tested without applying any milling force at the TCP. The case of the longitudinal pass (along y_{Base}) when the TCP moves away from the robot base is examined. A similar starting configuration as the one illustrated in Figure 6.3 is selected for the robot posture and is expressed in terms of joint angles in Table 6.1. TCP pose defining x_i can therefore be computed.

q_1 [°]	q_2 [°]	q_3 [°]	q_4 [°]	q_5 [°]	q_6 [°]
-91.84	37.97	121.07	0.04	-54.04	-0.01

Table 6.1: Initial joint positions to test the motion trajectories in the encoder reference

Input parameters to test the jerk trajectory are chosen to match realistic motion behaviour when a milling operation is achieved. Therefore, the desired TCP velocity is set at v_d=-0.15 m/s (along the negative y_{Base} axis). The TCP must travel a distance of 0.5 m between x_i (y_{Base}=-1.55 m) and x_f (y_{Base}=-2.05 m) during a time lapse of t_{Total}=5 s. Motion begins in t_1=2.5 s.

The jerk trajectory is first tested for the reference version of the Stäubli TX200 model (using robot manufacturer's data) which is composed of the joint, ground flexibilities and the DVA. Such a model is henceforth designated as the robot model with joint flexibilities. A perfect regularisation of the controller and rigid links are assumed. The reference version of the Stäubli TX200 model therefore comprises 22 DOFs. Figure 6.4 shows the time evolutions of the TCP position, velocity and acceleration along y_{Base} of the resulting motion. The three upper graphs illustrate the superimposition of the desired quantities (position, velocity and acceleration set points) with the simulated ones and the three lower graphs depict the time evolution of the corresponding tracking error along the feed direction (along y_{Base}). Starting value of the position error reflects the static deflection (Δx_{TCP} in Figure 6.3) due to the joint compliance. Position tracking error can therefore be deduced and worth about 200 μm. As expected, velocity profile exhibits a smooth transition before reaching desired velocity $\mathrm{v_d}$.

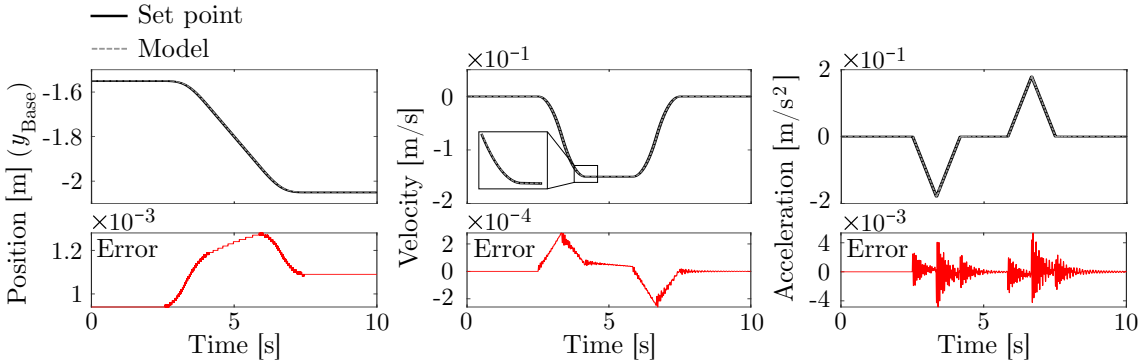


Figure 6.4: Trajectory with constant jerk profile when using the Stäubli TX200 model comprising the joint flexibilities

If the inverse dynamics controller is included in the robot dynamic model, it results in the time evolutions of the TCP position, velocity and acceleration presented in Figure 6.5. The velocity profile is still smooth even if its error is more noisy due to the high gain values set for the controller.

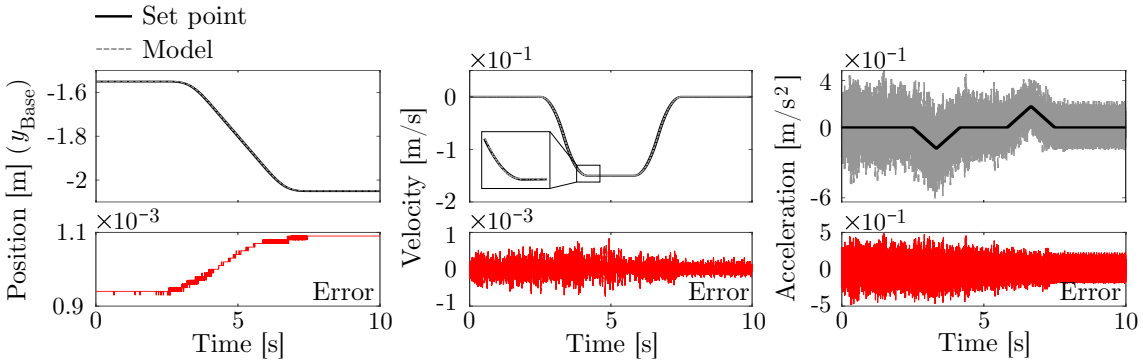


Figure 6.5: Constant jerk trajectory when using the Stäubli TX200 model comprising the joint flexibilities and the controller

If the controller and the link flexibilities are combined to the robot multibody model along with the joint flexibilities, the time evolutions of the TCP position, velocity and

acceleration become the ones displayed in Figure 6.6. Velocity profile is again smooth and desired velocity v_d is reached with a sufficiently small error.

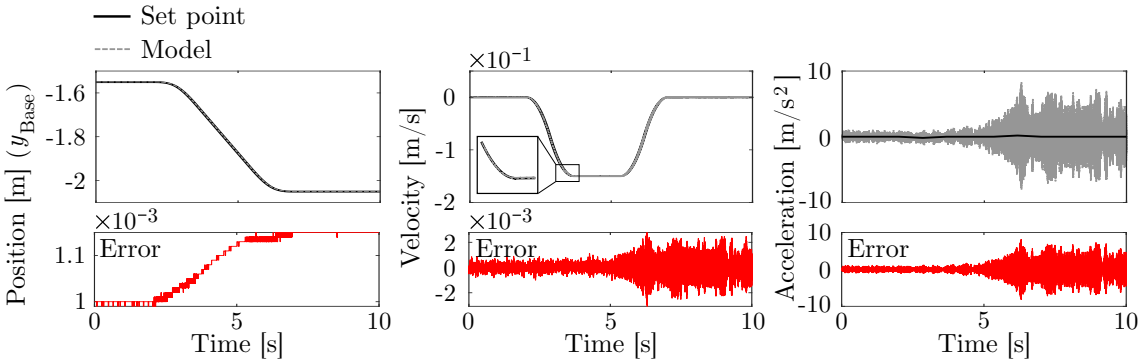


Figure 6.6: Constant jerk trajectory when using the Stäubli TX200 model comprising the joint and link flexibilities and the controller

Since desired velocity v_d is reached smoothly in all versions of the Stäubli multibody model (with joint deflections with or without controller and/or flexible links), the jerk trajectory is therefore used when dealing with robotic milling simulations.

6.2.2 Trajectory with trapezoidal velocity profile

Performances of the jerk trajectory are compared to the ones obtained with a classic trajectory with trapezoidal velocity profile. Equations of such a profile can be obtained in a similar way as for the jerk trajectory by assuming a null jerk constant throughout the motion. The relationships are also defined in [3]. The acceleration is set to non null constant values resulting in a velocity profile with a trapezoidal shape. As a baseline for comparison, the trajectory with trapezoidal velocity profile is implemented to command the TCP velocity of the Stäubli TX200 model only comprising the joint flexibilities. Robot configuration (longitudinal pass) and input parameters for the trajectory are the same as for the jerk trajectory (v_d=-0.15 m/s, x_i=-1.55 m, x_f=-2.05 m and t_{Total}=5 s).

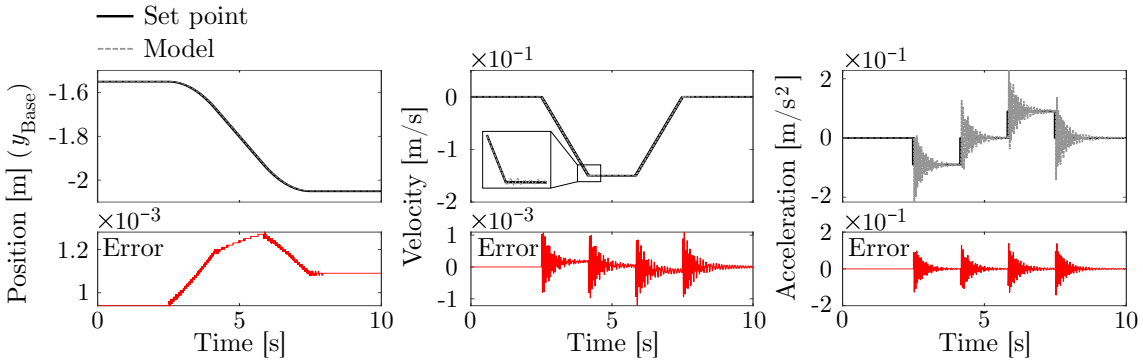


Figure 6.7: Trajectory with trapezoidal velocity profile when using the Stäubli TX200 model comprising the joint flexibilities

Figure 6.7 illustrates the resulting time evolutions of the TCP position, velocity and acceleration. This time, the transitions between the null velocity and desired velocity v_d,

and vice versa, are sharp. It thus produces a higher level of vibration in the velocity error compared with the same situation when using the jerk trajectory (Figure 6.4). Larger velocity errors are expected if the controller and the link flexibilities are added. It thus strengthens the retained choice of the jerk trajectory for the robotic milling simulations.

6.3 Validation on machine tool examples

Before using the simulator coupling multibody and milling codes as detailed in Section 6.1, it was first validated on milling examples carried out in conventional machine tool. The validation is accomplished on the basis of two literature examples. In conventional milling, the dynamics of machine tool can be reduced to a rotating tool whose vibrations occur in a plane. As a reminder, the direction along the tool axis is much stiffer than the ones in the cutting plane. Hence, the first treated example is dealing with a milling machine only owning a flexibility along the feed direction while the second one is concerned with vibrations along and perpendicular to the feed direction.

6.3.1 Single-axis vibrating milling machine

The studied system comprises one single degree of freedom which is the vibration of the end mill along the feed direction. The machine tool is modelled as a single mass-spring-damper system for which vibrations only occur along the feed direction which is chosen as the X-axis (Figure 6.8). In fact, the model is a representation of the first vibration mode of the machine tool such as m_{Tool} is the modal mass, k_x is the modal stiffness and d_x is the modal damping. Degree of freedom q_x designates the deflection of the spring. The moving mass can be treated as a tool rotating at constant angular velocity Ω into the workpiece. The mass-spring-damper system is connected to a rigid base moving at the desired feed rate v_{Feed} along the X-axis. Cutting forces F_x are therefore applied in the opposite direction.

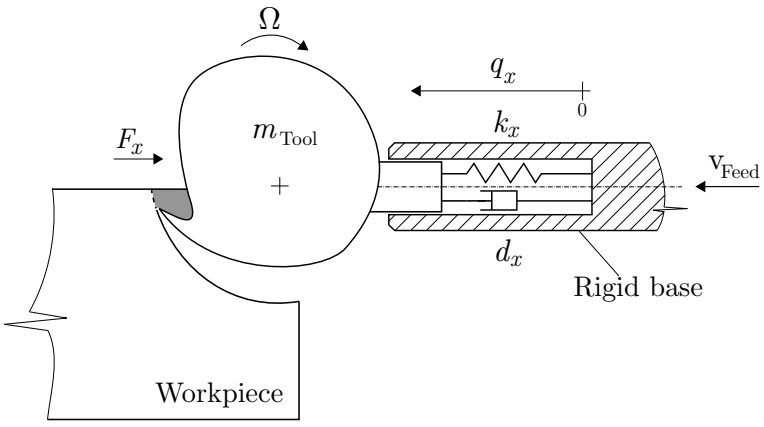


Figure 6.8: Single-axis vibrating mill model

The validation example is inspired from the milling case treated by Insperger et al. in [152]. It is concerned with the material removal of a workpiece made of 7075-T6 aluminium alloy. Reported cutting force coefficients are such as K_{tc}= 550 MPa and K_{rc}= 200 MPa (K_{ac}=

0 MPa). The cutting tool has a diameter of 10 mm and only possesses one tooth without any helix angle (N_z=1). It removes material in the up milling direction in half immersion (a_e=5 mm) at a feed per tooth f_z= 0.05 mm/tooth. Feed rate v_{Feed} is computed such as $v_{Feed}=N_z f_z \frac{\Omega}{60}$ with spindle speed Ω expressed in rev/min. The tested axial depth of cut is a_p= 2 mm. Finally, the modal characteristics of the mass-spring-damper system are the followings: modal mass m_{Tool}=2.573 kg, damping ratio ζ_n= 0.32 % and a natural frequency f_n of 146.4 Hz. Since simulations are carried out in the time domain, it is needed to specify the number of tool turns n_Ω, which are commanded to the rotating tool, as well as the time step Δt. The number of tool turns must be sufficiently large to detect instability during the material removal. In this case, the number of tool turns is chosen as n_Ω= 300. The selection of the time step can be a tricky problem, as discussed in [177], since it influences the accuracy of the material removal on the virtual workpiece. In other words, the machined surface is updated at each time step Δt. The workpiece updating must be accurate enough to detect chatter such that a trade-off between precision and computing time is adopted. To ease the selection of the time step, the constraint is reported into a geometric constraint by specifying the number of steps $n_{\text{Steps per turn}}$ to simulate over one revolution when at least one tooth is in the material. Hence, time step Δt, which is the rate of the integration of the equations of motion, is computed as follows

$$\Delta t = \frac{60}{n_{\text{Steps per turn}}\Omega}, \tag{6.9}$$

with Ω the spindle speed expressed in rev/min. A minimum of $n_{\text{Steps per turn}}$=30 steps per turn is highly recommended to be able to correctly predict the stability of a milling operation. For the simulation example, the number of steps per turn is selected as $n_{\text{Steps per turn}}$=120.

Three specific milling cases are simulated in order to reveal three particular configurations of cutting force time evolution. In other words, only the spindle speed is varied and successively becomes 16,000, 19,000 and 22,000 rev/min. Figure 6.9 plots the three resulting behaviours for the magnitude of cutting force F_x. The three time evolutions of cutting force F_x can be described as follows:

1. At 16,000 rev/min, the dynamic system is driven into an unstable behaviour which is the Hopf instability. The dynamics of the system, and more particularly the tool tip vibrations, become chaotic. The magnitude of cutting force F_x increases sharply and the frequency content of such signal is dominated by the chatter frequency close to the machine tool natural frequency.

2. At 19,000 rev/min, the dynamic system is also exposed to an instability which is the flip instability. An analysis in the frequency domain exhibits resonance peaks at half of the tooth passing frequency. The magnitude of cutting force F_x stands high.

3. At 22,000 rev/min, the dynamic system reaches stable conditions. The magnitude of cutting force F_x remains constant.

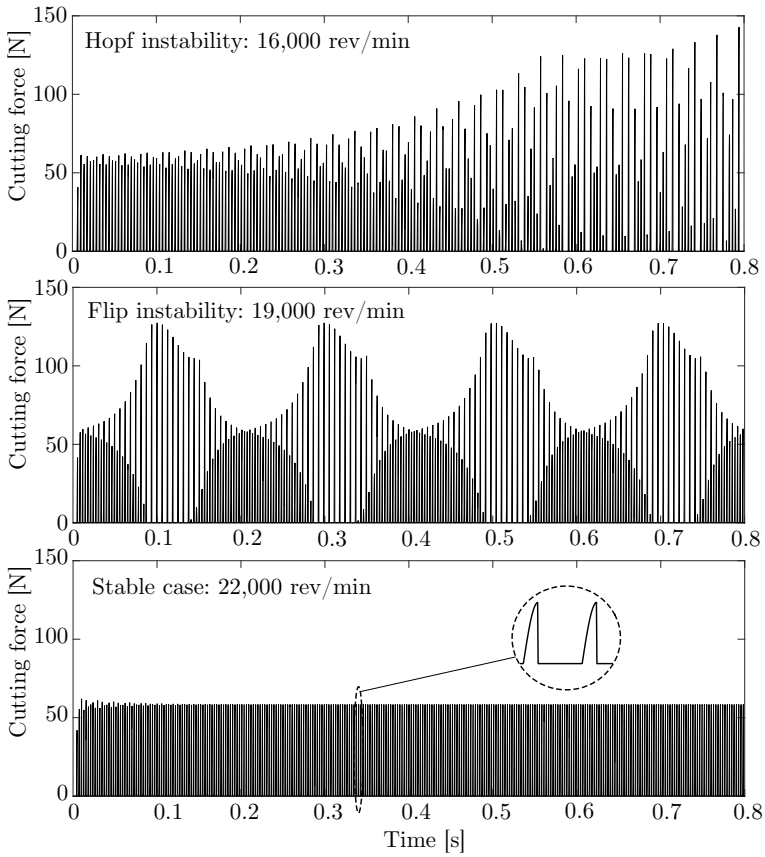


Figure 6.9: Simulated cutting forces by using the single-axis vibrating mill model for the stable case and the Hopf and flip instabilities

From the simulation results, it can be inferred that the coupled environment with multibody and milling codes is able to predict milling instabilities. As a complementary note, each simulation takes approximately 1 second with a 16-Gb RAM laptop equipped with an i7-8750H processor.

The presented results in the original article of Insperger et al. [152] are related to the simulation of stability lobe diagram. Hence, the simulator is used to replicate the results as a validation. Using time domain simulations of milling, the generation of stability lobe diagram can be achieved by assessing the stability of each pair of spindle speed and axial depth of cut. In other words, in the stability lobe diagram, the axes of spindle speed and of axial depth of cut are discretised with a chosen step and the stability of each pair is evaluated thanks to the stability criterion explained in Subsection 4.5.5. However, this process might be very long in terms of computing time. In order to accelerate the determination of the stability limit, stability lobe diagrams are instead generated via the algorithm illustrated in Figure 6.10 [160]. Input of the algorithm is the region of stability to study, bounded by the lower and upper limits for the spindle speed (Ω_{min} and Ω_{max}) and for the axial depth of cut ($a_{\text{p,min}}$ and $a_{\text{p,max}}$), and the discretisation step for the

spindle speed $\Delta\Omega$ and for the axial depth of cut Δa_p. The stability lobe diagram is swept vertically i.e. multiple evaluations of increasing axial depths of cut at a fixed spindle speed. At the beginning of the algorithm, the current axial depth of cut a_p and the current spindle speed Ω become $a_p = a_{p,min}$ and $\Omega = \Omega_{min}$. A first simulation is carried out using the specified cutting conditions and the stability of the milling operation is appraised. Depending on the outcome, the current axial depth of cut is compared with the maximum studied axial depth of cut $a_{p,max}$ (stable case) and with the minimum axial depth of cut $a_{p,min}$ (unstable case). While the current axial depth of cut a_p is within the limits of axial depth of cut $a_p > a_{p,min}$ and $a_p < a_{p,max}$, the domain is swept with a step of axial depth of cut Δa_p and stability is assessed at each incrementation. Four conditions allow switching to the next spindle speed to study. If the current axial depth of cut a_p is lower than $a_{p,max}$ and that the milling operation becomes unstable as the axial depth of cut increases or if the current axial depth of cut $a_p > a_{p,max}$, then current a_p and Ω are saved in memory. Analogously, if the current axial depth of cut a_p is greater than $a_{p,min}$ and that the milling operation becomes stable as the axial depth of cut decreases or if the current axial depth of cut $a_p < a_{p,min}$, then current a_p and Ω are also saved in memory. Current spindle speed is then increased by $\Delta\Omega$ if not greater than Ω_{max}, otherwise the algorithm ends.

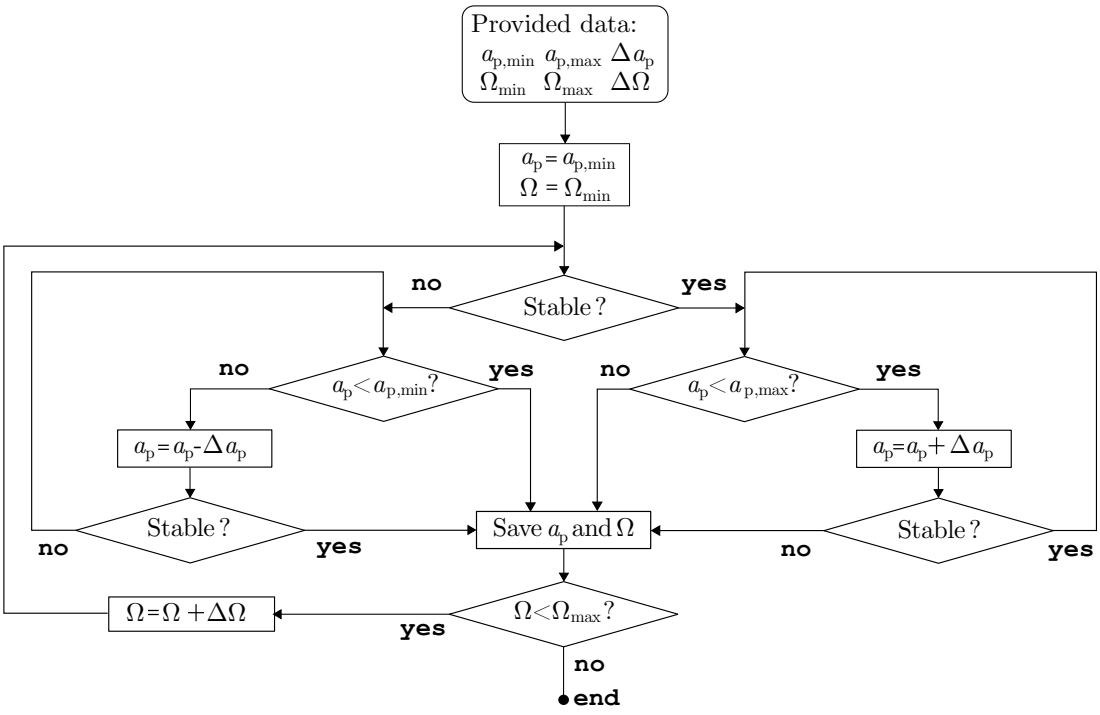


Figure 6.10: Generation of stability lobe diagram for time domain simulations

The presented algorithm generating stability lobe diagrams is applied to the example proposed by Insperger et al. in [152]. Investigated stability region is bounded as follows: $a_{p,min}$= 0.1 mm, $a_{p,max}$= 5.0 mm, Ω_{min}= 2,500 rev/min and Ω_{max}= 25,000 rev/min. Discretisation steps for the axial depth of cut and the spindle speed are chosen as follows: Δa_p= 0.1 mm and $\Delta\Omega$= 50 rev/min. The resulting stability limit proves to be similar to the one reported by Insperger et al. in [152]. Figure 6.11 compares the original digi-

talised stability limit with the simulated one using the simulator coupling multibody and milling codes. The typical shapes of the Hopf and flip lobes are correctly replicated. In Figure 6.11 are also highlighted the particular cutting conditions leading to the cutting force evolutions presented in Figure 6.9. Consequently, the example clearly indicates that coherent results can be delivered using the proposed simulator. For the record, it takes about 13 minutes to generate the simulated stability lobe.

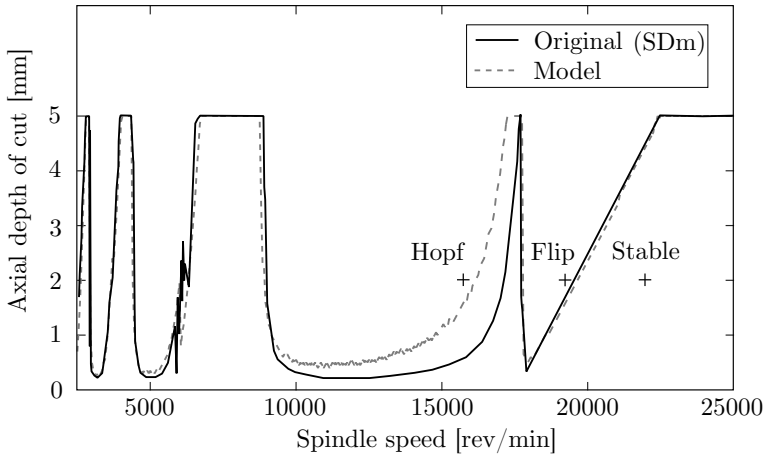


Figure 6.11: Comparison of the original and simulated stability lobe diagrams for the single-axis milling machine [152]

6.3.2 Bi-axis vibrating milling machine

The second validation example concerns a machine tool whose dynamics is modelled with two modes along the X-axis and two other modes along the Y-axis. It is proposed by Smith et al. in [178]. The corresponding milling system is illustrated in Figure 6.12 with springs and dampers in both directions. Degree of freedom $q_{x|y,\mathrm{i}}$ designates the deflection of the spring along the X or Y direction for mode i.

Again, each mode is modelled as a mass-spring-damper system. The modal characteristics of the four mass-spring-damper systems are provided in Table 6.2 (subscripts indicate the direction and the mode number). All the mass-spring-damper systems are connected to a rigid base moving at the desired feed rate $\mathrm{v_{Feed}}$ along the X-axis. Cutting forces F_x and F_y are applied along the X and Y axes, respectively.

Modal mass [kg]	$m_{x,1}$=84.7	$m_{x,2}$=9.3	$m_{y,1}$=239.8	$m_{y,2}$=4.5
Damping ratio [%]	$\zeta_{x,1}$=12	$\zeta_{x,2}$=4	$\zeta_{y,1}$=10	$\zeta_{y,2}$=10
Natural frequency [Hz]	$f_{x,1}$=260	$f_{x,2}$=389	$f_{y,1}$=150	$f_{y,2}$=348

Table 6.2: Modal characteristics of the milling system with four modes

The milling operation consists in a half immersion operation (a_{e}= 50 mm) into an aluminium alloy using a cylindrical end mill of diameter 100 mm, comprising eight teeth without any helix angle ($\mathrm{N_z}$=8). Reported cutting force coefficients are such as K_{tc}= 1500 MPa and K_{rc}= 450 MPa (K_{ac}= 0 MPa). Material is removed following the up

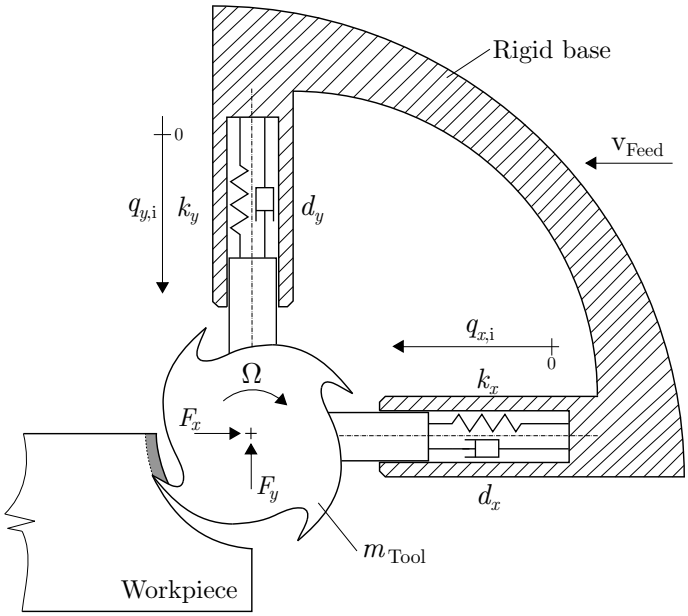


Figure 6.12: Bi-axis vibrating mill model

milling direction at a feed per tooth f_z= 0.1 mm/tooth. The investigated stability region is bounded by the following quantities: $a_{p,min}$= 2 mm, $a_{p,max}$= 12 mm, Ω_{min}= 1,000 rev/min and Ω_{max}= 8,000 rev/min. Discretisation steps for the axial depth of cut and the spindle speed are chosen as follows: Δa_p= 0.1 mm and $\Delta\Omega$= 100 rev/min. In Figure 6.13, the simulated stability lobe diagram is superimposed with the digitalised original stability limit computed by Altintas et al. in [45]. Although minor discrepancies are noticeable, the overall trend is again well captured by the simulator. The rising edge representing the Hopf bifurcation between 2,000 and 3,500 rev/min nearly coincides with the original curve. The stability lobe diagram is generated by using the algorithm proposed in Figure 6.10 and the stability of each studied pair of spindle and axial depth of cut is assessed with a number of tool turns n_Ω= 40 and a number of steps per turn $n_{\text{Steps per turn}}$= 1000. About 5 minutes are needed to complete all the calculations.

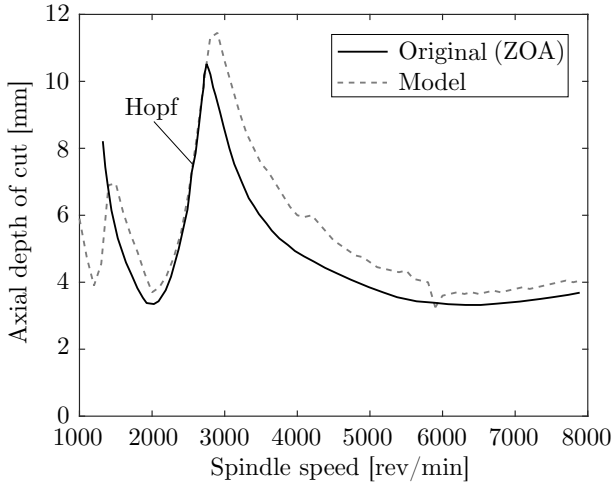


Figure 6.13: Comparison of the simulated and original stability lobe diagrams for the bi-axis milling machine [45]

To conclude with the milling examples from the literature, the simulator is used to reveal the evolution of the maximum value of the magnitude of the resultant cutting force F_{Rc} (Eq. 4.43) inside the investigated stability region. In order to display such a result, time domain simulations of milling are carried out for all the pairs of spindle speed and axial depth of cut to span the whole stability region. After each simulation, the maximum level of F_{Rc} is reported in the stability lobe diagram. Figure 6.14 shows the resulting evolution of F_{Rc} as the spindle speed and axial depth of cut are increased. The transition between lighter and darker colours coincides with the simulated stability limit. Using the same numbers of tool turns and steps per turn, it takes about seven hours to complete all the 7000 individual simulations. Similar stability lobe diagrams including the level of F_{Rc} will be shown in Chapter 7 dealing with the stability analysis in robotic milling.

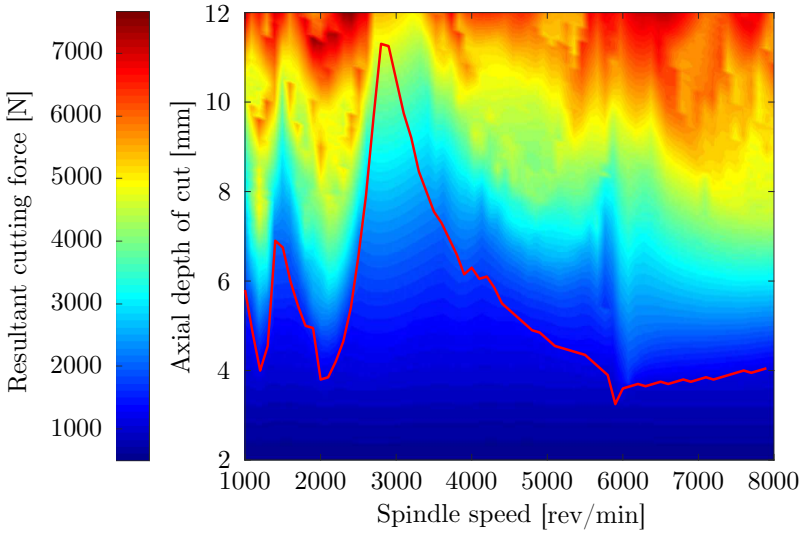


Figure 6.14: Simulated stability lobe diagram with the evolution of the maximum value of the magnitude of the resultant cutting force F_{Rc}

6.4 Simulation of robotic milling operations

The robotic milling simulator is now validated on an arbitrary robotic milling operation. The studied operation is a longitudinal pass achieved using the down milling direction as presented in Figure 5.58. The workpiece is the block and aluminium and steel milling are investigated. While the radial depth of cut is kept constant (a_{e}= 4 mm), the axial depth of cut is chosen such that the validation is carried out in stable conditions. Unstable milling operations are discussed in the next chapter dedicated to the stability analysis. All simulation results are generated with the Stäubli TX200 model based on the manufacturer's data and are compared, as far as possible, with the measured counterparts. The dynamic milling model is used and cutting forces are computed without the inclusion of the edge force coefficients (standard model). The influence of the consideration of the flexibilities of the joints, of the links and of the controller in the multibody model is also analysed. A complete analysis is provided for the aluminium part and a summary of the main results is outlined for the steel part.

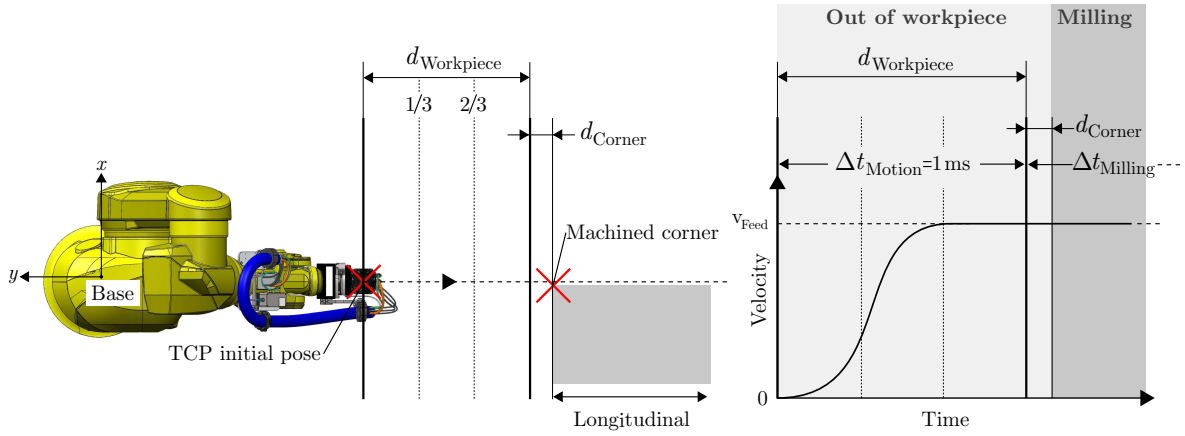


Figure 6.15: Progress of a typical robotic milling simulation (longitudinal pass in down milling)

The progress of a typical robotic milling operation using the simulator is depicted in Figure 6.15. As for example, the longitudinal pass in down milling is clarified regarding the positioning of the virtual workpiece with respect to the robot as well as the synchronisation of the trajectory. The positioning of the workpiece is defined with respect to the base frame and remains fixed for the considered milling situation i.e. for the longitudinal pass in down milling, the machined corner illustrated in Figure 6.15, which is the position of the workpiece corner that is firstly removed, is at coordinates x_{Corner}= 12.187, y_{Corner}= -1528.0 and z_{Corner}= 449.382 mm. Elevation of the workpiece top surface stays constant. In terms of joint angles, the position of the machined corner corresponds to q_1= -89.364°, q_2= 24.268°, q_3= 120.918°, q_4= 0.952°, q_5= -40.189° and q_6= -0.563° which leads to a robot configuration close to the one studied in the experimental modal analysis (milling pose). In Figure 6.15, the workpiece is slightly moved downwards along the negative x axis as the radial immersion is 40 % of the tool diameter to match the milling tests. As mentioned when presenting the jerk trajectory, a constant feed rate $\mathrm{v_{Feed}}$ must be reached before milling the workpiece. However, if the feed per tooth $\mathrm{f_z}$ is kept constant as it was the case for all the milling tests related to one material, the feed rate $\mathrm{v_{Feed}}$ changes with the spindle speed such as

$$\mathrm{v_{Feed}} = \mathrm{N_z f_z} \frac{\Omega}{60}, \tag{6.10}$$

with Ω the spindle speed expressed in rev/min. As it is desired to keep the same TCP initial pose (Figure 6.15) irrespectively from the selected spindle speed (e.g. to generate the stability lobe diagram), it means that the velocity profile must be adjusted each time the spindle speed is changed. Practically, the distance between the TCP initial pose and the workpiece corner is split in two. Distance d_{Corner} is defined as the sum of the tool diameter D_{Tool} and the distance travelled by the tool over one spindle rotation such as

$$d_{\text{Corner}} = \mathrm{v_{Feed}} \left(\frac{60}{\Omega} \right) + D_{\text{Tool}}. \tag{6.11}$$

Distance $d_{\text{Workpiece}}$ is then computed as the remaining segment in Figure 6.15. The velocity profile is modified for each spindle speed such that the desired feed rate v_{Feed} is reached at $\frac{2}{3}$ of distance $d_{\text{Workpiece}}$. In other words, instant t_3 in Figure 6.3 must be reached when the TCP is at $\frac{2}{3}$ of distance $d_{\text{Workpiece}}$ to ensure that feed rate v_{Feed} is stabilised well before milling the workpiece. The same strategy is used for the four situations of milling illustrated in Figure 5.58. Of course, the TCP initial pose and the machined corner of the workpiece are redefined according to the milling situation as presented in Table 6.3. The direction of motion is also modified accordingly. A radial depth of cut a_e= 4 mm is generated with an offset of the workpiece i.e. for the longitudinal pass in down milling, $x_{\text{Corner}} = x_{\text{TCP}}$ -1 mm knowing that the tool has a diameter D_{Tool}= 10 mm. After entering in the workpiece, the tool removes the material over the whole length of the workpiece ($L_{\text{Workpiece}}$= 100 mm in Figure 6.16a).

	TCP initial pose			Machined corner		
	x_{TCP}	y_{TCP}	z_{TCP}	x_{Corner}	y_{Corner}	z_{Corner}
Longitudinal down milling	13.187	-1427.458	449.382	12.187	-1528.0	449.382
Longitudinal up milling	13.187	-1757.458	449.382	12.187	-1628.0	449.382
Transversal down milling	-147.22	-1576.458	449.382	-47.22	-1577.458	449.382
Transversal up milling	152.78	-1576.458	449.382	52.78	-1577.458	449.382

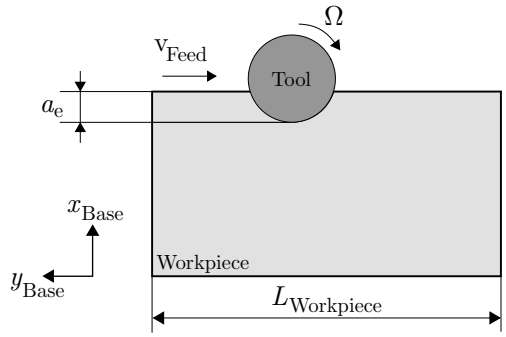
Table 6.3: Initial poses of TCP and machined corners for the four milling situations projected in the base frame (dimensions in mm)

Another important comment is issued on the chosen time step Δt for all the robotic milling simulations. Two different time steps are used per milling simulation. A fixed time step Δt_{Motion} is imposed to 1 ms during the rising of the velocity without milling and before that the TCP enters distance d_{Corner} as shown in Figure 6.15. When the TCP enters distance d_{Corner}, the time step is switched to $\Delta t_{\text{Milling}}$ established in Eq. 6.9. The milling time step is evidently much smaller than the motion time step in order to accurately model the virtual material removal ($\Delta t_{\text{Milling}} \ll \Delta t_{\text{Motion}}$). Managing two dedicated time steps allows a significant gain in computing time especially when repeating multiple milling simulations for the generation of stability lobe diagrams. For all the robotic milling simulations, a number of tool turns n_Ω= 500 is selected with a number of steps per turn $n_{\text{Steps per turn}}$= 50. The counting of tool turns starts when the time step is switched to $\Delta t_{\text{Milling}}$. Note that in the context of the stability analysis presented in the next chapter, the length of the workpiece is not limited to 100 mm but by the number of tool turns n_Ω= 500 in order to detect instability (should it occur) on longer length.

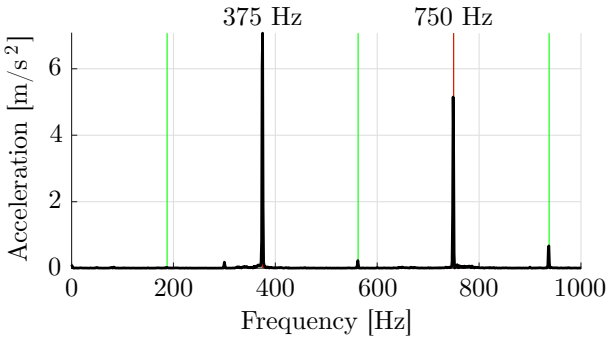
6.4.1 Aluminium milling with joint tri-axial flexibility

The longitudinal pass in down milling for the aluminium block is simulated using the robotic milling simulator. The milling operation to simulate is concerned with an axial depth of cut a_p= 2 mm at a feed per tooth f_z= 0.13 mm/tooth and a spindle speed Ω= 11,250 rev/min. Figure 6.16a depicts the milling operation while Figure 6.16b shows the

FFT applied to the measured vibrations perpendicular to the feed direction (along x_{Base} for the longitudinal passes). The latter mainly exhibits the tooth passing frequency (red vertical lines) at 375 Hz (2 teeth) and its harmonics. The green vertical lines indicate the spindle frequency and its harmonics.



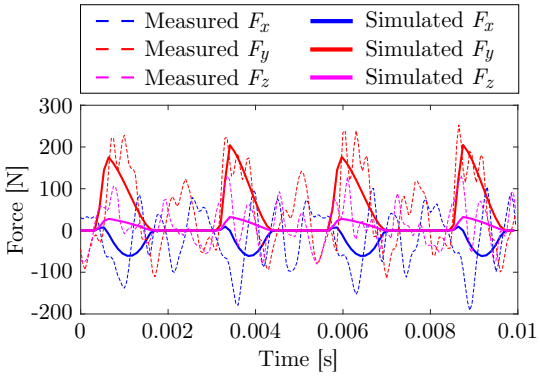
(a) Longitudinal pass in down milling



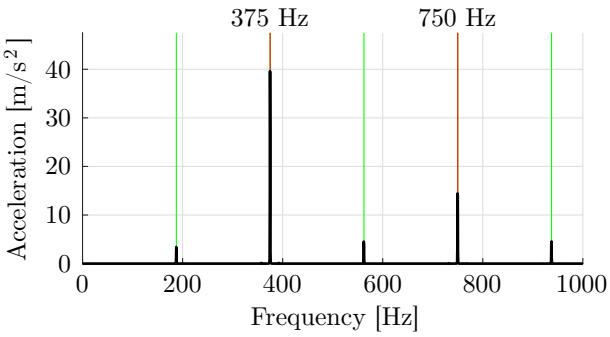
(b) FFT of the measured vibrations perpendicular to the feed direction

Figure 6.16: Validation example in aluminium for the Stäubli TX200 robot (a_{p}= 2 mm, a_{e}= 4 mm with a feed per tooth f_z = 0.13 mm/tooth at 11,250 rev/min)

As the influence of the consideration of the flexibility sources is studied, the reference model is chosen as the Stäubli TX200 model comprising the joint tri-axial flexibility presented in Subsection 3.3.4. As a reminder, the included flexibilities of such model were the ground flexibilities (q_{g_x}, q_{g_y} and q_{g_z}), the joint flexibilities ($q_{x,1\text{-}6}$, $q_{y,1\text{-}6}$ and $q_{z,1\text{-}6}$) and the DVA (q_{DVA}) for a total of 22 degrees of freedom. Since no controller is included, the degrees of freedom related to the motors are dropped. In other words, a perfect regularisation is assumed.



(a) Cutting force comparison



(b) FFT of the simulated vibrations perpendicular to the feed direction

Figure 6.17: Simulated signals using the joint tri-axial flexibility model for the longitudinal pass in down milling (a_{p}= 2 mm, a_{e}= 4 mm with a feed per tooth f_z = 0.13 mm/tooth at 11,250 rev/min)

Once the robotic milling simulation completed, the resulting cutting forces and vi-

brations are analysed and compared to the measured signals. Figure 6.17a shows the superimposition of the unfiltered measured and simulated cutting forces in which F_x is the force along the feed direction (along y_{Base} for the longitudinal pass) and F_y is the perpendicular force in the cutting plane (along x_{Base} for the longitudinal pass). The simulated cutting forces are computed at the robot TCP. A high correlation is noticed between the measured and simulated forces F_y. It is observed that the unfiltered signals are busy between the peaks representing the passing of the teeth. Cutting force F_y produces the main component of the force in down milling. The variable pitch between the teeth of the cutting tool is well accounted since one in every two peaks, larger forces (strongly noticeable on force F_y) are computed. Discrepancies arise for forces F_x and F_z in terms of amplitude for this particular milling experiment. The shape of measured force F_x seems different as noise sharpens the peak but simulated signal F_x exhibits its theoretical pattern. A better fit of force F_x was observed during the fitting process of the cutting force coefficients presented in Subsection 5.7.2. Force F_z is usually the most difficult to fit and to correctly replicate since it is mainly generated by the helix angle of the tool. Its magnitude is therefore somewhat similar to the noise level.

Regarding the vibration signals, as for their measured counterparts, they are analysed using the FFT to reveal their frequency content (Figure 6.17b). Comparison of the frequency content is carried out on the vibration signals generated perpendicularly to the feed direction (along x_{Base} for the longitudinal pass). As for the measured vibrations, the FFT of the simulated vibration signals mainly shows the tooth passing frequency and its harmonics. It is the expected behaviour for a stable milling operation. Although the dynamics of the spindle is not modelled, frequency peaks also appear at the spindle frequency and harmonics. The explanation lies in the fact that the variable pitch of the teeth is taken into account in the simulator. The analysis of the frequency content is of particular interest, especially when comparing the chatter frequency as it is the case in the next chapter dedicated to the stability analysis. In contrast, comparing the level of the peaks is not relevant in this case as vibrations are computed at the robot TCP while the measured signals are retrieved on the support on which the cutting force sensor was fixed (Figure 5.56). The accelerometer was far from the cutting zone which can explain the mismatch in amplitude between the measured and simulated peaks.

Overall, the robotic milling simulator faithfully reproduces the frequency content and the cutting forces of an arbitrary milling operation by using the robot multibody model with joint tri-axial flexibility. The computing time of such a simulation is about 1 minute. The effect of the consideration of additional flexibilities is now studied, especially in terms of induced TCP deviations under the application of the cutting forces.

6.4.2 Aluminium milling with joint flexibility and controller

The influence of the chosen controller (inverse dynamics) is now examined. The controller is appended to the robot multibody model with joint tri-axial flexibility. The current model therefore encompasses 28 degrees of freedom. Since no significant differ-

ence is observed between the measured and simulated cutting forces, their superimposition is not shown. Instead, Figure 6.18 presents global views of the measured and simulated cutting forces F_y during the milling of the longitudinal pass of 100 mm. Simulated cutting forces F_y are depicted for each studied effect related to the consideration of a particular flexibility source. As demonstrated in Figure 6.18, the overall shapes of the simulated cutting forces are similar irrespectively from the flexibility source. Measured cutting force F_y of course includes noise but the average amplitude matches the simulated ones.

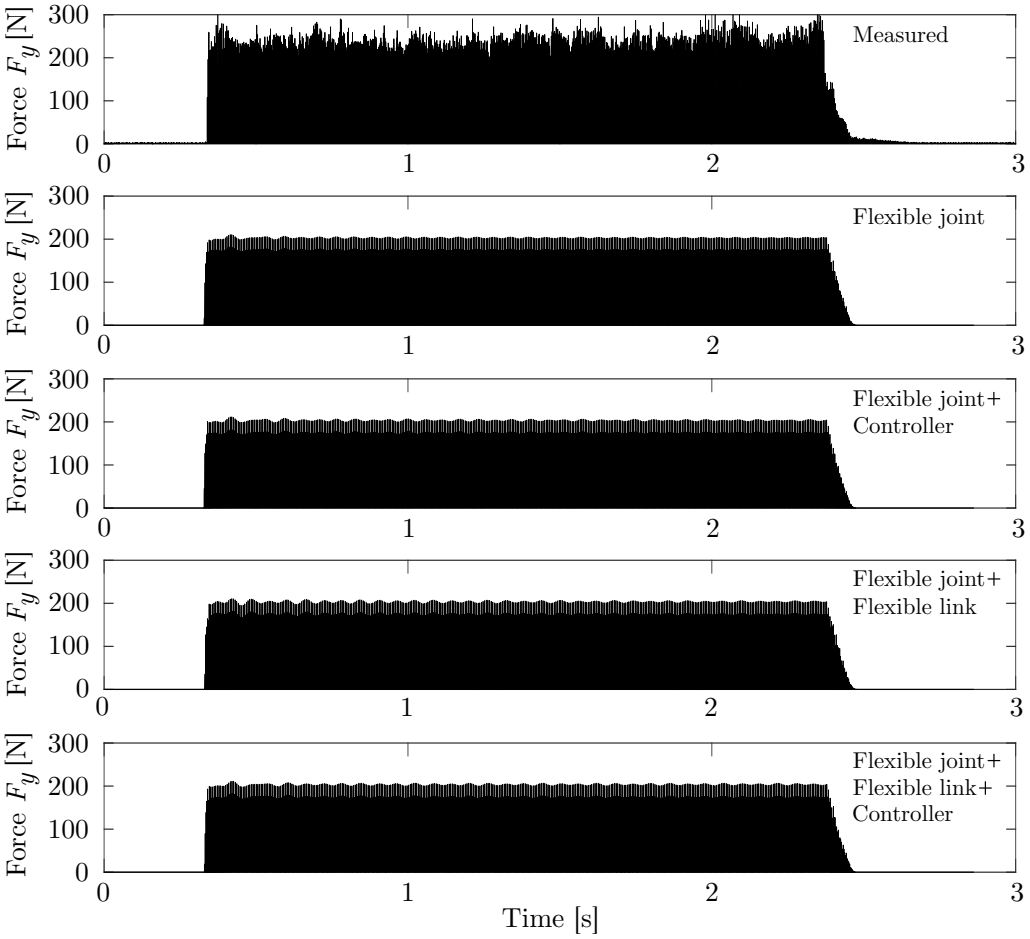
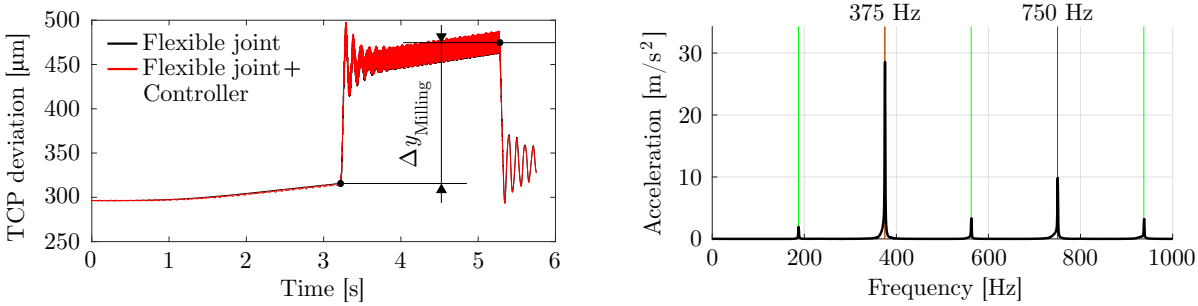


Figure 6.18: Influence of the type of robot modelling on cutting force F_y for the longitudinal pass in down milling (a_p= 2 mm, a_e= 4 mm with a feed per tooth f_z = 0.13 mm/tooth at 11,250 rev/min)

The impact of the controller flexibility is rather analysed through the FFT on the vibration signal perpendicular to the feed direction and via the TCP deviation under the process forces. The computing time of such a simulation is about 4 minutes. The studied TCP deviation is accounted perpendicular to the feed direction (along x_{Base} for the longitudinal pass). It was referenced as Δy_{TCP} in Figure 6.3 presenting the jerk trajectory. Figure 6.19a shows that the static deflection at the beginning of the jerk trajectory is about $\Delta y_{\text{TCP}} \approx 300\ \mu m$ whatever the robot multibody model includes or not the controller effect. The entry of the TCP in the material is strongly noticeable with a sharp increase in the deviation just after $t > 3$ s. It is chosen to assess the impact of the flexibility sources on the TCP deviation resulting from the application of the process forces without

considering the static component. Quantity $\Delta y_{\text{Milling}}$ is therefore defined as the difference of TCP deviation computed just before the tool enters the workpiece and at the end of the milling pass as illustrated in Figure 6.19a. In this case, the computed TCP deviation under the process forces is about $\Delta y_{\text{Milling}} \approx 160\ \mu m$. Again, the effect of the controller flexibility is barely noticeable while the TCP is submitted to the cutting forces. The same observation is formulated in view of the FFT of the vibration signal perpendicular to the feed direction. No additional frequency peak appears within the considered frequency bandwidth of 1000 Hz in Figure 6.19b. It means that the tuning of the simulated controller is such that its flexibility has little effect on the modelled robot dynamics. A similar trend was captured when comparing the FRFs at the TCP when the controller was in action or the brakes enabled (Figure 5.22).



(a) TCP deviation perpendicular to the feed direction

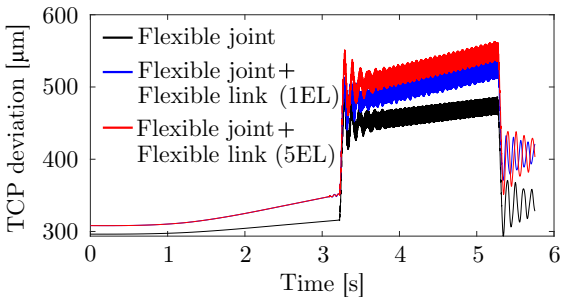
(b) FFT of the simulated vibrations perpendicular to the feed direction

Figure 6.19: Simulated signals using the joint tri-axial flexibility model with the controller for the longitudinal pass in down milling (a_{p}= 2 mm, a_{e}= 4 mm with a feed per tooth f_z = 0.13 mm/tooth at 11,250 rev/min)

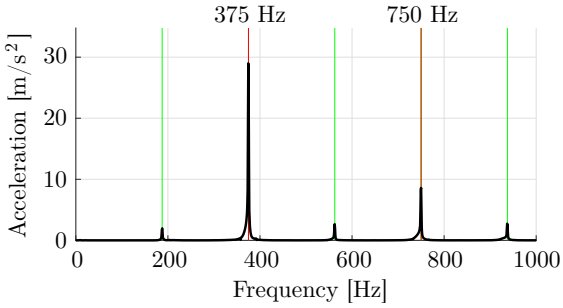
6.4.3 Aluminium milling with joint and link flexibilities

The flexible links are appended to the robot multibody model with joint tri-axial flexibility. As presented in Subsection 3.4.3, only the robot arm and robot forearm are modelled as equivalent flexible beams by using the corotational formulation. This results in a robot multibody model with a total of 34 degrees of freedom since each equivalent flexible beam is represented with one single flexible beam element gathering 6 degrees of freedom (named as the *1EL model*). Considering the alternative modelling of flexible beams presented in Appendix C, consisting in the assembly of successive flexible beam elements, another robot multibody model is derived. Its equivalent flexible beams are constituted of five successive beam elements for a total of 82 degrees of freedom, including the joint flexibilities (named as the *5EL model*).

Yet again, the TCP deviation and FFT on the vibrations perpendicular to the feed direction are shown in Figure 6.20. The computing time of such a simulation is about 2 minutes for the 1EL model and about 8 minutes for the 5EL model. The frequency content is only illustrated for the 1EL robot multibody model and do not exhibit any additional frequency peak in Figure 6.20b. Tooth passing and spindle frequencies as well as



(a) TCP deviation perpendicular to the feed direction



(b) FFT of the simulated vibrations perpendicular to the feed direction

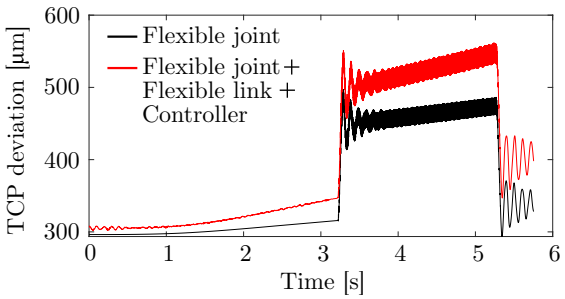
Figure 6.20: Simulated signals using the joint tri-axial flexibility model and the flexible links for the longitudinal pass in down milling (a_p= 2 mm, a_e= 4 mm with a feed per tooth f_z = 0.13 mm/tooth at 11,250 rev/min)

their harmonics are also observed when using the 5EL robot multibody model. In terms of TCP deviation, the inclusion of the flexible beams in the modelling clearly increases the simulated deflection (Figure 6.20a). Static deflection slightly increases but remains around $\Delta y_{\text{TCP}} \approx 300\ \mu m$. In contrast, TCP deviations resulting from the cutting process are now about $\Delta y_{\text{Milling}} \approx 200\ \mu$m for both models including flexible links (1EL and 5EL models). A slight offset of approximately 30 μm is noticed during the cutting force application between the model including flexible links and the one only comprising the joint flexibility. From a simulation point of view, it means that the consideration of the link flexibility leads to an increase of about 20 % in TCP deviation compared to the case in which the robot multibody model only gathers the joint flexibilities. A very few studies were accomplished in robotic milling with the consideration of flexible links. Even though interesting results were derived e.g. chatter prediction in static robot posture [107], a clear value of the effect of link flexibility, relatively to the joint flexibility, was not found (to the best of the author's knowledge).

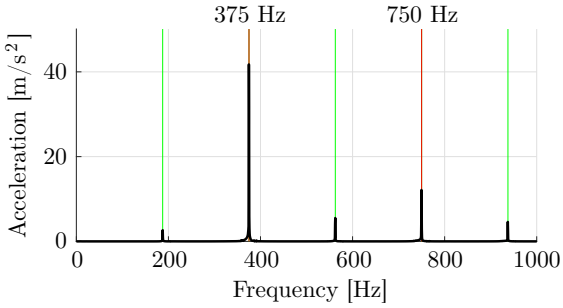
6.4.4 Aluminium milling with all flexibility sources

The effect of the controller and flexible beam flexibilities are now appended to the reference robot model. Considered flexible beams are constituted of one flexible beam element, as for the remainder of this work. The whole multibody model thus gathers a total of 40 degrees of freedom. The TCP deviation and the FFT of the simulated vibrations perpendicular to the feed direction are once more depicted in Figure 6.21. The computing time of such a simulation is about 7 minutes.

FFT of the simulated signals still exhibits frequency peaks at the tooth passing and spindle frequencies as well as their harmonics (Figure 6.21b). In Figure 6.21a, it is noticed that resulting TCP deviations perpendicular to the feed direction are very similar to the ones already observed when only appending the flexible links. Hence, the TCP deviation is about $\Delta y_{\text{Milling}} \approx 200\ \mu$m. It again shows that the controller effect is negligible relatively to the flexibilities originating from the joints and the links.



(a) TCP deviation perpendicular to the feed direction



(b) FFT of the simulated vibrations perpendicular to the feed direction

Figure 6.21: Simulated signals using the joint tri-axial flexibility model, the flexible links and the controller for the longitudinal pass in down milling (a_p= 2 mm, a_e= 4 mm with a feed per tooth f_z = 0.13 mm/tooth at 11,250 rev/min)

6.4.5 Machined lateral face in aluminium milling

One could ask if the simulated TCP deviations $\Delta y_{\text{Milling}}$ are in the correct order of magnitude with respect to the actual test. The longitudinal pass in down milling was therefore carried out in an aluminium plate of the same material. For this purpose, the tool first needs to remove material in slot milling as illustrated in Figure 6.22. The longitudinal pass in down milling is achieved with the same axial and radial depths of cut, a_p= 2 mm and a_e= 4 mm, respectively, same feed per tooth f_z= 0.13 mm/tooth and same spindle speed Ω= 11,250 rev/min. In Figure 6.22, the left side corresponds to the lateral face machined in down milling while the right side results from the slotting operation (the right side is machined in up milling). The profiles of the lateral faces are then measurable with the roughmeter and the coordinate measuring machine.

Using the CMM, each lateral face is discretised in 200 points whose coordinates are measured. The mean distance between the two profiles is evaluated to 13.86 mm instead of 14 mm. An offset of 140 μm is then measured relatively to the desired distance. It confirms that the simulated order of magnitude is not too far from the actual experiment.

The rugged profile left after the slotting operation originates from the up milling direction of the tool. For aluminium tests, it was generally observed a considerable degradation of the surface finish when adopting the up milling direction. Burrs were usually formed which made impracticable the roughness assessment of lateral faces machined in up milling. Figure 6.23 presents the corresponding photographs of the lateral faces machined in down milling (Figure 6.23a) and in up milling (Figure 6.23b).

The measured roughness in down milling can be compared to the one derived by the robotic milling simulator. Figure 6.24 illustrates a typical roughness assessment carried out on the simulated lateral profiles.

In Figure 6.24, the left illustration represents the desired situation in which a radial of

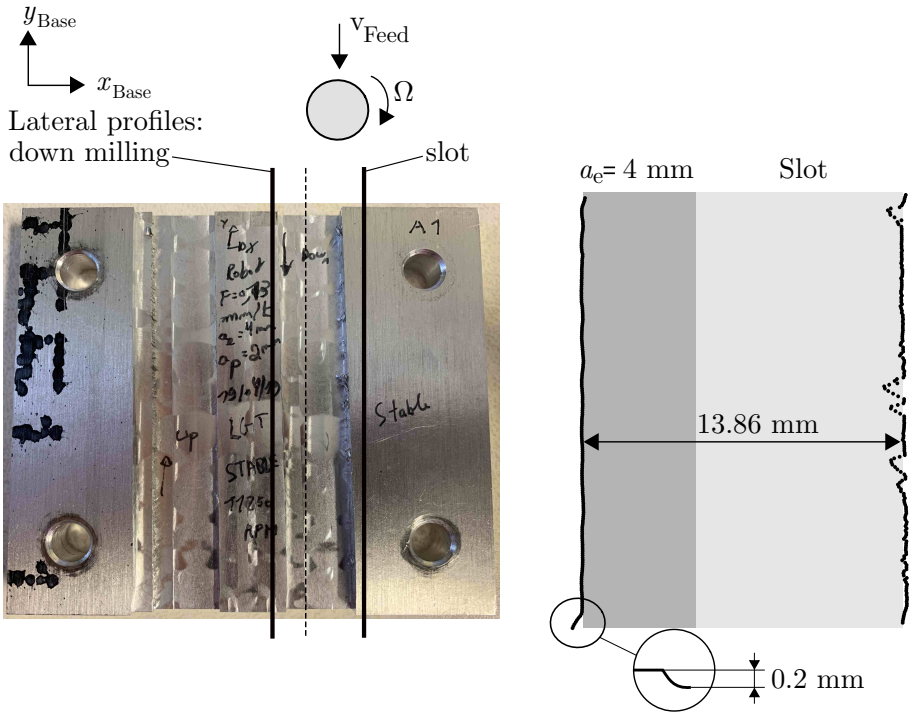


Figure 6.22: Measured lateral profiles with the coordinate measuring machine for the longitudinal pass (a_p= 2 mm, a_e= 4 mm and slot with a feed per tooth f_z = 0.13 mm/tooth at 11,250 rev/min)

(a) Down milling

(b) Up milling

Figure 6.23: Close-up on the aluminium lateral faces for the longitudinal pass (a_p= 2 mm, a_e= 4 mm with a feed per tooth f_z = 0.13 mm/tooth at 11,250 rev/min)

cut a_e= 4 mm is exactly removed. The right illustration is a close-up on the virtual machined profiles depending on the involved robot multibody model. It is observed that the closest profile relatively to the desired one is obtained by using the multibody model with the joint tri-axial flexibility. The offset, resulting from the joint deflections, is about 460 μm as already shown in Figure 6.19a with $\Delta y_{\text{Milling}}$. Larger deviations are naturally observed if the link flexibility is accounted for. The effect of the controller flexibility is again barely noticeable.

Although the length of the workpiece is $L_{\text{Workpiece}}$= 100 mm, the evaluation length for the kinematic roughness is l_n= 4.8 mm for the actual and simulated profiles. The roughness of the actual profile is measured at both ends of the profile. The mean

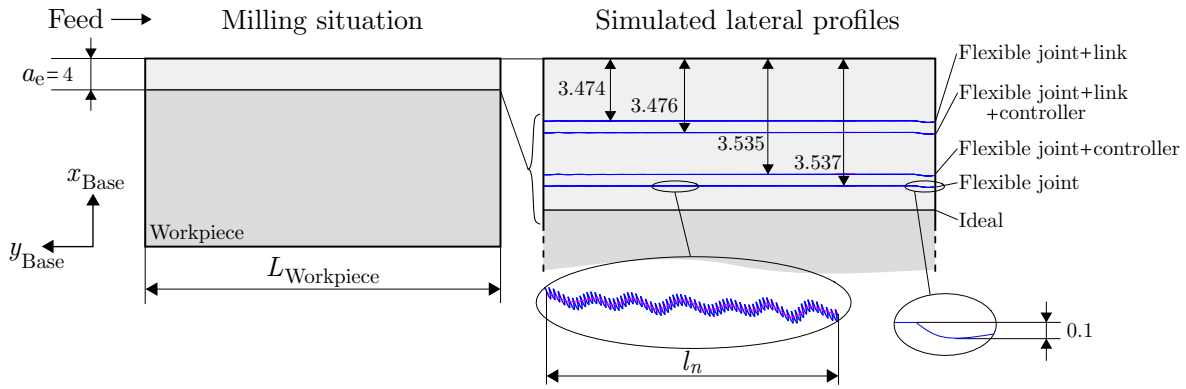


Figure 6.24: Influence of the type of robot modelling on the machined lateral profile for the longitudinal pass in down milling (dimensions in mm) with a_p= 2 mm, a_e= 4 mm and with a feed per tooth f_z = 0.13 mm/tooth at 11,250 rev/min

values of roughness of the actual profile are $R_{a,actual}$= 0.46 μm, $R_{q,actual}$= 0.55 μm and $R_{t,actual}$= 3.08 μm. Regarding the computation of roughness on the simulated profiles, the equations presented in Subsection 4.5.6 are utilised. Roughness is computed at the middle of the length of the simulated profile. An important note concerns the evaluation of the arithmetic R_a and quadratic R_q roughnesses: the mean line must follow the waviness of the lateral profile originating from the robot flexibility as presented in Figure 6.24 in the close-up of the simulated profile. Otherwise, an overestimation of the aforementioned quantities is found. If such a method is followed, simulated values of roughness, averaged from the different versions of robot multibody models, are the followings: $R_{a,simu}$= 0.46 μm, $R_{q,simu}$= 0.49 μm and $R_{t,simu}$= 4.14 μm. Simulated values of roughness are very similar to the measured ones for this arbitrary milling test. It confirms that the robotic milling simulator is able to predict the roughness for stable milling operations in down milling.

In Figure 6.24 is also highlighted a small deviation of 0.1 mm in the lateral profile which happened at the end of the pass. Same deviation was observed on the actual part as shown in Figures 6.22 and 6.23a and was estimated to 0.2 mm. The explanation of the phenomenon lies in the fact that less material needs to be removed by the tool at the end of the pass, hence less process forces are applied on the TCP which deflects less.

6.4.6 Aluminium milling in various feed directions

The presented milling operation in aluminium is simulated according to the four milling situations described in Figure 5.58. The cutting conditions remain the same (a_p= 2 mm, a_e= 4 mm with a feed per tooth f_z = 0.13 mm/tooth at 11,250 rev/min) and only the feed direction is therefore changed. For each milling situation, the assessment of the flexibility sources is also studied. Table 6.4 condenses all the main results in terms of TCP deviation $\Delta y_{Milling}$ and roughness (R_a, R_q and R_t).

Results for the longitudinal pass in down milling were presented through Subsections 6.4.1 to 6.4.5. For the longitudinal pass in up milling, TCP deviations range from 71.5 μm,

	Longitudinal down milling				Longitudinal up milling			
	$\Delta y_{\text{Milling}}$	R_{a}	R_{q}	R_{t}	$\Delta y_{\text{Milling}}$	R_{a}	R_{q}	R_{t}
Flexible joint	159.3	0.46	0.48	3.37	71.5	0.32	0.42	22.58
Flexible joint+controller	159.9	0.47	0.50	4.12	73.1	0.34	0.44	22.12
Flexible joint+link	197.6	0.46	0.48	4.79	94.5	0.36	0.45	13.9
Flexible joint+link+controller	201.9	0.47	0.48	4.28	105.8	0.37	0.55	24.37
	Transversal down milling				Transversal up milling			
	$\Delta y_{\text{Milling}}$	R_{a}	R_{q}	R_{t}	$\Delta y_{\text{Milling}}$	R_{a}	R_{q}	R_{t}
Flexible joint	67.8	0.27	0.5	3	65.9	0.25	0.32	12
Flexible joint+controller	73.1	0.26	0.49	3	63.8	0.24	0.33	10
Flexible joint+link	71.3	0.37	0.6	2	65.8	0.21	0.33	11
Flexible joint+link+controller	75.6	0.22	0.46	2	66.2	0.24	0.37	11

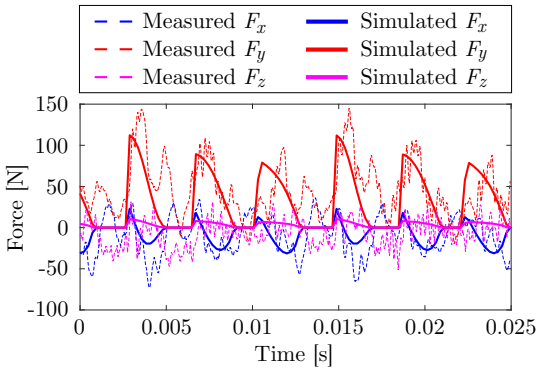
Table 6.4: Simulated milling deviation and roughness in μm amongst the four milling situations and the flexibility sources for aluminium milling (a_{p}= 2 mm, a_{e}= 4 mm with a feed per tooth f_{z} = 0.13 mm/tooth at 11,250 rev/min)

derived with the robot multibody model with joint tri-axial flexibility, to 105.8 μm if all flexibilities are included. Although it seems that TCP deviations are smaller than for the longitudinal pass in down milling, it must be pointed out that a perfect cut is assumed in the simulator. In other words, the simulated values are very unlikely and underestimated since burrs are not modelled in the robotic milling simulator. Same comments apply to the roughness which is not measurable on the actual parts in up milling because of the poor surface finish. Note that total roughness R_{t} sharply increases from a simulation point of view. The consideration of the link flexibility adds about 30 % of additional deviation relatively to the joint deflections. For the transversal pass in down milling, smaller TCP deviations are generated as it is observed on the machined workpiece with less than 100 μm of deflection. The different sources of flexibility produce a rather constant resulting TCP deflection. The roughness is however underestimated as actual values of $R_{\text{a,actual}}$= 0.59 μm, $R_{\text{q,actual}}$= 0.72 μm and $R_{\text{t,actual}}$= 4.47 μm were measured. For the transversal pass in up milling, TCP deflections are in the same order of magnitude irrespectively from the flexibility source. Nevertheless, it is very likely that they are underestimated as well as for the simulated roughness as surface finish was poor in up milling. The transversal direction leads to less TCP deflection overall as the main component of cutting force (F_y) acts along a direction in which the robot is stiffer. Indeed, for the transversal pass, cutting force F_y is applied along x_{TCP} which yielded small peaks in direct FRF H_{xx} depicted in Figure 5.17. In summary, the robotic milling simulator provides satisfying results in down milling. In up milling, since burrs are generally created during the pass in aluminium, the robotic simulator cannot deliver correct results as the phenomenon is not modelled.

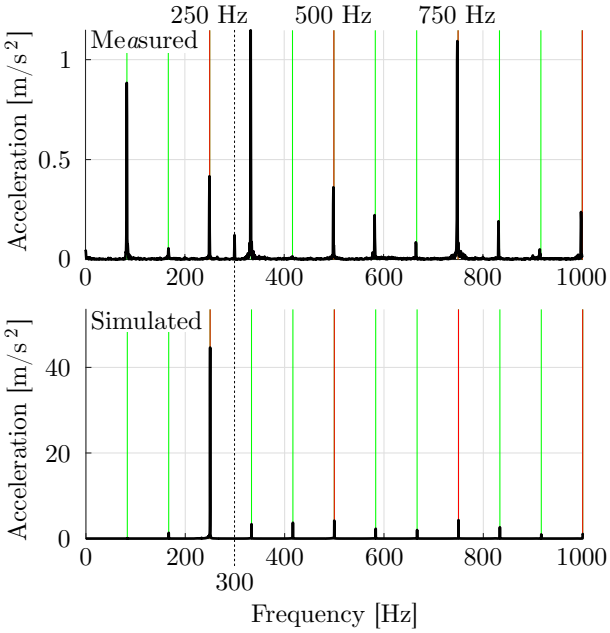
6.4.7 Steel milling with joint tri-axial flexibility

A longitudinal pass in down milling is conducted in steel by using the robotic milling simulator. Cutting conditions are modified to match the requirements of steel cutting. The feed rate becomes f_{z}= 0.09 mm/tooth and the radial depth of cut remains a_{e}= 4 mm.

As a stable pass is desired, the axial depth of cut is reduced to 0.5 mm and a spindle speed of 5,000 rev/min is selected. Figure 6.25 presents the comparison of the measured and simulated signals in terms of cutting forces and frequency content when relying on the multibody model only comprising the joint flexibilities (22 DOFs). Two tool turns are illustrated. The superimposition of the cutting forces in Figure 6.25a exhibits a strong correlation for the dominant signal related to force F_y acting perpendicularly to the feed direction (along x_{Base} for the longitudinal pass). Measured signals are busy but a periodic repetition of three peaks is remarked as the tool possesses three teeth with a variable pitch. The effect is well replicated in the robotic milling simulator. As for aluminium, larger discrepancies are observed for forces F_x and F_z due to the noise and the low levels. FFTs of the measured and simulated vibrations along a direction perpendicular to the feed motion are compared in Figure 6.25b. In this figure, the green lines are related to the spindle frequency and its harmonics and the red lines are paired with the tooth passing frequency and its harmonics. Overall, the tooth passing (250 Hz) and spindle (83.33 Hz) frequencies and their harmonics are well captured in the simulator since the modelled teeth feature a variable pitch. FFT applied on the measured vibrations however indicates a stronger harmonic related to the spindle frequency at 83.33 Hz. A peak at 300 Hz can also be noticed and corresponds to one of the harmonics of the rotating fan cooling the spindle. The effect was also present, to a lesser extent, in the FFT of the measured vibrations in aluminium cutting in Figure 6.16b.



(a) Cutting force comparison



(b) FFT of the measured and simulated vibrations perpendicular to the feed direction

Figure 6.25: Validation example in steel for the Stäubli TX200 robot for the longitudinal pass in down milling (a_{p}= 0.5 mm, a_{e}= 4 mm with a feed per tooth f_z = 0.09 mm/tooth at 5,000 rev/min)

6.4.8 Steel milling in various feed directions

The study regarding the assessment of the different flexibility sources is easily transposable to steel milling. A summary is proposed in Table 6.5 and shows the influence of the flexibility sources and of the milling situations (longitudinal or transversal and down or up milling). The simulated milling operations preserve the same cutting conditions (a_p= 0.5 mm, a_e= 4 mm with a feed per tooth f_z = 0.09 mm/tooth at 5,000 rev/min) and only the feed direction is changed. In stable conditions, as for aluminium cutting, the simulated cutting forces and the FFT applied on the modelled vibrations exhibit a similar trend as presented in Figure 6.25 irrespectively from the flexibility source. Small differences are nonetheless observed regarding the TCP deviations and the roughness of the simulated lateral profile (Table 6.5). For the longitudinal pass in down milling, the TCP deviations range from 127.4 μm, if the multibody model only comprises flexible joints, to 163.1 μm when all the flexibilities are involved. Again, it is noticed that the controller flexibility is negligible relatively to the one generated by the joints and the links. As in aluminium cutting, the flexible links induce an additional increase of 20 % in the TCP deviation relatively to the multibody model only comprising flexible joints. The average roughness of the lateral profile reaches the following values for the longitudinal pass in down milling: $R_{a,simu}$= 0.37 μm, $R_{q,simu}$= 0.43 μm and $R_{t,simu}$= 3.55 μm. Similar values are obtained throughout the four exposed milling situations. The simulated values for roughness cannot be compared to experimental data since the axial depth of cut (a_p= 0.5 mm) produces a lateral profile too thin for the size of the roughmeter probe. Larger axial depths of cut lead to unstable conditions as it is presented in the next chapter. Note that from an observable point of view, the lateral profiles in steel milling, either in down or up milling, exhibit a surface finish similar to the one obtained in down milling in aluminium. The same comment is applied to the four milling situations presented in Table 6.5.

	Longitudinal down milling				Longitudinal up milling			
	$\Delta y_{Milling}$	R_a	R_q	R_t	$\Delta y_{Milling}$	R_a	R_q	R_t
Flexible joint	127.4	0.38	0.45	2.89	47	0.25	0.33	2.65
Flexible joint+controller	129.4	0.39	0.46	2.6	48.3	0.3	0.34	3.1
Flexible joint+link	161.7	0.33	0.4	4.39	77	0.37	0.42	4.66
Flexible joint+link+controller	163.1	0.37	0.43	4.31	77.5	0.35	0.4	4.58
	Transversal down milling				Transversal up milling			
	$\Delta y_{Milling}$	R_a	R_q	R_t	$\Delta y_{Milling}$	R_a	R_q	R_t
Flexible joint	61.4	0.26	0.39	2	44.1	0.15	0.38	2
Flexible joint+controller	64.4	0.31	0.51	2	43.5	0.08	0.27	2
Flexible joint+link	63.3	0.24	0.49	2	45.3	0.34	0.41	2
Flexible joint+link+controller	66.7	0.25	0.41	2	44.3	0.35	0.41	2

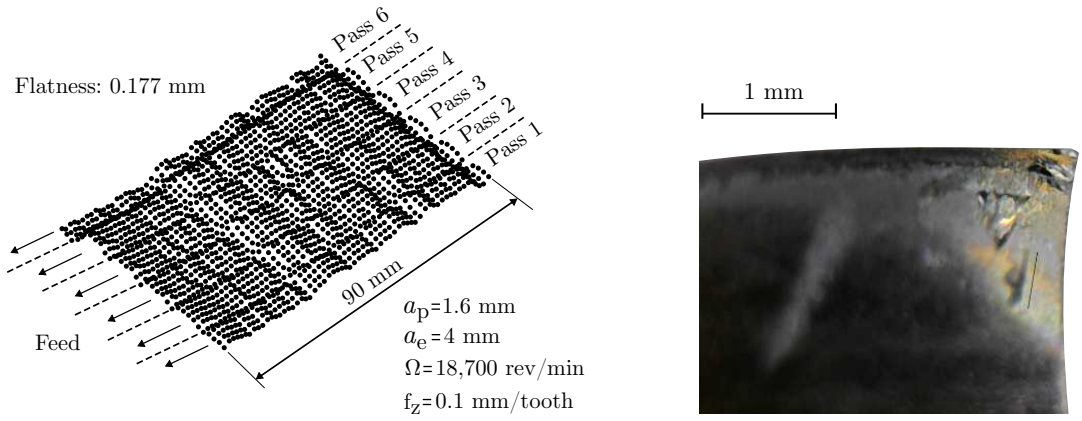
Table 6.5: Simulated milling deviation and roughness in μm amongst the four milling situations and the flexibility sources for steel milling (a_p= 0.5 mm, a_e= 4 mm with a feed per tooth f_z = 0.09 mm/tooth at 5,000 rev/min)

For the longitudinal pass in up milling, smaller TCP deviations are observed relatively to the down milling pass. The effect of link flexibility with respect to the joint flexibility is

about 30 % as also observed in aluminium milling for the longitudinal pass in up milling. The flexibility of the controller still looks negligible compared with the other flexibility sources. For the transversal passes, smaller TCP deviations are overall remarked since cutting force F_y is applied along a stiff robot direction whose dynamics is given by H_{xx} (Figure 5.17). Irrespectively from the flexibility source and the milling direction, simulated TCP deviations are rather constant.

6.5 Tool wear and flatness in face milling

Some additional information is provided regarding tool wear and flatness of the resulting machined bottom face. In order to assess the flatness, a layer of constant thickness was removed from the plates in aluminium and in steel. In other words, six passes following the longitudinal motion in down milling, with a radial depth of cut a_e= 4 mm, were carried out in order to achieve a face milling operation. In aluminium, the bottom profile of the machined workpiece, measured via the coordinate measuring machine, is depicted in Figure 6.26a for an axial depth of cut a_p= 1.6 mm, a spindle speed Ω= 18,700 rev/min and a feed per tooth f_z= 0.1 mm/tooth. By fitting the best root mean square (RMS) plane to the point cloud, a flatness of 0.177 mm is computed. Besides, it is observed that the bottom profile of each pass is somewhat wavy along the feed direction due to the high robot compliance along the tool axis. In addition, a significant offset in height (about 0.1 mm) is noticed at the overlapping of two passes (along the perpendicular direction to the feed motion). Note that the waviness was less pronounced for the transversal passes thus resulting in a better flatness around 0.09 mm. In steel, the measured flatness was in average about 0.4 mm for both milling directions.



(a) Flatness appraisal (b) Tool wear observed at the rake face

Figure 6.26: Flatness of a plane machined using longitudinal passes in down milling and illustration of tool wear for the end mill used in aluminium milling

Tool wear was analysed with a digital microscope (Opti-TekScope Model OT-HD). An illustration of tool wear is presented in Figure 6.26b for the end mill used for aluminium experiments. Using the tool for multiple milling experiments in order to derive the experimental stability limits, wear was typically localised at the tip of the end mill and small

crater could be recognized at the rake face. In severe conditions leading to chatter, some parts of the cutting edges could be broken.

6.6 Discussion

The interactions between the robot multibody model and the dynamic milling model were presented in this chapter. Overall, knowing the current robot TCP position and the previous ones, and the workpiece geometry, the dynamic milling model handles the computation of the cutting forces which are then applied to the robot multibody model. The simulation of robotic milling operation is carried out in the time domain by using the developed simulator coupling the multibody and milling models. The computation of such simulation relies on an implicit numerical integration algorithm solving the equations of motion over time. Once solved, the geometry of the workpiece is updated and the numerical integration process is repeated. The simulator was validated on two literature examples related to machine tool. The simulated stability lobes were compared to the original ones and a close match was observed for both examples. It confirmed that coherent results can be yielded using the developed robotic milling simulator.

Once the simulator validated on machine tool examples, it was used to replicate measurements obtained via robotic milling operations in aluminium and in steel. Stable milling conditions were tested for the validation of the simulator. Special attention was paid so that the robot TCP smoothly reaches the milling zone with a constant feed rate in the simulations. For this purpose, the straight line motion was accomplished by following a trajectory for which the acceleration and deceleration phases were managed with a constant jerk. After the simulation of the milling pass, the analysis of the results was carried out on the comparison of the measured and simulated cutting forces, FFT of the vibration signals and lateral profiles of the machined parts. The influence of the flexibility sources originating from the joints, the links and the controller was investigated as well as the effect of the feed direction. For simulations in aluminium milling, it was shown that the robotic milling simulator was able to accurately replicate the measured cutting forces, frequency content and roughness for down milling passes. Up milling passes in aluminium cannot be faithfully reproduced since actual milling tests formed burrs on the lateral face of the workpiece. The phenomenon is not taken into account in the robotic milling simulator since a perfect cut is assumed and therefore leads to incorrect prediction of TCP deviation and roughness for up milling passes in aluminium. In terms of TCP deviation, it was shown that the consideration of the flexible links induced 20 to 30 % of additional TCP deflections compared to the case in which only the joint flexibility is accounted for. This effect was observed when the component of cutting forces normal to the feed direction was acting on the TCP, along a direction perpendicular to the robot plane (longitudinal pass). When the component of cutting forces normal to the feed direction acts in the robot plane, TCP deviations proved to be rather constant irrespectively from the tested flexibility source (transversal pass). On the other hand, the controller flexibility was appraised negligible relatively to the joint and the link flexibilities. The same trends were overall captured in steel milling except that actual tests did not reveal burrs. Now

that the robotic milling simulator is globally validated on stable cases, a stability analysis is carried out to predict the stability limits of milling operations in aluminium and in steel.

Chapter 7
Stability analysis

In this chapter, a stability analysis is carried out on all the derived modellings, either with the multibody models of the robot or by using the conventional approaches for machine tool. The objective is to determine optimal cutting parameters leading to stable milling operations to increase the productivity. For this purpose, the stability lobe diagrams, delineating the stable and unstable operating conditions, are generated for all the milling situations (longitudinal and transversal passes in down or up milling) and for both materials, aluminium and steel. The generated stability charts are experimentally validated by conducting a series of milling experiments with the Stäubli TX200 robot.

A total of 88 stability charts are computed since 11 model variants were derived throughout Chapters 3 to 5. Accounting for the four milling situations and the two tested materials, one can generate the 88 stability charts. A summary of the 11 model variants is proposed below:

1. Stäubli TX200 model with joint tri-axial flexibility: the multibody model of the Stäubli TX200 robot is built by including three orthogonal elastic elements (torsional springs and dampers) at the interface between the links and the rotors as well as between the robot base and the ground, as explained in Subsection 3.3.4. As a reminder, the Stäubli TX200 model refers to the multibody model built on the basis of the provided manufacturer's data regarding the inertial properties. The model includes the effect of the supposed DVA for the duplicated modes around 21 Hz developing a mode shape which involves a rotation perpendicular to the motion axis of the first joint. Elastic parameters are identified in Subsection 5.3.3. The model with joint tri-axial flexibility therefore comprises 22 degrees of freedom $\mathbf{q} \in \mathbb{R}^{22\times 1}$.

2. UMons TX200 model with joint tri-axial flexibility: the multibody model is the same as the one described in the first item except that the inertial parameters are collected from the CAD models. Elastic parameters are identified in Subsection 5.3.3. The model thus also includes 22 degrees of freedom $\mathbf{q} \in \mathbb{R}^{22\times 1}$.

3. Stäubli TX200 model with joint and controller flexibilities: the flexibility of the controller is appended to the Stäubli TX200 model with joint tri-axial flexibility. Parameters of the inverse dynamics controller are found in Subsection 5.6. The model with joint flexibility and controller includes 28 degrees of freedom $\mathbf{q} \in \mathbb{R}^{28\times 1}$.

4. Stäubli TX200 model with joint and link flexibilities: the flexibilities of the robot arm and forearm are added to the Stäubli TX200 model with joint tri-axial flexibility.

Elastic parameters of the equivalent flexible beams are determined in Subsection 5.4. The model with joint and link flexibilities gathers 34 degrees of freedom $\mathbf{q} \in \mathbb{R}^{34\times1}$.

5. Stäubli TX200 model with joint, link and controller flexibilities: all the assessed flexibilities (joints, links and controller) are brought together in one single multibody model for a total of 40 degrees of freedom $\mathbf{q} \in \mathbb{R}^{40\times1}$.

6. Stäubli TX200 model with edge force coefficients: it is the Stäubli TX200 model with joint tri-axial flexibility except that the cutting forces applied to its tool tip are computed by adopting the force model with the edge force coefficients. The latter were identified in Subsection 5.7. Note that in the other multibody models, cutting forces are computed on the basis of the standard cutting force model which do not comprise the edge force coefficients. The model includes 22 degrees of freedom $\mathbf{q} \in \mathbb{R}^{22\times1}$.

7. Stäubli TX200 model fitted on 35 Hz: it is also the Stäubli TX200 model with joint tri-axial flexibility but for which elastic parameters are identified on the measured FRFs limited to a bandwidth of 35 Hz. The fitting of the measured and resulting simulated tool tip FRFs over 35 Hz were shown in Figure 5.48. Note that the multibody models presented in the previous items are fitted over a bandwidth of 100 Hz. Stability lobe diagrams are generated using the model fitted over 35 Hz to emphasise the effect of an overestimated joint damping during the model updating of the elastic parameters. The model also comprises 22 degrees of freedom $\mathbf{q} \in \mathbb{R}^{22\times1}$.

8. Semi-discretisation method (SDm): stability charts are computed using the classical time domain method in conventional milling. The semi-discretisation method was derived in Section 4.4. Inputs of the method are the modal parameters of the Stäubli TX200 robot which can be derived from the measured FRFs at the tool tip presented from Figures 5.17 to 5.19. The implemented method does not consider the cross coupling of FRFs and modal parameters used to generate the stability lobe diagrams are explicitly provided in this chapter.

9. 2D analytical zero-order approximation (ZOA): stability lobe diagrams are derived from the frequency domain method in conventional milling. The 2D analytical zero-order approximation method was presented in Subsection 4.3.1. It is named as the 2D ZOA since its inputs are the synthesized tool tip FRFs (Figures 5.17 and 5.18) but restricted to the ones in the cutting plane (H_{xx}, H_{xy}, H_{yx} and H_{yy}). The resolution of such method is developed in Section E.1 in Appendix E.

10. 3D analytical zero-order approximation: stability charts are calculated with the 3D zero-order approximation method for which the directional coefficients generate cutting forces in the XYZ directions. The 3D analytical ZOA was introduced in Subsection 4.3.2 and its inputs are naturally all the nine tool tip measured FRFs (Figures 5.17 to 5.19). The resolution of such method is developed in Section E.3 in Appendix E.

11. 3D digital zero-order approximation: it is basically the same method developed in the previous item but its resolution is based on a discrete solution from the measured FRFs as explained in Section E.4 in Appendix E.

Table 7.1 exposes the relationships between the eleven tested models which are used to compute the stability charts for all milling situations in aluminium and steel. Relationships between the models regard the assessed flexibility, the cutting force model and the consideration of the cross-FRFs in the robot dynamic model.

Model i	Joint tri-axial flexibility	Controller flexibility	Flexible links	Edge force	2D cutting forces	3D cutting forces	Cross-FRFs
1	×					×	×
2	×					×	×
3	×	×				×	×
4	×		×			×	×
5	×	×	×			×	×
6	×			×		×	×
7	×					×	×
8						×	
9					×		×
10						×	×
11						×	×

Table 7.1: Relationships between all tested models for deriving the stability charts

Stability lobe diagrams are first computed for the multibody models with joint tri-axial flexibility. A comparison of the measured and simulated cutting force and vibration signals is proposed for particular cutting conditions generating instability. The measured and simulated chatter frequencies and resulting machined lateral profiles are also examined and correlated. Afterwards, the effects of the flexibility options (joints, links and controller) for the multibody model on the stability charts are compared together. Finally, the efficiency of the classical stability methods in conventional milling is appraised when they are applied to robotic milling operations.

Note that due to the large number of graphs, all the 88 individual stability charts are gathered in Appendix M in which they are illustrated separately. In this chapter, the stability lobe diagrams obtained from the different models are superimposed to assess particular effects.

7.1 Stability lobes based on the multibody model

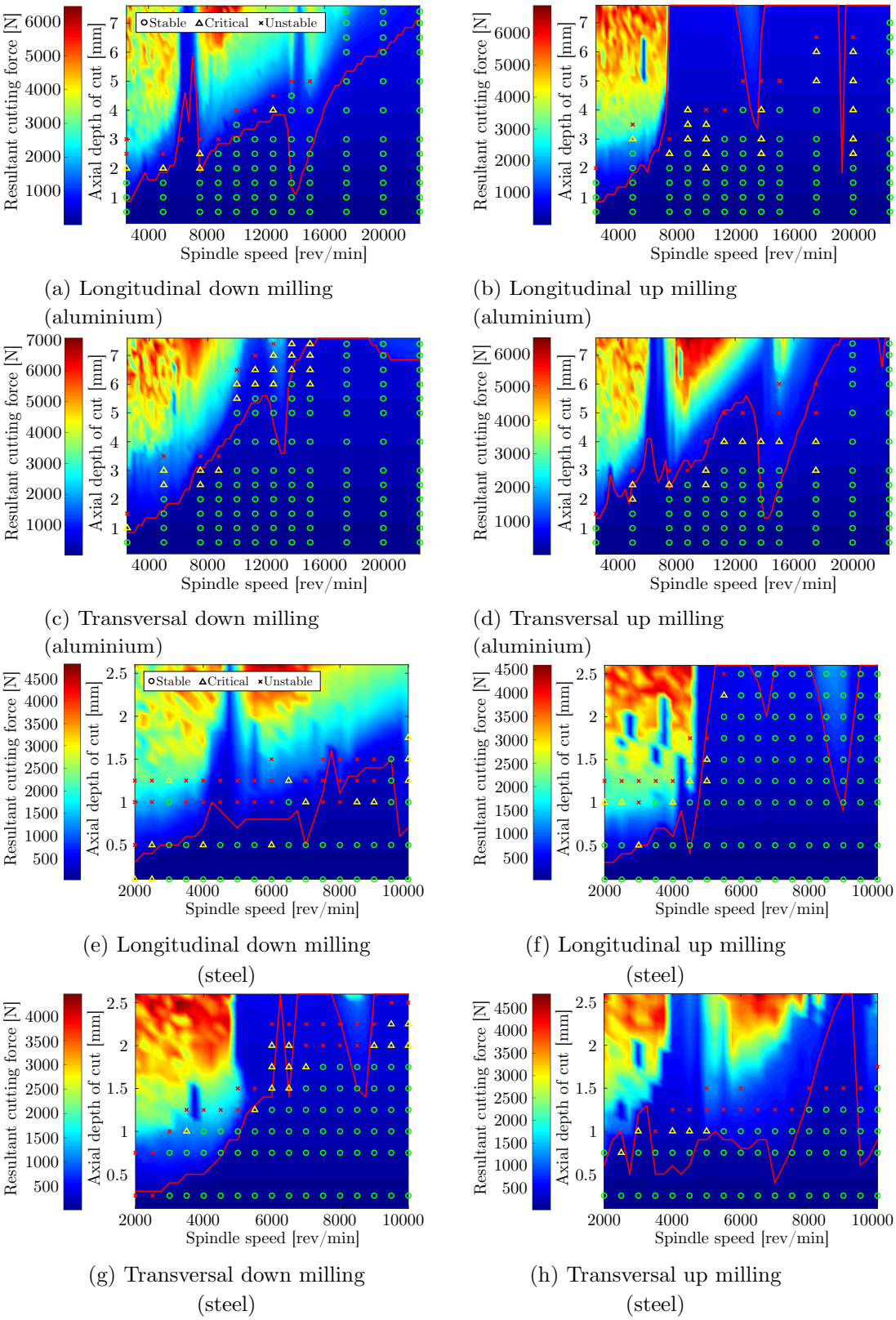
Stability charts are predicted by using the Stäubli and UMons multibody models with joint tri-axial flexibility for all the milling situations in aluminium and steel. Results are compared to the measured counterparts.

7.1.1 Generated lobes using the joint tri-axial flexibility

For the Stäubli TX200 model with joint tri-axial flexibility, the eight stability lobes are illustrated in Figure 7.1 for the longitudinal and transversal passes in aluminium and steel. In this figure, the simulated stability limit is highlighted with a red curve. As for the second developed example in machine tool in Chapter 6, the plot also shows the evolution of the maximum magnitude of the resultant cutting force F_{Rc} inside the investigated stability region. The stability of experimental milling tests is superimposed to the figure through some markers: a circle ($\circ$) indicates stable conditions, a cross ($\times$) designates unstable conditions and a triangle ($\triangle$) or a diamond ($\diamond$) refers to a critical case (slight chatter). The experimental stability criteria are presented in the next subsection when assessing the stability of particular cutting conditions from the generated charts offering a first global view.

From the simulation point of view, the depiction of the evolution of the maximum magnitude of the resultant cutting force involves to save its value after each simulation. The whole stability domain must therefore be swept to appraise the global evolution of the maximum value of F_{Rc}. For the simulations in aluminium, the loss of stability and $\max(F_{\mathrm{Rc}})$ are evaluated on a discretised stability domain from 2,500 rev/min to 22,500 rev/min with an increment of 250 rev/min and the axial depth of cut is progressively increased from 0.1 mm to 7.6 mm with an increment of 0.25 mm. It means that 2,400 individual robotic milling simulations need to be accomplished in the time domain in order to derive one stability chart for aluminium. The range of tested spindle speeds is guided by the tool manufacturer recommendations. For the simulations in steel, the spindle speed is varied between 2,000 to 10,000 rev/min at 250 rev/min increments and the axial depth of cut is ranged from 0.1 mm to 2.6 mm with a step of 0.1 mm. A total of 800 individual robotic milling simulations is therefore needed to generate one stability chart for steel. The feed per tooth $\mathrm{f_z}$ (0.13 mm/tooth in aluminium and 0.09 mm/tooth in steel) and the radial depth of cut a_{e}= 4 mm are maintained constant throughout all the simulations for one material. All the individual robotic milling simulations are carried out as described in Section 6.4 i.e. by starting from the adequate initial TCP pose depending on the milling situation (Table 6.3) and by implementing a jerk trajectory before proceeding to the material removal and the change of time step. The assessment of stability is finally given by the criterion explained in Subsection 4.5.5. For the record, it takes about three days to compute one stability chart with the backup of $\max(F_{Rc})$ in aluminium and about one day in steel using the robot multibody model with joint tri-axial flexibility. If the backup of $\max(F_{Rc})$ is not needed, the stability limit can be computed by relying on the algorithm shown in Figure 6.10, in order to reduce the computing time. In this case, the generation of one stability lobe in aluminium takes about 10 hours. For the sake of completeness, the evolution of $\max(F_{\mathrm{Rc}})$ is provided for all the individual lobes involving the robot multibody model in Appendix M.

From the experimental point of view, the stability is checked by the classical "step-cutting" strategy in which the axial depth of cut is kept constant during each cutting



(a) Longitudinal down milling (aluminium)

(b) Longitudinal up milling (aluminium)

(c) Transversal down milling (aluminium)

(d) Transversal up milling (aluminium)

(e) Longitudinal down milling (steel)

(f) Longitudinal up milling (steel)

(g) Transversal down milling (steel)

(h) Transversal up milling (steel)

Figure 7.1: Experimental and simulated stability charts using the Stäubli TX200 model with only the joint flexibility with a radial depth of cut a_e= 4 mm and standard cutting coefficients

test. In other words, one marker in any of the plots represents one milling pass in the material carried out with constant cutting conditions. Furthermore, the zig-zag or the one way strategy is chosen to achieve the face milling of the block at a constant axial depth of cut. The zig-zag strategy allows collecting cutting force and vibration signals for both down and up milling directions when completing one face milling at the desired axial depth of cut. The one way strategy is naturally deployed if the stability needs to be evaluated only along one direction, down or up milling. The investigated stability domain is the same as in the simulations for aluminium and steel. Tested cutting conditions are summarised in Table 7.2 for all milling situations and both materials. In order to spare some tool life, it is decided to carry out the milling experiments for one fixed axial depth of cut while varying the spindle speed i.e. the stability domain is swept from bottom to top. For one spindle speed, the axial depth of cut is increased till chatter appears. On site, the loss of stability is solely assessed by the produced sound.

From a global point of view, the presented stability lobe diagrams in Figure 7.1 do not seem to resemble the ones commonly found in machine tool. It is due to the robot flexible modes as opposed to the spindle modes which typically lead to chatter in conventional machine tool. As a reminder, the dynamics of the spindle is not included in any of the multibody models but still, the simulated stability limits appear to fairly correlate the experimental contour given by the cross-shaped markers.

For the *longitudinal pass in down milling in aluminium* (Figure 7.1a), the actual stability limit seems to split the stability domain diagonally from the lower-left corner to the upper-right corner. The simulated stability limit indicates a stability pocket around 6,000 rev/min. However, a milling experiment at 6,250 rev/min with an axial depth of cut of 3 mm invalidated the stability pocket. Another large discrepancy is observed between 12,000 rev/min and 16,000 rev/min in which the simulated stability limit underestimates the loss of stability. However, the shift in the colour mapping of $\max(F_{\mathrm{Rc}})$ appears to occur for higher depths of cut ($a_{\mathrm{p}} \approx$ 4 mm) at 15,000 rev/min which corroborates the experimental stability loss. The colour shift in the mapping of $\max(F_{\mathrm{Rc}})$ can be an interesting indicator as the colour transition is rather progressive. It gives a certain tolerance on the region where the stability is lost instead of relying on a clear limit provided by the red curve. Note that a time domain simulation of robotic milling at 14,000 rev/min and a_{p}= 3 mm (inside the pocket of instability) indeed exhibits cutting forces whose magnitudes contravene the stability criterion while keeping a steady oscillating behaviour.

For the *longitudinal pass in up milling in aluminium* (Figure 7.1b), the simulated stability limit correctly predicts the loss of stability below 8,000 rev/min. Beyond, the remaining of the simulated stability domain appears stable which is not correlated with the experimental tests. Around 14,000 rev/min and 19,000 rev/min, the simulator wrongly underestimates the loss of stability. Overall, the longitudinal pass in up milling in aluminium presented the largest discrepancies amongst all the milling situations (steel tests included). Author thinks that it is due to the fact that the phenomenon of built-up edge is not modelled in the robotic machining simulator. Indeed, the other reason for

Ω[rev/min] a_p[mm]	2500	5000	6250	7500	8750	10000	11250	12500	13750	15000	17500	20000	22500
7.4											$\times_d$	$\times_d$	$\times_d$
7											$\times_d$	$\times_d$	$\times$
6.5											$\times$	$\times$	$\times$
6.0											$\times$	$\times$	$\times$
5.0								$\times_u$	$\times$	$\times$	$\times$	$\times$	$\times$
4.5								$\times_d$	$\times_d$				
4.0					$\times_u$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
3.5		$\times_u$			$\times_u$	$\times$							
3.0	$\times_d$	$\times_u$	$\times_d$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
2.5	$\times_d$	$\times$		$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
2.0	$\times$	$\times$		$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
1.5	$\times$	$\times$		$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
1.0	$\times$	$\times$		$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
0.5	$\times$	$\times$		$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$

(a) Longitudinal passes in aluminium

Ω[rev/min] a_p[mm]	2500	5000	7500	8750	10000	11250	12500	13750	15000	17500	20000	22500
7.4							$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times$	$\times$
7						$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times$	$\times$
6.5					$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times$	$\times$
6.0					$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times$	$\times$	$\times$	$\times$
5.5					$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$
5.0					$\times_d$	$\times$	$\times$	$\times_d$	$\times$	$\times$	$\times$	$\times$
4.0					$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
3.5		$\times_d$	$\times_d$	$\times_d$								
3.0		$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
2.5		$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
2.0		$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
1.5	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
1.0	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
0.5	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$

(b) Transversal passes in aluminium

Ω[rev/min] a_p[mm]	2000	2500	3000	3500	4000	4500	5000	5500	6000	6500	7000	7500	8000	8500	9000	9500	10000
2.5								$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$
2.25								$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$
2.0								$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$
1.75						$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times_u$	$\times$
1.5						$\times_u$	$\times_u$	$\times_u$	$\times$	$\times_u$	$\times_u$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
1.25	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
1.0	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
0.5	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
0.1	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$

(c) Longitudinal passes in steel

Ω[rev/min] a_p[mm]	2000	2500	3000	3500	4000	4500	5000	5500	6000	6500	7000	7500	8000	8500	9000	9500	10000
2.5																$\times_d$	$\times_d$
2.25									$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$
2.0									$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$
1.75									$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times_d$	$\times$
1.5							$\times$	$\times_d$	$\times$	$\times_d$	$\times_d$	$\times_d$	$\times$	$\times$	$\times$	$\times$	$\times$
1.25				$\times_d$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
1.0			$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
0.75	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
0.25	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$

(d) Transversal passes in steel

Table 7.2: Tested spindle speeds and axial depths of cut to derive the stability charts (subscripts “d” down milling only, “u” up milling only and both directions otherwise)

stopping the increase in axial depth of cut when assessing the experimental stability loss was the appearance of built-up edge. The built-up edge phenomenon results from an accumulation of material against the rake face which can be prevented with the use of coolant (dry cutting in this work). The phenomenon only appeared in aluminium when achieving up milling passes. An illustration of unstable pass due to built-up edge is shown in Figure 7.2 for a longitudinal pass in up milling at 10,000 rev/min with an axial depth of cut a_{p}= 4 mm. In the figure, it is observed that material accumulated on both rake faces of the tool resulting in a very poor surface finish of the lateral profile. Consequently, the reason why the robotic milling simulator overestimates the stability limit beyond 8,000 rev/min is because a perfect cut, without built-up edge, is assumed in the simulator. Hence, the robot TCP wrongly less deviates (as shown in Table 6.4) and a larger stability is ensured.

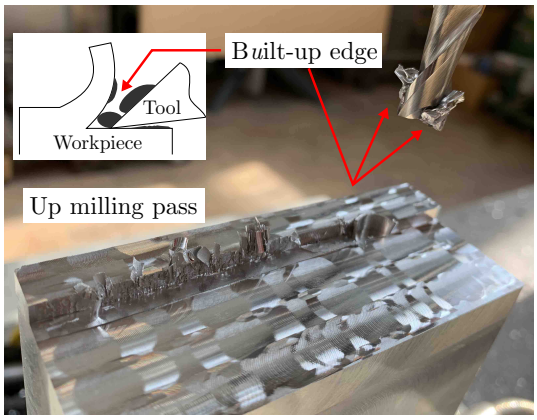


Figure 7.2: Built-up edge during the longitudinal pass in up milling in aluminium at 10,000 rev/min for a radial depth of cut a_{e}= 4 mm, an axial depth of cut a_{p}= 4 mm and a feed per tooth $\mathrm{f_z}$= 0.13 mm/tooth

For the *transversal pass in down milling in aluminium* (Figure 7.1c), the simulated stability limit correctly wraps the contour of the measured stable markers, despite the small discrepancy around 13,000 rev/min and the underestimate of stability loss beyond 20,000 rev/min. Note that the maximum level of force increased to 7,000 N. As a matter of fact, milling experiments following the transversal pass in down milling offered the largest stability domain. It can be explained by the fact that the major component of the cutting forces F_y (normal to the feed direction and in the cutting plane) is directed along the plane of the robot. It was indeed shown that FRF H_{xx} presented in Figure 5.17 was the most rigid amongst the three direct FRFs.

For the *transversal pass in up milling in aluminium* (Figure 7.1d), the robotic milling simulator correctly predicts the stability loss below 12,000 rev/min. Again, there is a wrong underestimate of stability loss around 15,000 rev/min but the colour shift of $\max(F_{\mathrm{Rc}})$ appears for higher axial depth of cut ($a_{\mathrm{p}} \approx$ 5 mm). Less built-up edges were generated when milling along the transversal direction in up milling which can explain the fair correlation between the simulated stability limit and the markers indicating an

unstable cut.

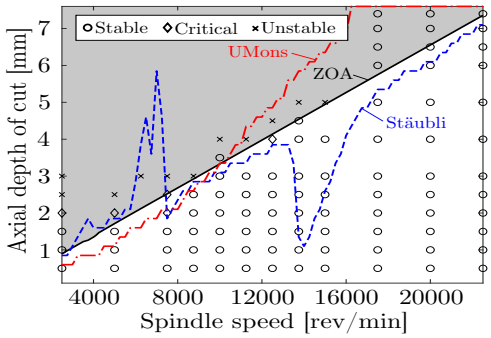
For the *longitudinal pass in down milling in steel* (Figure 7.1e), the simulated stability limit satisfactorily matches the contour delineated by the markers indicating unstable conditions over the whole range of tested spindle speeds, except around 3,000 rev/min where a stability pocket seems overridden.

For the *longitudinal pass in up milling in steel* (Figure 7.1f), the simulated stability limit correctly predicts the stability loss between 3,000 and 5,000 rev/min. Beyond and despite the underestimate of stability loss around 9,000 rev/min (slight colour shift for $\max(F_{\mathrm{Rc}})$), the robotic milling simulator indicates that milling operations are stable for the rest of the stability domain which is correlated with the experiments. In steel, no built-up edge was formed which yields a better correspondence. The investigated stability domain does not allow to appraise the milling stability beyond 6,000 rev/min since no chatter was experimentally detected.

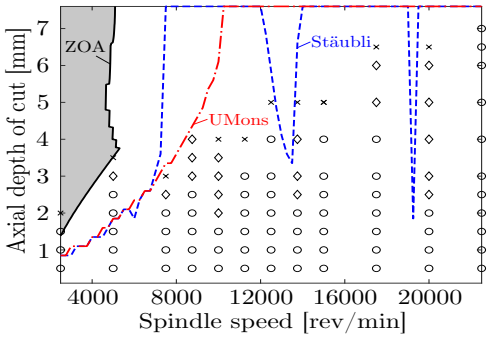
For the *transversal pass in down milling in steel* (Figure 7.1g), the experimental stable markers wrongly enter the predicted unstable zone between 3,000 and 5,000 rev/min. An overestimate of stability loss is predicted between 7,000 and 8,000 rev/min while an underestimate of instability is present around 9,000 rev/min.

For the *transversal pass in up milling in steel* (Figure 7.1h), the simulated stability limit reasonably follows the contour of the experimental unstable markers. Nevertheless, a large overestimate of stability is observed between 8,000 and 9,500 rev/min which is not corroborated with the milling experiments.

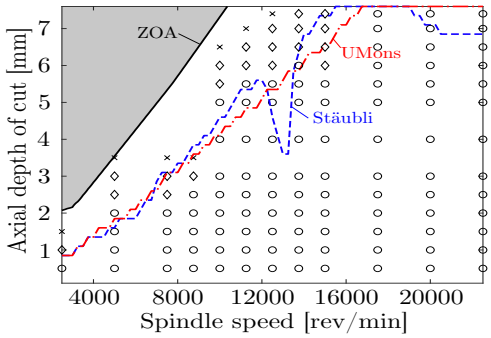
Using the UMons TX200 model with joint tri-axial flexibility, the eight stability lobe diagrams are generated for all the milling situations in down and up milling. They are superimposed to the ones computed with the Stäubli TX200 model in Figure 7.3. Overall, the simulated stability limits yielded by the UMons TX200 model are smooth compared with the ones derived with the Stäubli TX200 model. It means that a small misestimation in the joint elastic parameters can cause large discrepancies in the resulting stability charts. On the other hand, it also shows that as long as the inertial parameters of the robot arm are in the correct orders of magnitude, the associated elastic parameters must be well tuned to deliver the correct stability charts. The latter issue remains very challenging as it often leads to an underdetermined system to solve i.e. more elastic parameters to fit than measured frequency peaks. Nevertheless, it seems that the elastic parameters of the UMons TX200 model are luckily better estimated. With the UMons TX200 model, there is no underestimate of stability loss with a reversed peak (except in Figure 7.3f around 5,000 rev/min). On the contrary, the simulated stability limits often overestimate the contour shaped by the unstable markers (except in Figure 7.3g). The individual stability lobes for the UMons TX200 model can be found in Figure M.6 in Appendix M.



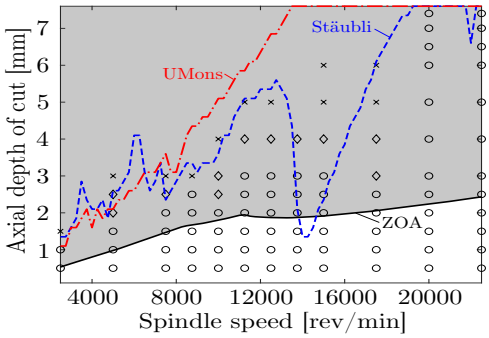
(a) Longitudinal down milling (aluminium)



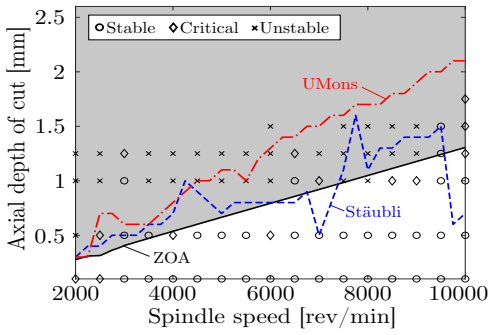
(b) Longitudinal up milling (aluminium)



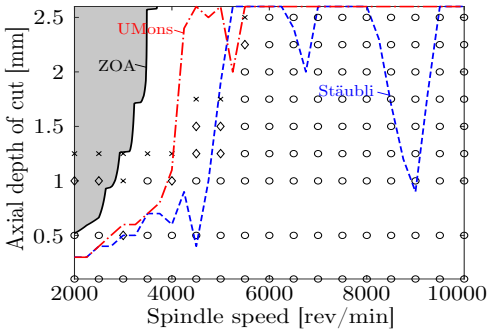
(c) Transversal down milling (aluminium)



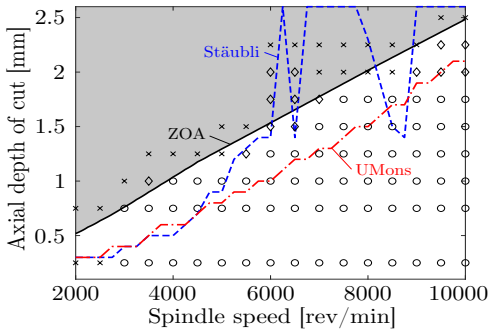
(d) Transversal up milling (aluminium)



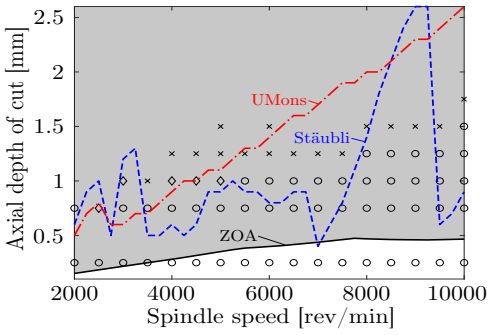
(e) Longitudinal down milling (steel)



(f) Longitudinal up milling (steel)



(g) Transversal down milling (steel)



(h) Transversal up milling (steel)

Figure 7.3: Experimental and simulated stability charts using the Stäubli and UMons TX200 multibody models with joint tri-axial flexibility compared to the 3D ZOA using the standard cutting coefficients

In Figure 7.3 are also shown the stability lobes generated by using the 3D digital zero-order approximation whose inputs are the measured FRFs over a bandwidth of 200 Hz in the milling posture. The 3D digital ZOA is retained as it yielded the best estimates of the stability limits **amongst the conventional methods** used in milling (Figure 7.17), relative to the experimental contour delimited by the cross-shaped markers. The computed lobes with the 3D digital ZOA therefore serve as a reference to compare the stability limits produced by the multibody models and the other conventional methods used in milling. Individual stability charts obtained by using the 3D digital ZOA are illustrated in Figure M.10 in Appendix M.

Compared with the multibody models, the 3D digital ZOA provides good correlations of the experimental stability contour for the longitudinal passes in down milling in aluminium (Figure 7.3a) and for the longitudinal and transversal passes in down milling in steel (Figures 7.3e and 7.3g). For the longitudinal passes in up milling (Figures 7.3b and 7.3f), the trends of stability limits are the same as for the multibody models i.e. a sharp rise of the stability limits for low spindle speeds followed by a large stability area. In aluminium, the ZOA contour is logically incorrect since the phenomenon of built-up edge is not included in the conventional method. In steel, the ZOA contour somewhat overestimates the experimental limit but its shape seems suitable. In order to generate the stability lobes for the transversal passes, the measured FRFs, which are the inputs of the method, are simply switched to match the feed direction i.e. H_{xx} becomes H_{yy}. For the transversal passes, only the ZOA contour in down milling and in steel seems to reasonably approach the experimental limit (Figure 7.3g). The other ZOA contours for the transversal passes either underestimate or overestimate the experimental stability limits (Figures 7.3c, 7.3d and 7.3h).

7.1.2 Analysis of arbitrary unstable cutting conditions

Clarifications are provided concerning the determination of unstable milling conditions from the measured cutting force and vibration signals. As mentioned earlier, the produced sound during the milling experiments was used as an early indicator to increase or not the axial depth of cut. Nevertheless, it is not sufficient to conclude if a pass chatters or not. Therefore, a more in-depth analysis of each pass was carried out by relying on the measured cutting force and vibration signals.

The stability of each experimental milling pass was assessed on the basis of three criteria:

1. The same stability criterion used in the simulations and explained in Section 4.5.5 is applied to the measured cutting forces. For each of the milling experiments, it is possible to compute the magnitude of the resultant of the cutting forces $F_{\mathrm{s,Rc}}$ if the mechanical system is completely rigid. The magnitude of the resultant of the measured cutting forces can also be computed and named $F_{\mathrm{exp,Rc}}$. The stability of the milling pass is ensured if

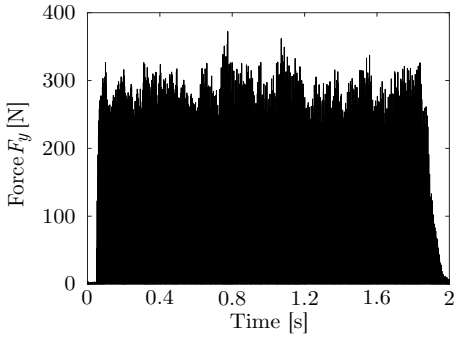
$$\frac{F_{\text{exp,Rc}}}{F_{\text{s,Rc}}} < 1.25 \rightarrow \text{stable.} \tag{7.1}$$

2. The second criterion is still related to the measured cutting forces and more precisely to the visual assessment of their time evolution. A sharp rise in the cutting forces reveals that self-excited vibrations are generated and produce a higher chip thickness. On the other hand, if the length of the milling pass, which was only 100 mm, is too short to witness the fully developed self-excited vibrations, the rise in the cutting forces can be used as an indicator of instability. Hence, if the level of the cutting forces remains constant throughout the whole length of the milling pass, the stability is ensured.

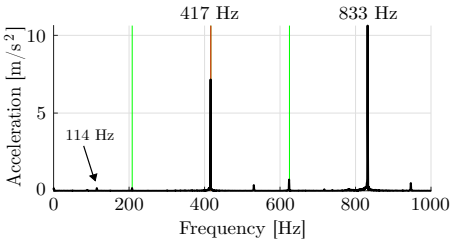
3. Vibration signals are classically examined by using FFT. If other peaks than the tooth passing and spindle frequencies and associated harmonics appear in the Fourier spectra, it undoubtedly means that the pass chatters. For some researchers investigating chatter stability with the measured signals from a microphone, chatter only occurs if the unstable frequency peaks (other than tooth passing and spindle frequencies and associated harmonics) are higher, in amplitude, than the ones corresponding to the tooth passing and spindle frequencies and associated harmonics. In this work, since an accelerometer was used and was placed far from the cutting area, a pass is said to chatter when unstable peaks significantly emerge. Therefore, stability is ensured if only the tooth passing and spindle frequencies and associated harmonics appear in the Fourier spectra (as observed in Figure 6.16b for a stable case). In addition, the use of the FFT allows evaluating the chatter frequency f_{c}.

The loss of stability for the experimental milling passes is stated when all the three above criteria are violated: a cross-shaped marker ($\times$) is drawn at the corresponding spindle speed and axial depth of cut in the stability chart. In contrast, when all the three criteria are respected, the experimental milling pass is said to be stable: a circle-shaped marker ($\circ$) is drawn in the stability chart. When only one or two of the criteria are respected, it is an indecisive case for which the experimental milling pass is said to slightly chatter: a triangle-shaped (or diamond-shaped $\diamond$) marker ($\triangle$) is drawn in the stability chart. Some researchers also describe the slight chatter condition as a critical case. An example of critical experimental milling pass is illustrated in Figure 7.4 for a spindle speed of 12,500 rev/min and an axial depth of cut a_{p}= 4 mm for the longitudinal pass in down milling in aluminium. The first stability criterion is respected since ratio $\frac{F_{\text{exp,Rc}}}{F_{\text{s,Rc}}}$= 0.91. In Figure 7.4a is reported the major component of the cutting forces F_y, which is perpendicular to the feed direction in the cutting plane. It is clear that the time evolution of F_y cannot be stated as constant. Indeed, the magnitude of F_y stands high and its time evolution is rather chaotic. The second stability criterion is therefore violated. On the other hand, regarding the FFT of the corresponding vibration signal in Figure 7.4b, very small peaks appear alongside the ones representing the tooth passing and spindle frequencies and associated harmonics. The third criterion is respected. Since two of the three criteria are respected, the experimental pass slightly chatters and a triangle-shaped marker is drawn

at 12,500 rev/min and a_p= 4 mm in the stability chart of the longitudinal pass in down milling in aluminium (Figure 7.1a).



(a) Measured force F_y



(b) FFT of the measured vibrations perpendicular to the feed direction

Figure 7.4: Critical cutting conditions in aluminium at 12,500 rev/min for a radial depth of cut a_e= 4 mm, an axial depth of cut a_p= 4 mm and a feed per tooth f_z= 0.13 mm/tooth for the longitudinal pass in down milling

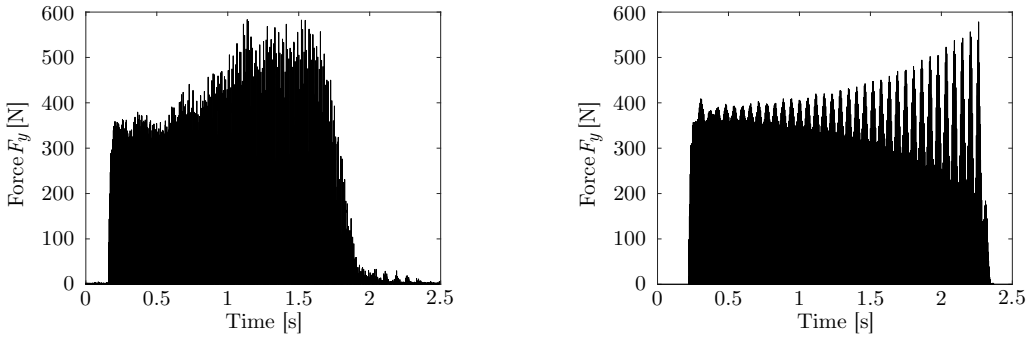
In contrast, two examples, one in aluminium and one in steel, are presented to illustrate clear chatter conditions from the measured cutting force and vibration signals. Measured signals are compared with the simulated counterparts.

In aluminium, the milling case treated in Subsection 6.4.1 is continued by increasing the axial depth of cut until unstable conditions are triggered. For the longitudinal pass in down milling at 11,250 rev/min, clear milling instability arises for an axial depth of cut a_p= 4 mm. As a matter of fact, a sharp rise in the major component of the measured cutting forces F_y is witnessed. In Figure 7.5a, it is observed that when the cutting tool enters the material the major component of the cutting forces F_y arrives at about 350 N and it reaches about 600 N at the end of the pass. Ratio $\frac{F_{\text{exp,Rc}}}{F_{\text{s,Rc}}}$ is therefore high and attains a value of 1.7.

Regarding the FFT of the corresponding measured vibrations perpendicular to the feed direction (Figure 7.6a), the Fourier spectrum clearly exhibits peaks different from the tooth passing and spindle frequencies and associated harmonics. Consequently, the three stability criteria are violated and chatter conditions are clearly recognised: a cross-shaped marker is drawn at 11,250 rev/min and a_p= 4 mm for the longitudinal pass in down milling in aluminium (Figure 7.1a).

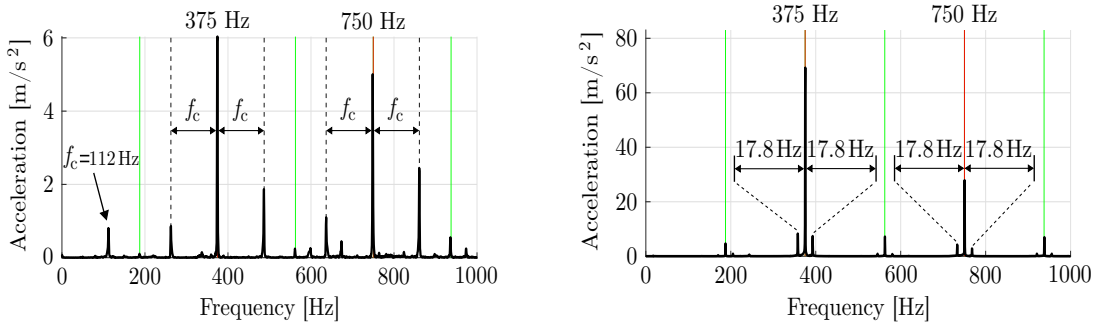
The Stäubli TX200 model with joint tri-axial flexibility is used to replicate the longitudinal pass in down milling in aluminium at 11,250 rev/min and a_p= 4 mm. The time evolution of the simulated cutting force F_y is presented in Figure 7.5b and also features a sharp rise in its magnitude. Measured and simulated levels of cutting force F_y are similar for the tested unstable milling conditions.

Discrepancies arise when comparing the Fourier spectra of the measured and simulated



(a) Measured force F_y (b) Simulated force F_y

Figure 7.5: Comparison of experimental and simulated (Stäubli TX200 model with joint tri-axial flexibility) force F_y for unstable cutting conditions in aluminium at 11,250 rev/min for a radial depth of cut a_e= 4 mm, an axial depth of cut a_p= 4 mm and a feed per tooth f_z= 0.13 mm/tooth (standard cutting coefficients) for the longitudinal pass in down milling



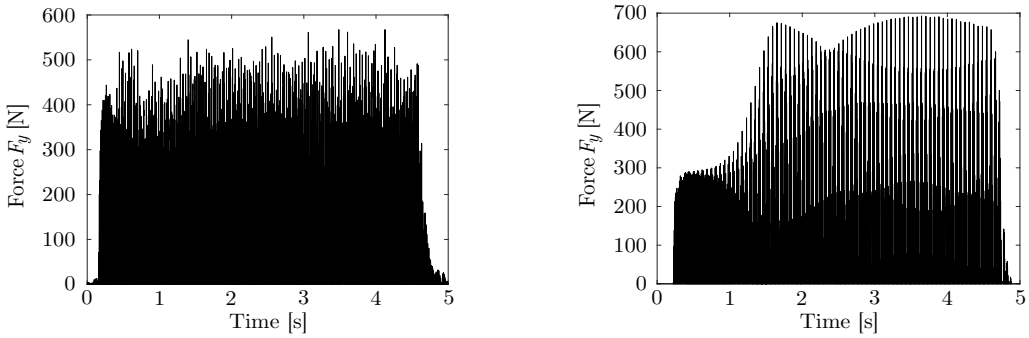
(a) FFT on measured vibrations (b) FFT on simulated vibrations

Figure 7.6: Comparison of experimental and simulated (Stäubli TX200 model with joint tri-axial flexibility) Fourier spectra of the vibration signal perpendicular to the feed direction for unstable cutting conditions in aluminium at 11,250 rev/min for a radial depth of cut a_e= 4 mm, an axial depth of cut a_p= 4 mm and a feed per tooth f_z= 0.13 mm/tooth (standard cutting coefficients) for the longitudinal pass in down milling

vibration signals perpendicular to the feed direction. FFT on the simulated vibrations is depicted in Figure 7.6b and also reveals frequency peaks other than the tooth passing and spindle frequencies and associated harmonics. It reveals that the simulated pass clearly chatters as for the actual milling experiment. However, the chatter frequency f_c does not match between the one observed in the Fourier spectrum of the measured vibrations and the one that appears in the Fourier spectrum of the simulated vibrations. In the Fourier spectrum of the measured vibrations, a chatter frequency $f_{exp,c}$ of 112 Hz is noticed under the form of modulations around the tooth passing frequency (375 Hz) and harmonics but also as a sole peak emerging at 112 Hz. It is not the case in the Fourier spectrum of the simulated vibrations which exhibits modulations of $f_{s,c}$= 17.8 Hz around the tooth passing frequency and associated harmonics. From the experimental modal analysis presented in Chapter 5, it is known that chatter frequency $f_{exp,c}$ of 112 Hz is close to a robot mode around 115 Hz whose mode shape involves wrist motions (eleventh mode). In

the robotic milling simulator, chatter frequency $f_{s,c}$ of 17.8 Hz corresponds to the third robot mode ($f_{s,3}$= 17.5 Hz) whose mode shape implies the onward motion of the TCP. Even though an eleventh mode is present in the robotic milling simulator around 100 Hz, it does not trigger the instability at 11,250 rev/min and a_p= 4 mm for the simulated longitudinal pass in down milling in aluminium. The reason is that the simulated eleventh mode shape, still involving wrist motions, is wrongly fitted. Nevertheless, despite that the chatter frequencies are not the same, the simulated stability limit is still close to the contour suggested by the cross-shaped markers indicating unstable conditions for the longitudinal pass in down milling in aluminium in Figure 7.1a. Note that the observed chatter frequency $f_{exp,c}$ of 112 Hz issued from the measured vibrations would suggest that all the modes below 200 Hz must be updated for the robot multibody model to correctly predict the stability limit. However, it will be shown in the next subsection that the 3D digital ZOA, which reasonably predicts the experimental stability limits in Figure 7.3, also predicts a chatter frequency close to the simulated one.

Second example pursues the milling case discussed in Subsection 6.4.7 concerning the longitudinal pass in down milling in steel at 5,000 rev/min. Analogously to the aluminium test, the axial depth of cut was increased till unstable conditions were provoked during the milling experiments. Clear unstable conditions are detected for an axial depth of cut a_p= 1.25 mm. As observed in Figure 7.7a, the time evolution of measured cutting force F_y exhibits a chaotic behaviour. On the other hand, besides the chaotic behaviour, simulated cutting force F_y in Figure 7.7b reveals a sharp rise in its magnitude matching the level developed by the measured counterpart. From the point of view of the cutting forces, instability is well captured by the robotic milling simulator.



(a) Measured force F_y (b) Simulated force F_y

Figure 7.7: Comparison of experimental and simulated (Stäubli TX200 model with joint tri-axial flexibility) force F_y for unstable cutting conditions in steel at 5,000 rev/min for a radial depth of cut a_e= 4 mm, an axial depth of cut a_p= 1.25 mm and a feed per tooth f_z= 0.09 mm/tooth (standard cutting coefficients) for the longitudinal pass in down milling

Regarding the Fourier spectrum derived from the measured vibrations perpendicular to the feed direction, frequency peaks different from the tooth passing (250 Hz) and spindle frequencies and associated harmonics are noticed in Figure 7.8. One clear frequency peak stands out around 355 Hz, which does not correspond to one of the harmonics of

the spindle frequency (close-up in Figure 7.8). The chatter frequency can therefore be computed from the nearest harmonic of the spindle frequency around 333 Hz and is therefore equal to $f_{\text{exp,c}}$= 24 Hz. Smaller modulations in amplitude of chatter frequency $f_{\text{exp,c}}$= 24 Hz are also witnessed around the tooth passing frequency (250 Hz). From the results of the experimental modal analysis presented in Chapter 5, it is inferred that chatter frequency of $f_{\text{exp,c}}$= 24 Hz is close to the fifth experimental mode shape which portrayed a robot motion involving a rotation perpendicular to the motion axis of the first joint. It is coherent that this mode gets excited during a longitudinal pass in down milling since the major component of the cutting force F_y is parallel to the direction of deflection of the mode. Indeed, high flexible frequency peaks between 20 Hz and 24 Hz were remarked in direct FRF H_{yy} shown in Figure 5.18.

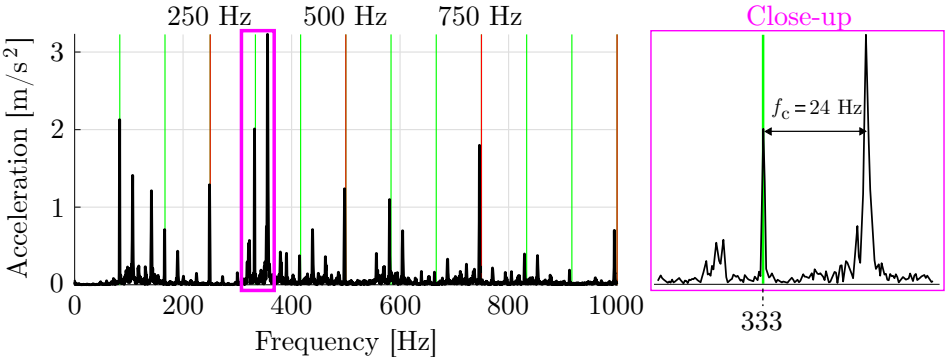


Figure 7.8: FFT of the measured vibrations perpendicular to the feed direction for unstable cutting conditions in steel at 5,000 rev/min for a radial depth of cut a_{e}= 4 mm, an axial depth of cut a_{p}= 1.25 mm and a feed per tooth f_z= 0.09 mm/tooth for the longitudinal pass in down milling

Still using the Stäubli TX200 model with joint tri-axial flexibility, FFT is applied to the simulated vibration signals perpendicular to the feed direction. The resulting Fourier spectrum is depicted in Figure 7.9. It again exhibits frequency peaks other than the tooth passing and spindle frequencies and associated harmonics, thus confirming that the simulated pass chatters as in the milling experiment. However, the simulated chatter frequency $f_{\text{s,c}}$= 16.6 Hz is still not the same as in the Fourier spectrum derived from the measured vibrations ($f_{\text{exp,c}}$= 24 Hz). The simulated chatter frequency at 16.6 Hz is again close to the third fitted mode of the Stäubli TX200 model at 17.5 Hz involving the onward motion of the TCP. Nevertheless, the simulated stability limit for the longitudinal pass in down milling in steel was close to the contour suggested by the cross-shaped markers in Figure 7.1e.

Such stability analysis for particular cutting conditions can obviously be repeated to all the cross-shaped markers indicating instability in the milling experiments. The two examples that were treated above gave some insights regarding the correlations between the measured and simulated signals in unstable conditions. Similar trends are expected for the other unstable milling experiments at various degrees of correlation.

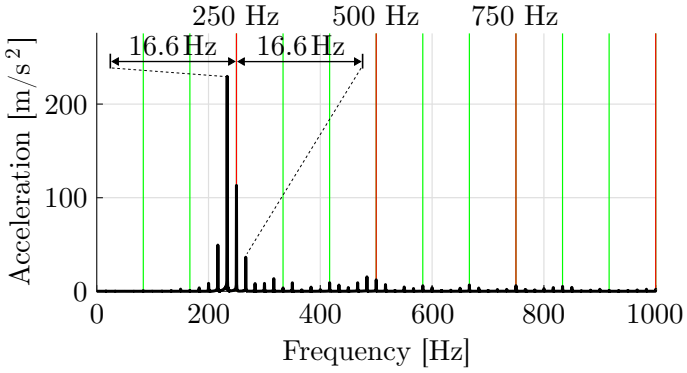
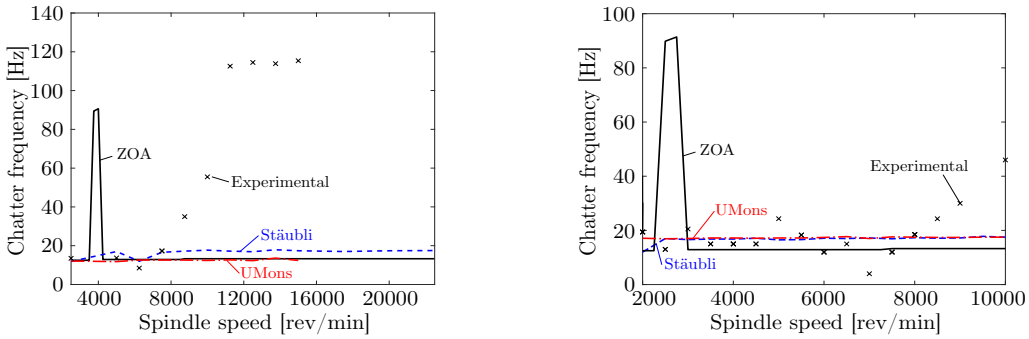


Figure 7.9: FFT of the simulated vibrations perpendicular to the feed direction for unstable cutting conditions in steel at 5,000 rev/min for a radial depth of cut a_e= 4 mm, an axial depth of cut a_p= 1.25 mm and a feed per tooth f_z= 0.09 mm/tooth for the longitudinal pass in down milling using the Stäubli TX200 model with joint tri-axial flexibility (standard cutting coefficients)

7.1.3 Chatter frequency analysis

From the analyses of the two arbitrary examples in unstable conditions, it is noticed that the mismatch is mainly due to a wrong prediction of the chatter frequency. The analysis of the chatter frequency is carried out for the longitudinal passes in down milling in aluminium and in steel since their stability charts gather the largest number of cross-shaped markers. Deduced chatter frequencies from the Fourier spectra on the measured vibrations are also noted with cross-shaped markers in Figure 7.10. It was witnessed from Figures 7.3a and 7.3e that the 3D digital ZOA reasonably correlated the contour delineated by the cross-shaped markers for the longitudinal passes in down-milling. The chatter frequency is easily retrieved using the 3D digital ZOA at the loss of stability thanks to its numerical implementation. Resulting plots are depicted in Figure 7.10a for aluminium and in Figure 7.10b for steel. They present the evolution of the chatter frequency with respect to the tested spindle speeds. In aluminium, the chatter frequency yielded by the 3D digital ZOA remains constant at 12.9 Hz except around 4,000 rev/min where it reaches 90.6 Hz. In steel, the same behaviour is remarked with a constant chatter frequency of 12.9 Hz except around 2,500 rev/min where it attains 91.4 Hz. From the results of the experimental modal analysis explained in Chapter 5, it is known that mode at around 12.9 Hz corresponds to a rotation of the second joint around its motion axis and mode at around 90 Hz is associated with a deflection around the motion axis of the sixth joint.

It is also possible to retrieve the chatter frequencies by using the robot multibody models by first specifying unstable cutting conditions from the simulated stability charts and then, analysing the Fourier spectra of the resulting vibrations. Hence, chatter frequencies are computed for the longitudinal passes in down milling in aluminium and in steel for the Stäubli and UMons multibody models with joint tri-axial flexibility. Their evolutions, which are rather constant, are superimposed to the ones from the 3D digital ZOA in Figure 7.10.



(a) Aluminium (b) Steel

Figure 7.10: Experimental and simulated chatter frequency charts using the 3D digital ZOA and the Stäubli and UMons multibody models with joint tri-axial flexibility for the longitudinal pass in down milling

In aluminium, a close match is observed between the UMons TX200 model and the 3D digital ZOA for the whole range of spindle speeds, except at around 4,000 rev/min. On the other hand, the Stäubli TX200 model overestimates the chatter frequency at around 17.8 Hz (third robot mode). Another observation lies in the fact that the non smooth stability limit simulated in Figure 7.1a for the Stäubli TX200 model does not generate variations in the chatter frequencies. The simulated chatter frequency for the Stäubli TX200 model remains rather constant. Relative to the experimental chatter frequencies, a sufficient correlation is only observed below 7,000 rev/min for the UMons model and 3D digital ZOA. Beyond, experimental chatter frequencies can attain high values around 112 Hz (eleventh robot mode), as presented in Figures 7.4b and 7.6a. The experimental chatter frequency at 112 Hz is not captured by any of the models which nonetheless provide a satisfying stability limit relatively to the experimental stability contour in Figure 7.3a.

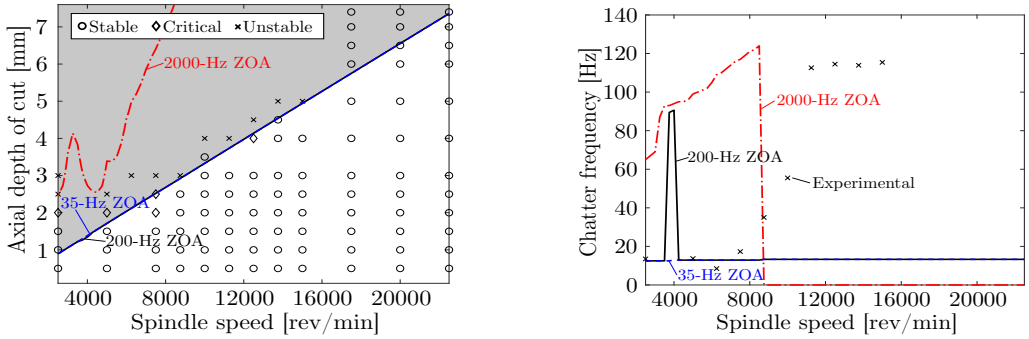
In steel, the Stäubli and UMons models converge towards the same chatter frequency around 17 Hz for the whole range of spindle speeds. It does not match the chatter frequency predicted by the 3D digital ZOA at 12.9 Hz, which does not correspond either to the measurements. Predicted chatter frequencies for the three models in steel are still close to the experimental values (around 15 Hz). Regarding the experimental chatter frequencies, as opposed to aluminium, they are rather concentrated in the lower-frequency range below 50 Hz. Most of the experimental chatter frequencies seem to oscillate between modes located below 35 Hz, which represent the structural modes of the robot. Even if the cross-shaped markers are not exactly on the predicted chatter frequencies, a better correlation is still noticed in steel in Figure 7.10b.

The discrepancy at around 4,000 rev/min in the chatter frequency chart in aluminium shown in Figure 7.10a is studied by considering various frequency bandwidths in the input measured FRFs fed to the 3D digital ZOA. A similar study can be conducted in steel. Three frequency bandwidths are considered and the utilised 3D digital ZOA is named after the selected bandwidth in Figure 7.11 such as

1. 35-Hz ZOA: the inputs of the method are the measured FRFs over a bandwidth of 200 Hz but only the range below 35 Hz is actually fed to the 3D digital ZOA.

2. 200-Hz ZOA: the inputs of the method are the measured FRFs over a bandwidth of 200 Hz, without truncation. It is the reference bandwidth used in this work for all ZOA methods.

3. 2000-Hz ZOA: the inputs of the method are the measured FRFs over a bandwidth of 4000 Hz but only the range below 2000 Hz is actually fed to the 3D digital ZOA. Hence, the frequency resolution was $\Delta f = 1$ Hz (instead of $\Delta f = 0.39$ Hz). The corresponding direct FRFs in receptance format were presented in Figure 5.24.



(a) Stability lobe diagram (b) Chatter frequency

Figure 7.11: Stability and chatter frequency charts using the 3D digital ZOA for the longitudinal pass in down milling in aluminium by varying the considered frequency bandwidth in the measured FRFs: over 35 Hz, 200 Hz and 2000 Hz

In Figure 7.11 are displayed the stability charts and the associated chatter frequency evolutions for the three tested frequency bandwidths. It is immediately noticed that the ZOA method is dependent on the frequency resolution of the input measured FRFs. Hence, the 2000-Hz 3D digital ZOA overestimates the stability limit and predicts chatter frequencies over 100 Hz between 6,000 and 9,000 rev/min, which is not validated experimentally (Figure 7.11b). The 200-Hz 3D digital ZOA produces the results illustrated in Figures 7.3a and 7.10a which were already commented. Interestingly, the stability limit predicted by the 35-Hz 3D digital ZOA is very similar to the one generated by the 200-Hz 3D digital ZOA. Using the 35-Hz 3D digital ZOA, the chatter frequency cannot rise beyond 35 Hz. Instead, the chatter frequency comes down to 12.9 Hz at around 4,000 rev/min. The chatter frequency of the 35-Hz 3D digital ZOA now becomes closer to the ones predicted by the multibody models for spindles speeds below 7,000 rev/min. Consequently, as exposed in Chapter 5, it also suggests that updating robot modes below 35 Hz is sufficient to predict the stability limits (Figure 7.11a), at the expense of correlation in the chatter frequency i.e. the simulated stability contour might be close to the experimental one without necessarily matching the measured chatter frequencies.

The study regarding the chatter frequency can be easily transposed to the other milling situations. Since the excited modes belong to the structural modes of the robot, it is

not surprising that other modes get excited in the other milling situations. Same kinds of mismatch are therefore noticed for the other milling situations and are therefore not reported.

7.1.4 Comparison of unstable machined lateral profiles

Machined lateral profiles resulting from unstable milling conditions are shown in Figure 7.12 for the longitudinal pass in down milling in aluminium and in steel. To highlight the wavy profiles resulting from the self-excited vibrations, the presented limit axial depths of cut are increased by 1 mm. Hence, the aluminium plate is machined at 11,250 rev/min with a_p= 5 mm and the steel plate is machined at 5,000 rev/min with a_p = 2.25 mm, using a constant feed per tooth and a radial depth of cut a_e= 4 mm.

(a) Aluminium at 11,250 rev/min and a_p= 5 mm (f_z= 0.13 mm/tooth)

(b) Steel at 5,000 rev/min and a_p= 2.25 mm (f_z= 0.09 mm/tooth)

Figure 7.12: Close-up of the lateral faces for the longitudinal pass in down milling after unstable conditions with a radial depth of cut a_e= 4 mm

In Figure 7.12a for the unstable pass in aluminium, it is clearly observed that the pass progressively chatters as the cutting tool progresses in the material. The cutting tool enters the material on the right-hand side leaving a straight lateral face (almost a mirror finish). Imprinted waves are recognised on the left-hand side of the figure. They progressively grow in width as the tool moves forward in the material. The trend is corroborated with the progressive increase in cutting forces shown in Figure 7.5a. The observed behaviour is different in steel as pictured in Figure 7.12b. As suggested by the time evolution of the cutting forces in Figure 7.7a, chatter occurs directly as soon as the tool enters the material. The level of instability is preserved throughout the whole length of the pass. Hence, the machined lateral profile in steel after an unstable pass outlines imprinted waves with a rather constant width.

The milling pass in aluminium that generated the lateral profile shown in Figure 7.12a was simulated with the robotic milling simulator. Naturally, the simulation of the machined lateral profile cannot be carried out via frequency methods i.e. the ZOA. The simulated lateral profile is presented in Figure 7.13 with the blue curve in the right drawing. Of course, roughness is not assessed for an unstable pass. As for the actual pass, the width of

the imprinted waves progressively grows as the cutting tool moves forward in the material. A large deflection of 0.4 mm is simulated at the end of the pass whereas the measured counterpart reaches 0.5 mm. The virtual machined profile in steel (not shown) outlines a rather constant width of imprinted waves as suggested by the time evolution of the simulated cutting forces in Figure 7.7b.

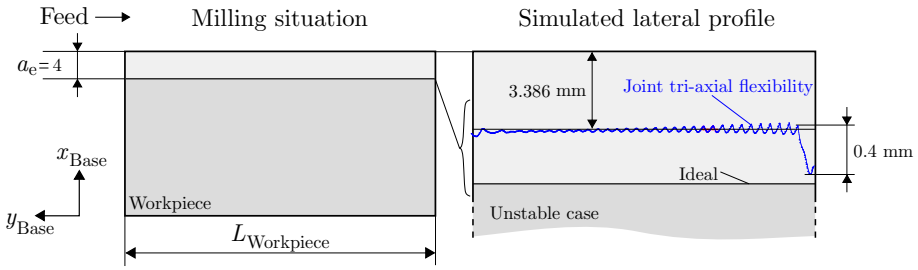
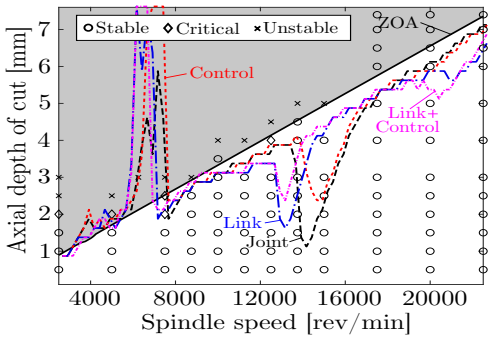


Figure 7.13: Shape of the virtual workpiece for unstable cutting conditions in aluminium at 11,250 rev/min for a radial depth of cut a_e= 4 mm, an axial depth of cut a_p= 5 mm and a feed per tooth f_z= 0.13 mm/tooth for the longitudinal pass in down milling using the Stäubli TX200 model with joint tri-axial flexibility and the standard cutting coefficients

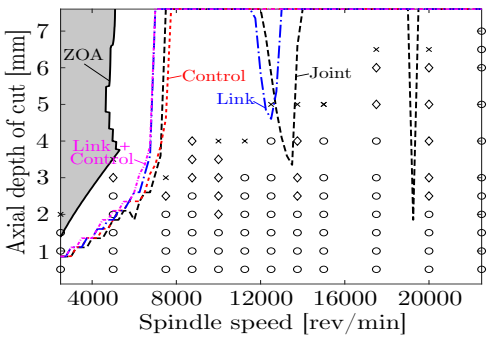
7.2 Impact of various flexibility sources on the lobes

The influence of the various flexibility sources is appraised on the shape of the stability limits. Namely, the flexibility of the controller, the flexibility of the links or the combination of both are appended to the Stäubli TX200 model with joint tri-axial flexibility. For each model variant, the stability chart of all milling situations in aluminium or in steel is drawn. The superimpositions of the resulting stability limits assessing all flexibility effects for all milling situations are presented in Figure 7.14. Note that the individual stability limit for each flexibility effect and for each milling situation is illustrated in Appendix M along with its mapping of the maximum magnitude of resultant cutting force F_{Rc}. Thus, the individual stability limits assessing the flexibility of the controller are depicted in Figure M.1, the ones dealing with the flexibility of the links are shown in Figure M.2 and their combined effects are provided in Figure M.3.

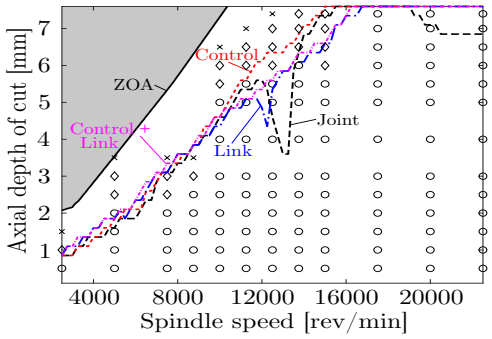
In Figure 7.14, the stability limits computed from the 3D digital ZOA are also shown as they constitute the reference solutions for the stability methods found in machine tool. Overall, it is observed that the introduction of additional flexibilities brings damping and slightly shifts the positioning of the lobe (e.g. inverted stability pocket around 15,000 rev/min in Figure 7.14a) relatively to the spindle speed axis. As a matter of fact, the stable area becomes wider and the stability pockets lightly move towards the left. The additional flexibilities do not yield the apparition of a new stability pocket in the stability charts as long as the flexible links and the controller are realistically tuned. Also remember that the appearance of the stability pockets computed from the Stäubli TX200 model with joint tri-axial flexibility was due to an inadequate tuning of the joint elastic parameters as shown by comparison of the sta-



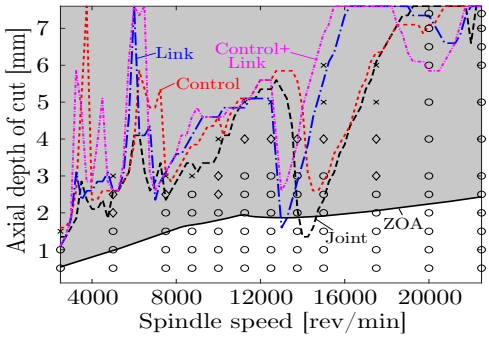
(a) Longitudinal down milling (aluminium)



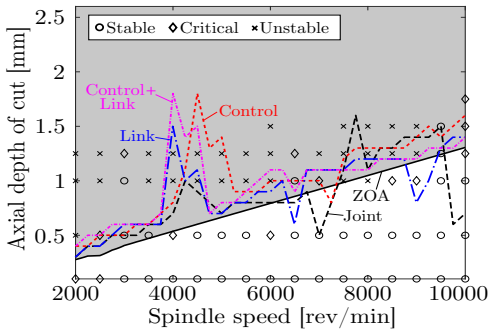
(b) Longitudinal up milling (aluminium)



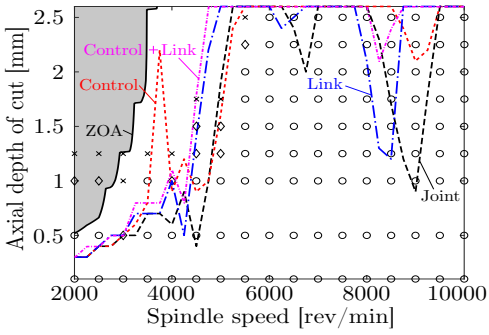
(c) Transversal down milling (aluminium)



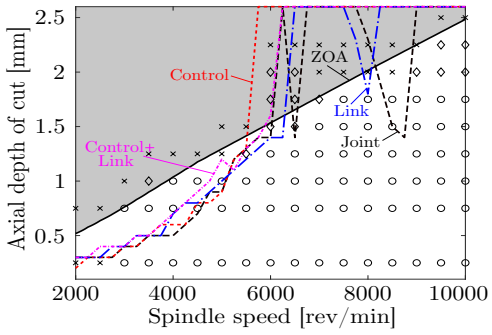
(d) Transversal up milling (aluminium)



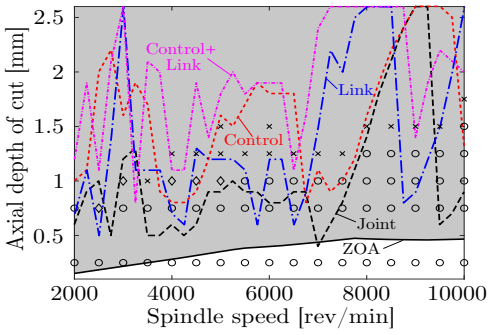
(e) Longitudinal down milling (steel)



(f) Longitudinal up milling (steel)



(g) Transversal down milling (steel)



(h) Transversal up milling (steel)

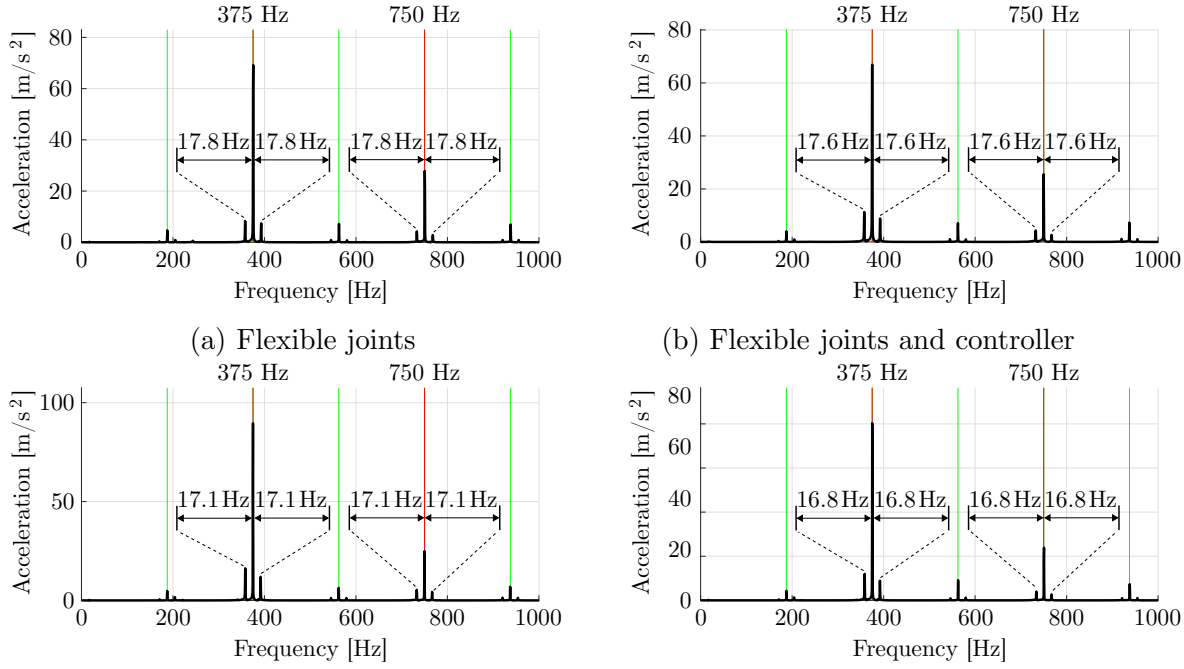
Figure 7.14: Experimental and simulated stability charts using the Stäubli TX200 multi-body model with joint tri-axial flexibility, flexible joints and controller, flexible joints and links, flexible joints and links and controller with the standard cutting coefficients

bility limits generated by the UMons TX200 model (smooth stability limits) in Figure 7.3.

As for instance, for the longitudinal pass in down milling in aluminium (Figure 7.14a), the effect of the controller flexibility (red curve) damps out the prominence of the "inverted" stability pocket at around 15,000 rev/min and thus expands the stability area. Same phenomenon appears to the wrongly predicted stability pocket at around 6,000 rev/min. Concerning the effect of the flexible links (blue curve), it mainly shifts the positioning of the stability pocket towards the left and also adds some damping. When both flexibilities are appended to the joint tri-axial flexibility model (magenta curve), the stability area is still larger due to the increased damping but the slight shift of stability pocket is less evident. A similar trend is observed in the other milling situations in aluminium and in steel. In particular, these effects are strongly noticed for the longitudinal passes in up milling in aluminium and in steel (Figures 7.14b and 7.14f). The combined effects of the link and controller flexibilities damp out the wrongly predicted stability pockets. Hence, smooth stability limits are generated (magenta curves) as for the UMons TX200 model with joint tri-axial flexibility (Figures 7.3b and 7.3f). It should also be pointed out that the overall shape of the stability limits generated by the joint tri-axial flexibility model is preserved even with the additional flexibilities of the links or the controller.

In order to further investigate the shift of stability pockets when additional flexibilities are introduced, the Fourier spectra are compared for one combination of spindle speed and axial depth of cut generating instability for all the multibody model variants. The selected cutting conditions refer to the longitudinal pass in down milling in aluminium at 11,250 rev/min and with an axial depth of cut a_{p}= 4 mm. The Fourier spectra assessing the effects of each flexibility source are depicted in Figure 7.15. The Fourier spectrum computed on the basis of the Stäubli TX200 model with joint tri-axial flexibility was already presented and commented using Figure 7.6b (Figure 7.15a). In Figure 7.15b is evaluated the effect of the controller flexibility on the chatter frequency f_{c} i.e. the frequency modulations around the peaks representing the tooth passing frequency (375 Hz) and harmonics. The controller flexibility lightly reduces the chatter frequency from 17.8 Hz to 17.6 Hz. The largest chatter frequency shift is noticed when the link flexibility is appended to the joint tri-axial flexibility model. The chatter frequency drops from 17.8 Hz to 17.1 Hz. This shift was reflected in the corresponding stability chart in Figure 7.14a (e.g. shift towards the left of the inverted stability pocket around 15,000 rev/min). When both flexibilities are appended to the reference multibody model, the chatter frequency sharply decreases to 16.8 Hz. Note that the same robot structural mode remains responsible for the instability: the third robot mode identified at 17.5 Hz whose mode shape was an onward motion of the TCP. Overall, the distribution of the natural frequencies and damping ratios of the robot is reflected in the shape of the stability limits. In fact, when a flexibility in series is appended to the model, such as the controller flexibility, it naturally lowers the natural frequencies of the structural modes and affects the shape of the stability chart.

Important considerations must be paid to the computing time when additional flex-



(a) Flexible joints

(b) Flexible joints and controller

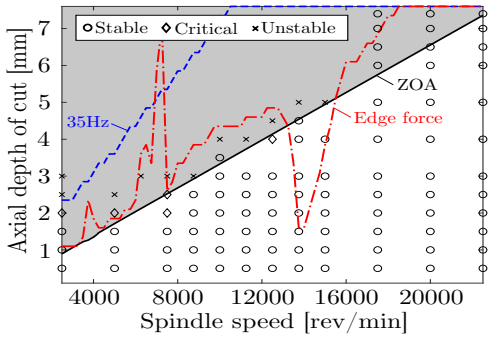
(c) Flexible joints and links

(d) Flexible joints, links and controller

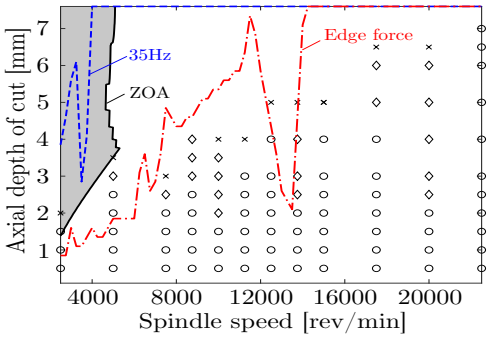
Figure 7.15: FFT of the simulated vibrations perpendicular to the feed direction for unstable cutting conditions in aluminium at 11,250 rev/min for a radial depth of cut a_e= 4 mm, an axial depth of cut a_p= 4 mm and a feed per tooth f_z= 0.13 mm/tooth for the longitudinal pass in down milling for assessing the various flexibility sources of the Stäubli TX200 model (standard cutting coefficients)

ibilities are added to the multibody model with joint tri-axial flexibility. With the current implementation (discretisation of the stability domain), it takes about six days to generate one stability chart in aluminium considering the joint and link flexibilities if the whole stability domain is swept for the saving of the maximum magnitude of the resultant cutting force F_{Rc} (Figure M.2). Three days were necessary to deliver the same stability charts in aluminium when only the flexible joints were incorporated in the multibody model. The controller flexibility further burdens the computing time. The computing time of one stability chart in aluminium with the appended controller flexibility rises to 12 days (Figure M.1). It is naturally even worse when all the flexibilities are included for which the computing time rises to 16 days (Figure M.3). Less time was needed in steel since the number of points to evaluate in the stability domain was smaller. From an industrial point of view, the consideration of additional flexibilities (links and controller) must therefore result from a thoughtful trade-off between the computing time and the cost saving on expensive parts.

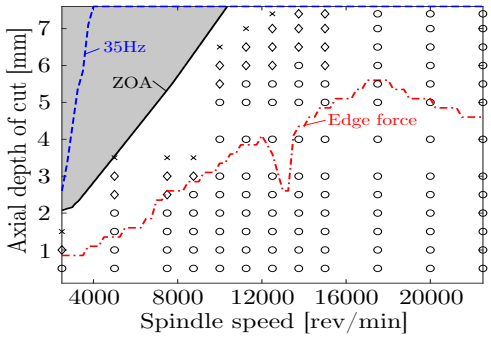
Beside the link and controller flexibilities, two other variants of the Stäubli TX200 model with joint tri-axial flexibility are discussed regarding their generated stability charts. The first discussed model is the Stäubli TX200 model with joint tri-axial flexibility for which the elastic parameters result from the sole implementation of the first identification step (curve fitting of the tool tip FRFs) on a reduced frequency



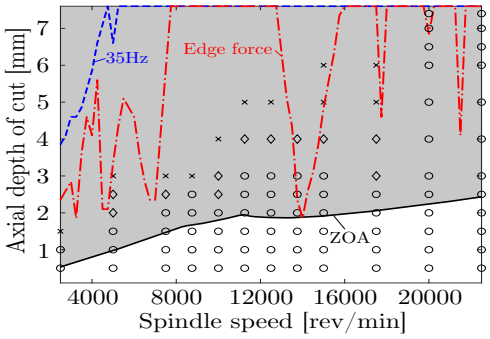
(a) Longitudinal down milling (aluminium)



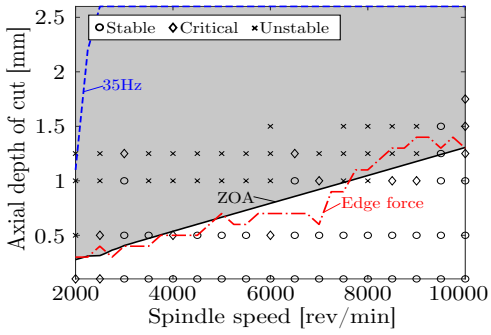
(b) Longitudinal up milling (aluminium)



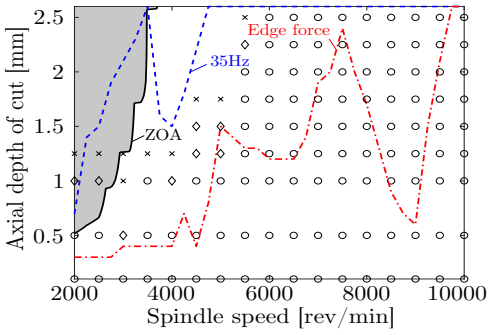
(c) Transversal down milling (aluminium)



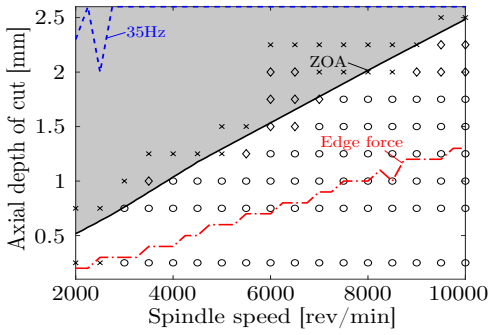
(d) Transversal up milling (aluminium)



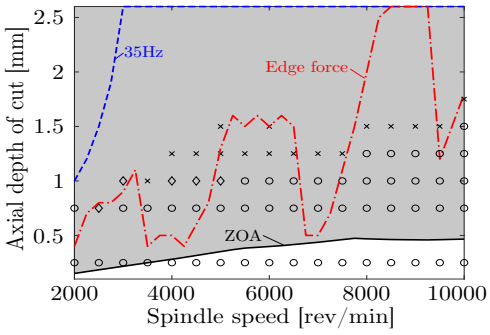
(e) Longitudinal down milling (steel)



(f) Longitudinal up milling (steel)



(g) Transversal down milling (steel)



(h) Transversal up milling (steel)

Figure 7.16: Experimental and simulated stability charts using the Stäubli TX200 multi-body model with joint tri-axial flexibility with a FRF fitting over 35 Hz (standard cutting coefficients) and with a fitting over 100 Hz with edge force coefficients

bandwidth of 35 Hz. The resulting tool tip FRFs were compared to the measured ones in Figure 5.47 and exhibited a strong correlation in the frequency peaks. Nevertheless, the damping of some modes seemed overestimated as dull peaks were fitted to the measured ones, which are mainly sharp. The effect of an overestimated joint damping on the stability limits is observed in Figure 7.16. It is clear that overestimation in the joint damping increases the simulated stability limits to higher axial depths of cut ($\max(F_{\mathrm{Rc}})$ also sharply decreases). The observation is populated to all milling situations for aluminium and steel. It means that in the identification procedure of the joint elastic parameters, the fitting of the damping ratios is as much as important as the fitting of the natural frequencies. Matching the correct damping ratios is even more complicated since they are dependent on the robot posture and the controller action.

The second discussed variant is the Stäubli TX200 model with joint tri-axial flexibility (elastic parameters identified over 100 Hz found in Table 5.17) for which the cutting forces applied to its TCP include the edge force effects. The generated stability charts are also plotted in Figure 7.16 for aluminium and steel. Compared with the stability charts obtained with the same model and the standard cutting coefficients (Figure 7.1), identifying common trends is somewhat tricky. For the longitudinal passes in down milling in aluminium and in steel (Figures 7.16a and 7.16e), it appears that the stability limits are very similar to the ones derived with the standard cutting coefficients. It should be highlighted that the stability limit with the edge forces is almost the same as the one yielded by the 3D digital ZOA for the longitudinal pass in down milling in steel (Figure 7.16e). For the transversal passes in down milling in aluminium and in steel, the edge forces lower the stability area and the resulting limits are thus far under the experimental contour (Figures 7.16c and 7.16g). The friction modelled with the edge forces grants most of its effect for the longitudinal passes in up milling, which is the expected behaviour (Figures 7.16b and 7.16f): the resulting stability limits reduce the stable zone. Interpretation is less obvious for the transversal passes in up milling in aluminium and in steel (Figures 7.16d and 7.16h): beside the fact that the stability limits are non smooth due to an inadequate fitting of the joint elastic parameters, it seems that edge forces enlarge the stable area. Overall, it can be stated that the inclusion of the edge force coefficients in the cutting force model is not necessary to generate coherent stability limits with respect to the experimental contour. Note that regarding the computing time, as no additional flexibility is appended to the multibody model with joint tri-axial flexibility, it also takes about three days to generate one stability chart in aluminium with the backup of the maximum magnitude of the resultant cutting force F_{Rc}. Individual lobes using the Stäubli TX200 model fitted over a frequency bandwidth of 35 Hz are shown in Figure M.5 and the ones produced with the edge force effects are presented in Figure M.4.

7.3 Stability lobes by using conventional methods

The efficiency of the conventional stability methods used in machine tool is assessed in terms of prediction of the experimental stability contours and in terms of computing time. As anticipated, the 3D digital ZOA yielded the highest correlations with the experimental contour amongst the tested conventional methods. Hence, it was already

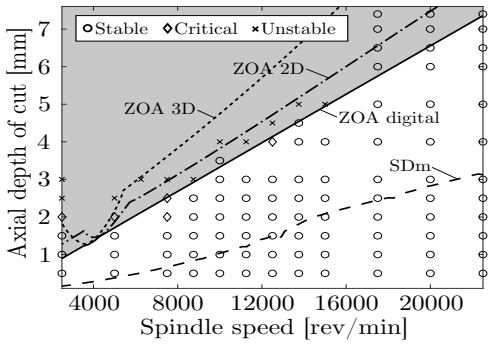
compared with the stability charts generated by the Stäubli and UMons models in Figure 7.3. Instead, the stability charts generated by the conventional stability methods are compared together in Figure 7.17. In the exposed sub-figures, *ZOA 2D* refers to the 2D analytical ZOA, *ZOA 3D* corresponds to the 3D analytical ZOA and *ZOA digital* is the 3D digital ZOA. As before, individual stability lobe diagrams are found in Appendix M: stability charts for SDm in Figure M.7, for the 2D analytical ZOA in Figure M.8, for the 3D analytical ZOA in Figure M.9 and for the 3D digital ZOA in Figure M.10.

The inputs of the four tested methods are now discussed in detail. The same discretisation in terms of spindle speed and axial depth of cut is used for the investigated region of stability. The semi-discretisation method requires as inputs the modal properties of the Stäubli TX200 robot in terms of modal mass, modal damping and modal stiffness. This information can be gathered from the experimental modal analysis results presented in Section 5.2. In this work, modal properties of the Stäubli TX200 robot are identified using the LMS Test.Lab software from the FRFs measured over a frequency bandwidth of 200 Hz in the milling posture while the controller was in action (Figures 5.17 to 5.19). The ten identified modes are therefore transposed in modal properties which are presented in Table 7.3.

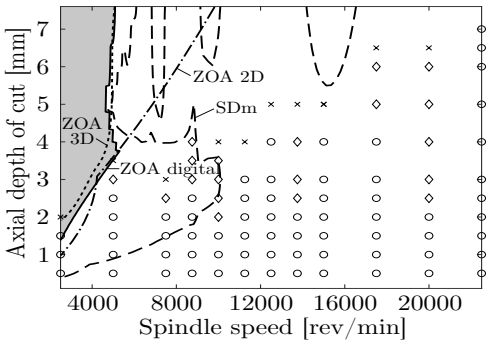
Modal mass [kg]			Modal damping [kg/s]			Modal stiffness [N/m]		
$m_{x,1}$=54.5	$m_{y,1}$=24.0	$m_{z,1}$=41.9	$c_{x,1}$=313.5	$c_{y,1}$=149.7	$c_{z,1}$=196.7	$k_{x,1}$=0.7e^6	$k_{y,1}$=0.09e^6	$k_{z,1}$=0.3e^6
$m_{x,2}$=48.6	$m_{y,2}$=21.5	$m_{z,2}$=1.0	$c_{x,2}$=655.7	$c_{y,2}$=80.4	$c_{z,2}$=1e^4	$k_{x,2}$=2.0e^6	$k_{y,2}$=0.3e^6	$k_{z,2}$=1e^8
$m_{x,3}$=91.2	$m_{y,3}$=54.0	$m_{z,3}$=1.0	$c_{x,3}$=495.5	$c_{y,3}$=149.9	$c_{z,3}$=1e^4	$k_{x,3}$=6.7e^6	$k_{y,3}$=1.1e^6	$k_{z,3}$=1e^8
$m_{x,4}$=1.0	$m_{y,4}$=8.9	$m_{z,4}$=1.0	$c_{x,4}$=1e^4	$c_{y,4}$=416.9	$c_{z,4}$=1e^4	$k_{x,4}$=1e^8	$k_{y,4}$=1.4e^8	$k_{z,4}$=1e^8
$m_{x,5}$=1.0	$m_{y,5}$=12.1	$m_{z,5}$=1.0	$c_{x,5}$=1e^4	$c_{y,5}$=384.1	$c_{z,5}$=1e^4	$k_{x,5}$=1e^8	$k_{y,5}$=2.8e^8	$k_{z,5}$=1e^8
$m_{x,6}$=1.0	$m_{y,6}$=7.7	$m_{z,6}$=1.0	$c_{x,6}$=1e^4	$c_{y,6}$=221.0	$c_{z,6}$=1e^4	$k_{x,6}$=1e^8	$k_{y,6}$=2.3e^8	$k_{z,6}$=1e^8

Table 7.3: Modal characteristics of the Stäubli TX200 robot for the 3D uncoupled SDm with the X, Y, Z directions referring to the TCP frame

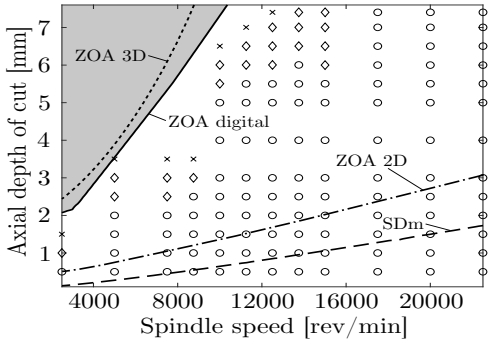
The uncoupled semi-discretisation method is implemented and therefore needs that one particular direction (X, Y or Z) is associated to each of the modes. The X, Y and Z directions are referred to the TCP frame. The dominant direction indicated by the measured mode shapes (Figure 5.20) is used as criterion to associate one direction to the concerned mode. Thus, the first mode, whose mode shape exhibited a rotation of the first joint around its motion axis, mainly contributes to the deflection of the TCP along the Y direction in the TCP frame. Hence, the first mode is associated to subscript $y, 1$ in Table 7.3. The second mode corresponds to subscript $z, 1$ as its mode shape presented a deflection of the TCP along the Z direction due to the rotation of the second joint around its motion axis. The third mode refers to subscript $x, 1$ as its mode shape resulted from the combined motions of the second and third joints moving the TCP onwards. Mode shapes of modes four and five were caused by a deflection perpendicular to the motion axis of the first joint and resulted in a deviation of the TCP along the Y direction. Subscripts $y, 2$ and $y, 3$ are respectively associated to modes four and five. For modes six and seven, their mode shapes showed a TCP deflection along the X direction. Therefore,



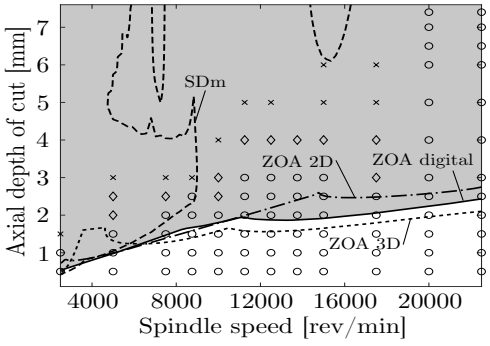
(a) Longitudinal down milling (aluminium)



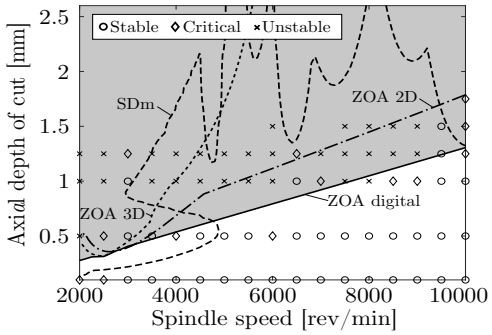
(b) Longitudinal up milling (aluminium)



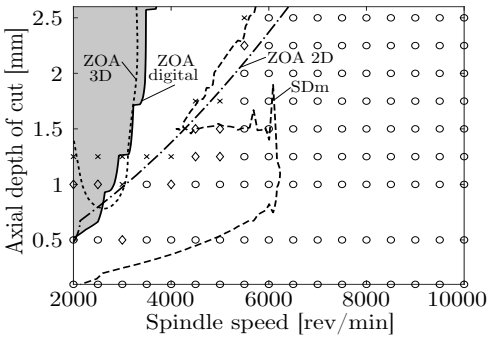
(c) Transversal down milling (aluminium)



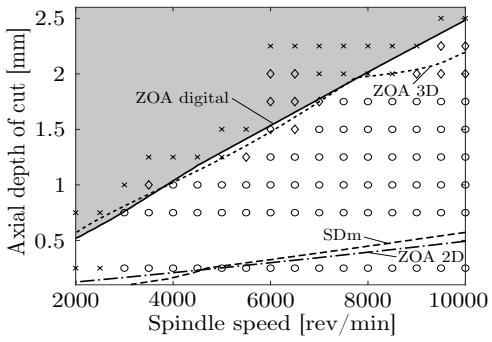
(d) Transversal up milling (aluminium)



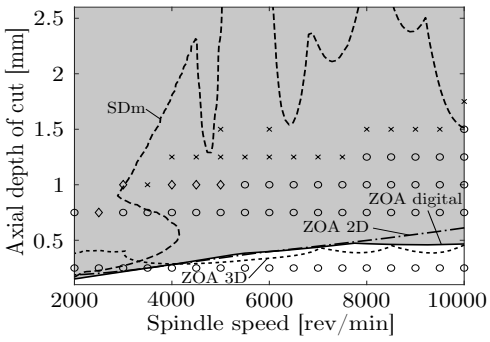
(e) Longitudinal down milling (steel)



(f) Longitudinal up milling (steel)



(g) Transversal down milling (steel)



(h) Transversal up milling (steel)

Figure 7.17: Experimental and simulated stability charts using the 2D analytical ZOA, the 3D analytical ZOA, the digital 3D ZOA and the semi-discretisation method

subscripts $x,2$ and $x,3$ are respectively associated to modes six and seven. The last three modes involve mode shapes for which the TCP mainly deflects along the Y direction due to wrist motions. Subscripts $y,4$, $y,5$ and $y,6$ are therefore respectively related to modes eight, nine and ten. Knowing that the uncoupled semi-discretisation method also requires square mass, damping and stiffness matrices, the modal properties are completed with virtual modes presenting very high damping and stiffness characteristics. In Table 7.3, the highest number of modes is found in the Y direction. The diagonal system matrices are therefore such as $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K} \in \mathbb{R}^{18\times18}$. Identified modal properties are introduced in the system matrices along their diagonal as presented in Eq. F.18 in Appendix F. In Table 7.3, it is assumed that the pass is along the longitudinal direction. For a transversal pass, modal properties along the X direction must be switched with the ones along the Y direction.

For the variants of the ZOA, the inputs are the FRFs at the tool tip. Directly measured FRFs over a frequency bandwidth of 200 Hz (Figures 5.17 to 5.19) are used as inputs for the 3D analytical ZOA and the 3D digital ZOA. For the 2D analytical ZOA, the synthesized FRFs over a frequency bandwidth of 100 Hz are considered with a smaller frequency resolution Δf= 0.01 Hz (as opposed to 0.39 Hz for the measured FRFs) and shown in Figure 7.18 along with the real part of the FRF matrix $\mathbf{H}(\omega)$.

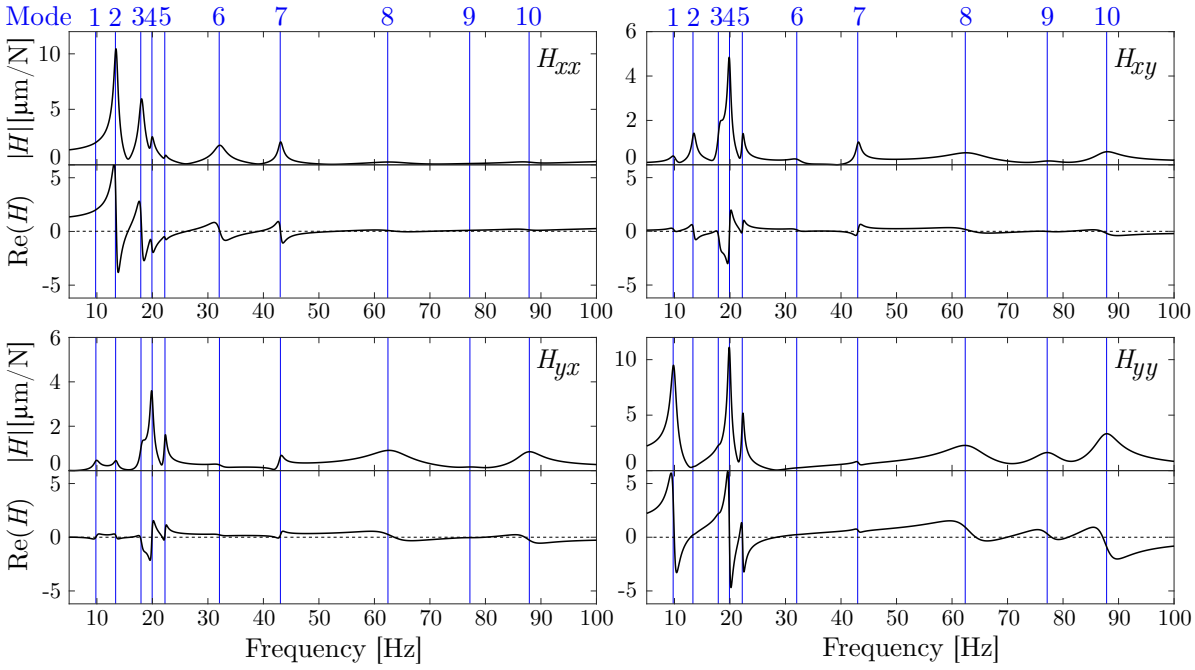


Figure 7.18: Amplitude and real part of synthesized FRF matrix $\mathbf{H}(\omega)$ used to derive the stability lobe diagram for the 2D analytical ZOA

Measured and synthesized tool tip FRFs are valid for longitudinal passes. For transversal passes, the X and Y directions in the FRFs must be switched. All the stability limits computed using the variants of ZOA suppose that there is less than one full vibration wave imprinted on cutting arc when chatter occurs i.e. N_c= 0. It was indeed observed that the lobes generated for $N_c > 0$ were clustered in the low-spindle-speed region which is not investigated in this work. For instance in Figure 7.19, stability lobes computed

for N_c=1 are appended to Figure 7.17a using the 2D analytical ZOA for the longitudinal pass in down milling in aluminium with a starting spindle speed of 100 rev/min instead of 2,500 rev/min. Higher order lobes were thus not observable within the chosen spindle speed ranges for aluminium and steel, as recommended by the tool manufacturer for such materials.

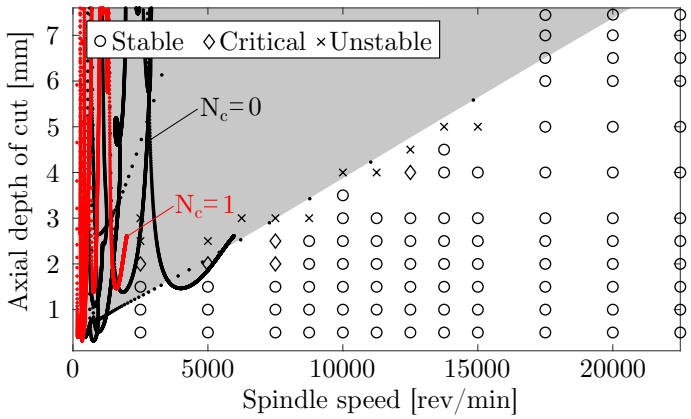


Figure 7.19: Experimental and simulated stability charts for the longitudinal pass in down milling in aluminium using the 2D analytical ZOA with the first two lobe orders

Still referring to the stability limits derived by the 2D analytical ZOA shown in Figures 7.17a and 7.19 for the longitudinal pass in down milling in aluminium, a classical stability analysis is carried out in relation with the synthesized tool tip FRFs (Figure 7.18) to assess which modes contribute to the overall stability for this milling operation. The analysis is achieved using the 2D analytical ZOA since the stability limit computed by the 3D analytical ZOA is far from the experimental limit (Figure 7.17a) and that the digital implementation does not allow the plot of individual lobe associated to one particular mode. Focussing on the spindle range below 10,000 rev/min, individual lobes of order N_c=0 are displayed in Figure 7.20 in combination with the labels of the associated modes.

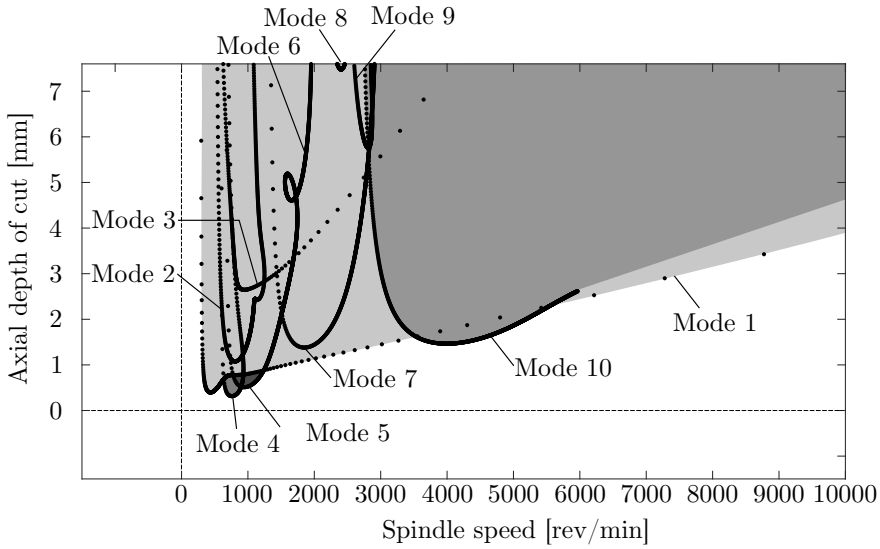


Figure 7.20: Close-up of the simulated stability lobe diagram of order N_c=0 for the longitudinal pass in down milling in aluminium using the 2D analytical ZOA

Overall, only four of the ten modes below 100 Hz contribute to the overall stability: modes 1, 4, 5 and 10. The excited modes are unsurprisingly excited since they exhibit mode shapes involving a motion perpendicular to the feed direction (Figure 5.20) and prominent negative real parts in H_{yy} (Figure 7.18). Mode 1 (f_{exp}=9.4 Hz and ζ_{exp}=7.9%) is responsible for the loss of stability over large ranges of spindle speed: between 1,500 and 3,500 rev/min and all the way after 5,500 rev/min. Its mode shape was associated with the rotation of the first robot joint. Between 3,500 and 5,500 rev/min, mode 10 (f_{exp}=88.7 Hz and ζ_{exp}=2.3%), whose mode shape was related to the spindle motion, becomes excited. The same trend of chatter frequency variation was observed in Figure 7.10a by using the digital ZOA on the same milling operation. Nevertheless, the switching to a chatter frequency related to mode 10 was not experimentally verified using cutting conditions of 5,000 rev/min at an axial depth of cut of 2.5 mm for the longitudinal pass in down milling in aluminium.

On the efficiency of the conventional stability methods, it is clear that the semi-discretisation method yielded the worst results in terms of stability limit prediction. For the longitudinal and transversal passes in down milling in aluminium (Figures 7.17a and 7.17c), the stability limits are underestimated as well as for the transversal pass in down milling in steel (Figure 7.17g). Nonetheless, the trend of the limits is the same as the one suggested by the ZOA variants. It is not the case for the remaining milling situations in which the semi-discretisation method generates curious shapes for the stability limits (Figures 7.17b, 7.17d, 7.17e, 7.17f and 7.17h). Part of the reason of the mismatch introduced by the SDm is that the implementation is based on uncoupled modes. The 2D analytical ZOA provides acceptable results for the longitudinal passes in down milling in aluminium and in steel with respect to the experimental contours (Figures 7.17a and 7.17e). Results are also admissible for the longitudinal pass in up milling in steel (Figure 7.17f). The stability limits are however underestimated for the transversal passes in down and up milling in aluminium and in steel (Figures 7.17c, 7.17d, 7.17g and 7.17h). Reasons of the discrepancies are mainly due to the ignorance of the tool axial direction which exhibited one very flexible mode in H_{zz} (Figure 5.19). The 3D analytical and digital ZOAs unsurprisingly generated similar stability limits (except in steel for the longitudinal pass in down milling due to the alternative implementation). Due to the unmodelled built-up edge, the method is obviously not able to fit the experimental stability contour for the up milling passes in aluminium (Figures 7.17b and 7.17d). Other strong discrepancies are remarked for the transversal passes in up milling for aluminium and steel in which the stability limits are underestimated (Figures 7.17d and 7.17h). Nevertheless, in the other milling situations, the 3D digital ZOA provided the highest correlations with the experimental stability contour compared with the other conventional stability methods. It suggests than only Hopf lobes are responsible for instability using this particular robotic milling cell and cutting conditions, being known than ZOA is unable to predict the flip lobes.

In terms of computing time, in addition to deliver unsatisfying results, the generation of one stability chart with the semi-discretisation method takes about 1.5 days since it is

also a time domain method. In contrast, it takes less than one minute to compute one stability chart with any of the ZOA variants in the frequency domain.

7.4 Discussion

A stability analysis in robotic milling was conducted with the developed models. It consisted in comparing the simulated and experimental stability charts derived from milling experiments. The tested models gathered multibody representations of the Stäubli TX200 robot as well as conventional milling methods developed to analyse the stability in machine tool. The stability of model variants was also assessed. The model variants aimed at evaluating a particular feature on the yielded stability limits such as the consideration of the link or the controller flexibility. Since the structural modes of the robot are pose dependent, a sensitivity analysis was also carried out to assess the influence of the feed direction on the stability limits. Altogether, a total of 88 stability lobe diagrams were computed to evaluate all the effects.

A summary of the appraisals of all models regarding the stability limits relatively to the experimental contours is proposed in Table 7.4. A "+" symbol indicates a reasonable agreement with the experimental contour and a "-" symbol means the opposite. An indication regarding the computing time is also provided in the worst situation (simulations in aluminium). Overall, it can be stated that none of the tested model is able to correctly predict the experimental stability contour for all milling situations.

Regarding the stability charts yielded by the multibody models, it can be concluded that appending the flexibility of the links or the controller, or the combination of both do not lead to a significant modification in the prediction of the stability limits. The controller flexibility mainly extends the stable area by providing additional mode damping. Adding flexibilities tends to lower the natural frequencies of the simulated robot modes which is reflected in the stability chart by a shift of the stability pocket towards the left with respect to the spindle speed axis. Their consideration is however important when dealing with static deflections. It is therefore observed in Table 7.4 that the multibody models with flexible joints and variants (models 1, 3, 4 and 5) share the same appreciations. The first and second models respectively refer to the Stäubli and UMons TX200 models, which are multibody models with joint tri-axial flexibility. Although the UMons TX200 model provided smoother stability limits, it tended to overestimate them relatively to the Stäubli TX200 model. Smooth stability limits were obtained using the UMons TX200 model thanks to a better tuning of the joint elastic parameters. In other words, it shows that the tuning of joint elastic parameters is of prime importance in order to predict the stability limits. The inertia matrix can be reasonably well approximated using the reshaped CAD models. The sixth model is the Stäubli TX200 model with joint tri-axial flexibility but with the inclusion of the edge forces in the cutting force model. It was shown that their inclusion did not improve the prediction of the stability limits. The next model is the Stäubli TX200 model with joint tri-axial flexibility but for which elastic parameters are only tuned on the basis of a FRF fitting thus resulting in

over-damped modes. It highlights the fact that the identified elastic parameters must accurately predict both the robot natural frequencies and damping ratios, otherwise the stability area is overestimated.

	Longitudinal down milling			Longitudinal up milling		
	Aluminium	Steel	Duration	Aluminium	Steel	Duration
1. Flexible joint	+	+	3 days	-	+	3 days
2. UMons TX200	+	-	3 days	-	+	3 days
3. Flexible joint+controller	+	+	12 days	-	+	12 days
4. Flexible joint+link	+	+	6 days	-	+	6 days
5. Flexible joint+link+controller	+	+	16 days	-	+	16 days
6. Edge force	+	+	3 days	+	-	3 days
7. Stäubli 35 Hz	-	-	3 days	-	-	3 days
8. Uncoupled SDm	-	-	1.5 days	-	-	1.5 days
9. 2D analytical ZOA	+	+	<1 min.	-	+	<1 min.
10. 3D analytical ZOA	-	-	<1 min.	-	-	<1 min.
11. 3D digital ZOA	+	+	<1 min.	-	-	<1 min.
	Transversal down milling			Transversal up milling		
	Aluminium	Steel	Duration	Aluminium	Steel	Duration
1. Flexible joint	+	-	3 days	+	-	3 days
2. UMons TX200	+	+	3 days	-	-	3 days
3. Flexible joint+controller	+	-	12 days	+	-	12 days
4. Flexible joint+link	+	-	6 days	+	-	6 days
5. Flexible joint+link+controller	+	-	16 days	+	-	16 days
6. Edge force	-	-	3 days	-	-	3 days
7. Stäubli 35 Hz	-	-	3 days	-	-	3 days
8. Uncoupled SDm	-	-	1.5 days	-	-	1.5 days
9. 2D analytical ZOA	-	-	<1 min.	-	-	<1 min.
10. 3D analytical ZOA	-	+	<1 min.	-	-	<1 min.
11. 3D digital ZOA	-	+	<1 min.	-	-	<1 min.

Table 7.4: Summary of the correlations of the simulated stability limits with the experimental contours for each model and each milling situation with an estimation of the computing time (worst case) using the proposed discretisations of the stability domain (min. stands for minute)

Regarding the conventional stability methods, the stability limits generated by the semi-discretisation method made clear that cross-FRFs must be included in order to reasonably fit the experimental stability limits. The 2D analytical ZOA showed the necessity to consider the tool tip direct and cross-FRFs along the X, Y and Z directions, otherwise the experimental limits are wrongly predicted. All the tool tip measured FRFs are the inputs of the 3D digital ZOA which consequently yields decent stability predictions in a very short computing time. In Table 7.4, it is observed that only one "+" symbol is found in column regarding the longitudinal pass in up milling in aluminium due to the unmodelled built-up edge which mainly appeared for this particular feed direction and material. The consideration of edge forces seems able to emulate the built-up edge phenomenon though. For some reason, conventional methods do not handle well the transversal passes in up milling. Since no method provides satisfying results for all

the milling situations, it points out the challenging task to model robotic milling processes.

A study was also carried out on the chatter frequency at the loss of stability as the axial depth of cut increases. In this work, it was remarked that only the pose-dependent structural robot modes were excited when chatter occurred. The findings partially agrees with the study of Cordes et al. [104]. They also stated that the low-frequency modes pertaining to the robot structure are responsible for the self-excited vibrations in hard material such as steel (or titanium). In fact, in low-speed milling, the pose dependent low frequency robot modes chatter which is also verified in this work. However, they also affirm that in high-speed milling, such as in aluminium, the spindle modes, which are independent of the robot configuration, chatter. In this work, it was verified that higher frequency modes were excited in high-speed milling but the milling experiments in aluminium revealed chatter frequency in the range of the robot structural modes. The excited modes were around 100 Hz and were related to mode shapes involving a wrist motion. In fact, the spindle modes are not considered in any of the modellings since their flexible modes were negligible compared with the robot modes in this work. Still, the stability limits were reasonably replicated in aluminium by comparison with the experimental stability contours. Variability in the results thus strongly depends on the considered robot and spindle and the mounting between both. The spindle absolutely needs to be considered if its flexible modes are as significant as the robot structural modes.

Still related to the chatter frequency, it was observed, with the use of the 3D digital ZOA, that stability limits were almost similar whether considering the measured FRFs over a frequency bandwidth of 200 Hz or 35 Hz. Under 35 Hz, the structural modes of the robot are highly pose dependent and their mode shapes mainly involve the motion of its first three joints. This trend tends to confirm that a fitting of the pose-dependent robot structural modes over a frequency bandwidth of 35 Hz is sufficient in the context of a multibody modelling. To robot modes must be appended the spindle modes if they are significant.

Chapter 8
Conclusions

Final conclusions regarding the development and the validation of a numerical model of robotic milling to optimise the cutting parameters are drawn in this last chapter. A summary of the research work is first proposed before outlining the main findings and future work. The list of publications that were issued during the thesis is also provided.

8.1 Summary of the research

Due to their open mechanical structure, milling robots profit from a larger workspace and can handle parts with complex geometry thanks to their high dexterity. Compared with conventional machine tools sharing the same workspace, substantial savings are expected with milling robots when the requirements on part quality are moderate. However, their productivity remains limited by self-excited vibrations, known as chatter, since they are considerably less rigid than machine tools.

In this work, the stability lobe diagrams of milling operations are computed based on a dynamic multibody model of the Stäubli TX200 robot. Its equations of motion are numerically solved in the time domain. Several variants of the multibody model were set up to assess the effects of different flexibility sources on the stability limits. Since past studies showed that the flexibility of industrial robots mainly originated from its joints, the reference multibody model of the Stäubli TX200 robot comprises flexible joints at the interface between the rotors and the links. Flexible joints are modelled with three torsional springs and viscous dampers which are orthogonal to each other. In the first variant of the multibody model, the link flexibility is appended to the reference model. In particular, the flexibility of the robot arm and forearm is modelled by equivalent flexible beams using the so-called corotational formulation. The second model variant adds the controller flexibility to the reference multibody model. Finally, the third multibody model variant combines the flexibilities of the considered links and the selected controller relatively to the reference one.

The dynamics of milling must also be modelled to derive the stability charts using the developed multibody models. Conducting virtual manufacturing operations necessitates the close interaction of three essential components: the dynamic model of the milling machine, a cutting force model and the representation of the workpiece and the tool. The model of the machine is naturally handled by the developed multibody models.

The parameters of the multibody model and variants as well as of the dynamic milling model were identified through experimental tests. Concerning the multibody models,

the sought parameters are related to the robot inertial properties, the elastic parameters of its joints and its links and the gains of the controller. Regarding the inertial properties, they were obtained from the robot manufacturer, from CAD models and on the basis of a rigid body identification method. Afterwards, the modal characteristics of the actual robot were measured through hammer tests using experimental modal analysis technique while the robot controller was in action. The elastic parameters of the joints were identified with the use of a curve fitting applied on the measured tool tip FRFs. Developed multibody models were able to accurately predict the mode shapes and natural frequencies, that were situated below 35 Hz, in other robot postures as long as the arm kept a configuration close to the one in which it was identified. The elastic parameters pertaining to the equivalent flexible beams representing the robot arm and forearm were determined on the basis of finite element models. The remaining controller gains were tuned on the basis of the measured FRFs. Concerning the dynamic milling model, the cutting force and edge force coefficients were identified through milling experiments with the Stäubli TX200 robot in aluminium and in steel.

The validated robotic milling simulator was used to eventually derive stability charts which were experimentally validated. Since the structural modes of the robot are pose dependent, it was decided to study the impact of the feed direction on the stability. Hence, milling experiments in aluminium and in steel were carried out with a feed direction parallel and perpendicular to the robot plane, both in down and up milling. Generated stability limits were compared with their experimental contours as well as relative to the stability boundaries provided by the classical methods used in machine tool. Besides the stability limits, the simulated chatter frequencies, time evolutions of cutting forces and virtual machined lateral faces resulting from unstable conditions were also examined and matched with the experimental results.

8.2 Main findings

The thesis sheds light on some findings which can be listed as follows:

1. Joint model with three orthogonal elastic elements: it was shown that only considering the joint flexibility around the axes of motion of the robot was not sufficient in order to capture mode shapes involving perpendicular deflection relative to motion axes. Hence, in this work, the multibody models gather flexible joints represented by three torsional springs and viscous dampers which are made perpendicular to each other (joint tri-axial flexibility model).
2. Dynamic vibration absorber: experimental modal analysis through hammer tests on the Stäubli TX200 robot revealed two closely spaced modes with nearly the same mode shapes (around 23 Hz). The phenomenon was modelled using a dynamic vibration absorber.
3. Gravity compensator model: it was shown that its effect could be neglected in the context of model updating.

4. Inertial properties from CAD models: a robot multibody model whose inertia parameters were derived from reshaped CAD models could reproduce the first five mode shapes, natural frequencies and damping ratios and handle the posture dependency issue with a satisfactory precision.

5. 3D experimental modal analysis: multiple experimental modal analyses were carried out on the Stäubli TX200 robot in different postures. Hammer impacts were distributed all over the surfaces of the robot in multiple directions to later animate 3D mode shapes. Overall, mode shapes under a frequency bandwidth of 200 Hz mainly exhibited joint motions and, depending on the tested robot posture, changes in their order of appearance could be observed, despite that their shapes remained the same.

6. Controller effect: it was shown that the controller had few influence on the structural modes. As a matter of fact, from the comparison of the measured tool tip FRFs when the brakes were enabled, it was noticed that the controller did not modify significantly the natural frequencies of the robot under 100 Hz and only added a slight damping, mainly for the modes beyond 35 Hz.

7. Modelled non-linearities: in this work, some non-linear effects are considered in the robotic milling simulator. In the robot multibody model are accounted the Coriolis, centrifugal and gyroscopic terms. In addition, when chatter develops during the simulation of unstable milling conditions, the time evolution of self-excited vibrations clearly exhibit a non-linear pattern. The marks left on the virtual workpiece therefore progressively grow along the pass which is experimentally verified.

8. Influence of the link and controller flexibilities: it was concluded that appending the flexibility of the links or the controller, or the combination of both to the multibody model with the joint tri-axial flexibility model did not lead to a significant modification in the prediction of the stability charts. On the other hand, their consideration is important when dealing with static deflections. It was recorded that the consideration of flexible links induced 20 to 30 % of additional TCP deflections, compared with the case in which only the joint flexibility was accounted for.

9. Consideration of robot structural modes under 35 Hz: from the comparison between the simulated and experimental stability charts, it was observed that satisfying correlations could already be obtained by only considering the robot structural modes under 35 Hz. Corresponding mode shapes mainly involved the deflections of the first three joints and were consequently highly pose dependent.

10. Chatter frequency: since robot structural modes are pose dependent, the chatter frequency naturally depends on the feed direction.

11. Excitation of robot structural modes: in low-speed milling, the pose-dependent low-frequency robot modes chatter as suggested by recent studies. However, it was discovered that in high-speed milling, although higher frequency modes were excited, they did not belong to the spindle modes. They rather corresponded to wrist motions in this work.

12. Conventional stability methods: in addition of being computationally efficient, the zero-order approximation proved to reasonably predict the experimental stability contours in most feed directions.

8.3 Future work

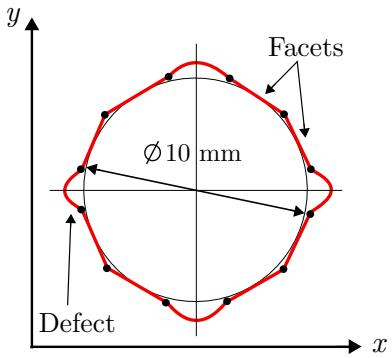
Ideas for future work can be clustered into three categories:

Non-linearity modelling

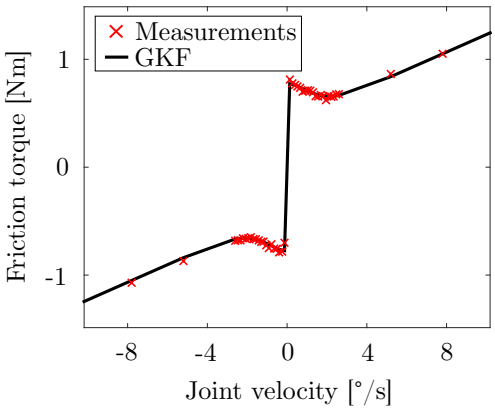
1. Joint model: since measured tool tip cross FRFs exhibited large amplitudes and a significant asymmetry, it would be interesting to exactly determine the sources of non-linearity resulting in this phenomenon. A finer model and understanding of joint mechanics seem to be a necessity in order to faithfully reproduce the modal behaviour of the robot in its whole workspace.

2. Hardening effect in gears: a finite element gear modelling could be integrated to the multibody model.

3. Friction: in this work, friction effects were completely neglected. However, friction might introduce significant TCP deviations when the joint motion reverses. At joint reversal, dry friction effects are significant since velocity comes to zero and robot TCP might deviate. This phenomenon is typically observed when the robot TCP is driven on small-diameter circles (Figure 8.1a). Defects are easily noticeable in the four quadrants. Such small circles were machined with the Stäubli TX200 robot and other defects appeared under the form of facets approximating the contour of the circle. So far, the identification of friction was carried out for the first joint of the robot using the general kinetic friction (GKF) model (Figure 8.1b).



(a) Quadrant glitches at motion direction reversals due to pre-sliding regime friction disturbance



(b) Measured and fitted friction torque for the first joint of the Stäubli TX200 robot

Figure 8.1: Friction effects

4. Non-linear stiffness characteristics: adopting non-linear stiffness characteristics might improve the prediction of modal parameters in arbitrary configurations.

Nevertheless, identifying the parameters of non-linear stiffness profile is also challenging with an assembled robot. Although relying on simple principle, the set-up to measure deflections and to apply a torque at the joint is not easy to implement in practice. The procedure was attempted on the Stäubli TX200 robot in order to identify the stiffness characteristics of the second joint (Figure 8.2). A force sensor and a cable tightener were inserted before the force application point. The cable tightener was introduced in order to apply a progressive force to the arm. On the other side of the arm, a comparator whose tip was in contact with the arm was installed to measure the deformation. Its support was fixed on a vertical stand resting on the ground.

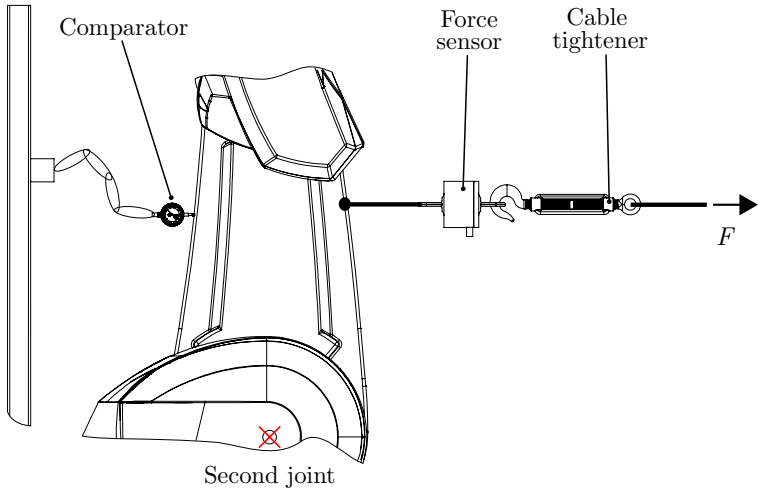


Figure 8.2: Tensile tests for the non-linear stiffness characterisation of the second joint

It was observed that a change in the robot posture greatly influenced the value of the non-linear stiffness around the axis of motion of the joint.

Parameter identification

5. Model updating: the identification of joint elastic parameters is very challenging. In this work was presented a FRF curve fitting in the frame pertaining to the accelerometer. Another alternative would be to transpose the measured FRFs in the joint space of the robot. In this way, fewer peaks would be needed to be fitted in the FRFs connecting the configuration parameters of the multibody model.

6. Operational modal analysis: since it was observed that chatter frequency was slightly offset relatively to the modes measured in a static robot pose, operational modal analysis could confirm that the robotic milling simulator accurately captures the frequency shift during a milling operation.

Improvement in stability assessment for robotic milling operations

7. Force control: the flexibility of the force sensor could be modelled in the simulator. Besides, the measurement of the virtual forces could be used to develop a force control reacting in case of unstable conditions.

8. 3D geometries of the workpiece and tool: although a $2D^{1/2}$ model of cutting seems sufficient to capture most of the effects to build the stability limits, robotic milling is intended for large parts with complex geometries. In order to simulate and optimise the whole process, a 3D material removal process must be implemented, for instance using the dexel method as presented in Appendix G.

9. Built-up edge: it would be interesting to model the built-up edge phenomenon in order to predict chatter along up milling passes in aluminium.

10. Chatter detection: in this work, the accelerometer used to measure the vibrations during the milling passes was located far from the cutting zone. Hence, chatter frequency was difficult to identify from the Fourier spectra. It would therefore be advised to mount the accelerometer on the spindle for future milling experiments. Particular care will be paid to the accelerometer wires which can be easily broken by an unexpected robot motion.

11. Faster stability analysis: in this work, the stability charts are obtained using a robot multibody model by assessing the stability of milling passes in the time domain which can lead to excessive computing time. In the context of an industrial implementation, the mass, damping and stiffness matrices could be derived from a multibody model of the robot that correctly predicts the pose dependency of its structural modes in the whole workspace. Then, the computed constitutive matrices would be used to emulate the tool tip FRFs and finally provided to the zero-order approximation (ZOA) to quickly compute the stability chart. This implementation assumes an accurate joint modelling taking into account all the non-linearities.

In order to enhance the stability prediction of the current multibody model in milling operations, the following three-step methodology could be adopted:

- Step 1: a novel flexible joint modelling should be developed in order to better fit the measured tool tip FRFs, in terms of natural frequency, damping ratio and mode shape. The new model should also be able to accurately reproduce the non symmetry appearing in the cross terms as well as the variability in the modes due to their posture dependency.

- Step 2: once the prediction of the significant modes can be ensured in the whole robot workspace, unstable milling experiments could be carried out with the objective to correctly fit the chatter frequency. In particular, the excited mode during chatter could be measured through accelerometers mounted on the robot arm and later correlated with the results provided by the robotic milling simulator.

- Step 3: predicted stable conditions using the resulting robotic milling simulator would be ultimately checked on the basis of milling experiments.

8.4 Publications

Results presented in this work conducted to scientific contributions that were either published in journal papers or conference proceedings. Next subsections list the written journal papers, conference papers as main author and conference papers as co-author.

8.4.1 Journal papers

1. H.N. Huynh, E. Rivière-Lorphèvre, O. Verlinden, "Milling Simulations with a 3-DOF Flexible Planar Robot," in *Engineering and Technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, vol. 95, pp. 1543-1552, 2016.

2. H.N. Huynh, O. Verlinden, A. V. Wouwer, "Comparative application of model predictive control strategies to a wheeled mobile robot," in *Journal of Intelligent & Robotic Systems*, vol. 87, pp. 81-95, 2017.

3. H.N. Huynh, E. Rivière-Lorphèvre, F. Ducobu, A. Ozcan, O. Verlinden, "Dystamill: a framework dedicated to the dynamic simulation of milling operations for stability assessment," in *The International Journal of Advanced Manufacturing Technology*, vol. 98, pp. 2109-2126, 2018.

4. H.N. Huynh, H. Assadi, E. Rivière-Lorphèvre, O. Verlinden, K. Ahmadi, "Modelling the dynamics of industrial robots for milling operations," in *Robotics and Computer-Integrated Manufacturing*, vol. 61, 2020.

5. O. Verlinden, H.N. Huynh, G. Kouroussis, E. Rivière-Lorphèvre, "Modelling of flexible bodies with minimal coordinates by means of the corotational formulation," in *Multibody System Dynamics*, vol. 42, pp. 495-514, 2018.

6. E. Rivière-Lorphèvre, H.N. Huynh, O. Verlinden, "Influence of the time step selection on dynamic simulation of milling operation," in *The International Journal of Advanced Manufacturing Technology*, vol. 95, pp. 4497-4512, 2018.

8.4.2 Conference papers as main author

7. H.N. Huynh, E. Rivière-Lorphèvre, O. Verlinden, "Integration of machining simulation within a multibody framework: application to milling," in *The Joint International Conference on Multibody System Dynamics*, Montréal, Canada, 2016.

8. H.N. Huynh, E. Rivière-Lorphèvre, O. Verlinden, "Report of Robotic Machining Measurements Using a Stäubli TX200 Robot: Application to Milling," in *ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Cleveland, USA, 2017.

9. H.N. Huynh, O. Verlinden, E. Rivière-Lorphèvre, "Robotic Machining Simulation using a Simplified Multibody Model," in *Annals of DAAAM & Proceedings*, vol. 28, Croatia, 2017.

10. H.N. Huynh, E. Rivière-Lorphèvre, O. Verlinden, "Multibody modelling of a flexible 6-axis robot dedicated to robotic machining," in *The Joint International Conference on Multibody System Dynamics*, Lisboa, Portugal, 2018.

11. H.N. Huynh, G. Kouroussis, O. Verlinden, E. Rivière-Lorphèvre, "Modal Updating of a 6-axis robot for Milling Application," in *25th International Congress on Sound and Vibration*, Hiroshima, Japan, 2018.

8.4.3 Conference papers as co-author

12. A. Ozcan, E. Rivière-Lorphèvre, E. Filippi, O. Verlinden, H.N. Huynh, "Modélisation du mouvement dynamique d'une machine-outil à l'aide d'un logiciel de simulation dynamique multi-corps – Première approche," in *Machines et Usinage À Grande Vitesse*, Tours, France, 2016.

13. E. Rivière-Lorphèvre, H.N. Huynh, O. Verlinden, "Time step selection for the dynamic simulation of milling operations," in *International conference on COmputational methods in Manufacturing Processes*, Liège, Belgique, 2016.

14. A. Ozcan, E. Rivière-Lorphèvre, H.N. Huynh, F. Ducobu, O. Verlinden, E. Filippi, "Modelling of pocket milling operation considering cutting forces and CNC control inputs," in *Procedia CIRP*, vol. 58, pp. 239-244, 2017.

15. O. Verlinden, M. Hajžman, H.N. Huynh, M. Byrtus, "Multibody modelling of friction based interaction between turbine blades," in *ECCOMAS Thematic Conference on Multibody Dynamics*, Prague, Czech Republic, 2017.

16. E. Rivière-Lorphèvre, H.N. Huynh, F. Ducobu, O. Verlinden, "Cutting Force Prediction in Robotic Machining," in *Procedia CIRP*, vol. 82, pp. 509-514, 2019.

Bibliography

[1] An End Mill Designed to Control Chatter. https://www.mmsonline.com/blog/post/getting-an-edge-on-chatter-control, Consulted in May 2019.

[2] CNC Machining of Impeller. http://mechanicstips.blogspot.com/2016/03/cnc-machining-of-impeller.html, Consulted in May 2019.

[3] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer, 2010.

[4] E. Appleton and D.J. Williams, *Industrial robot applications*. Halsted press, 1987.

[5] V. Daniel Hunt, *Industrial robotics handbook*. Industrial Press, Inc., 1983.

[6] J. F. Engelberger, *Robotics in Practice: Management and applications of industrial robots*. Springer US, 1980.

[7] R. K. Miller, *Industrial robot handbook: Case histories of effective robot use in 70 industries*. Competitve Manufacturing, 1989.

[8] Janome Industrial Equipment, *Janome's Cartesian Robot Deburs Metal Parts*. http://trends.directindustry.com/janome-industrial-equipment/project-21823-154408.html, Consulted in May 2019.

[9] Milling with SCARA Robotic Arm. https://www.youtube.com/watch?v=lSWo7v0rduo, Consulted in May 2019.

[10] FANUC Delta Robot Deburrs A Machined Casting Using Force Control. https://www.youtube.com/watch?v=1JdH_ytpzpk, Consulted in May 2019.

[11] Mobile milling Gantry machine. https://www.pietrocarnaghi.cn/fr/22/fraiseuse-a-portique-mobile.html, Consulted in May 2019.

[12] Robotics for nuclear manufacturing. https://www.neimagazine.com/features/featurerobotics-for-nuclear-manufacturing-4655013/, Consulted in May 2019.

[13] Kuka KR Quantec loads and unloads machine tool in its own production. https://www.kuka.com/en-au/industries/solutions-database/2016/06/solution-robotics-kuka, Consulted in May 2019.

[14] International Federation of Robotics, "Executive Summary World Robotics 2018 Industrial Robots," 2018.

[15] Y. Chen and F. Dong, "Robot machining: recent development and future research issues," *International Journal of Advanced Manufacturing Technology*, 2013. DOI 10.1007/s00170-012-4433-4.

[16] E. Guizzo, "The rise of the machines," *IEEE Spectrum*, vol. 45, 2008.

[17] J. Lee, "Apply Force/Torque Sensors to Robotic Applications," *Robotic Vision Systems*, vol. 3, pp. 189–194, 1987.

[18] T. Izumi, T. Narikiyo and Y. Fukui, "Teachingless grinding robot depending on three force information," *Advanced robotics*, vol. 2, no. 1, pp. 55–67, 1987.

[19] M. Jinno, F. Ozaki, T. Yoshimi and K. Tatsuno, "Development of a Force Controlled Robot for Grinding, Chamfering and Polishing," *IEEE International Conference on robotics and Automation*, 1995.

[20] Y. H. Chen and Y. N. Hu, "Implementation of a Robot System for Sculptured Surface Cutting. Part 1. Rough Machining," *International Journal of Advanced Manufacturing Technology*, 1999.

[21] Y. Song and Y. H. Chen, "Feature-based robot machining for rapid prototyping," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, pp. 451–459, 1999. https://doi.org/10.1243/0954405991516921.

[22] S. Muto and K. Shimokura, "Teaching and Control of Robot Contour-Tracking Using Contact Point Detection," *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, 1994.

[23] W. Ji and L. Wang, "Industrial Robotic Machining: A Review," *International Journal of Advanced Manufacturing Technology*, Feb. 2019. DOI: 10.1007/s00170-019-03403-z.

[24] I. Iglesias, M.A. Sebastian, J.E. Ares, "Overview of the state of robotic machining: Current situation and future potential," *Procedia Engineering*, vol. 132, pp. 911–917, 2015.

[25] J. Pandremenos, C. Doukas, P. Stravopoulos, G. Chryssolouris, "Machining with robots: a critical review," *7th International Conference on Digital Enterprise Technology*, 2011.

[26] L. Yuan, Z. Pan, D. Ding, S. Sun, W. Li, "Review on Chatter in Robotic Machining Process Regarding Both Regenerative and Mode Coupling Mechanism," *IEEE/ASME Transactions on mechatronics, October, 2018.*, vol. 23, no. 5.

[27] S. Marie, E. Courteille, P. Maurine, "Elasto-geometrical modeling and calibration of robot manipulators: Application to machining and forming applications," *Mechanism and Machine Theory*, 2013. 10.1016/j.mechmachtheory.2013.05.00.

[28] S. Mejri, "Identification et modélisation du comportement dynamique des robots d'usinage (EN: Identification and modelling of milling robot dynamic behaviour)," *PhD Thesis, Blaise Pascal University - Clermont-Ferrand II*, 2016.

[29] Airbus reveals the latest recruits to its A320 production line. https://www.dailymail.co.uk/sciencetech/article-5843983/Robots-Luise-Renate-join-Airbus-A320-production-line-Hamburg.html, Consulted in May 2019.

[30] Boeing Aerostructures Australia leads large-scale development of resin infusion as an industrial process. https://www.compositesworld.com/articles/resin-infusion-taking-off, Consulted in May 2019.

[31] Oversized plastic pipes milled to perfection. https://www.maschinenmarkt.international/oversized-plastic-pipes-milled-to-perfection-a-569061/, Consulted in May 2019.

[32] Carbon fiber trimming. https://www.robotic-hitechsolutions.com/applications/, Consulted in May 2019.

[33] Quantum leap towards Industry 4.0 in mold manufacturing. https://www.staubli.com/en-be/robotics/solution-application/automotive/chassis-safety/success-story-audi/, Consulted in May 2019.

[34] Robotic systems integration presses forward with Parliament Hill restoration. https://www.automationmag.com/technology/robotics/5477-canadian-can-do, Consulted in May 2019.

[35] For high quality surface finishing and cast cleaning. https://new.abb.com/products/robotics/application-software/machining/robotware-machining-fc, Consulted in May 2019.

[36] Robotic deburring from Abtex. http://www.abtex.com/robotic-deburring-news.html?no_redirect=true, Consulted in May 2019.

[37] SprutCAM. https://sprutcamrobot.com/what-is-sprutcamrobot, Consulted in May 2019.

[38] Robot Made: Large-Scale Robotic Timber Fabrication in Architecture Workshop. https://www.woodinnovatesbc.ca/featured-topic/robot-made-large-scale-robotic-timber-fabrication-in-architecture-workshop/, Consulted in May 2019.

[39] C. Moeller, H. Schmidt, P. Koch, C. Boehlmann et al., "Real Time Pose Control of an Industrial Robotic System for Machining of Large Scale Components in Aerospace Industry Using Laser Tracker System," *SAE International Journal of Aerospace*, 2017. https://doi.org/10.4271/2017-01-2165.

[40] J. DePree and C. Gesswein, "Robotic machining white paper project," *Halcyon Development*, 2008. Robotic Industries Association.

[41] T.L. Schmitz, K.S. Smith, *Machining Dynamics*. Springer Science and Business Media, 2009. DOI 10.1007/978-0-387-09645-2_2.

[42] S.A. Tobias and W. Fishwick, "Theory of Regenerative Machine Tool Chatter," *Engineering*, vol. 205, 1958.

[43] Y. Altintas, *Manufacturing Automation: Metal Cutting Mechanics, Machine Tool Vibrations, and CNC Design, 2e edition*. Cambridge University Press, 2012.

[44] T. Insperger, D. Lehotzky, and G. Stepan, "Regenerative delay, parametric forcing and machine tool chatter: a review," *IFAC-PapersOnLine*, pp. 322–327, 2015.

[45] Y. Altintas, E. Budak, "Analytical prediction of stability lobes in milling," *Annals of CIRP*, vol. 44, pp. 357–362, 1995.

[46] J. Tlusty, M. Polacek, "The stability of machine tools against self excited vibrations in machining," *International Research in Production Engineering ASME*, pp. 465–474, 1963.

[47] S. Seguy, L. Arnaud and T. Insperger, "Chatter in interrupted turning with geometrical defects: an industrial case study," *International Journal of Advanced Manufacturing Technology*, 2014.

[48] A. Gasparetto, "A System Theory Approach to Mode Coupling Chatter in Machining," *Journal of Dynamic Systems, Measurement, and Control*, vol. 120, pp. 545–547, 1998.

[49] Z. Pan, H. Zhang, Z. Zhu, J. Wang, "Chatter analysis of robotic machining process," *Journal of Material Processing Technology*, vol. 173, pp. 301–309, 2006.

[50] Z. Pan, H. Zhang, "Analysis and Suppression of Chatter in Robotic Machining Process," *International Conference on Control, Automation and systems ICCAS*, pp. 595–600, 2007.

[51] U. Schneider, M. Drust, M. Ansaloni, C. Lehmann, M. Pellicciari, F. Leali, J.W. Gunnink and A. Verl, "Improving robotic machining accuracy through experimental error investigation and modular compensation," *International Journal of Advanced Manufacturing Technology*, 2013.

[52] J.D. Barnfather, M.J. Goodfellow, T. Abram, "A performance evaluation methodology for robotic machine tools used in large volume manufacturing," *Robotics and Computer Integrated Manufacturing*, vol. 37, pp. 49–56, 2016.

[53] C. Dumas, S. Caro, S. Garnier, B. Furet, "Joint stiffness identification of six-revolute industrial serial robots," *Robotics and Computer Integrated Manufacturing*, vol. 27, pp. 881–888, 2011.

[54] S.F. Chen, "The 6x6 stiffness formulation and transformation of serial manipulators via the CCT theory," *IEEE international conference on robotics and automation*, 2003.

[55] T. Hardeman, *Modeling and Identification of Industrial Robots Including Drive and Joint Flexibilities*. PhD thesis, University of Twente, 2008.

[56] J. Wang, H. Zhang, T. Fuhlbrigge, "Improving Machining Accuracy with Robot Deformation Compensation," *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2011.*

[57] H. Hage, *Identication et simulation physique d'un robot Staubli TX90 pour le fraisage a grande vitesse.* PhD thesis, Universite Pierre et Marie Curie - Paris VI, 2012.

[58] A. Klimchik, Y. Wu, S. Caro, B. Furet, "Accuracy Improvement of Robot-Based Milling Using an Enhanced Manipulator Model," *Advances on Theory and Practice of Robots and Manipulators*, pp. 73–81, 2014. DOI: 10.1007/978-3-319-07058-2_9.

[59] M. Cordes and W. Hintze, "Offline simulation of path deviation due to joint compliance and hysteresis for robot machining," *International Journal of Advanced Manufacturing Technology*, vol. 90, pp. 1075–1083, 2016.

[60] C. Doukas, J. Pandremenos, P. Stravopoulos, P. Foteinopoulos, G. Chryssolouris, "On an empirical investigation of the structural behaviour of robots," *45th CIRP Conference on Manufacturing System*, pp. 501–506, 2012.

[61] C. Lehmann, B. Olofsson, K. Nilsson, M. Halbauer, M. Haage, A. Robertsson, O. Sörnmo, U. Berger, "Robot Joint Modeling and Parameter Identification Using the Clamping Method," *7th IFAC Conference on Manufacturing Modelling, Managemen, and Control*, 2013.

[62] C. Reinl, M. Friedmann, J. Bauer, M. Pischan, E. Abele, O. Von Stryk, "Model-based Off-line Compensation of Path Deviation for Industrial Robots in Milling Applications," *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2011.

[63] E. Abele, K. Schützer, J. Bauer and M. Pischan, "Tool path adaption based on optical measurement data for milling with industrial robots," *Production Engineering - Research and Development*, 2012. DOI 10.1007/s11740-012-0383-9.

[64] L.F.F. Furtado, E. Villani, L.G. Trabasso, R. Sutério, "A method to improve the use of 6-dof robots as machine tools," *International Journal of Advanced Manufacturing Technology, 2017.*

[65] N.R. Slavkovic, D.S. Milutinovic, M.M. Glavonjic, "A method for off-line compensation of cutting force-induced errors in robotic machining by tool path modification," *International Journal of Advanced Manufacturing Technology*, vol. 70, pp. 2083–2096, 2014.

[66] S.S. Makhanova, D. Batanovb, E. Bohezb, K. Sonthipaumpoonc, W. Anotaipaiboona and M. Tabucanon, "On the tool-path optimization of a milling robot," *Computers & Industrial Engineering*, no. 43, pp. 455–472, 2002.

[67] A. Atmosudiro, M. Keinert, A. Karim, A. Lechler, A. Verl, A. Csizar, "Productivity Increase through Joint Space Path Planning for Robot Machining," *UKSim-AMSS 8th European Modelling Symposium*, 2014.

[68] Y.V. Ilyukhin, Yu.V. Poduraev, A.V. Tatarintseva, "Nonlinear Adaptive Correction of Continuous Path Speed of the Tool for High Efficiency Robotic Machining," *25th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAAM 2014, Procedia Engineering*, vol. 100, 2014.

[69] C. Lehmann, M. Halbauer, D. Euhus, D. Overbeck, "Milling with industrial robots: Strategies to reduce and compensate process force induced accuracy influences," *17th IEEE International Conference on Emerging Technologies and Factory Automation*, 2012.

[70] U. Schneider, M. Momeni-K, M. Ansaloni and A. Verl, "Stiffness Modeling of Industrial Robots for Deformation Compensation in Machining," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014.

[71] U. Schneider, M. Drust, A. Puzik, A. Verl, "Compensation of errors in robot machining with a parallel 3d-piezo compensation mechanism," *Procedia CIRP*, vol. 7, pp. 305–310, 2013.

[72] J. R. D. Posada, U. Schneider, A. Sridhar and A. Verl, "Automatic Motion Generation for Robotic Milling Optimizing Stiffness with Sample-Based Planning," *Special Issue Robotic Machine Tools*, 2017. doi:10.3390/machines5010003.

[73] O. Sörnmo, B. Olofsson, U. Schneider, A. Robertsson and R. Johansson, "Increasing the Milling Accuracy for Industrial Robots Using a Piezo-Actuated High-Dynamic Micro Manipulator," *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2012.

[74] H. Bo, M. Azhar, D. M. Mohan and D. Campolo, "Review of robotic control strategies for industrial finishing operations," *10th International Symposium on Mechatronics and its Applications (ISMA)*, 2015.

[75] O. Sörnmo, B. Olofsson, A. Robertsson and R. Johansson, "Increasing Time-Efficiency and Accuracy of Robotic Machining Processes Using Model-Based Adaptive Force Control," *Proceedings of 10th IFAC Symposium on Robot Control*, 2012.

[76] F. Tian, C. Lv, Z. Li and G. Liu, "Modeling and control of robotic automatic polishing for curved surfaces," *CIRP Journal of Manufacturing Science and Technology*, 2016.

[77] A. Olabi, M. Damak, R. Bearée, O. Gibaru, S. Leleu, "Improving the accuracy of industrial robots by offline compensation of joints errors," *IEEE International Conference on Industrial Technology*, pp. 492–497, 2012.

[78] M. Oueslati, "Contribution à la modélisation dynamique, l'identification et la synthèse de lois de commande adaptées aux axes flexibles d'un robot industriel (translated: contribution to the dynamic modelling, the identification and the synthesis of command laws adapted for industrial robot flexible joints)," *PhD Thesis*, L'école Nationale Supérieure d'Arts et Métiers, ParisTech - Centre de Lille, 2013.

[79] COMET, 2011, EU/FP7-project: Plug-and-produce COmponents and METhods for adaptive control of industrial robots enabling cost effective, high precision manufacturing in factories of the future. http://www.cometproject.eu.

[80] Hephastos: Hard Material Small-Batch Industrial Machining Robot. http://hephestosproject.eu, Last access in May 2019.

[81] F. A. S. for Robotic Machining. https://www.cnr.it/en/bilateral-agreements/project/1953/advanced-solutions-for-robotic-machining.

[82] COROMA: Cognitively Enhanced Robot for flexible manufacturing of Metal and Composite parts. https://www.coroma-project.eu/project-overview/.

[83] S. Caro, C. Dumas, S. Garnier, B. Furet, "Workpiece placement optimization for machining operations with a kuka kr270-2 robot," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2921–2926, May, 2013.

[84] H. Qin, Y. Li and X. Xiong, "Workpiece Pose Optimization for Milling with Flexible-Joint Robots to Improve Quasi-Static Performance," *Applied sciences*, 2019. DOI: 10.3390/app9061044.

[85] J. Li, B. Li, N.Y. Shen, H. Qian and Z.M. Guo, "Effect of the cutter path and the workpiece clamping position on the stability of the robotic milling system," *International Journal of Advanced Manufacturing Technology*, 2017. DOI 10.1007/s00170-016-9759-x.

[86] M. Slamani, S. Gauthier, J.-F. Chatelain, "Comparison of surface roughness quality obtained by high speed CNC trimming and high speed robotic trimming for CFRP laminate," *Robotics and Computer Integrated Manufacturing*, vol. 42, pp. 63–72, 2016.

[87] S. Caro, S. Garnier, B. Furet, A. Klimchik and A. Pashkevich, "Workpiece Placement Optimization for Machining Operations with Industrial Robots," *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2014.

[88] S.H.H. Zargarbashi, W. Khan, J. Angeles, "Posture optimization in robot-assisted machining operations," *Mechanism and Machine Theory*, vol. 51, pp. 74–86, 2012.

[89] Y. Lin, H. Zhao, H. Ding, "Posture optimization methodology of 6R industrial robots for machining using performance evaluation indexes," *Robotics and Computer Integrated Manufacturing*, 2017. http://dx.doi.org/10.1016/j.rcim.2017.02.002.

[90] Y. Lin, H. Zhao and H. Ding, "Spindle configuration analysis and optimization considering the deformation in robotic machining applications," *Robotics and Computer Integrated Manufacturing*, 2018.

[91] G. Xiong, Y. Ding, L.M. Zhu, "Stiffness-based pose optimization of an industrial robot for five-axis milling," *Robotics and Computer Integrated Manufacturing*, 2017.

[92] N.Y. Shen, Z.M. Guo, J. Li, L. Tong and K. Zhu, "A practical method of improving hole position accuracy in the robotic drilling process," *International Journal of Advanced Manufacturing Technology*, 2018. https://doi.org/10.1007/s00170-018-1776-5.

[93] Y. Guo, H. Dong, Y. Ke, "Stiffness-oriented posture optimization in robotic machining applications," *Robotics and Computer Integrated Manufacturing*, 2015.

[94] G.-C. Vosniakos and E. Matsas, "Improving feasibility of robotic milling through robot placement optimisation," *Robotics and Computer Integrated Manufacturing*, 2010. doi:10.1016/j.rcim.2010.04.001.

[95] Y. Wu, *Optimal pose selection for the identification of geometric and elastostatic parameters of machining robots*. PhD thesis, Ecole des Mines de Nantes, 2014.

[96] G. Alici, B. Shirinzadeh, "Enhanced stiffness modeling, identification and characterization for robot manipulators," *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 554–564, 2005.

[97] A. Klimchik, A. Pashkevich, S. Caro, D. Chablat, "Stiffness matrix of manipulators with passive joints: computational aspects," *IEEE Transactions on Robotics*, 2012.

[98] A. Klimchik, A. Ambiehl, S. Garnier, B. Furet, A. Pashkevich, "Efficiency evaluation of robots in machining applications using industrial performance measure," *Robotics and Computer Integrated Manufacturing*, 2017.

[99] L. Cen, S.N. Melkote, "Effect of Robot Dynamics on the Machining Forces in Robotic Milling," *45th SME North American Manufacturing Research Conference, Procedia Manufacturing*, vol. 10, pp. 486–496, 2017.

[100] Dávid Hajdu, "Analysis of the sensitivity of milling stability on the modal parameters," Master's thesis, Budapest University of Technology and Economics: Faculty of Mechanical Engineering, 2014.

[101] L.T. Tunc and D. Stoddart, "Tool path pattern and feed direction selection in robotic milling for increased chatter-free material removal rate," *International Journal of Advanced Manufacturing Technology*, vol. 89, pp. 2907–2918, 2017. DOI 10.1007/s00170-016-9896-2.

[102] G. Wang, H. Dong, Y. Guo and Y. Ke, "Chatter mechanism and stability analysis of robotic boring," *International Journal of Advanced Manufacturing Technology*, 2016. DOI 10.1007/s00170-016-9731-9.

[103] G. Wang, H. Dong, Y. Guo and Y. Ke, "Early chatter identification of robotic boring process using measured force of dynamometer," *International Journal of Advanced Manufacturing Technology*, vol. 94, pp. 1243–1252, 2017. DOI 10.1007/s00170-017-0941-6.

[104] M. Cordes, W. Hintze, Y. Altintas, "Chatter stability in robotic milling," *Robotics and Computer Integrated Manufacturing*, vol. 55, pp. 11–18, 2019. https://doi.org/10.1016/j.rcim.2018.07.004.

[105] S. Mejri, V. Gagnol, Thien-Phu Le, L. Sabourin, P. Ray, P. Paultre, "Dynamic characterization of machining robot and stability analysis," *International Journal Advanced Manufacturing Technology, 2015.* DOI 10.1007/s00170-015-7336-3.

[106] S. Mousavi, V. Gagnol, B.C. Bouzgarou, P. Ray, "Dynamic behaviour model of a machining robot," *ECCOMAS Multibody Dynamics*, pp. 771–779, July, 2013.

[107] S. Mousavi, V. Gagnol, B. C. Bouzgarrou, P. Ray, "Dynamic model and stability prediction in robotic machining," *International Journal of Advanced Manufacturing Technology*, pp. 1–13, June 2016.

[108] S. Mousavi, V. Gagnol, B.C. Bouzgarou, P. Ray, "Control of a multi degrees functional redundancies robotic cell for optimization of the machining stability," *Procedia CIRP*, vol. 58, pp. 269–274, 2017.

[109] S. Mousavi, V. Gagnol, B.C. Bouzgarrou, P. Ray, "Stability optimization in robotic milling through the control of functional redundancies," *Robotics and Computer Integrated Manufacturing*, 2018.

[110] C. Chen, F. Peng, R. Yan, Z. Fan, Y. Li, D. Wei, "Posture-dependent stability prediction of a milling industrial robot based on inverse distance weighted method," *Procedia manufacturing*, pp. 993–1000, 2018.

[111] S. Baglioni, F. Cianetti, C. Braccesi and D. M. De Micheli, "Multibody modelling of N DOF robot arm assigned to milling manufacturing: Dynamic analysis and position errors evaluation," *Journal of Mechanical Science and Technology*, vol. 30(1), pp. 405–420, 2016.

[112] A. Klimchik, S. Caro, Y. Wu, D. Chablat, B. Furet, A. Pashkevich, "Stiffness modeling of robotic manipulator with gravity compensator," *In Proceedings of the 6th International Workshop on Computational Kinematics*, 2013.

[113] A. Klimchik, D. Bondarenko, A. Pashkevich, S. Briot, B. Furet, "Compliance error compensation in robotic-based milling," *Lectures Notes in Electrical Engineering, 2014.*

[114] W. Khalil and E. Dombre, *Modeling, identification and control of robots.* Butterworth-Heinemann, 2004.

[115] J. Craig, *Introduction to robotics: Mechanics and control.* Pearson Education, 2005.

[116] R. Hartenberg, J. Denavit, "A kinematic notation for lowerpair mechanisms based on matrices," *Journal of Applied Mechanics*, pp. 215–221, 1955.

[117] W. Khalil, J. Kleinfinger, "A new geometric notation for open and closed loop robots," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1174–1180, 1986.

[118] M.W. Spong, *Robot Dynamics and Control*. John Wiley & Sons, 1989.

[119] D.L. Pieper, *The kinematics of manipulators under computer control*. PhD thesis, Stanford university, 1968.

[120] B. Siciliano and O. Khatib, *Handbook of Robotics*. Springer-Verlag, 2016.

[121] W. Schiehlen, *Multibody System Handbook*. Springer, New York, 1990.

[122] M. Hiller, A. Kecskeméthy, "Dynamics of multibody systems with minimal coordinates," *In: Pereira, M.F.O.S., Ambrosio, J.A.C. (eds.) Computer-Aided Analysis of Rigid and Flexible Mechanical Systems*, vol. 268, pp. 61–100, 1994.

[123] E. Abele, S. Rothenbücher, M. Weigold, "Cartesian compliance model for industrial robots using virtual joints," *Production Engineering - Research and Development*, vol. 2, no. 3, pp. 339–343, 2008.

[124] M. Neubauer, H. Gattringer, A. Müller, A.Steinhauser and W. Höbarth, "A two-stage calibration method for industrial robots with joint and drive flexibilities," *Mechanical sciences*, vol. 6, pp. 191–201, 2015.

[125] O. Verlinden, H.N. Huynh, G. Kouroussis and E. Rivière-Lorphèvre, "Modelling flexible bodies with minimal coordinates by means of the corotational formulation," *Multibody System Dynamics*, vol. 42(4), pp. 495–514, 2018. https://doi.org/10.1007/s11044-017-9609-0.

[126] A.A. Shabana, *Dynamics of Multibody Systems*. Cambridge University Press, New York, 2005.

[127] A. Cardona, M. Géradin, "Modelling of superelements in mechanism analysis," *International Journal for Numerical Methods in Engineering*, vol. 32, pp. 1565–1593, 1991.

[128] A. Cardona, "Superelements modelling in flexible multibody dynamics," *Multibody System Dynamics*, vol. 4, pp. 245–266, 2000.

[129] T.N. Le, J.M. Battini, M. Hjiaj, "Efficient formulation for dynamics of corotational 2D beams," *Computational Mechanics*, vol. 48, no. 2, pp. 153–161, 2011.

[130] W.J. Book, "Recursive Lagrangian Dynamics of Flexible Manipulator Arms via Transformation Matrices," *The International Journal of Robotics Research*, vol. 3, no. 3, pp. 87–101, 1984.

[131] Nabtesco, "Catalogue," *Precision Reduction Gear RV*, 2015.

[132] M. Ruderman and T. Bertram, "Modeling and observation of hysteresis lost motion in elastic robot joints," *In Proceedings of 10th IFAC Symposium on Robot Control International Federation of Automatic Control*, 2012.

[133] T. Tjahjowidodo, F. Al-Bender, H. Van Brussel, "On the chaotic response in a robot joint mechanism due to backlash," *Proceedings of European Nonlinear Oscillations Conference ENOC, Eindohoven, Netherlands, 2005.*

[134] F. Ma and H. Zhang, "Parameter Analysis of the Differential Model of Hysteresis," *Transactions of the ASME*, 2004.

[135] H.N. Huynh, E. Rivière-Lorphèvre, O. Verlinden, "Multibody modelling of a flexible 6-axis robot dedicated to robotic machining," *IMSD: Proceedings of the 5th Joint International Conference on Multibody System Dynamics*, Portugal, June, 2018.

[136] R. Waiboer, *Dynamic modelling, identification and simulation of industrial robots, for off-line programming of robotised laser welding.* PhD thesis, University of Twente, 2007.

[137] S. Bissey, *Development of a cutting forces model applicable to a family of tools: case of milling of hard steel.* PhD thesis, Arts et Métiers ParisTech, 2005.

[138] Y. Kaynak, T. Lu & I. S. Jawahir, "Cryogenic Machining-Induced Surface Integrity: A Review and Comparison with Dry, MQL, and Flood-Cooled Machining," *Machining Science and Technology: An International Journal*, vol. 18, no. 2, pp. 149–198, 2014. http://dx.doi.org/10.1080/10910344.2014.897836.

[139] E. Shamoto and Y. Altintas, "Prediction of shear angle in oblique cutting with maximum stress and minimum energy principle," *Journal of manufacturing science and engineering*, vol. 121, pp. 399–407, 1999.

[140] M. Wiercigroch and A.M. Kristov, "Frictional chatter in orthogonal metal cutting," *Philosophical Transactions of The Royal Society A Mathematical Physical and Engineering Sciences*, vol. 359, no. 1781, pp. 713–738, 2001.

[141] M. Wiercigroch and E. Budak, "Sources of nonlinearities, chatter generation and suppression in metal cutting," *Philosophical Transactions of The Royal Society A Mathematical Physical and Engineering Sciences*, vol. 359, no. 1781, pp. 663–693, 2001.

[142] Y. Altintas, "Manufacturing Automation: Metal Cutting Mechanics, Machine Tool Vibrations and CNC Design," *1st ed., Cambridge University Press, USA*, 2000.

[143] Y. Altintas, "Analytical prediction of three dimensional chatter stability in milling," *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, vol. 44, no. 3, pp. 717–723, 2001. https://doi.org/10.1299/jsmec.44.717.

[144] M. Eynian, *Chatter stability of turning and milling with process damping*. PhD thesis, University of British Columbia, 2010.

[145] T. Insperger, G. Stépán, P.V. Bayly, B.P. Mann, "Multiple chatter frequencies in milling processes," *Journal of sound and vibration*, vol. 262, pp. 333–345, 2003.

[146] T. Insperger, G. Stepan, *Semi-Discretization for Time-Delay Systems: Stability and Engineering applications*. Springer Science Business Media, 2011.

[147] M.G. Floquet, *Équations différentielles linéaires à coefficients périodiques*. Annales Scientifiques de l'École Normale Supérieure, 1883.

[148] M. A. Davies, B. Dutterer, J. R. Pratt, A. J. Schaut, J. B. Bryan, "On the Dynamics of High-Speed Milling with Long, Slender Endmills," *Annals of the CIRP*, vol. 47, no. 1, pp. 55–60, 1998. https://doi.org/10.1016/S0007-8506(07)62784-X.

[149] S.D. Merdol, Y. Altintas, "Multi frequency solution of chatter stability for low immersion milling," *Journal of Manufacturing Science and Engineering*, vol. 126, no. 3, pp. 459–466, 2004. doi:10.1115/1.1765139.

[150] T. Insperger and G. Stépán, "Updated semi-discretization method for periodic delay-differential equations with discrete delay," *International Journal for Numerical Methods in Engineering*, vol. 61, pp. 117–141, 2004.

[151] D. Hajdu, T. Insperger, G. Stepan, "The effect of non-symmetric FRF on machining: a case study," *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Boston, Massachusetts, USA, August 2-5 (2015)*.

[152] T. Insperger, B.P. Mann, G. Stépán, P.V. Bayly, "Stability of up-milling and down-milling, part I: alternative analytical methods," *International Journal of Machine Tools and Manufacture*, vol. 43, pp. 25–34, 2003.

[153] X. Long, B. Balachandran, "Stability of Up-milling and Down-milling Operations with Variable Spindle Speed," *Journal of Vibration and Control*, 2017. https://doi.org/10.1177/1077546309341131.

[154] S. Mousavi, *Analyse de la stabilité de la coupe d'un procédé d'usinage robotisé*. PhD thesis, Université Blaise Pascal - Clermont-Ferrand II, 2016.

[155] S. Engin, Y. Altintas, "Mechanics and dynamics of general milling cutters. Part I: helical end mills," *International Journal of Machine Tools and Manufacture*, vol. 41, pp. 2195–2212, 2001.

[156] S. Engin, Y. Altintas, "Mechanics and dynamics of general milling cutters. Part II: inserted cutters," *International Journal of Machine Tools and Manufacture*, vol. 41, pp. 2213–2231, 2001.

[157] G. Peigné, H. Paris, D. Brissaud, "A model of milled surface generation for time domain simulation of high-speed cutting," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 217, no. 7, pp. 919–930, 2003. https://doi.org/10.1243/09544050360686798.

[158] Y. Altintas and P. Lee, "Mechanics and Dynamics of Ball End Milling," *Journal of Manufacturing Science and Engineering*, vol. 120, pp. 684–692, 1998.

[159] M.L. Campomanes, Y. Altintas, "An improved time domain simulation for dynamic milling at small radial immersion," *Transactions of the ASME*, 2003.

[160] E. Rivière-Lorphèvre, *Study and simulation of high speed machining processes: cutting forces, stability, surface finish; Étude et simulation de procédés de fraisage grande vitesse: efforts de coupe, stabilité, états de surface.* PhD thesis, UMons: Faculty of Engineering, 2007.

[161] Wittenstein, *High Performance Servo Motors, Compact and Efficient Motors*, 2011.

[162] H. Mayeda, K. Yoshida, K. Osuka, "Base parameters of manipulator dynamic models," *IEEE Transactions on robotics and automation*, vol. 6, no. 3, pp. 312–321, June 1990.

[163] J. Swevers, C. Ganseman, J. De Schutter, H. Van Brussel, "Experimental robot identification using optimised periodic trajectories," *Mechanical Systems and Signal Processing*, vol. 10, no. 5, pp. 561–577, 1996.

[164] J. Swevers, C. Ganseman, D.B. Tükel, J. De Schutter, H. Van Brussel, "Optimal Robot Excitation and Identification," *IEEE Transactions on robotics and automation*, vol. 13, no. 5, pp. 730–740, 1997.

[165] L. Ljung, *Signal Analysis and Prediction*, ch. System identification, pp. 163–173. Birkhauser,, 1998.

[166] R. Brincker and C. Ventura, *Introduction to operational modal analysis.* Wiley, 2015.

[167] N.M.M. Maia and J.M.M. Silva, "Theoretical and experimental modal analysis," *John Wiley & Sons*, 1997.

[168] P. Elosegui, "Measurement of the Dynamic Model of a PUMA 560 Robot Using Experimental Modal Analysis," *Journal of Mechanical Design*, 1994.

[169] J. J. Uicker, B. Ravani, P.N. Sheth, *Matrix Methods in the Design Analysis of Mechanisms and Multibody Systems.* Cambridge University Press, 2013.

[170] J. C. Lagarias, J. A. Reeds, M. H. Wright and P. E. Wright, "Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions," *SIAM Journal of Optimization*, vol. 9, no. 1, pp. 112–147, 1998.

[171] Edouard Rivière-Lorphèvre, Enrico Filippi, Pierre Dehombreux, "Inverse method for cutting forces parameters evaluation," *Engineering MECHANICS*, vol. 14(5), pp. 1–13, 2007.

[172] O. Verlinden, L. Ben Fékih and G. Kouroussis, "Symbolic generation of the kinematics of multibody systems in EasyDyn: From MuPAD to Xcas/Giac," *Theoretical and Applied Mechanics Letters*, vol. 3, no. 1, pp. 013012, doi: 10.1063/2.13013012, 2013.

[173] P. Masarati, M. Morandini, P. Mantegazza, "An efficient formulation for general-purpose multibody/multiphysics analysis," *Journal of Computational and Nonlinear Dynamics*, vol. 9, no. 4, 2014.

[174] J. Gerstmayr, A. Dorninger, R. Eder, P. Gruber, D. Reischl, M. Saxinger, M. Schörgenhumer, A. Humer, K. Nachbagauer, A. Pechstein, Y. Vetyukov, "HOTINT - a script language based framework for the simulation of multibody dynamics systems," *In proceedings of the ASME Design Engineering Technical Conference*, vol. 7B, 2013.

[175] H.N. Huynh, E. Rivière-Lorphèvre, F. Ducobu, A. Ozcan, O. Verlinden, "Dystamill: a framework dedicated to the dynamic simulation of milling operations for stability assessment," *International Journal of Advanced Manufacturing Technology*, 2018.

[176] P. Corke, *Robotics, Vision and Control, Fundamental Algorithms in MATLAB*. Springer, Berlin, Heidelberg, 2011. https://doi.org/10.1007/978-3-642-20144-8.

[177] E. Rivière-Lorphèvre, H.N. Huynh and O. Verlinden, "Optimal time step selection for dynamic simulation of milling operation," *International Journal of Advanced Manufacturing Technology*, 2018.

[178] S. Smith, J. Tlusty, "Update on High Speed Milling Dynamics," *Annals of CIRP*, vol. 39, pp. 517–521, 1990.

[179] R. Krishnan, *Electric Motor Drives Modeling, Analysis and Control*. Prentice Hall, 2001.

[180] Y. Kiran, P.S. Puttaswamy, "Field Oriented Control of a Permanent Magnet Synchronous Motor using a DSP," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 2014.

[181] Y. Yang, Q. Liu and B. Zhang, "Three-dimensional chatter stability prediction of milling based on the linear and exponential cutting force model," *International Journal of Advanced Manufacturing Technology*, vol. 72, pp. 1175–1185, 2014. DOI 10.1007/s00170-014-5703-0.

[182] W. Zhang, M.C. Leu, "Surface Reconstruction Using Dexel Data From Three Sets of Orthogonal Rays," *Journal of Computing and Information Science in Engineering*, 2009.

[183] John J. Craig, *Introduction to Robotic: Mechanics and Control*. Pearson, Prentice Hall, 2005.

Appendix A

Inverse kinematics detail: Paul's method

A.1 Modified Denavit-Hartenberg convention

The inverse kinematics problem is solved using the modified Denavit-Hartenberg parameters. Compared to the standard notation, the modified convention is said to be clearer and tidier and is also commonly used. The modified convention brings back frame O_{i} on joint i and changes the order of mathematical operations. In this convention, there are still four parameters but parameters a_{i} and d_{i} were substituted for $d_{\mathrm{i\text{-}1}}$ and r_{i}, respectively. Hence, the usage of either convention is easily recognised. Figure A.1 depicts the application of the modified Denavit-Hartenberg convention to define the relative position and orientation of two consecutive links.

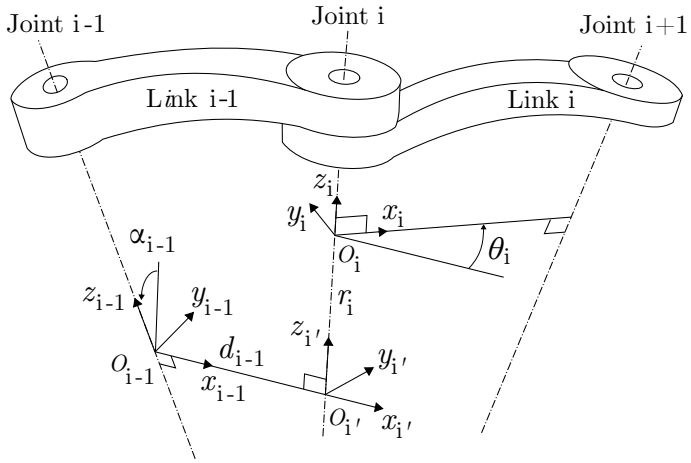


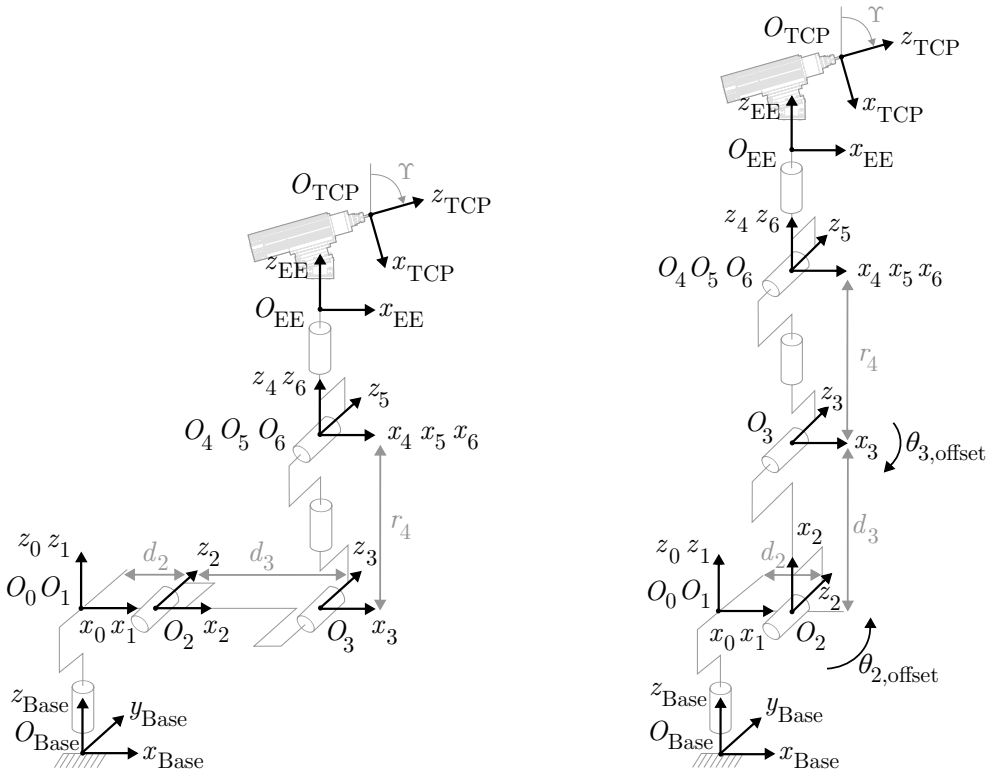
Figure A.1: Modified Denavit-Hartenberg convention

The modified convention can also be transposed in terms of successive multiplications of homogeneous transformation matrices as follows

$$\mathbf{T}_{\mathrm{i\text{-}1,i}} = \mathbf{T}_{\mathrm{rotx}}(\alpha_{\mathrm{i\text{-}1}}) \cdot \mathbf{T}_{\mathrm{disp}}(d_{\mathrm{i\text{-}1}}, 0, 0) \cdot \mathbf{T}_{\mathrm{rotz}}(\theta_{\mathrm{i}}) \cdot \mathbf{T}_{\mathrm{disp}}(0, 0, r_{\mathrm{i}}). \tag{A.1}$$

Note that the order of operations was changed and now involves parameters $\alpha_{\mathrm{i\text{-}1}}$ and $d_{\mathrm{i\text{-}1}}$.

The modified DH parameters of the Stäubli TX200 robot are defined using the same geometric pose (Figure A.2a). The numbering of the joint frame was changed with respect to the adopted convention. Similarly to the standard convention, the modified convention starts from robot frame O_0 to frame O_6. Therefore, the pose of frame O_6 is defined. In Figure A.2b, the Stäubli TX200 robot was positioned in its encoder pose by adding the joint variable offset $\mathbf{\Theta}_{\mathrm{Offset}} = [0 \ -\frac{\pi}{2} \ \frac{\pi}{2} \ 0 \ 0 \ 0]^T$.



(a) Geometric pose (b) Encoder pose

Figure A.2: Modified Denavit-Hartenberg convention applied to the Stäubli TX200 robot

The determination of the modified DH table is achieved in the same way as the standard one. Using the modified convention, frames O_0 and O_1 are coincident. Thus, no motion is needed between the two frames. Frame O_2 is at a distance d_2 from frame O_1 along x_1. As before, the rotation axis of the second joint is aligned with direction z_2. Frame O_3 is at a distance d_3 from frame O_2 along x_2 without rotation. Finally, the centre point of the spherical wrist is positioned at a distance r_4 from frame O_3 along the rotation axis of the fourth joint in the direction z_4. Axes z_5 and z_6 are oriented such that they correspond to the axes of rotation of joints five and six, respectively. Table A.1 summarises the modified DH parameters for the Stäubli TX200 robot.

i	d_i [m]	α_i [rad]	r_i [m]	θ_i [rad]
1	0	0	0	θ_1
2	0.25	-$\pi/2$	0	θ_2
3	0.95	0	0	θ_3
4	0	$\pi/2$	0.8	θ_4
5	0	-$\pi/2$	0	θ_5
6	0	$\pi/2$	0	θ_6

Table A.1: Stäubli TX200 modified Denavit-Hartenberg parameters

Using either the standard or the modified DH convention, the complete direct kinematics problem is solved by successively multiplying the homogeneous transformation matrices from frame O_0 to frame O_6 without omitting the base and tool transformations

such as

$$\mathbf{T}_{\text{Base,TCP}}(\mathbf{\Theta}) = \mathbf{T}_{\text{Base},0} \cdot \mathbf{T}_{0,1} \cdot \mathbf{T}_{1,2} \cdot \mathbf{T}_{2,3} \cdot \mathbf{T}_{3,4} \cdot \mathbf{T}_{4,5} \cdot \mathbf{T}_{5,6} \cdot \mathbf{T}_{6,\text{EE}} \cdot \mathbf{T}_{\text{EE,TCP}}. \quad \text{(A.2)}$$

Detail of the elementary homogeneous transformation matrices from frame O_0 to frame O_6 are provided below, knowing that $\mathbf{T}_{\text{Base},0}$, $\mathbf{T}_{6,\text{EE}}$ and $\mathbf{T}_{\text{EE,TCP}}$ are the same as defined in Chapter 3 (Eqs. 3.15 and 3.16), such as

$$\begin{aligned}
\mathbf{T}_{0,1} &= \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \mathbf{T}_{1,2} &= \begin{bmatrix} c_2 & -s_2 & 0 & d_2 \\ 0 & 0 & 1 & 0 \\ -s_2 & -c_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
\mathbf{T}_{2,3} &= \begin{bmatrix} c_3 & -s_3 & 0 & d_3 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \mathbf{T}_{3,4} &= \begin{bmatrix} c_4 & -s_4 & 0 & 0 \\ 0 & 0 & -1 & -r_4 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
\mathbf{T}_{4,5} &= \begin{bmatrix} c_5 & -s_5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s_5 & -c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \mathbf{T}_{5,6} &= \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
\end{aligned} \quad \text{(A.3)}$$

In a more general and compact form, the resulting pose of frame O_6 can be expressed as

$$\mathbf{T}_{0,6} = \begin{bmatrix} S_x & N_x & A_x & p_x \\ S_y & N_y & A_y & p_y \\ S_z & N_z & A_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad \text{(A.4)}$$

A.2 Inverse kinematics derivation

Using the Paul's method to solve the inverse kinematics problem for the Stäubli TX200 robot, joint angles are computed in ascending order from θ_1 to θ_6 as shown in Figure A.3.

The inverse kinematics problem is solved by determining a valid set of joint variables $\mathbf{\Theta}$ corresponding to the desired TCP pose such that $\mathbf{\Theta} = f^{-1}(\mathbf{x}_{\text{TCP}})$. Since the inverse problem uses the modified DH parameters, it is defined for the manipulator from frame O_0 to frame O_6 in the geometric pose. In that case, the base and tool transformations must first be removed from the expression providing the direct kinematics as follows

$$\mathbf{T}_{0,6} = \mathbf{T}_{\text{Base},0}^{-1} \cdot \mathbf{T}_{\text{Base,TCP}} \cdot (\mathbf{T}_{6,\text{EE}} \cdot \mathbf{T}_{\text{EE,TCP}})^{-1}. \quad \text{(A.5)}$$

The solution to the inverse problem is divided into two parts. First, the three angles determining the position of the centre point of the spherical wrist are computed i.e. joint

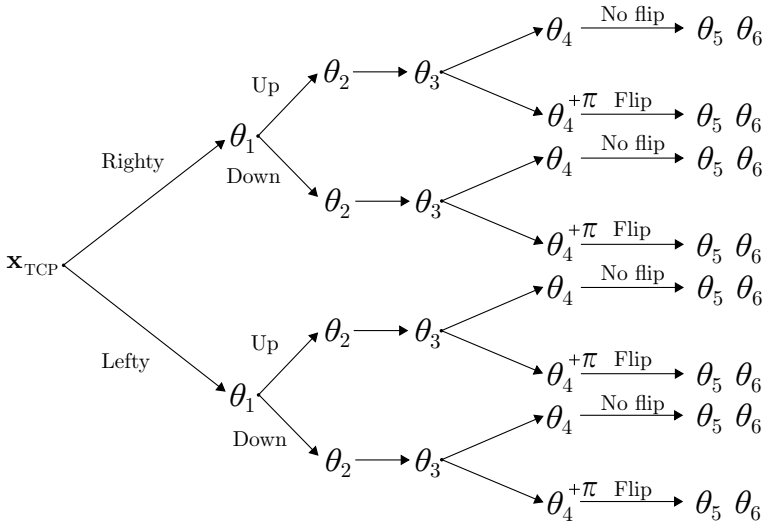


Figure A.3: Possible solutions to the inverse kinematics problem for an industrial robot

angles θ_1, θ_2 and θ_3 are found. Then, joint angles θ_4, θ_5 and θ_6 providing the orientation of frame O_6 are computed.

1. Position of the spherical wrist

To solve for the spherical wrist position, the application of the Paul's method to find a valid set of joint variables consists in successively pre-multiplying by homogeneous transformation matrix $\mathbf{T}_{\text{i+1,i}}$ the left hand side of equation A.5 to determine the first three joint angles θ_1, θ_2 and θ_3. It allows finding relationships with only one unknown which is the sought angle. From equations A.3 and A.5, the position of the spherical wrist can be expressed as follows

$$\begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} = \mathbf{T}_{0,1} \cdot \mathbf{T}_{1,2} \cdot \mathbf{T}_{2,3} \cdot \mathbf{T}_{3,4} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}. \tag{A.6}$$

• **Resolution for θ_1**

By pre-multiplying equation A.6 by $\mathbf{T}_{1,0}$ as follows

$$\mathbf{T}_{1,0} \cdot \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} = \mathbf{T}_{1,0} \cdot \mathbf{T}_{0,1} \cdot \mathbf{T}_{1,2} \cdot \mathbf{T}_{2,3} \cdot \mathbf{T}_{3,4} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \tag{A.7}$$

three equalities can be derived

1. $p_x c_1 + p_y s_1 = d_3 c_2 + r_4 s_{23} + d_2$,
2. $-p_x s_1 + p_y c_1 = 0$,
3. $p_z = -d_3 s_2 + r_4 c_{23}$.

The second equality is a particular type of equation in $Xs_\mathrm{i} + Yc_\mathrm{i} = Z$ whose solution is $\theta_\mathrm{i} = \mathrm{atan2}(s_\mathrm{i}, c_\mathrm{i})$, in which c_i and s_i are expressed as follows

$$\begin{cases} s_\mathrm{i} = \dfrac{XZ + \eta Y\sqrt{X^2+Y^2-Z^2}}{X^2+Y^2} \\ c_\mathrm{i} = \dfrac{XZ - \eta Y\sqrt{X^2+Y^2-Z^2}}{X^2+Y^2}. \end{cases} \tag{A.8}$$

For the Stäubli TX200 robot

$$\begin{cases} s_1 = \dfrac{+\eta p_y\sqrt{p_x^2+p_y^2}}{p_x^2+p_y^2} \\ c_1 = \dfrac{-\eta p_y\sqrt{p_x^2+p_y^2}}{p_x^2+p_y^2}, \end{cases} \tag{A.9}$$

which leads to $\theta_1 = \mathrm{atan2}(s_1, c_1)$. A singularity happens when $p_x = p_y = 0$ that is when the targeted pose crosses the first robot axis. Then, an infinite number of solutions for θ_1 arises (shoulder singularity). Variable η can take value +1 or -1 and influences the robot configuration. If η=+1, the robot is in a righty configuration otherwise it is in a lefty configuration.

• **Resolution for θ_2**

By pre-multiplying equation A.6 by $\mathbf{T}_{2,1}$ as follows

$$\mathbf{T}_{2,1} \cdot \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} = \mathbf{T}_{2,1} \cdot \mathbf{T}_{0,1} \cdot \mathbf{T}_{1,2} \cdot \mathbf{T}_{2,3} \cdot \mathbf{T}_{3,4} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \tag{A.10}$$

three other equalities can be derived

1. $-p_z s_2 - d_2 c_2 + p_x c_1 c_2 + p_y c_2 s_1 = r_4 s_3 + d_3$,
2. $-p_z c_2 + d_2 s_2 - p_x s_2 - p_x c_1 s_2 - p_y s_1 s_2 = -r_4 c_3$,
3. $-p_x s_1 + p_y c_1 = 0$.

From the first and second new equalities, terms $r_4 s_3$ and $r_4 c_3$ can be extracted such as

$$\begin{cases} \underbrace{r_4}_{\mathrm{W}}\, s_3 = \underbrace{(p_x c_1 + p_y s_1 - d_2)}_{\mathrm{X}}\, c_2 + \underbrace{(-p_z)}_{\mathrm{Y}}\, s_2 - \underbrace{d_3}_{\mathrm{Z}} \\ \underbrace{r_4}_{\mathrm{W}}\, c_3 = \underbrace{(p_x c_1 + p_y s_1 - d_2)}_{\mathrm{X}}\, s_2 - \underbrace{(-p_z)}_{\mathrm{Y}}\, c_2. \end{cases} \tag{A.11}$$

By squaring and adding the two previous equations to eliminate θ_3, the following equation can be found

$$\underbrace{-2ZY}_{B_1}\, s_2 + \underbrace{-2ZX}_{B_2}\, c_2 = \underbrace{W^2 - X^2 - Y^2 - Z^2}_{B_3}, \tag{A.12}$$

which has the same solution as presented in equation A.8. Then, θ_2 can be solved by θ_2 = $\mathrm{atan2}(s_2, c_2)$ in which

$$\begin{cases} s_2 = \dfrac{B_1 B_3 + \upsilon(-B_2)\sqrt{B_1^2 + B_2^2 - B_3^2}}{B_1^2 + B_2^2} \\ c_2 = \dfrac{B_2 B_3 - \upsilon B_1 \sqrt{B_1^2 + B_2^2 - B_3^2}}{B_1^2 + B_2^2}. \end{cases} \tag{A.13}$$

A singularity can arise if B_1=B_2=0 meaning that the manipulator is completely outstretched or retracted (elbow singularity). Variable υ can take value +1 or -1 and also influences the robot configuration. If υ=+1, the robot arm is pointing to an upwards direction otherwise it is pointing to a downwards direction.

• **Resolution for θ_3**

Knowing θ_2, the resolution for θ_3 is straightforward with equation A.11, that is, $\theta_3 = \text{atan2}(s_3, c_3)$ in which

$$\begin{cases} s_3 = \dfrac{X c_2 + Y s_2 - Z}{W} \\ c_3 = \dfrac{X s_2 - Y c_2}{W}. \end{cases} \tag{A.14}$$

2. Orientation of the spherical wrist

To solve for the spherical wrist orientation and thus determining θ_4, θ_5 and θ_6, the application of the Paul's method also imply to pre-multiply by homogeneous transformation matrix $\mathbf{T}_{\text{i+1,i}}$. From equation A.4, it can be recalled that the final rotation matrix orienting the spherical wrist and depending on all joint angles (θ_1 to θ_6) can be expressed as follows

$$\mathbf{R}_{0,6} = [SNA]. \tag{A.15}$$

Since the first three angles are known, it is advised to pose matrix $[FGH]$ which contains the last three unknowns

$$\mathbf{R}_{3,0}(\theta_1, \theta_2, \theta_3) \cdot [SNA] = \mathbf{R}_{3,6}(\theta_4, \theta_5, \theta_6) = [FGH]. \tag{A.16}$$

The idea is to find equalities between matrix $[FGH]$ expressed with the known angles (θ_1, θ_2, θ_3) and the same matrix expressed with (θ_4, θ_5, θ_6). $\mathbf{R}_{3,0}$ is expressed as the product of the first three rotation matrices $\mathbf{R}_{3,0} = [R_{0,1} \cdot R_{1,2} \cdot R_{2,3}]^T$:

$$\mathbf{R}_{3,0} = \begin{bmatrix} c_1c_2c_3 - c_1s_2s_3 & s_1c_2c_3 - s_1s_2s_3 & -s_2c_3 - c_2s_3 \\ -c_1c_2s_3 - c_1s_2c_3 & -s_1c_2s_3 - s_1s_2c_3 & s_2s_3 - c_2c_3 \\ -s_1 & c_1 & 0 \end{bmatrix}. \tag{A.17}$$

The developed form of matrix $[FGH]$ depending on the first three angles can then be generated such as

$$\mathbf{R}_{3,0} \cdot [SNA] = \begin{bmatrix} F_x & G_x & H_x \\ F_y & G_y & H_y \\ F_z & G_z & H_z \end{bmatrix}$$
$$= \begin{bmatrix} c_{23}(S_x c_1 + S_y s_1) - s_{23} S_z & c_{23}(N_x c_1 + N_y s_1) - s_{23} N_z & c_{23}(A_x c_1 + A_y s_1) - s_{23} A_z \\ -s_{23}(S_x c_1 + S_y s_1) - c_{23} S_z & -s_{23}(N_x c_1 + N_y s_1) - c_{23} N_z & -s_{23}(A_x c_1 + A_y s_1) - c_{23} A_z \\ -S_x s_1 + S_y c_1 & -N_x s_1 + N_y c_1 & -A_x s_1 + A_y c_1 \end{bmatrix}. \quad \text{(A.18)}$$

• **Resolution for θ_4**

Pre-multiplying by $\mathbf{R}_{4,3}$ matrix $[FGH]$ allows finding an expression in θ_4 only depending on the three known angles. The developed terms of calculus $\mathbf{R}_{4,3} \cdot [FGH] = \mathbf{R}_{4,6}$ are expressed below

$$\mathbf{R}_{4,3} \cdot [FGH] = \begin{bmatrix} c_4 F_x + s_4 F_z & c_4 G_x + s_4 G_z & c_4 H_x + s_4 H_z \\ -s_4 F_x + c_4 F_z & -s_4 G_x + c_4 G_z & -s_4 H_x + c_4 H_z \\ -F_y & -G_y & -H_y \end{bmatrix}, \quad \text{(A.19)}$$

$$\mathbf{R}_{4,6} = \begin{bmatrix} c_5 c_6 & -c_5 c_6 & s_5 \\ s_6 & c_6 & 0 \\ -s_5 c_6 & s_5 s_6 & c_5 \end{bmatrix}. \quad \text{(A.20)}$$

The sought expression in the unknown θ_4 is then found through the equality of element [2,3] in the two developed terms above: $-s_4 H_x + c_4 H_z = 0$ with H_x and H_z only depending on the first three angles as shown in equation A.18 such as

1. $H_x = c_{23}(A_x c_1 + A_y s_1) - A_z s_{23}$,
2. $H_z = -A_x s_1 + A_y c_1$.

The two solutions for θ_4 are eventually found

$$\begin{cases} \theta_4 = \text{atan2}(H_z, H_x), \\ \theta_{4'} = \text{atan2}(H_z, H_x) + \pi. \end{cases} \quad \text{(A.21)}$$

In the first solution, the wrist is said *not flipped* (natural positioning of the wrist) while the other solution leads to a *flipped* wrist.

• **Resolution for θ_5**

Similarly, a solution for angle θ_5 can be found through the equalities of elements [1,3] and [3,3] from the developed terms A.19 and A.20 such as

$$\begin{cases} s_5 = H_x c_4 + H_z s_4, \\ c_5 = -H_y, \end{cases} \quad \text{(A.22)}$$

with $H_y = -s_{23}(A_x c_1 + A_y s_1) - A_z c_{23}$.

The unique angle θ_5 is computed as follows

$$\theta_5 = \text{atan2}(s_5, c_5). \tag{A.23}$$

Singularity arises when $\theta_5 = 0$, that is, when the rotation axes of the forearm and the flange are aligned since an infinity of combinations of θ_4 and θ_6 leads to the targeted TCP pose. In case of singularity, last angle θ_6 is set to its previous value, just before the singularity.

• **Resolution for θ_6**

Again, the solution for angle θ_6 is derived from the equalities of elements [2,1] and [2,2] from the developed terms A.19 and A.20 such as

$$\begin{cases} s_6 = -F_x s_4 + F_z c_4, \\ c_6 = -G_x s_4 + G_z c_4, \end{cases} \tag{A.24}$$

with

1. $F_x = c_{23}(S_x c_1 + S_y s_1) - S_z s_{23}$,
2. $F_z = -S_x s_1 + S_y c_1$,
3. $G_x = c_{23}(N_x c_1 + N_y s_1) - N_z s_{23}$,
4. $G_z = -N_x s_1 + N_y c_1$.

The complete inverse geometric model of the Stäubli TX200 robot is finally solved with the computation of the last angle

$$\theta_6 = \text{atan2}(s_6, c_6). \tag{A.25}$$

Having computed a valid set of joint angles θ_1 to θ_6 referenced in the geometric pose of the manipulator, it remains to add the offset angles $\boldsymbol{\Theta}_{\text{Offset}}$ to retrieve the same joint angles as the ones read by the robot encoders.

Appendix B

Corotational formulation detail

B.1 Kinetic energy with the corotational formulation

Using the corotational formulation, kinetic energy $\mathcal{T}_{\mathrm{f}}$ of flexible body f can be set under the following form

$$\mathcal{T}_{\mathrm{f}} = \frac{1}{2}\sum_{\mathrm{f}=1}^{n_{\mathrm{F}}}\sum_{\mathrm{g}=1}^{\mathrm{N}}\sum_{\mathrm{h}=1}^{\mathrm{N}}\Big[\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\cdot[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\cdot[\mathbf{M}_{\mathrm{f,T_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\cdot[\mathbf{M}_{\mathrm{f,R_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\cdot[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\Big], \tag{B.1}$$

where subscripts g and h are related to the nodes of the flexible body, n_{F} is related to the number of flexible bodies and N is the number of nodes pertaining to flexible body f. Curly brackets for vectors $\{\bullet\}_{\mathrm{f}*}$ and square brackets for matrices $[\bullet]_{\mathrm{f}*}$ mean that the quantity is projected in the so-called corotational frame $O_{\mathrm{f}*}$. It is assumed that matrix $\mathbf{M}_{\mathrm{f}}$ is constant in corotational frame $O_{\mathrm{f}*}$ which rotates with flexible body f (Figure B.1).

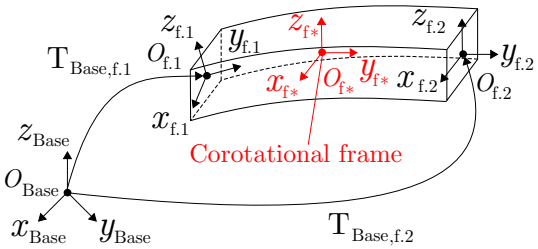


Figure B.1: Flexible beam

Each sub-matrix of mass matrix $\mathbf{M}_{\mathrm{f}}$ can be considered as a (3×3) tensor of the following form

$$\mathbf{M}_{\mathrm{f}} = \begin{bmatrix} [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*} & [\mathbf{M}_{\mathrm{f,T_g,R_g}}]_{\mathrm{f}*} & \vdots & [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*} & [\mathbf{M}_{\mathrm{f,T_g,R_h}}]_{\mathrm{f}*} \\ [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*} & [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} & \vdots & [\mathbf{M}_{\mathrm{f,R_g,T_h}}]_{\mathrm{f}*} & [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*} & [\mathbf{M}_{\mathrm{f,T_h,R_g}}]_{\mathrm{f}*} & \vdots & [\mathbf{M}_{\mathrm{f,T_h,T_h}}]_{\mathrm{f}*} & [\mathbf{M}_{\mathrm{f,T_h,R_h}}]_{\mathrm{f}*} \\ [\mathbf{M}_{\mathrm{f,R_h,T_g}}]_{\mathrm{f}*} & [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*} & \vdots & [\mathbf{M}_{\mathrm{f,R_h,T_h}}]_{\mathrm{f}*} & [\mathbf{M}_{\mathrm{f,R_h,R_h}}]_{\mathrm{f}*} \end{bmatrix}. \tag{B.2}$$

Kinetic energy (Eq. B.1) can be projected in an arbitrary frame, as far as the tensor variance is applied. As for illustration for sub-matrix $[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}$, its projection in corotational frame $O_{\mathrm{f}*}$ frame reads

$$[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*} = \mathbf{R}_{\mathrm{Base,f}*}\mathbf{M}_{\mathrm{f,R_g,R_h}}\mathbf{R}^{T}_{\mathrm{Base,f}*}. \tag{B.3}$$

B.2 Application of the Lagrange's theorem

Lagrange's theorem is expressed as follows

$$-\boldsymbol{\xi}_l(-m\mathbf{a}) = \frac{d}{dt}(\frac{\partial \mathcal{T}_\mathrm{f}}{\partial \dot{q}_l}) - \frac{\partial \mathcal{T}_\mathrm{f}}{\partial q_l}. \tag{B.4}$$

The aim is to find the reactions of inertia of generalised force $\boldsymbol{\xi}_l$ in degree of freedom q_l using the Lagrange's theorem by relying on the corotational formulation.

B.2.1 Pure translation contributions: $[\mathbf{M}_{\mathbf{f,T_g,T_g}}]_{\mathbf{f*}}$

The matrix $[\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}}$ is dependent on $\mathbf{q}$ but not on $\dot{\mathbf{q}}$. Using the Lagrange's theorem, the partial derivative of the kinetic energy with respect to generalised coordinate q_l is given by

$$\frac{\partial \mathcal{T}_\mathrm{f}}{\partial \dot{q}_l} = \frac{1}{2}(\frac{\partial \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{\partial \dot{q}_l} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}} \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} + \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}} \frac{\partial \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{\partial \dot{q}_l}), \tag{B.5}$$

with $\dfrac{\partial \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{\partial \dot{q}_l} = \mathbf{d}^{\mathrm{f.g,l}}$ the partial contribution of $\dot{q}_l$ in the translational velocity of node g of flexible body f. As $[\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}}$ is symmetric, Eq. B.5 becomes

$$\frac{\partial \mathcal{T}_\mathrm{f}}{\partial \dot{q}_l} = \mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}} \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}. \tag{B.6}$$

By taking the time derivative with respect to an arbitrary frame $O_{\mathrm{f*}}$, assuming $\mathcal{T}_\mathrm{f}$ is a scalar, the first term of the Lagrange's theorem reads

$$\left.\frac{d}{dt}\frac{\partial \mathcal{T}_\mathrm{f}}{\partial \dot{q}_l}\right|_{O_{\mathrm{f*}}} = \left.\frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f*}}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}} \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} + \mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}} \left.\frac{d\,\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{dt}\right|_{O_{\mathrm{f*}}}, \tag{B.7}$$

with $\dfrac{d(\mathbf{d}^{\mathrm{f.g,l}})}{dt} = \sum_{\mathrm{k}}^{n} \dfrac{\partial \mathbf{d}^{\mathrm{f.g,l}}}{\partial q_\mathrm{k}} \dot{q}_\mathrm{k}$.
With further development, it becomes

$$\left.\frac{d}{dt}\frac{\partial \mathcal{T}_\mathrm{f}}{\partial \dot{q}_l}\right|_{O_{\mathrm{f*}}} = \sum_{\mathrm{k}}^{n} \left.\frac{\partial \mathbf{d}^{\mathrm{f.g,l}}}{\partial q_\mathrm{k}}\right|_{O_{\mathrm{f*}}} \dot{q}_\mathrm{k} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}} \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} + \mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}} \left.\frac{d\,\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{dt}\right|_{O_{\mathrm{f*}}}. \tag{B.8}$$

Second term of Lagrange's theorem, assuming that $[\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}}$ is symmetric, reads

$$\left.\frac{\partial \mathcal{T}_\mathrm{f}}{\partial q_l}\right|_{O_{\mathrm{f*}}} = \left.\frac{\partial \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{\partial q_l}\right|_{O_{\mathrm{f*}}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f*}} \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}, \tag{B.9}$$

with $\dfrac{\partial \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{\partial q_l} = \dfrac{\partial \sum_{\mathrm{k}}^{n} \mathbf{d}^{\mathrm{f.g,k}}}{\partial q_l} \dot{q}_\mathrm{k} = \sum_{\mathrm{k}}^{n} \dfrac{\partial \mathbf{d}^{\mathrm{f.g,k}}}{\partial q_l} \dot{q}_\mathrm{k}$.
With further development, it becomes

$$\left.\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} = \sum_{\mathrm{k}}^{n} \left.\frac{\partial \mathbf{d}^{\mathrm{f.g,k}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} \dot{q}_{\mathrm{k}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*} \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}. \tag{B.10}$$

The Lagrange's theorem is computed by subtracting equations (B.8) - (B.10):

$$\begin{aligned}\left.\frac{d}{dt}\left(\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l}\right)\right|_{O_{\mathrm{f}*}} - \left.\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} =& \mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*} \left.\frac{d\,\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} + \\ & \left(\sum_{\mathrm{k}}^{n} \left.\frac{\partial \mathbf{d}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}\right|_{O_{\mathrm{f}*}} \dot{q}_{\mathrm{k}} - \sum_{\mathrm{k}}^{n} \left.\frac{\partial \mathbf{d}^{\mathrm{f.g,k}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} \dot{q}_{\mathrm{k}}\right) \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*} \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}.\end{aligned} \tag{B.11}$$

One reminds here the theorem of the derivative composition (Eqs. B.12 and B.13):

$$\left.\frac{d\mathcal{F}}{dt}\right|_{O_{\mathrm{Base}}} = \left.\frac{d\mathcal{F}}{dt}\right|_{O_{\mathrm{S}}} + \boldsymbol{\omega}_{\mathrm{f.g}} \times \mathcal{F}, \tag{B.12}$$

$$\left.\frac{d\mathcal{F}}{dq_l}\right|_{O_{\mathrm{Base}}} = \left.\frac{d\mathcal{F}}{dq_l}\right|_{O_{\mathrm{S}}} + \frac{\partial \boldsymbol{\omega}_{\mathrm{f.g}}}{\partial \dot{q}_l} \times \mathcal{F} = \left.\frac{d\mathcal{F}}{dq_l}\right|_{O_{\mathrm{S}}} + \mathbf{w}^{\mathrm{f.g,l}} \times \mathcal{F}. \tag{B.13}$$

with $\mathcal{F}$ a (3×1) arbitrary vector and O_{S} an arbitrary frame, hence

- $\left.\frac{\partial \mathbf{d}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}\right|_{O_{\mathrm{f}*}} = \left.\frac{\partial \mathbf{d}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{k}} \times \mathbf{d}^{\mathrm{f.g,l}} = \frac{\partial^2 \mathbf{p}_{\mathrm{f.g}}}{\partial q_{\mathrm{k}} \partial q_l} - \mathbf{w}^{\mathrm{f}*,\mathrm{k}} \times \mathbf{d}^{\mathrm{f.g,l}};$
- $\left.\frac{\partial \mathbf{d}^{\mathrm{f.g,k}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} = \frac{\partial^2 \mathbf{p}_{\mathrm{f.g}}}{\partial q_{\mathrm{k}} \partial q_l} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \mathbf{d}^{\mathrm{f.g,k}};$
- $\left.\frac{d\,\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} = \left.\frac{d\,\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} = \{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}.$

By replacing in Eq. B.11, the generalised force $\boldsymbol{\xi}_l$ in degree of freedom q_l reads

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =& \mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*} (\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \\ & \left(\sum_{\mathrm{k}}^{n} \frac{\partial^2 \mathbf{p}_{\mathrm{f.g}}}{\partial q_{\mathrm{k}} \partial q_l} \dot{q}_{\mathrm{k}} - \sum_{\mathrm{k}}^{n} \mathbf{w}^{\mathrm{f}*,\mathrm{k}} \times \mathbf{d}^{\mathrm{f.g,l}} \dot{q}_{\mathrm{k}} - \right. \\ & \left. \sum_{\mathrm{k}}^{n} \frac{\partial^2 \mathbf{p}_{\mathrm{f.g}}}{\partial q_{\mathrm{k}} \partial q_l} \dot{q}_{\mathrm{k}} + \sum_{\mathrm{k}}^{n} \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \mathbf{d}^{\mathrm{f.g,k}} \dot{q}_{\mathrm{k}}\right) \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*} \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}.\end{aligned} \tag{B.14}$$

After simplifying the summations of the mixed partial derivatives and knowing that

- $\sum_{\mathrm{k}}^{n} \mathbf{w}^{\mathrm{f}*,\mathrm{k}} \dot{q}_{\mathrm{k}} = \boldsymbol{\omega}_{\mathrm{f}*},$
- $\sum_{\mathrm{k}}^{n} \mathbf{d}^{\mathrm{f.g,k}} \dot{q}_{\mathrm{k}} = \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*},$

the following relationship is obtained

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})+ \\ &\left(-\boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{d}^{\mathrm{f.g,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right) \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}.\end{aligned} \tag{B.15}$$

After distributing the second term, it reads

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*}- \\ &\boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) - \left(\boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{d}^{\mathrm{f.g,l}}\right) \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}+ \\ &\left(\mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right) \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}.\end{aligned} \tag{B.16}$$

Using the properties of the triple scalar product, it reads

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*}- \\ &\boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) - \left[([\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \times \boldsymbol{\omega}_{\mathrm{f}*}\right] \cdot \mathbf{d}^{\mathrm{f.g,l}}+ \\ &\left[\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] \cdot \mathbf{w}^{\mathrm{f}*,\mathrm{l}}.\end{aligned} \tag{B.17}$$

By isolating $\mathbf{d}^{\mathrm{f.g,l}}$ and $\mathbf{w}^{\mathrm{f}*,\mathrm{l}}$, the relationship becomes

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{d}^{\mathrm{f.g,l}} \cdot \left[[\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] + \\ &\mathbf{w}^{\mathrm{f}*,\mathrm{l}} \cdot \left[\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right].\end{aligned} \tag{B.18}$$

The contribution of $[\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}$ is equivalent to

- a force on node g of flexible body f $O_{\mathrm{f.g}}$ equals to: $-[[\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})]$;
- a torque on the corotational frame $O_{\mathrm{f}*}$ equals to: $-[\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})]$.

B.2.2 Rigid case in translation

In the case of a rigid body f with one single node g, $[\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*} = m_{\mathrm{f}}\mathbf{I}$, $\boldsymbol{\omega}_{\mathrm{f}*} = \mathbf{0}$ and $\mathbf{w}^{\mathrm{f}*,\mathrm{l}} = \mathbf{0}$. One recognises a force applied on f equivalent to $\boldsymbol{\mathcal{R}}_{\mathrm{f.g}} = -m_{\mathrm{f}}\mathbf{I}\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*}$ by replacing in Eq. B.18.

Verification:

Starting from the translational part of the Eq. B.18 to find $\boldsymbol{\mathcal{R}}_{\mathrm{f.g}} = -m_{\mathrm{f}}\mathbf{I}\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*}$, the following condition must be satisfied

$$-\mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}(\boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \mathbf{d}^{\mathrm{f.g,l}} \cdot (\boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})) = 0. \tag{B.19}$$

As $[\mathbf{M}_{\mathrm{f,T_g,T_g}}]_{\mathrm{f}*} = m_{\mathrm{f}}\mathbf{I}$ for the rigid case, it leads to the following development

$$-\mathbf{d}^{\mathrm{f.g,l}} \cdot [m_{\mathrm{f}}\mathbf{I}(\boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) - \boldsymbol{\omega}_{\mathrm{f}*} \times (m_{\mathrm{f}}\mathbf{I}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})] = 0, \tag{B.20}$$

$$-\mathbf{d}^{\mathrm{f.g,l}} \cdot [m_{\mathrm{f}}(\boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) - \boldsymbol{\omega}_{\mathrm{f}*} \times (m_{\mathrm{f}}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})] = 0, \tag{B.21}$$

$$-\mathbf{d}^{\mathrm{f.g,l}} \cdot m_{\mathrm{f}} [\boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}] = 0 \quad \forall \boldsymbol{\omega}_{\mathrm{f}*}. \tag{B.22}$$

Finally, the generalised force $\boldsymbol{\xi}_l$ in degree of freedom q_l is expressed

$$\frac{d}{dt}\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l} - \frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial q_l} = \mathbf{d}^{\mathrm{f.g,l}} \cdot m_{\mathrm{f}} \mathbf{I} \{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*}, \tag{B.23}$$

as it was the case for the rigid case.

B.2.3 Pure rotation contributions: $[\mathbf{M}_{\mathbf{f,R_g,R_g}}]_{\mathbf{f}*}$

The partial derivative of the kinetic energy with respect to generalised coordinate q_l is given by Eq. B.24. Considering that matrix $[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}$ is symmetric, it leads to

$$\begin{aligned} \frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l} =& \frac{1}{2}\frac{\partial \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial \dot{q}_l} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} + \frac{1}{2}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \frac{\partial \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial \dot{q}_l} \\ =& \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}, \end{aligned} \tag{B.24}$$

with $\dfrac{\partial \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial \dot{q}_l} = \mathbf{w}^{\mathrm{f.g,l}}$ the partial contribution of $\dot{q}_l$ in the rotational velocity of node g of flexible body f. By taking the time derivative with respect to corotational frame $O_{\mathrm{f}*}$ assuming that kinetic energy $\mathcal{T}$ is a scalar, the first term of the Lagrange's theorem is obtained for the rotational part as

$$\left.\frac{d}{dt}\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l}\right|_{O_{\mathrm{f}*}} = \left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} + \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}}, \tag{B.25}$$

with $\dfrac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt} = \displaystyle\sum_{\mathrm{k}}^{n} \frac{\partial \mathbf{w}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}} \dot{q}_{\mathrm{k}}$.
By substitution, it leads to

$$\left.\frac{d}{dt}\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l}\right|_{O_{\mathrm{f}*}} = \sum_{\mathrm{k}}^{n} \left.\frac{\partial \mathbf{w}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}\right|_{O_{\mathrm{f}*}} \dot{q}_{\mathrm{k}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} + \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} \tag{B.26}$$

Second term of the Lagrange's theorem, by considering that matrix $[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}$ is symmetric, is expressed as

$$\begin{aligned} \left.\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} =& \frac{1}{2} \left.\frac{\partial \vec{\omega}_i^T}{\partial q_l}\right|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} + \frac{1}{2}\vec{\omega}_i^T \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \left.\frac{\partial \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}} \\ =& \left.\frac{\partial \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}. \end{aligned} \tag{B.27}$$

with $\dfrac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l} = \sum_{\mathrm{k}}^{n} \dfrac{\partial \mathbf{w}^{\mathrm{f.g,k}}}{\partial q_l}\dot{q}_{\mathrm{k}}$.
By substitution, it leads to

$$\left.\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} = \sum_{\mathrm{k}}^{n} \left.\frac{\partial \mathbf{w}^{\mathrm{f.g,k}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} \dot{q}_{\mathrm{k}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}. \tag{B.28}$$

The complete expression of the Lagrange's theorem is computed by subtracting Eqs. (B.26) - (B.28) such as

$$\begin{aligned}\left.\frac{d}{dt}\left(\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l}\right)\right|_{O_{\mathrm{f}*}} - \left.\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} =& \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} + \\ & \left(\sum_{\mathrm{k}}^{n} \left.\frac{\partial \mathbf{w}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}\right|_{O_{\mathrm{f}*}} \dot{q}_{\mathrm{k}} - \sum_{\mathrm{k}}^{n} \left.\frac{\partial \mathbf{w}^{\mathrm{f.g,k}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} \dot{q}_{\mathrm{k}}\right) \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*},\end{aligned} \tag{B.29}$$

with

- $\left.\dfrac{\partial \mathbf{w}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}\right|_{O_{\mathrm{f}*}} = \left.\dfrac{\partial \mathbf{w}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{k}} \times \mathbf{w}^{\mathrm{f.g,l}}$,
- $\left.\dfrac{\partial \mathbf{w}^{\mathrm{f.g,k}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} = \left.\dfrac{\partial \mathbf{w}^{\mathrm{f.g,k}}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \mathbf{w}^{\mathrm{f.g,k}}$,
- $\left.\dfrac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} = \left.\dfrac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}} - (\boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})$.

By replacing in Eq. B.29 the derived relationships, it leads to generalised force

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =& \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \left(\left.\frac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}} - (\boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right) + \\ & \left(\sum_{\mathrm{k}}^{n} \left.\frac{\partial \mathbf{w}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}\right|_{O_{\mathrm{Base}}} \dot{q}_{\mathrm{k}} - \sum_{\mathrm{k}}^{n} \mathbf{w}^{\mathrm{f}*,\mathrm{k}} \times \mathbf{w}^{\mathrm{f.g,l}}\dot{q}_{\mathrm{k}} - \right. \\ & \left. \sum_{\mathrm{k}}^{n} \left.\frac{\partial \mathbf{w}^{\mathrm{f.g,k}}}{\partial q_l}\right|_{O_{\mathrm{Base}}} \dot{q}_{\mathrm{k}} + \sum_{\mathrm{k}}^{n} \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \mathbf{w}^{\mathrm{f.g,k}}\dot{q}_{\mathrm{k}}\right) \cdot [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*},\end{aligned} \tag{B.30}$$

with

- $\sum_{\mathrm{k}}^{n} \mathbf{w}^{\mathrm{f}*,\mathrm{k}}\dot{q}_{\mathrm{k}} = \boldsymbol{\omega}_{\mathrm{f}*}$,
- $\sum_{\mathrm{k}}^{n} \mathbf{w}^{\mathrm{f}*,\mathrm{l}}\dot{q}_{\mathrm{k}} = \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}$.

By substitution, it leads to

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}}\cdot[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\left(\left.\frac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}}-(\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right)+\\&\left[\sum_{\mathrm{k}}^{n}\left(\left.\frac{\partial\mathbf{w}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}\right|_{O_{\mathrm{Base}}}-\left.\frac{\partial\mathbf{w}^{\mathrm{f.g,k}}}{\partial q_l}\right|_{O_{\mathrm{Base}}}\right)\dot{q}_{\mathrm{k}}-\boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{w}^{\mathrm{f.g,l}}+\mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right]\cdot\\&[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*},\end{aligned}\tag{B.31}$$

with $\dfrac{\partial\mathbf{w}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}-\dfrac{\partial\mathbf{w}^{\mathrm{f.g,k}}}{\partial q_l}=\mathbf{w}^{\mathrm{f.g,k}}\times\mathbf{w}^{\mathrm{f.g,l}}$.
Hence, the generalised force is rewritten as

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}}\cdot[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\left(\left.\frac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}}-(\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right)+\\&\left[\sum_{\mathrm{k}}^{n}\left(\mathbf{w}^{\mathrm{f.g,k}}\times\mathbf{w}^{\mathrm{f.g,l}}\right)\dot{q}_{\mathrm{k}}-\boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{w}^{\mathrm{f.g,l}}+\mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right]\cdot\\&[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*},\end{aligned}\tag{B.32}$$

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}}\cdot[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\left(\left.\frac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}}-(\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right)+\\&\left(\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times\mathbf{w}^{\mathrm{f.g,l}}-\boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{w}^{\mathrm{f.g,l}}+\mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\cdot[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*},\end{aligned}\tag{B.33}$$

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}}\cdot[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\left(\left.\frac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}}-(\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right)+\\&\left(\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times\mathbf{w}^{\mathrm{f.g,l}}\right)\cdot[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}-\\&\left(\boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{w}^{\mathrm{f.g,l}}\right)\cdot[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}+\\&\left(\mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\cdot[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}.\end{aligned}\tag{B.34}$$

By using the properties of the triple scalar product, it leads to

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}}\cdot[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\left(\left.\frac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}}-(\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right)+\\&\left([\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\cdot\mathbf{w}^{\mathrm{f.g,l}}-\\&\left([\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times\boldsymbol{\omega}_{\mathrm{f}*}\right)\cdot\mathbf{w}^{\mathrm{f.g,l}}+\\&\left(\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\cdot\mathbf{w}^{\mathrm{f}*,\mathrm{l}}.\end{aligned}\tag{B.35}$$

By isolating $\mathbf{w}^{\mathrm{f.g,l}}$ and $\mathbf{w}^{\mathrm{f}*,\mathrm{l}}$, it leads to

$$
\begin{aligned}
-\boldsymbol{\xi}_l(-m\mathbf{a}) =& \mathbf{w}^{\mathrm{f.g,l}} \cdot \left[[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \left(\left. \frac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt} \right|_{O_{\mathrm{Base}}} - (\boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \right) - \right. \\
& \left. \left(\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right) + \left(\boldsymbol{\omega}_{\mathrm{f}*} \times [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right) \right] + \\
& \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \cdot \left[\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right].
\end{aligned}
\tag{B.36}
$$

The contribution of $[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}$ is equivalent to:

- a torque on node g of flexible body f equals to:
  $-[[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} \left(\left. \frac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt} \right|_{O_{\mathrm{Base}}} - (\boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \right) +$
  $(\boldsymbol{\omega}_{\mathrm{f}*} - \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \times \left([\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)]$,
- a torque on corotational frame $O_{\mathrm{f}*}$ equals to: $-[\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times [\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}]$.

B.2.4 Rigid case in rotation

In the case of rigid body f with one single node g: $[\mathbf{M}_{\mathrm{f,R_g,R_g}}]_{\mathrm{f}*} = \boldsymbol{\Phi}_{\mathrm{G,f}}$, $\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} = \boldsymbol{\omega}_{\mathrm{f}*}$ and $\mathbf{w}^{\mathrm{f.g,l}} = \mathbf{w}^{\mathrm{f}*,\mathrm{l}}$. Starting from Eq. B.36, one replaces with the assumptions of the rigid case such as

$$
\begin{aligned}
-\boldsymbol{\xi}_l(-m\mathbf{a}) =& \mathbf{w}^{\mathrm{f.g,l}} \cdot \boldsymbol{\Phi}_{\mathrm{G,f}} \left[\left. \frac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt} \right|_{O_{\mathrm{Base}}} - (\underbrace{\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}_{=\mathbf{0}}) \right] + \\
& \mathbf{w}^{\mathrm{f.g,l}} \cdot \left[\underbrace{\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times \boldsymbol{\Phi}_{\mathrm{G,f}}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times \boldsymbol{\Phi}_{\mathrm{G,f}}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}_{=\mathbf{0}} + \right. \\
& \left. \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times \boldsymbol{\Phi}_{\mathrm{G,f}}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right].
\end{aligned}
\tag{B.37}
$$

It finally leads to the same expression as in the rigid case for the expression of the generalised force such as

$$
-\boldsymbol{\xi}_l(-m\mathbf{a}) = \mathbf{w}^{\mathrm{f.g,l}} \cdot \left(\boldsymbol{\Phi}_{\mathrm{G,f}} \left. \frac{d\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt} \right|_{O_{\mathrm{Base}}} + \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times \boldsymbol{\Phi}_{\mathrm{G,f}}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right). \tag{B.38}
$$

B.2.5 Contribution in translation for mixed terms: $[\mathbf{M}_{\mathbf{f,T_g,T_h}}]_{\mathbf{f}*}$

The expression of kinetic energy $\mathcal{T}_{\mathrm{f}}$ of flexible body f is given by

$$
\mathcal{T}_{\mathrm{f}} = \frac{1}{2}(\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}), \tag{B.39}
$$

$$
\mathcal{T}_{\mathrm{f}} = \frac{1}{2}(\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} + \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}[\mathbf{M}^{T}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}), \tag{B.40}
$$

$$\mathcal{T}_{\mathrm{f}} = \frac{1}{2}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot ([\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*} + [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*})\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}, \tag{B.41}$$

$$\mathcal{T}_{\mathrm{f}} = \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}, \tag{B.42}$$

with $[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*} = \frac{1}{2}([\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*} + [\mathbf{M}^T_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*})$, a skew-symmetric matrix.

The partial derivative of the kinetic energy with respect to generalised coordinate q_l is given by

$$\begin{aligned}\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l} &= \mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\mathbf{d}^{\mathrm{f.h,l}} \\ &= \mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \mathbf{d}^{\mathrm{f.h,l}} \cdot [\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}.\end{aligned} \tag{B.43}$$

By taking the time derivative, the first term of the Lagrange's theorem is obtained as

$$\begin{aligned}\frac{d}{dt}\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l}\bigg|_{O_{\mathrm{f}*}} &= \frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\bigg|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) \\ &+ \frac{d\,\mathbf{d}^{\mathrm{f.h,l}}}{dt}\bigg|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} + \mathbf{d}^{\mathrm{f.h,l}} \cdot [\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}).\end{aligned} \tag{B.44}$$

The second term of the Lagrange's theorem is expressed by

$$\begin{aligned}\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial q_l}\bigg|_{O_{\mathrm{f}*}} &= \frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\bigg|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\,\frac{\partial\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{\partial q_l}\bigg|_{O_{\mathrm{f}*}} \\ &= \frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\bigg|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \frac{\partial\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{\partial q_l}\bigg|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}.\end{aligned} \tag{B.45}$$

The Lagrange's theorem is computed by subtracting Eqs. (B.44) - (B.45)

$$\begin{aligned}\frac{d}{dt}\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l}\bigg|_{O_{\mathrm{f}*}} - \frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial q_l}\bigg|_{O_{\mathrm{f}*}} =& \mathbf{d}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ & \mathbf{d}^{\mathrm{f.h,l}} \cdot [\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \\ & \left[\frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\bigg|_{O_{\mathrm{f}*}} - \frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\bigg|_{O_{\mathrm{f}*}}\right] \cdot [\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \\ & \left[\frac{d\,\mathbf{d}^{\mathrm{f.h,l}}}{dt}\bigg|_{O_{\mathrm{f}*}} - \frac{\partial\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{\partial q_l}\bigg|_{O_{\mathrm{f}*}}\right] \cdot [\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*},\end{aligned} \tag{B.46}$$

with

•

$$\begin{aligned}\left[\left.\frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f}*}} - \left.\frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}}\right] &= \left(\left.\frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{d}^{\mathrm{f.g,l}}\right) - \\ &\quad \left(\left.\frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right) \\ &= \sum_{\mathrm{k}}^{n}\frac{\partial\mathbf{d}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}}\dot{q}_{\mathrm{k}} - \boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{d}^{\mathrm{f.g,l}} - \\ &\quad \sum_{\mathrm{k}}^{n}\frac{\partial\mathbf{d}^{\mathrm{f.g,k}}}{\partial q_l}\dot{q}_{\mathrm{k}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \\ &= -\boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{d}^{\mathrm{f.g,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*},\end{aligned}$$

•

$$\begin{aligned}\left[\left.\frac{d\,\mathbf{d}^{\mathrm{f.h,l}}}{dt}\right|_{O_{\mathrm{f}*}} - \left.\frac{\partial\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}}\right] &= \left(\left.\frac{d\,\mathbf{d}^{\mathrm{f.h,l}}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{d}^{\mathrm{f.h,l}}\right) - \\ &\quad \left(\left.\frac{\partial\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right) \\ &= \sum_{\mathrm{k}}^{n}\frac{\partial\mathbf{d}^{\mathrm{f.h,l}}}{\partial q_{\mathrm{k}}}\dot{q}_{\mathrm{k}} - \boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{d}^{\mathrm{f.h,l}} - \\ &\quad \sum_{\mathrm{k}}^{n}\frac{\partial\vec{d}_{j,k}}{\partial q_l}\dot{q}_{\mathrm{k}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} \\ &= -\boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{d}^{\mathrm{f.h,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}.\end{aligned}$$

By substituting the derived expressions, it leads to

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =& \mathbf{d}^{\mathrm{f.g,l}}\cdot[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}\times\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ & \left[-\boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{d}^{\mathrm{f.g,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right]\cdot[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \\ & \mathbf{d}^{\mathrm{f.h,l}}\cdot[\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}\times\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \\ & \left[-\boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{d}^{\mathrm{f.h,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right]\cdot[\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}.\end{aligned} \tag{B.47}$$

The distribution of terms $[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}$ and $[\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}$ leads to

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =& \mathbf{d}^{\mathrm{f.g,l}}\cdot\left[[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}\times\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*})\right] - \\ & \left(\boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{d}^{\mathrm{f.g,l}}\right)\cdot[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \left(\mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\cdot[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \\ & \mathbf{d}^{\mathrm{f.h,l}}\cdot\left[[\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}\times\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] - \\ & \left(\boldsymbol{\omega}_{\mathrm{f}*}\times\mathbf{d}^{\mathrm{f.h,l}}\right)\cdot[\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} + \left(\mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\cdot[\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}.\end{aligned} \tag{B.48}$$

The permutation property of the scalar triple product leads to

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{d}^{\mathrm{f.g,l}} \cdot \left[[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*})\right] - \\ &\left[([\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) \times \boldsymbol{\omega}_{\mathrm{f}*}\right] \cdot \mathbf{d}^{\mathrm{f.g,l}} + \left[\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*})\right] \cdot \mathbf{w}^{\mathrm{f}*,\mathrm{l}} + \\ &\mathbf{d}^{\mathrm{f.h,l}} \cdot \left[[\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] - \\ &\left[([\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \times \boldsymbol{\omega}_{\mathrm{f}*}\right] \cdot \mathbf{d}^{\mathrm{f.h,l}} + \left[\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] \cdot \mathbf{w}^{\mathrm{f}*,\mathrm{l}}.\end{aligned} \tag{B.49}$$

The separation of the terms for nodes g and h of flexible body f for the force and the torque leads to

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{d}^{\mathrm{f.g,l}} \cdot \left[[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*})\right] + \\ &\left[\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*})\right] \cdot \mathbf{w}^{\mathrm{f}*,\mathrm{l}} + \\ &\mathbf{d}^{\mathrm{f.h,l}} \cdot \left[[\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] + \\ &\left[\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] \cdot \mathbf{w}^{\mathrm{f}*,\mathrm{l}}.\end{aligned} \tag{B.50}$$

The contribution of $[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}$ is equivalent to:

- a force on node g of flexible body f equals to:
  $-\left[[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*})\right]$;
- a force on node h of flexible body f equals to:
  $-\left[[\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right]$;
- a torque on $O_{\mathrm{f}*}$ equals to:
  $-\left[\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right]$.

B.2.6 Contribution in rotation for mixed terms: $[\mathbf{M}_{\mathbf{f,R_g,R_h}}]_{\mathbf{f}*}$

Expression of the kinetic energy is given by

$$\begin{aligned}\mathcal{T}_{\mathrm{f}} &= \frac{1}{2}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \frac{1}{2}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \\ &= \frac{1}{2}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot ([\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*} + [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*})\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} \\ &= \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}.\end{aligned} \tag{B.51}$$

The contribution in the Lagrange's theorem is computed as the partial derivative of kinetic energy $\mathcal{T}_{\mathrm{f}}$ with respect to generalised coordinate q_l such as

$$\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l} = \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \mathbf{w}^{\mathrm{f.h,l}} \cdot [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}. \tag{B.52}$$

The first term of the Lagrange's theorem is obtained by taking the time derivative as

$$\begin{aligned}\frac{d}{dt}\left.\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l}\right|_{O_{\mathrm{f}*}} = &\left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*} \left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} + \\ &\left.\frac{d\,\mathbf{w}^{\mathrm{f.h,l}}}{dt}\right|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} + \mathbf{w}^{\mathrm{f.h,l}} \cdot [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*} \left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}}.\end{aligned} \tag{B.53}$$

The second term of the Lagrange's theorem is expressed by

$$\left.\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} = \left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}} \cdot [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}. \tag{B.54}$$

The Lagrange's theorem is computed by subtracting Eqs. (B.53) - (B.54):

$$\begin{aligned}\frac{d}{dt}\left.\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial \dot{q}_l}\right|_{O_{\mathrm{f}*}} - \left.\frac{\partial \mathcal{T}_{\mathrm{f}}}{\partial q_l}\right|_{O_{\mathrm{f}*}} = &\mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*} \left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} + \\ &\mathbf{w}^{\mathrm{f.h,l}} \cdot [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*} \left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} + \\ &\left[\left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f}*}} - \left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}}\right] \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \\ &\left[\left.\frac{d\,\mathbf{w}^{\mathrm{f.h,l}}}{dt}\right|_{O_{\mathrm{f}*}} - \left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}}\right] \cdot [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*},\end{aligned} \tag{B.55}$$

with

- $$\left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} = \left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*},$$
- $$\left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} = \left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*},$$
- $$\begin{aligned}\left[\left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f}*}} - \left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}}\right] = &\left(\left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}}\right) - \\ &\left(\left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right),\end{aligned}$$
- $$\begin{aligned}\left[\left.\frac{d\,\mathbf{w}^{\mathrm{f.h,l}}}{dt}\right|_{O_{\mathrm{f}*}} - \left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}}\right] = &\left(\left.\frac{d\,\mathbf{w}^{\mathrm{f.h,l}}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.h,l}}\right) - \\ &\left(\left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right).\end{aligned}$$

Then, by replacing in Eq. B.55, it leads to

$$
\begin{aligned}
-\boldsymbol{\xi}_l(-m\mathbf{a}) =& \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*} \left(\left. \frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{dt} \right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} \right) + \\
& \mathbf{w}^{\mathrm{f.h,l}} \cdot [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*} \left(\left. \frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt} \right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right) + \\
& \left[\left(\left. \frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt} \right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}} \right) - \left(\left. \frac{\partial \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l} \right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right) \right] \cdot \\
& [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*} \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \\
& \left[\left(\left. \frac{d\,\mathbf{w}^{\mathrm{f.h,l}}}{dt} \right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.h,l}} \right) - \left(\left. \frac{\partial \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{\partial q_l} \right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} \right) \right] \cdot \\
& [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*} \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*},
\end{aligned}
\tag{B.56}
$$

with for node g (the same applied for node h)

- $$
\begin{aligned}
& \left[\left(\left. \frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt} \right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}} \right) - \left(\left. \frac{\partial \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l} \right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right) \right] \\
&= \left. \frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt} \right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}} - \left. \frac{\partial \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l} \right|_{O_{\mathrm{Base}}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \\
&= \sum_{\mathrm{k}}^{n} \frac{\partial \mathbf{w}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}} \dot{q}_{\mathrm{k}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}} - \sum_{\mathrm{k}}^{n} \frac{\partial \mathbf{w}^{\mathrm{f.g,k}}}{\partial q_l} \dot{q}_{\mathrm{k}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \\
&= \sum_{\mathrm{k}}^{n} \underbrace{\left(\frac{\partial \mathbf{w}^{\mathrm{f.g,l}}}{\partial q_{\mathrm{k}}} - \frac{\partial \mathbf{w}^{\mathrm{f.g,k}}}{\partial q_l} \right)}_{= \mathbf{w}^{\mathrm{f.g,k}} \times \mathbf{w}^{\mathrm{f.g,l}}} \dot{q}_{\mathrm{k}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \\
&= \left(\sum_{\mathrm{k}}^{n} \mathbf{w}^{\mathrm{f.g,k}} \dot{q}_{\mathrm{k}} \right) \times \mathbf{w}^{\mathrm{f.g,l}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \\
&= \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \\
&= (\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}) \times \mathbf{w}^{\mathrm{f.g,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}.
\end{aligned}
$$

Then, the found expression can be replaced in Eq. B.56 to give

$$
\begin{aligned}
-\boldsymbol{\xi}_l(-m\mathbf{a}) =& \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*} \left(\left. \frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{dt} \right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} \right) + \\
& \left[(\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}) \times \mathbf{w}^{\mathrm{f.g,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right] \cdot [\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*} \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} + \\
& \mathbf{w}^{\mathrm{f.h,l}} \cdot [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*} \left(\left. \frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt} \right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right) + \\
& \left[(\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}) \times \mathbf{w}^{\mathrm{f.h,l}} + \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} \right] \cdot [\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*} \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}.
\end{aligned}
\tag{B.57}
$$

The distribution of terms $[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}$ and $[\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}$ and by posing $\left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}} = \{\dot{\boldsymbol{\omega}}_{\mathrm{f.h}}\}_{\mathrm{f}*}$ leads to

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}}\cdot\left[[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\left(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.h}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\right]+\\ &\left[\left(\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\right)\times\mathbf{w}^{\mathrm{f.g,l}}\right]\cdot[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}+\\ &\left(\mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\cdot\left([\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)+\\ &\mathbf{w}^{\mathrm{f.h,l}}\cdot\left[[\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\left(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\right]+\\ &\left[\left(\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\right)\times\mathbf{w}^{\mathrm{f.h,l}}\right]\cdot[\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}+\\ &\left(\mathbf{w}^{\mathrm{f}*,\mathrm{l}}\times\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\cdot\left([\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right).\end{aligned}\tag{B.58}$$

The permutation property of the triple scalar product leads to

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}}\cdot\left[[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\left(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.h}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\right]+\\ &\left[\left([\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\times\left(\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\right)\right]\cdot\mathbf{w}^{\mathrm{f.g,l}}+\\ &\left[\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times\left([\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\right]\cdot\mathbf{w}^{\mathrm{f}*,\mathrm{l}}+\\ &\mathbf{w}^{\mathrm{f.h,l}}\cdot\left[[\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\left(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\right]+\\ &\left[\left([\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\times\left(\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\right)\right]\cdot\mathbf{w}^{\mathrm{f.h,l}}+\\ &\left[\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\times\left([\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\right]\cdot\mathbf{w}^{\mathrm{f}*,\mathrm{l}}.\end{aligned}\tag{B.59}$$

The separation of the terms related to the torque on nodes g and h and for corotational frame $O_{\mathrm{f}*}$ leads to

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}}\cdot\left[[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\left(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.h}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)+\right.\\ &\left.\left(\boldsymbol{\omega}_{\mathrm{f}*}-\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\times\left([\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\right]+\\ &\left[\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times\left([\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\right]\cdot\mathbf{w}^{\mathrm{f}*,\mathrm{l}}+\\ &\mathbf{w}^{\mathrm{f.h,l}}\cdot\left[[\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\left(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)+\right.\\ &\left.\left(\boldsymbol{\omega}_{\mathrm{f}*}-\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\times\left([\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\right]+\\ &\left[\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\times\left([\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\right]\cdot\mathbf{w}^{\mathrm{f}*,\mathrm{l}}.\end{aligned}\tag{B.60}$$

The contribution of $[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}$ is equivalent to:

- a torque on node g of flexible body f equals to:
  $-\left[[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\left(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.h}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)+\left(\boldsymbol{\omega}_{\mathrm{f}*}-\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\times\left([\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\right]$;
- a torque on node h of flexible body f equals to:
  $-\left[[\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\left(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*}-\boldsymbol{\omega}_{\mathrm{f}*}\times\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)+\left(\boldsymbol{\omega}_{\mathrm{f}*}-\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)\times\left([\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\right]$;
- a torque on corotational frame $O_{\mathrm{f}*}$ equals to:
  $-\left[\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\times\left([\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\right)+\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}\times\left([\mathbf{M}_{\mathrm{f,R_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}\right)\right]$.

B.2.7 Mixed contributions in translation and rotation: $[\mathbf{M}_{\mathbf{f,R_g,T_g}}]_{\mathbf{f*}}$

The expression of kinetic energy $\mathcal{T}_\mathrm{f}$ of flexible body f is

$$\begin{aligned}\mathcal{T}_\mathrm{f} &= \frac{1}{2}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} + \frac{1}{2}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} \cdot [\mathbf{M}_{\mathrm{f,T_g,R_g}}]_{\mathrm{f*}}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}} \\ &= \frac{1}{2}(\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} + \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}} \cdot [\mathbf{M}^T_{\mathrm{f,T_g,R_g}}]_{\mathrm{f*}}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}) \\ &= \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}},\end{aligned} \tag{B.61}$$

with $[\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}} = \frac{1}{2}([\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}} + [\mathbf{M}^T_{\mathrm{f,T_g,R_g}}]_{\mathrm{f*}})$, a skew-symmetric matrix.

The contribution in the Lagrange's theorem is the partial derivative of the kinetic energy with respect to generalised coordinates q_l and is given by

$$\frac{\partial \mathcal{T}_\mathrm{f}}{\partial \dot{q}_l} = \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} + \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\mathbf{d}^{\mathrm{f.g,l}}. \tag{B.62}$$

By taking the time derivative of Eq. B.62, the first term of the Lagrange's theorem is derived as

$$\begin{aligned}\frac{d}{dt}\left.\frac{\partial \mathcal{T}_\mathrm{f}}{\partial \dot{q}_l}\right|_{O_{\mathrm{f*}}} = &\left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f*}}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} + \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f*}} - \boldsymbol{\omega}_{\mathrm{f*}} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}) + \\ &\left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{dt}\right|_{O_{\mathrm{f*}}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\mathbf{d}^{\mathrm{f.g,l}} + \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}} \left.\frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f*}}}.\end{aligned} \tag{B.63}$$

The second term of the Lagrange's theorem is expressed by

$$\left.\frac{\partial \mathcal{T}_\mathrm{f}}{\partial q_l}\right|_{O_{\mathrm{f*}}} = \left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{\partial q_l}\right|_{O_{\mathrm{f*}}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} + \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}} \left.\frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{\partial q_l}\right|_{O_{\mathrm{f*}}}. \tag{B.64}$$

The Lagrange's theorem is computed by subtracting Eq. B.64 from B.63 and it leads to

$$\begin{aligned}\frac{d}{dt}\left.\frac{\partial \mathcal{T}_\mathrm{f}}{\partial \dot{q}_l}\right|_{O_{\mathrm{f*}}} - \left.\frac{\partial \mathcal{T}_\mathrm{f}}{\partial q_l}\right|_{O_{\mathrm{f*}}} = &\mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f*}} - \boldsymbol{\omega}_{\mathrm{f*}} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}) + \\ &\left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{dt}\right|_{O_{\mathrm{f*}}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\mathbf{d}^{\mathrm{f.g,l}} + \\ &\left[\left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f*}}} - \left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{\partial q_l}\right|_{O_{\mathrm{f*}}}\right] \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}} + \\ &\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f*}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f*}}\left[\left.\frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f*}}} - \left.\frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f*}}}{\partial q_l}\right|_{O_{\mathrm{f*}}}\right],\end{aligned} \tag{B.65}$$

with

- $$\left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{f}*}} = \left.\frac{d\,\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{dt}\right|_{O_{\mathrm{Base}}} - (\boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) = \{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*} - (\boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}),$$

- $$\left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f}*}} = \left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}},$$

- $$\left.\frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{f}*}} = \left.\frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{d}^{\mathrm{f.g,l}},$$

- $$\left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}} = \left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*},$$

- $$\left.\frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{f}*}} = \left.\frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}.$$

By replacing in Eq B.65, it leads to

$$\begin{aligned}
-\boldsymbol{\xi}_l(-m\mathbf{a}) = {} & \mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \\
& (\{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\mathbf{d}^{\mathrm{f.g,l}} + \\
& \left[\left(\left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}} \right) - \left(\left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right) \right] \cdot \\
& [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} + \\
& \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot \\
& [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*} \left[\left(\left.\frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{d}^{\mathrm{f.g,l}} \right) - \left(\left.\frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right) \right].
\end{aligned} \tag{B.66}$$

with

- $$\begin{aligned}
& \left[\left(\left.\frac{d\,\mathbf{w}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{w}^{\mathrm{f.g,l}} \right) - \left(\left.\frac{\partial\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right) \right] \\
& = (\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}) \times \mathbf{w}^{\mathrm{f.g,l}} + (\mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}),
\end{aligned}$$

- $$\begin{aligned}
& \left[\left(\left.\frac{d\,\mathbf{d}^{\mathrm{f.g,l}}}{dt}\right|_{O_{\mathrm{Base}}} - \boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{d}^{\mathrm{f.g,l}} \right) - \left(\left.\frac{\partial\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}}{\partial q_l}\right|_{O_{\mathrm{Base}}} - \mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \right) \right] \\
& = (-\boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{d}^{\mathrm{f.g,l}}) + (\mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}).
\end{aligned}$$

Generalised force $\boldsymbol{\xi}_l$ in degree of freedom q_l is then given by

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})+\\ &(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\mathbf{d}^{\mathrm{f.g,l}}+\\ &\left[(\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}) \times \mathbf{w}^{\mathrm{f.g,l}} + (\mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}+\\ &\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\left[(-\boldsymbol{\omega}_{\mathrm{f}*} \times \mathbf{d}^{\mathrm{f.g,l}}) + (\mathbf{w}^{\mathrm{f}*,\mathrm{l}} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right].\end{aligned} \tag{B.67}$$

Using the permutation property of the triple scalar product, it leads to

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}} \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})+\\ &(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \cdot [\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\mathbf{d}^{\mathrm{f.g,l}}+\\ &\left[([\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \times (\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*})\right] \cdot \mathbf{w}^{\mathrm{f.g,l}}+\\ &\left[\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] \cdot \mathbf{w}^{\mathrm{f}*,\mathrm{l}}-\\ &\left[([\mathbf{M}_{\mathrm{f,T_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \times \boldsymbol{\omega}_{\mathrm{f}*}\right] \cdot \mathbf{d}^{\mathrm{f.g,l}} + \left[\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] \cdot \mathbf{w}^{\mathrm{f}*,\mathrm{l}},\end{aligned} \tag{B.68}$$

$$\begin{aligned}-\boldsymbol{\xi}_l(-m\mathbf{a}) =&\mathbf{w}^{\mathrm{f.g,l}} \cdot \left[[\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})-\right.\\ &\left.(\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}) \times ([\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] +\\ &\left[\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right] \cdot \mathbf{w}^{\mathrm{f}*,\mathrm{l}}+\\ &\left[(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})[\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*} - ([\mathbf{M}_{\mathrm{f,T_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \times \boldsymbol{\omega}_{\mathrm{f}*}\right] \cdot \mathbf{d}^{\mathrm{f.g,l}}.\end{aligned} \tag{B.69}$$

The contribution of $[\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}$ is equivalent to:

- a force on node g of flexible body f equals to:
  $-\left[(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})[\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*} - ([\mathbf{M}_{\mathrm{f,T_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \times \boldsymbol{\omega}_{\mathrm{f}*}\right]$;
- a torque on node g of flexible body f equals to:
  $-\left[[\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) - (\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*}) \times ([\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right]$;
- a torque on corotational frame $O_{\mathrm{f}*}$ equals to:
  $-\left[\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,R_g,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\right]$.

Similarly, the contribution of $[\mathbf{M}_{\mathrm{f,T_g,R_h}}]_{\mathrm{f}*}$ is equivalent to:

- a force on node g of flexible body f equals to:
  $-\left[[\mathbf{M}_{\mathrm{f,T_g,R_h}}]_{\mathrm{f}*}(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*})\right]$;
- a torque on node h of flexible body f equals to:
  $-\left[[\mathbf{M}_{\mathrm{f,R_h,T_g}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + ([\mathbf{M}_{\mathrm{f,R_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \times (\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*})\right]$;
- a torque on corotational frame $O_{\mathrm{f}*}$ equals to:
  $-\left[\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,R_h,T_g}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*})\right]$.

Finally, the contribution of $[\mathbf{M}_{\mathrm{f,R_g,T_h}}]_{\mathrm{f}*}$ is equivalent to:

- a force on node h of flexible body f equals to:
  $-\big[[\mathbf{M}_{\mathrm{f,T_h,R_g}}]_{\mathrm{f}*}(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) + \boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\big]$;
- a torque on node g of flexible body f equals to:
  $-\big[[\mathbf{M}_{\mathrm{f,R_g,T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + ([\mathbf{M}_{\mathrm{f,R_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) \times (\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*})\big]$;
- a torque on corotational frame $O_{\mathrm{f}*}$ equals to:
  $-\big[\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,R_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_h,R_g}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*})\big]$.

B.3 Total reactions of inertia

The total equivalent force on node h of flexible body f projected in corotational frame $O_{\mathrm{f}*}$ is provided by Eqs. B.70 and B.71 such as

$$\begin{aligned}\{\boldsymbol{\mathcal{R}}_{\mathrm{f.h}}(-m\mathbf{a})\}_{\mathrm{f}*} = -\sum_{\mathrm{h}=1}^{\mathrm{N}} \big[[\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ \boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*})\big]\,,\end{aligned} \tag{B.70}$$

$$\begin{aligned}\{\boldsymbol{\mathcal{R}}_{\mathrm{f.h}}(-m\mathbf{a})\}_{\mathrm{f}*} = -\sum_{\mathrm{h}=1}^{\mathrm{N}} \big[[\mathbf{M}_{\mathrm{f,T_g,R_h}}]_{\mathrm{f}*}(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ \boldsymbol{\omega}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*})\big]\,.\end{aligned} \tag{B.71}$$

The total equivalent torque on node h of flexible body f projected in corotational frame $O_{\mathrm{f}*}$ is provided by Eqs. B.72 and B.73 and reads

$$\begin{aligned}\{\boldsymbol{\mathcal{M}}_{\mathrm{f.h}}(-m\mathbf{a})\}_{\mathrm{f}*} = -\sum_{\mathrm{h}=1}^{\mathrm{N}} \big[[\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\,(\{\dot{\boldsymbol{\omega}}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ (\boldsymbol{\omega}_{\mathrm{f}*} - \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \times ([\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*})\big]\,,\end{aligned} \tag{B.72}$$

$$\begin{aligned}\{\boldsymbol{\mathcal{M}}_{\mathrm{f.h}}(-m\mathbf{a})\}_{\mathrm{f}*} = -\sum_{\mathrm{h}=1}^{\mathrm{N}} \big[[\mathbf{M}_{\mathrm{f,R_g,T_h}}]_{\mathrm{f}*}(\{\mathbf{a}_{\mathrm{f.h}}\}_{\mathrm{f}*} - \boldsymbol{\omega}_{\mathrm{f}*} \times \{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) + \\ (\boldsymbol{\omega}_{\mathrm{f}*} - \{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*}) \times ([\mathbf{M}_{\mathrm{f,R_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*})\big]\,.\end{aligned} \tag{B.73}$$

The total equivalent torque on $O_{\mathrm{f}*}$ is eventually found by Eqs. B.74 and B.75 and reads

$$\begin{aligned}\{\boldsymbol{\mathcal{M}}_{\mathrm{f}*}(-m\mathbf{a})\}_{\mathrm{f}*} = -\sum_{\mathrm{g}=1}^{\mathrm{N}}\sum_{\mathrm{h}=1}^{\mathrm{N}}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) - \\ \sum_{\mathrm{g}=1}^{n}\sum_{\mathrm{h}=1}^{n}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,R_g,R_h}}]_{\mathrm{f}*}, \{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}),\end{aligned} \tag{B.74}$$

$$\{\boldsymbol{\mathcal{M}}_{\mathrm{f}*}(-m\mathbf{a})\}_{\mathrm{f}*} = -\sum_{\mathrm{g}=1}^{\mathrm{N}}\sum_{\mathrm{h}=1}^{\mathrm{N}}\{\boldsymbol{\omega}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,R_g,T_h}}]_{\mathrm{f}*}\{\mathbf{v}_{\mathrm{f.h}}\}_{\mathrm{f}*}) - \sum_{\mathrm{g}=1}^{n}\sum_{\mathrm{h}=1}^{n}\{\mathbf{v}_{\mathrm{f.g}}\}_{\mathrm{f}*} \times ([\mathbf{M}_{\mathrm{f,T_g,R_h}}]_{\mathrm{f}*}\{\boldsymbol{\omega}_{\mathrm{f.h}}\}_{\mathrm{f}*}). \tag{B.75}$$

The final form of generalised force $\boldsymbol{\xi}_l$ accounting for the inertia forces and for degree of freedom l is eventually expressed

$$\boldsymbol{\xi}_l(-m\mathbf{a}) = \sum_{\mathrm{f}=1}^{n_{\mathrm{F}}}\sum_{\mathrm{g}=1}^{\mathrm{N}}\left(\{\boldsymbol{\mathcal{R}}_{\mathrm{f.g}}(-m\mathbf{a})\}_{\mathrm{f}*}\cdot\mathbf{d}^{\mathrm{f.g,l}} + \{\boldsymbol{\mathcal{M}}_{\mathrm{f.g}}(-m\mathbf{a})\}_{\mathrm{f}*}\cdot\mathbf{w}^{\mathrm{f.g,l}}\right) + \{\boldsymbol{\mathcal{M}}_{\mathrm{f}*}(-m\mathbf{a})\}_{\mathrm{f}*}\cdot\mathbf{w}^{\mathrm{f}*,\mathrm{l}}. \tag{B.76}$$

Appendix C

Flexible beam elements

Flexible beam elements are used in the corotational formulation in order to model the flexibility of the robot arm and forearm. They are constituted of two nodes (N=2) and the motion of each of the nodes involves 3 displacements and 3 rotations. It is the imposed condition to apply the corotational formulation. Figure C.1 defines the six degrees of freedom $\mathbf{q}_{\mathrm{f.g}}$ allotted per node for a uniform beam. Local axis $\mathrm{X_f}$ is the neutral axis of the beam element and local axes $\mathrm{Y_f}$ and $\mathrm{Z_f}$ are the principal inertial axes. For node g of flexible beam f, the first three degrees of freedom $q_{\mathrm{f.g},x}$, $q_{\mathrm{f.g},y}$ and $q_{\mathrm{f.g},z}$ are dedicated to the translational motion of the node while the last three degrees of freedom $q_{\mathrm{f.g,Rx}}$, $q_{\mathrm{f.g,Ry}}$ and $q_{\mathrm{f.g,Rz}}$ accounts for its orientation. As flexible beam f has two nodes, twelve degrees of freedom describe its deformation along the defined local axes.

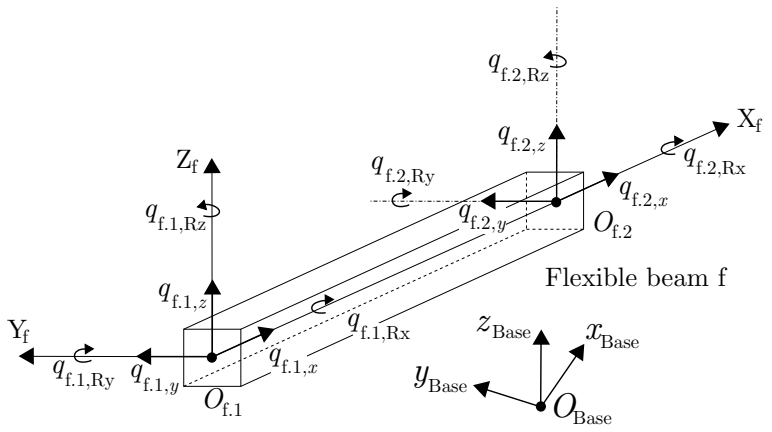


Figure C.1: Representation of a flexible beam element

C.1 Mass matrix of a flexible beam element

Mass matrix $\mathbf{M}_{\mathrm{f}}$ of a beam element is a (12×12) matrix. From the order of the allotted degrees of freedom per node, the matrix can be subdivided into (3×3) sub-matrices as follows

$$\mathbf{M}_{\mathrm{f}} = \begin{bmatrix} \mathbf{M}_{\mathrm{f,T_1,T_1}} & \mathbf{M}_{\mathrm{f,T_1,R_1}} & \mathbf{M}_{\mathrm{f,T_1,T_2}} & \mathbf{M}_{\mathrm{f,T_1,R_2}} \\ \mathbf{M}_{\mathrm{f,R_1,T_1}} & \mathbf{M}_{\mathrm{f,R_1,R_1}} & \mathbf{M}_{\mathrm{f,R_1,T_2}} & \mathbf{M}_{\mathrm{f,R_1,R_2}} \\ \mathbf{M}_{\mathrm{f,T_2,T_1}} & \mathbf{M}_{\mathrm{f,T_2,R_1}} & \mathbf{M}_{\mathrm{f,T_2,T_2}} & \mathbf{M}_{\mathrm{f,T_2,R_2}} \\ \mathbf{M}_{\mathrm{f,R_2,T_1}} & \mathbf{M}_{\mathrm{f,R_2,R_1}} & \mathbf{M}_{\mathrm{f,R_2,T_2}} & \mathbf{M}_{\mathrm{f,R_2,R_2}} \end{bmatrix}. \tag{C.1}$$

For instance, $\mathbf{M}_{\mathrm{f,T_1,R_2}}$ links the translational degrees of freedom of node 1 with the rotational degrees of freedom of node 2. The complete expression of the mass matrix for a beam element is developed such as

$$\mathbf{M}_{\mathrm{f}} = \rho A L \left[\begin{array}{ccc|ccc|ccc|ccc} \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & \frac{1}{6} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{13}{35} & 0 & 0 & 0 & \frac{11L}{210} & 0 & \frac{9}{70} & 0 & 0 & 0 & \frac{-13L}{420} \\ 0 & 0 & \frac{13}{35} & 0 & \frac{-11L}{210} & 0 & 0 & 0 & \frac{9}{70} & 0 & \frac{13L}{420} & 0 \\ \hline 0 & 0 & 0 & \frac{I_{\mathrm{P}}}{3A} & 0 & 0 & 0 & 0 & 0 & \frac{I_{\mathrm{P}}}{6A} & 0 & 0 \\ 0 & 0 & \frac{-11L}{210} & 0 & \frac{L^2}{105} & 0 & 0 & 0 & \frac{-13L}{420} & 0 & \frac{-L^2}{140} & 0 \\ 0 & \frac{11L}{210} & 0 & 0 & 0 & \frac{L^2}{105} & 0 & \frac{13L}{420} & 0 & 0 & 0 & \frac{-L^2}{140} \\ \hline \frac{1}{6} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{9}{70} & 0 & 0 & 0 & \frac{13L}{420} & 0 & \frac{13}{35} & 0 & 0 & 0 & \frac{-11L}{210} \\ 0 & 0 & \frac{9}{70} & 0 & \frac{-13L}{420} & 0 & 0 & 0 & \frac{13}{35} & 0 & \frac{11L}{210} & 0 \\ \hline 0 & 0 & 0 & \frac{I_{\mathrm{P}}}{6A} & 0 & 0 & 0 & 0 & 0 & \frac{I_{\mathrm{P}}}{3A} & 0 & 0 \\ 0 & 0 & \frac{13L}{420} & 0 & \frac{-L^2}{140} & 0 & 0 & 0 & \frac{11L}{210} & 0 & \frac{L^2}{105} & 0 \\ 0 & \frac{-13L}{420} & 0 & 0 & 0 & \frac{-L^2}{140} & 0 & \frac{-11L}{210} & 0 & 0 & 0 & \frac{L^2}{105} \end{array}\right] \tag{C.2}$$

with $I_{\mathrm{P}} = I_y + I_z$ the polar inertia, ρ the density, A the cross section and L the length of the beam element. Equatorial inertias are designated by I_y and I_z. If the beam, whose total length is L_{Beam}, is subdivided into (N - 1) beam elements, then the length of one beam element becomes $L = \frac{L_{\mathrm{Beam}}}{\mathrm{N}-1}$. In this case, the mass matrix of each beam element must be assembled to span all the degrees of freedom of all the nodes (see Section C.4).

C.2 Stiffness matrix of a flexible beam element

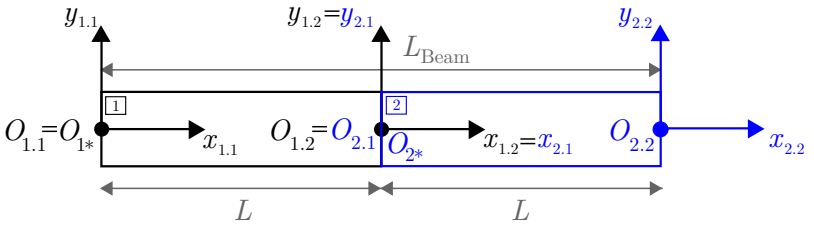
Similarly, stiffness matrix $\mathbf{K}_{\mathrm{f}}$ of a beam element is also a (12 × 12) matrix. From the order of the allotted degrees of freedom per node, the matrix can be subdivided into (3 × 3) sub-matrices as follows

$$\mathbf{K}_{\mathrm{f}} = \left[\begin{array}{ccc|ccc|ccc|ccc} \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & \frac{-EA}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} & 0 & \frac{-12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} \\ 0 & 0 & \frac{12EI_y}{L^3} & 0 & \frac{-6EI_y}{L^2} & 0 & 0 & 0 & \frac{-12EI_y}{L^3} & 0 & \frac{-6EI_y}{L^2} & 0 \\ \hline 0 & 0 & 0 & \frac{GI_{\mathrm{P}}}{L} & 0 & 0 & 0 & 0 & 0 & \frac{-GI_{\mathrm{P}}}{L} & 0 & 0 \\ 0 & 0 & \frac{-6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} & 0 & \frac{-6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} \\ \hline \frac{-EA}{L} & 0 & 0 & 0 & 0 & 0 & \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{-12EI_z}{L^3} & 0 & 0 & 0 & \frac{-6EI_z}{L^2} & 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{-6EI_z}{L^2} \\ 0 & 0 & \frac{-12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 & 0 & 0 & \frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 \\ \hline 0 & 0 & 0 & \frac{-GI_{\mathrm{P}}}{L} & 0 & 0 & 0 & 0 & 0 & \frac{GI_{\mathrm{P}}}{L} & 0 & 0 \\ 0 & 0 & \frac{-6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} & 0 & \frac{-6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} \end{array}\right] \tag{C.3}$$

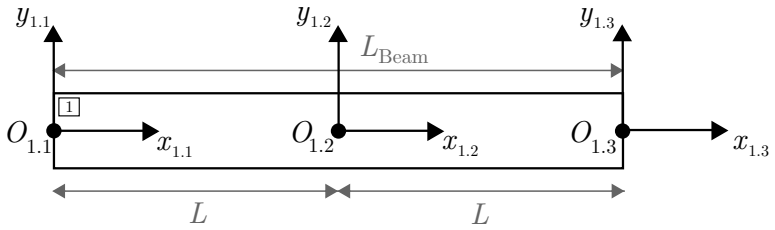
with $G = \frac{E}{2(1+\nu)}$ the Coulomb's modulus, E the Young's modulus, ν the Poisson's ratio (typically ν=0.3). The length of each beam element L must also be computed following Section C.4 if the total beam length L_{Beam} is subdivided into multiple beam elements.

C.3 Application of the corotational formulation

Using the corotational formulation, flexible beams can be modelled following two different approaches. Considering a flexible beam of total length L_{Beam}, the first approach (case a. in Figure C.2a) consists in assembling several two-node beam elements of length L i.e. the behaviour of the flexible beam is represented by several flexible beam elements. Of course the sum of the lengths of all beam elements is L_{Beam}. If all the two-node beam elements have the same length L, their length is therefore computed by $L = \frac{L_{\text{Beam}}}{n_{\text{F}}}$. The second approach (case b. in Figure C.2b) models the behaviour of the flexible beam with one single beam element comprising N nodes. The N nodes are equally spaced along the length of the flexible beam. In this case, length L between each node is computed by $L = \frac{L_{\text{Beam}}}{\text{N}-1}$. Since structural matrices $\mathbf{M}_{\text{f}}$, $\mathbf{K}_{\text{f}}$ and $\mathbf{C}_{\text{f}}$ representing the behaviour of a flexible element define the interaction between two nodes, they must be assembled (Appendix C.4). The two modelling approaches are illustrated in the case of two beam elements in Figure C.2.



(a) Connected two-node beam elements



(b) One flexible beam element with 3 nodes

Figure C.2: Modelling approaches for a corotational flexible beam

The derivation of the kinematics of the nodes also relies on the concept of homogeneous transformation matrices. The two approaches are examined hereafter:

a. Connected two-node beam elements: the flexible beam is composed of two two-node beam elements connected together (two flexible bodies $n_{\text{F}} = 2$). As a result, three homogeneous transformation matrices must be defined per beam elements: two for locating the end nodes ($\mathbf{T}_{1.1}$ and $\mathbf{T}_{1.2}$) and one for the corotational frame ($\mathbf{T}_{1*}$). The position of the first node $O_{1.1}$ of the first beam element is given by

$$\mathbf{T}_{\text{Base},1.1} = \mathbf{T}_{\text{disp}}(q_{1.1,x}, q_{1.1,y}, q_{1.1,z}) \cdot \mathbf{T}_{\text{rotx}}(q_{1.1,\text{Rx}}) \cdot \mathbf{T}_{\text{roty}}(q_{1.1,\text{Ry}}) \cdot \mathbf{T}_{\text{rotz}}(q_{1.1,\text{Rz}}), \quad \text{(C.4)}$$

with $q_{1.1,x}$, $q_{1.1,y}$ and $q_{1.1,z}$ the translational degrees of freedom of the first node of the first beam element and $q_{1.1,\text{Rx}}$, $q_{1.1,\text{Ry}}$ and $q_{1.1,\text{Rz}}$ its rotational degrees of freedom.

The first node of the first beam element is assumed to be located on base frame O_{Base}.

Relatively to the first node of the first element, the situation of the second node of the first beam element is expressed as follows

$$\begin{aligned}\mathbf{T}_{\text{Base,1.2}} =&\mathbf{T}_{\text{Base,1.1}} \cdot \mathbf{T}_{\text{disp}}(L + q_{1.2,x}, q_{1.2,y}, q_{1.2,z}) \cdot \\ &\mathbf{T}_{\text{rotx}}(q_{1.2,\text{Rx}}) \cdot \mathbf{T}_{\text{roty}}(q_{1.2,\text{Ry}}) \cdot \mathbf{T}_{\text{rotz}}(q_{1.2,\text{Rz}}).\end{aligned} \tag{C.5}$$

with $q_{1.2,x}$, $q_{1.2,y}$ and $q_{1.2,z}$ the translational degrees of freedom of the second node of the first beam element and $q_{1.2,\text{Rx}}$, $q_{1.2,\text{Ry}}$ and $q_{1.2,\text{Rz}}$ its rotational degrees of freedom. Note that the second node could have been defined relatively to base frame O_{Base} as well, like in the Cartesian coordinates. Eq. C.5 defines the second node in a relative coordinate way. Finally, the situation of corotational frame O_{1*} can be chosen either on one of the defined nodes or near the centre of the elements. For the sake of simplicity, the corotational frame was located on the first node such as

$$\mathbf{T}_{\text{Base,1}*} = \mathbf{T}_{\text{disp}}(q_{1.1,x}, q_{1.1,y}, q_{1.1,z}) \cdot \mathbf{T}_{\text{rotx}}(q_{1.1,\text{Rx}}) \cdot \mathbf{T}_{\text{roty}}(q_{1.1,\text{Ry}}) \cdot \mathbf{T}_{\text{rotz}}(q_{1.1,\text{Rz}}). \tag{C.6}$$

Since the two beam elements are connected together via one of the end nodes of the first element, first node of the second beam element $O_{2.1}$ is clamped to the second node of the first beam element $O_{1.2}$ such as

$$\mathbf{T}_{\text{Base,2.1}} = \mathbf{T}_{\text{Base,1.2}}. \tag{C.7}$$

In the same way, the positions of second node $O_{2.2}$ and corotational frame O_{2*} of the second beam element are defined by

$$\begin{aligned}\mathbf{T}_{\text{Base,2.2}} =&\mathbf{T}_{\text{Base,2.1}} \cdot \mathbf{T}_{\text{disp}}(L + q_{2.2,x}, q_{2.2,y}, q_{2.2,z}) \cdot \\ &\mathbf{T}_{\text{rotx}}(q_{2.2,\text{Rx}}) \cdot \mathbf{T}_{\text{roty}}(q_{2.2,\text{Ry}}) \cdot \mathbf{T}_{\text{rotz}}(q_{2.2,\text{Rz}}),\end{aligned} \tag{C.8}$$

$$\mathbf{T}_{\text{Base,2}*} = \mathbf{T}_{\text{Base,1.2}}. \tag{C.9}$$

Degrees of freedom $q_{2.2,x}$, $q_{2.2,y}$, $q_{2.2,z}$, $q_{2.2,\text{Rx}}$, $q_{2.2,\text{Ry}}$ and $q_{2.2,\text{Rz}}$ belong to the second node of the second element with obvious meaning. In total, 18 degrees of freedom and 6 homogeneous transformation matrices are required to describe the behaviour of the flexible beam of length L_{Beam} composed of two beam elements of length L.

b. Flexible beam element with 3 nodes: in the second approach, the flexible beam is composed of one single beam element comprising 3 nodes (one flexible body n_{F}=1). In this case, four homogeneous transformation matrices must be defined for the whole flexible beam: three for the nodes ($\mathbf{T}_{1.1}$, $\mathbf{T}_{1.2}$ and $\mathbf{T}_{1.3}$) and one for the corotational frame ($\mathbf{T}_{1*}$). The positions of the three nodes are expressed as follows

$$\begin{aligned}\mathbf{T}_{\mathrm{Base},1.1} =&\mathbf{T}_{\mathrm{disp}}(q_{1.1,x}, q_{1.1,y}, q_{1.1,z})\cdot \\ &\mathbf{T}_{\mathrm{rotx}}(q_{1.1,\mathrm{Rx}}) \cdot \mathbf{T}_{\mathrm{roty}}(q_{1.1,\mathrm{Ry}}) \cdot \mathbf{T}_{\mathrm{rotz}}(q_{1.1,\mathrm{Rz}}),\end{aligned} \tag{C.10}$$

$$\begin{aligned}\mathbf{T}_{\mathrm{Base},1.2} =&\mathbf{T}_{\mathrm{disp}}(\frac{L}{2} + q_{1.2,x}, q_{1.2,y}, q_{1.2,z})\cdot \\ &\mathbf{T}_{\mathrm{rotx}}(q_{1.2,\mathrm{Rx}}) \cdot \mathbf{T}_{\mathrm{roty}}(q_{1.2,\mathrm{Ry}}) \cdot \mathbf{T}_{\mathrm{rotz}}(q_{1.2,\mathrm{Rz}}),\end{aligned} \tag{C.11}$$

$$\begin{aligned}\mathbf{T}_{\mathrm{Base},1.3} =&\mathbf{T}_{\mathrm{disp}}(L + q_{1.3,x}, q_{1.3,y}, q_{1.3,z})\cdot \\ &\mathbf{T}_{\mathrm{rotx}}(q_{1.3,\mathrm{Rx}}) \cdot \mathbf{T}_{\mathrm{roty}}(q_{1.3,\mathrm{Ry}}) \cdot \mathbf{T}_{\mathrm{rotz}}(q_{1.3,\mathrm{Rz}}),\end{aligned} \tag{C.12}$$

while the position of corotational frame is set at the middle of the beam such as

$$\begin{aligned}\mathbf{T}_{\mathrm{Base},1*} =&\mathbf{T}_{\mathrm{disp}}(\frac{L}{2} + \sum_{\mathrm{g}=1}^{3} \frac{q_{1.\mathrm{g},x}}{3}, \sum_{\mathrm{g}=1}^{3} \frac{q_{1.\mathrm{g},y}}{3}, \sum_{\mathrm{g}=1}^{3} \frac{q_{1.\mathrm{g},z}}{3})\cdot \\ &\mathbf{T}_{\mathrm{rotx}}(\sum_{\mathrm{g}=1}^{3} \frac{q_{1.\mathrm{g},\mathrm{Rx}}}{3}) \cdot \mathbf{T}_{\mathrm{roty}}(\sum_{\mathrm{g}=1}^{3} \frac{q_{1.\mathrm{g},\mathrm{Ry}}}{3}) \cdot \mathbf{T}_{\mathrm{rotz}}(\sum_{\mathrm{g}=1}^{3} \frac{q_{1.\mathrm{g},\mathrm{Rz}}}{3}).\end{aligned} \tag{C.13}$$

Again 18 degrees of freedom but only 4 homogeneous transformation matrices are required to describe the behaviour of the flexible beam of length L_{Beam} composed of 3 nodes for one single flexible beam element of length L.

Once the situations of each node and corotational frames are known, their partial contributions and accelerations are computed using Eqs. 3.28 to 3.30 and Eq. 3.66. The equations of motion are built using Eq. 3.65 and the constitutive matrices of the beam elements $\mathbf{M}_{\mathrm{f}}$, $\mathbf{K}_{\mathrm{f}}$ and $\mathbf{C}_{\mathrm{f}}$.

Of course, the two illustrative examples can be extended to several beam elements. The two modelling approaches can even be mixed together. A flexible beam could be modelled using several beam elements containing several nodes. Note that the beam stiffening effect is better accounted if the beam elements are interconnected to form the beam of length L_{Beam} (first approach). Beam stiffening effect may arise if the flexible beam is subjected to large torques. Also, the corotational formulation can be used to model mechanical systems involving large displacements, as illustrated in Figure C.3. Although the displacements are large, the system involves only small local deformation as it is modelled by 20 successive flexible beam elements.

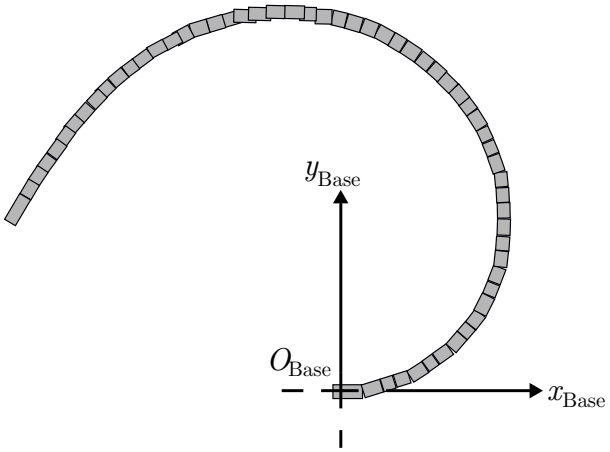


Figure C.3: Large displacement modelling of a flexible beam

C.4 Assembly of structural matrices

A flexible beam of total length L_{Beam} can be subdivided into beam elements of length L using $L = \frac{L_{\text{Beam}}}{\text{N}-1}$. Hence, all beam elements share the same length L. Assembling beam elements can be useful in the context of model updating. Although all beam elements have the same length, their constitutive properties (ρ, E, etc.) can be seen as tuning parameters to fit the model. For example, a beam of total length L_{Beam} with 3 nodes is depicted in Figure C.4. The length of each beam element becomes $L = \frac{L_{\text{Beam}}}{2}$. Although corotational frame $O_{\text{f}*}$ can be located on one of the end nodes, the results are generally of better quality if it is near the centre of mass of the flexible beam.

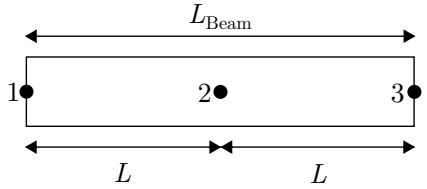


Figure C.4: Subdivision in beam elements of length L

Constitutive matrices $\mathbf{M}_{\text{f}}$, $\mathbf{K}_{\text{f}}$ and $\mathbf{C}_{\text{f}}$ must be assembled accordingly to span all the degrees of freedom. In the case of 3 nodes, the total number of degrees of freedom for the entire flexible beam f is $\mathbf{q}_{\text{f}} \in \mathbb{R}^{18\times 1}$. Hence, the assembled matrix must be (18×18) and is derived as follows

$$\mathbf{A}_{\text{f}} = \begin{bmatrix} \mathbf{A}_{\text{f,T}_1,\text{T}_1} & \mathbf{A}_{\text{f,T}_1,\text{R}_1} & \mathbf{A}_{\text{f,T}_1,\text{T}_2} & \mathbf{A}_{\text{f,T}_1,\text{R}_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{A}_{\text{f,R}_1,\text{T}_1} & \mathbf{A}_{\text{f,R}_1,\text{R}_1} & \mathbf{A}_{\text{f,R}_1,\text{T}_2} & \mathbf{A}_{\text{f,R}_1,\text{R}_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{A}_{\text{f,T}_2,\text{T}_1} & \mathbf{A}_{\text{f,T}_2,\text{R}_1} & \mathbf{A}_{\text{f,T}_2,\text{T}_2} + \mathbf{A}_{\text{f,T}_2,\text{T}_2} & \mathbf{A}_{\text{f,T}_2,\text{R}_2} + \mathbf{A}_{\text{f,T}_2,\text{R}_2} & \mathbf{A}_{\text{f,T}_2,\text{T}_3} & \mathbf{A}_{\text{f,T}_2,\text{R}_3} \\ \mathbf{A}_{\text{f,R}_2,\text{T}_1} & \mathbf{A}_{\text{f,R}_2,\text{R}_1} & \mathbf{A}_{\text{f,R}_2,\text{T}_2} + \mathbf{A}_{\text{f,R}_2,\text{T}_2} & \mathbf{A}_{\text{f,R}_2,\text{R}_2} + \mathbf{A}_{\text{f,R}_2,\text{R}_2} & \mathbf{A}_{\text{f,R}_2,\text{T}_3} & \mathbf{A}_{\text{f,R}_2,\text{R}_3} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{\text{f,T}_3,\text{T}_2} & \mathbf{A}_{\text{f,T}_3,\text{R}_2} & \mathbf{A}_{\text{f,T}_3,\text{T}_3} & \mathbf{A}_{\text{f,T}_3,\text{R}_3} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{\text{f,R}_3,\text{T}_2} & \mathbf{A}_{\text{f,R}_3,\text{R}_2} & \mathbf{A}_{\text{f,R}_3,\text{T}_3} & \mathbf{A}_{\text{f,R}_3,\text{R}_3} \end{bmatrix} \tag{C.14}$$

with a $\mathbf{A}_{\text{f}}$ a generic matrix either representing mass $\mathbf{M}_{\text{f}}$, stiffness $\mathbf{K}_{\text{f}}$ or damping $\mathbf{C}_{\text{f}}$ matrix.

Appendix D

Field oriented control

D.1 Objective of the control strategy

Actuators embedded in the Stäubli TX200 robot were simplified into brushless DC motors. However, the actual actuators are AC servo motors most probably built on the permanent magnet synchronous motor (PMSM) technology. In a permanent magnet synchronous motor, the rotor is constituted of permanent magnets and is surrounded by three equally spaced fixed stator windings. When the motor is powered, the windings produce a rotating magnetic field which induces an attraction and repulsion mechanism with the rotor magnets. By controlling the current flowing in the three windings, a magnetic field of arbitrary direction and magnitude can be produced by the stator to control the electromagnetic torque.

For any position of the rotor, there is an optimal orientation of the magnetic field produced by the stator which maximises the torque produced [179]. If the fields generated by the stator and the rotor are parallel, no torque is produced. In contrast, the electromagnetic torque is maximum when the latter are perpendicular. The stator field is produced by the current flowing in three equally spaced windings which are 120° apart referenced as i_a, i_b and i_c in Figure D.1. The direction and orientation of the generated magnetic field can be represented with a current space vector being the sum of the components of the windings.

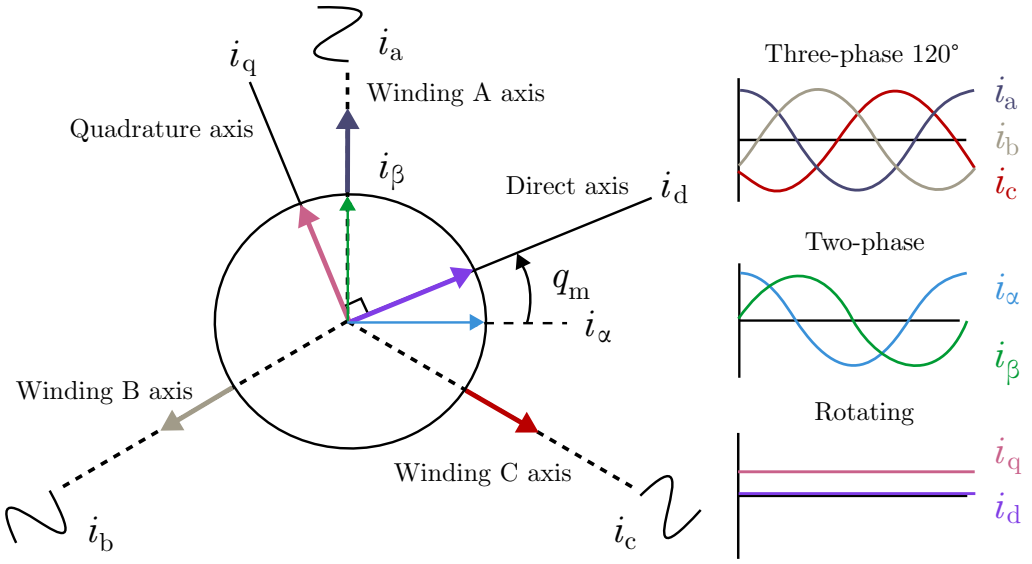


Figure D.1: Three-phase to direct-quadrature reference frame

Just like the stator field, the resulting current space vector can be decomposed in a

stationary reference frame in which currents i_α and i_β flow sinusoidally. Currents i_α and i_β are used in sinusoidal control whose main drawback is the difficulty to control their magnitude and direction at the same time. Hence, field oriented control proposes to transform currents i_α and i_β into the rotating frame of the rotor into currents i_d and i_q. The orthogonal (quadrature) component i_q is responsible for producing torque and the parallel (direct) component i_d produces undesired heat and compression forces. Hence, an ideal drive should aim at minimising the parallel component while maximising the quadrature component. Finally, a smooth torque is delivered in the ideal case that the stator current space vector is constant in magnitude and always oriented towards the quadrature direction, irrespective of the rotor angle and speed. Field oriented control takes advantage of the rotating frame of the rotor in which constant currents are monitored.

D.2 Field oriented control equations

The most prevalent control strategies for brushless motors are trapezoidal control, sinusoidal control and field oriented control. Trapezoidal control is relatively simple and offers smooth operation at high speed but causes torque ripple at low speeds. On the other hand, sinusoidal control eliminates torque ripple and provides efficient operation at low speeds. Nevertheless, it operates in the two-phase reference frame in which the sinusoidal nature of currents limit the performance at high speeds. Field oriented control combines the best aspects of the previous two control strategies, offering smooth and efficient operation with fast dynamic response at both low and high speeds [180].

Using the Clarke and Park transformations and their inverse, it is possible to transpose quantities from the three-phase 120° reference frame to the rotating rotor frame. Field oriented control strategy relies on both transformations as it can be observed in its block diagram (Figure D.2). The field oriented control is illustrated for a rotor velocity control using a PI controller to cancel steady state errors.

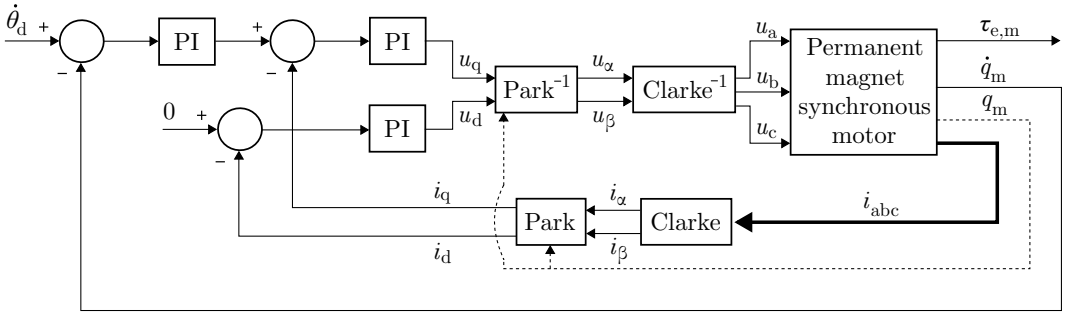


Figure D.2: Block diagram of the field oriented control [180]

The internal loops consist of two control loops for the quadrature current i_q and the direct current i_d. PI current controllers thus monitor time-invariant quantities, rather than controlling sinusoidal signals if they were expressed in the two-phase reference frame. Current controllers are somehow isolated from time variant winding currents and voltages, which simplifies the control structure. Of course, the objective of the controllers

is to maximise the quadrature component over the direct component to minimise the losses. Therefore, null reference current is assigned for the PI controller monitoring direct current i_{d}.

The inputs of the permanent magnet synchronous motor model are nonetheless expressed in the three-phase 120° reference frame (u_{a}, u_{b} and u_{c}). The electric equations of a PMSM motor are expressed in a direct-quadrature reference frame as follows

$$\frac{di_{\mathrm{d,m}}}{dt} = \frac{1}{L_{\mathrm{d}}}u_{\mathrm{d,m}} - \frac{R_{\mathrm{s}}}{L_{\mathrm{d}}}i_{\mathrm{d,m}} + \frac{L_{\mathrm{q}}}{L_{\mathrm{d}}}\mathrm{p}\dot{q}_{\mathrm{m}}i_{\mathrm{q,m}}, \tag{D.1}$$

$$\frac{di_{\mathrm{q,m}}}{dt} = \frac{1}{L_{\mathrm{q}}}u_{\mathrm{q,m}} - \frac{R_{\mathrm{s}}}{L_{\mathrm{q}}}i_{\mathrm{d,m}} - \frac{L_{\mathrm{d}}}{L_{\mathrm{q}}}\mathrm{p}\dot{q}_{\mathrm{m}}i_{\mathrm{d,m}} - \frac{\lambda \mathrm{p}\dot{q}_{\mathrm{m}}}{L_{\mathrm{q}}}, \tag{D.2}$$

$$\tau_{\mathrm{e,m}} = 1.5\mathrm{p}\left[\lambda i_{\mathrm{q,m}} + (L_{\mathrm{d}} - L_{\mathrm{q}})i_{\mathrm{d,m}}i_{\mathrm{q,m}}\right], \tag{D.3}$$

with $\tau_{\mathrm{e,m}}$ the electromagnetic torque, $i_{\mathrm{d,m}}$ the motor direct current, $i_{\mathrm{q,m}}$ the motor quadrature current, L_{d} the direct inductance, L_{q} the quadrature inductance, p the number of pole pairs, R_{s} the stator resistance and λ the flux linkage which is computed as $\lambda = \frac{k_{\mathrm{t}}}{\mathrm{p}}$, with k_{t}, the torque constant. Mechanical equation of the rotor reads

$$\phi_{zz,\mathrm{m}}\frac{d^2q}{dt^2} = \tau_{\mathrm{e,m}} - \tau_{\mathrm{L}}, \tag{D.4}$$

with τ_{L} the torque load.

To convert the three-phase input voltages of the PMSM (u_{a}, u_{b} and u_{c}), Clarke and Park transformations are successively applied. Clarke transformation for voltages reads

$$\begin{aligned} u_{\mathrm{d'}} &= \sqrt{\frac{2}{3}}\left(u_{\mathrm{a}} - \frac{1}{2}u_{\mathrm{b}} - \frac{1}{2}u_{\mathrm{c}}\right), \\ u_{\mathrm{q'}} &= \sqrt{\frac{2}{3}}\left(\frac{\sqrt{3}}{2}u_{\mathrm{b}} - \frac{\sqrt{3}}{2}u_{\mathrm{c}}\right), \end{aligned} \tag{D.5}$$

while Park transformation for voltages is

$$\begin{bmatrix} u_{\mathrm{d,m}} \\ u_{\mathrm{q,m}} \end{bmatrix} = \begin{bmatrix} \cos(\mathrm{p}\,q_{\mathrm{m}}) & \sin(\mathrm{p}\,q_{\mathrm{m}}) \\ -\sin(\mathrm{p}\,q_{\mathrm{m}}) & \cos(\mathrm{p}\,q_{\mathrm{m}}) \end{bmatrix} \begin{bmatrix} u_{\mathrm{d'}} \\ u_{\mathrm{q'}} \end{bmatrix}. \tag{D.6}$$

Motor equations (Eqs. D.1 to D.4) are then used to produce the corresponding direct and quadrature currents in the motor ($i_{\mathrm{d,m}}$ and $i_{\mathrm{q,m}}$). Successively applying the inverse Park transformation in current,

$$\begin{bmatrix} i_{\mathrm{d'}} \\ i_{\mathrm{q'}} \end{bmatrix} = \begin{bmatrix} \cos(\mathrm{p}\,q_{\mathrm{m}}) & -\sin(\mathrm{p}\,q_{\mathrm{m}}) \\ \sin(\mathrm{p}\,q_{\mathrm{m}}) & \cos(\mathrm{p}\,q_{\mathrm{m}}) \end{bmatrix} \begin{bmatrix} i_{\mathrm{q,m}} \\ i_{\mathrm{d,m}} \end{bmatrix}, \tag{D.7}$$

and the inverse Clarke transformation in current,

$$\begin{bmatrix} i_{\mathrm{a}} \\ i_{\mathrm{b}} \\ i_{\mathrm{c}} \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & 0 & 1 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & 1 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 1 \end{bmatrix} \begin{bmatrix} i_{\mathrm{d}_\alpha} \\ i_{\mathrm{q}_\alpha} \\ 0 \end{bmatrix}, \tag{D.8}$$

leads to the output three-phase current of the PMSM (i_{a}, i_{b} and i_{c}), referenced as i_{abc} in the field oriented control block diagram (Figure D.2).

At the output of the PMSM, quantities must again be transformed into the direct-quadrature reference frame for control purposes. Therefore, Clarke and Park transformations are successively applied in current. Clarke transformation in current reads

$$\begin{aligned} i_\alpha &= \sqrt{\frac{3}{2}} i_{\mathrm{a}}, \\ i_\beta &= \frac{\sqrt{2}}{2} \left(i_{\mathrm{a}} + 2 i_{\mathrm{b}} \right), \end{aligned} \tag{D.9}$$

and Park transformation current is

$$\begin{bmatrix} i_{\mathrm{d}} \\ i_{\mathrm{q}} \end{bmatrix} = \begin{bmatrix} \cos(\mathrm{p}\, q_{\mathrm{m}}) & \sin(\mathrm{p}\, q_{\mathrm{m}}) \\ -\sin(\mathrm{p}\, q_{\mathrm{m}}) & \cos(\mathrm{p}\, q_{\mathrm{m}}) \end{bmatrix} \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix}. \tag{D.10}$$

At the output of the PI controllers, the direct and quadratures voltages are converted into the three-phase 120° reference frame to generate u_{a}, u_{b} and u_{c} which are the inputs of the PMSM motor. This time, Park and Clarke inverse transformations in voltage are applied. They respectively read

$$\begin{bmatrix} u_\alpha \\ u_\beta \end{bmatrix} = \begin{bmatrix} \cos(\mathrm{p}\, q_{\mathrm{m}}) & -\sin(\mathrm{p}\, q_{\mathrm{m}}) \\ \sin(\mathrm{p}\, q_{\mathrm{m}}) & \cos(\mathrm{p}\, q_{\mathrm{m}}) \end{bmatrix} \begin{bmatrix} u_{\mathrm{d}} \\ u_{\mathrm{q}} \end{bmatrix}, \tag{D.11}$$

and the Clarke inverse transformation in voltage is

$$\begin{bmatrix} u_{\mathrm{a}} \\ u_{\mathrm{b}} \\ u_{\mathrm{c}} \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & 0 & 1 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & 1 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 1 \end{bmatrix} \begin{bmatrix} u_\alpha \\ u_\beta \\ 0 \end{bmatrix}. \tag{D.12}$$

Presented field oriented control block diagram in Figure D.2 depicts a motor whose inputs are continuous sinusoidal waves. Their frequency and amplitude are varied in order to control the speed and torque of the motor. Since fixed frequency sinusoidal supply powers the motor, the sinusoidal waves are reconstructed using sinusoidal pulse width modulation (PWM). Interested readers are redirected to [179].

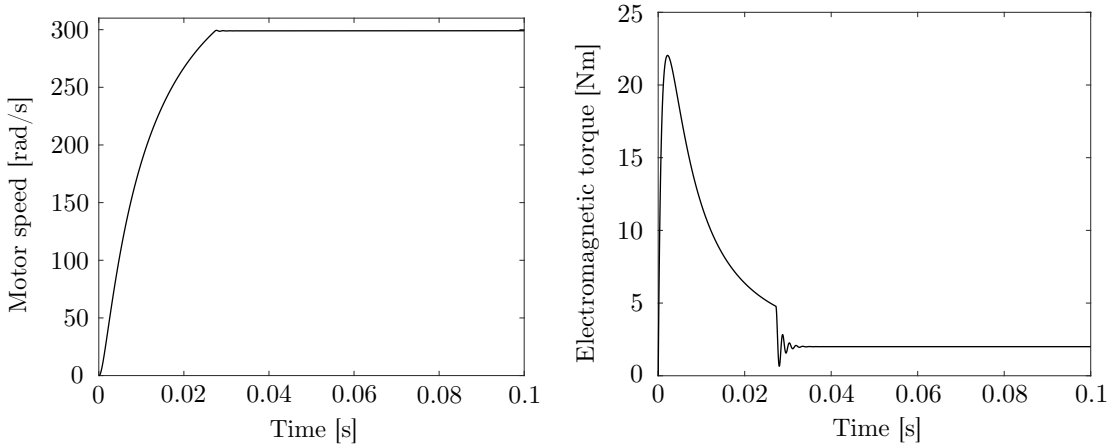
D.3 Simulation results

The field oriented control was applied to the control of a permanent magnet synchronous motor. Data set used to simulate its behaviour is provided in Table D.1.

R_{s}	Stator resistance	2.875 [Ω]
L_{q}	Quadrature inductance	0.0085 [H]
L_{d}	Direct inductance	0.0085 [H]
k_{t}	Torque constant	0.7 [Nm/A]
p	Number of pole pairs	4
τ_{L}	Torque load	2 [Nm]
$\phi_{zz,\mathrm{m}}$	Rotor inertia	0.0008 [kg.m^2]
K_{p_i}	Proportional gain for current loop	10
K_{i_i}	Integral gain for current loop	5
$K_{\mathrm{p_v}}$	Proportional gain for velocity loop	10
$K_{\mathrm{i_v}}$	Integral gain for velocity loop	5

Table D.1: Data set used for field oriented control simulation

The commanded speed of the motor was set at $\dot{\theta}_{\mathrm{d}}$=300 rad/s. Figure D.3a shows that the desired velocity is indeed reached after 0.03 s and sustained afterwards. Regarding the electromagnetic torque in Figure D.3b, it stabilises at 2 Nm which is the value of the torque load τ_{L}.



(a) Controlled motor speed (b) Electromagnetic torque

Figure D.3: Field oriented control simulation results

In view of the numerous parameters presented in Table D.1 to simulate the field oriented control strategy, it is clear that equivalent data for an actual servo motor are difficult to find. Hence, the actuators of the Stäubli TX200 robot were simplified to DC motors with only two variables: the armature resistance R_{a} and the torque constant k_{t} (voltage constant k_{v} is equal to torque constant k_{t} if expressed in coherent units).

Appendix E

Zero-order method detail

E.1 Stability in 2D milling without cross-coupling

Using the *zero-order* method for a two-dimensional milling [45], the machine dynamics is modelled with an end mill having two degrees of freedom: one along the feed direction, chosen as X direction and the other one, in the perpendicular direction, chosen as Y direction (Figure E.1). The end mill is assumed to have N_z teeth with zero helix angle (straight flutes).

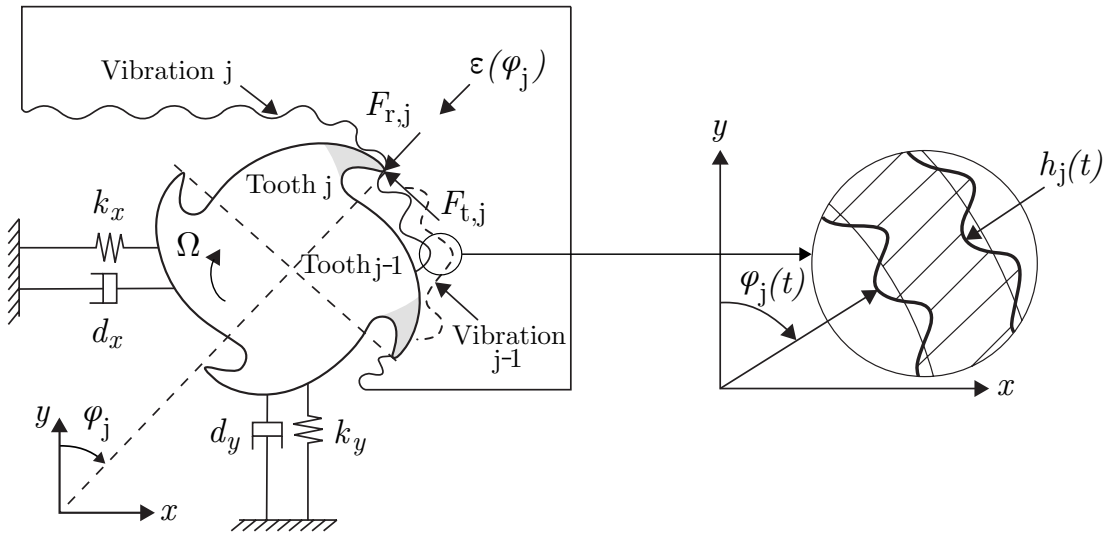


Figure E.1: Regenerative chatter in 2D milling [142]

Chatter vibrations are expressed using a delay differential equation such as the dynamic chip thickness along the radial direction reads

$$\begin{aligned} h(\varphi_j) &= [(x(t) - x(t-T))\sin\varphi_j + (y(t) - y(t-T))\cos\varphi_j]\, g(\varphi_j) \\ &= [\Delta x \sin\varphi_j + \Delta y \cos\varphi_j]\, g(\varphi_j), \end{aligned} \tag{E.1}$$

with $\Delta x = x - x_T$ and $\Delta y = y - y_T$ the relative displacements between two waves imprinted by two consecutive teeth, $T = \frac{60}{N_z \Omega}$ the tooth passing period (if the spindle speed Ω is in [rev/min]), N_c an integer representing the number of waves imprinted by two consecutive teeth, φ_j the instantaneous angular immersion of tooth j and $g(\varphi_j)$ a unit step function that determines whether the tooth is in or out of cut defined as

$$\begin{cases} g(\varphi_j) = 1 & \leftarrow \quad \varphi_{st} < \varphi_j < \varphi_{ex}, \\ g(\varphi_j) = 0 & \leftarrow \quad \varphi_j < \varphi_{st} \ \text{ or } \ \varphi_j > \varphi_{ex}, \end{cases} \tag{E.2}$$

with φ_{st} and φ_{ex}, the start and exit immersion angles, respectively.

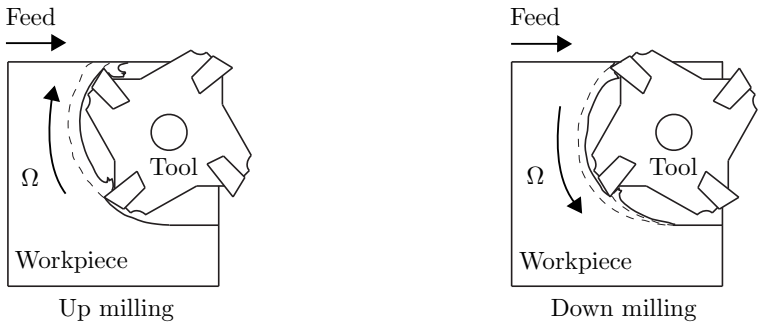


Figure E.2: Up milling and down milling operations [100]

Two main different milling operations can be distinguished based on the geometrical properties, up milling and down milling, which can be seen in Figure E.2. The entering and exiting angles can be calculated in up milling as

$$\varphi_{\mathrm{st}} = 0 \quad \text{and} \quad \varphi_{\mathrm{ex}} = \arccos\left(1 - \frac{2a_{\mathrm{e}}}{D}\right), \tag{E.3}$$

and in down milling such as

$$\varphi_{\mathrm{st}} = \arccos\left(\frac{2a_{\mathrm{e}}}{D} - 1\right) \quad \text{and} \quad \varphi_{\mathrm{ex}} = \pi, \tag{E.4}$$

with a_{e} the radial depth of cut and D the diameter of the cutting tool.

Milling forces are defined proportional to the dynamic chip thickness such as

$$F_{\mathrm{t,j}} = K_{\mathrm{t}} a_{\mathrm{p}} h(\varphi_{\mathrm{j}}) \quad \text{and} \quad F_{\mathrm{r,j}} = K_{\mathrm{r}} F_{\mathrm{t,j}}, \tag{E.5}$$

with $F_{\mathrm{t,j}}$ and $F_{\mathrm{r,j}}$ the tangential and radial forces exerted on tooth j, respectively, K_{t} and K_{r} are the cutting force coefficients for the tangential (subscript t) and the radial directions (subscript r). Cutting forces can be projected in a fixed reference frame (x, y) attached to the milling machine such as

$$\begin{aligned} F_{x,\mathrm{j}} &= -F_{\mathrm{t,j}} \cos\varphi_{\mathrm{j}} - F_{\mathrm{r,j}} \sin\varphi_{\mathrm{j}} \\ F_{y,\mathrm{j}} &= +F_{\mathrm{t,j}} \sin\varphi_{\mathrm{j}} - F_{\mathrm{r,j}} \cos\varphi_{\mathrm{j}}. \end{aligned} \tag{E.6}$$

By summing the contributions of all forces exerted by all teeth, the total dynamic milling forces acting on the end mill reads

$$F_x = \sum_{\mathrm{j}=0}^{\mathrm{N_z}-1} F_{x,\mathrm{j}}, \qquad \text{and} \qquad F_y = \sum_{\mathrm{j}=0}^{\mathrm{N_z}-1} F_{y,\mathrm{j}}. \tag{E.7}$$

Rearranging Eqs. E.1 to E.7, Cartesian cutting forces can be expressed in a matrix form as follows

$$\underbrace{\begin{bmatrix} F_x \\ F_y \end{bmatrix}}_{\mathbf{F}_{\mathrm{c}}(\mathrm{t})} = \frac{1}{2} a_{\mathrm{p}} K_{\mathrm{t}} \underbrace{\begin{bmatrix} a_{xx} & a_{xy} \\ a_{yx} & a_{yy} \end{bmatrix}}_{\mathbf{A}(\mathrm{t})} \underbrace{\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}}_{\boldsymbol{\Delta}(t)}, \tag{E.8}$$

with a_{p} the current axial depth of cut. The directional dynamic milling force coefficients are given by

$$\begin{aligned}
a_{xx} &= \sum_{\mathrm{j}=0}^{\mathrm{N_z}-1} -g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[\sin 2\varphi_{\mathrm{j}} + K_{\mathrm{r}}(1 - \cos 2\varphi_{\mathrm{j}})\right] \\
a_{xy} &= \sum_{\mathrm{j}=0}^{\mathrm{N_z}-1} -g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[(1 + \cos 2\varphi_{\mathrm{j}}) + K_{\mathrm{r}} \sin 2\varphi_{\mathrm{j}}\right] \\
a_{yx} &= \sum_{\mathrm{j}=0}^{\mathrm{N_z}-1} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[(1 - \cos 2\varphi_{\mathrm{j}}) - K_{\mathrm{r}} \sin 2\varphi_{\mathrm{j}}\right] \\
a_{yy} &= \sum_{\mathrm{j}=0}^{\mathrm{N_z}-1} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[\sin 2\varphi_{\mathrm{j}} - K_{\mathrm{r}}(1 + \cos 2\varphi_{\mathrm{j}})\right].
\end{aligned} \tag{E.9}$$

Directional dynamic milling force coefficients vary with time and angular velocity. Equation E.8 can therefore be expressed in the time domain as

$$\mathbf{F}_{\mathrm{c}}(t) = \frac{1}{2} a_{\mathrm{p}} K_{\mathrm{t}} \mathbf{A}(t) \boldsymbol{\Delta}(t). \tag{E.10}$$

Matrix $\mathbf{A}(t)$ is periodic at tooth passing frequency $\omega = \mathrm{N_z}\Omega$ or tooth period $T = \frac{60}{\omega}$ (if spindle speed Ω is in [rev/min]) and can therefore be expressed into Fourier series as

$$\mathbf{A}(t) = \sum_{\mathrm{k}=-\infty}^{\infty} \mathbf{A}_{\mathrm{k}} e^{ir\omega t}, \qquad \mathbf{A}_{\mathrm{k}} = \frac{1}{T} \int_0^T \mathbf{A}(t) e^{-ir\omega t} dt. \tag{E.11}$$

Limiting the Fourier series to the average component, the zero-order term, for r=0, is expressed as

$$\mathbf{A}(t) = \begin{bmatrix} a_{xx} & a_{xy} \\ a_{yx} & a_{yy} \end{bmatrix} \approx \mathbf{A}_0 = \frac{1}{T} \int_0^T [A(t)]\, dt. \tag{E.12}$$

Since $\mathbf{A}_0$ is only valid between the entry φ_{st} and the exit φ_{ex} angles, it becomes equal to the average value of $\mathbf{A}(t)$ at cutter pitch angle $\varphi_{\mathrm{p}} = \frac{2\pi}{\mathrm{N_z}}$ such as

$$\mathbf{A}_0 = \frac{1}{\varphi_{\mathrm{p}}} \int_{\varphi_{\mathrm{st}}}^{\varphi_{\mathrm{ex}}} \mathbf{A}(\varphi) d\varphi = \frac{\mathrm{N_z}}{2\pi} \begin{bmatrix} \alpha_{xx} & \alpha_{xy} \\ \alpha_{yx} & \alpha_{yy} \end{bmatrix}, \tag{E.13}$$

where the integrated functions are provided by

$$\begin{aligned}
\alpha_{xx} &= \frac{1}{2}\left[\cos 2\varphi - 2K_{\mathrm{r}}\varphi + K_{\mathrm{r}}\sin 2\varphi\right]_{\varphi_{\mathrm{st}}}^{\varphi_{\mathrm{ex}}} \\
\alpha_{xy} &= \frac{1}{2}\left[-\sin 2\varphi - 2\varphi + K_{\mathrm{r}}\cos 2\varphi\right]_{\varphi_{\mathrm{st}}}^{\varphi_{\mathrm{ex}}} \\
\alpha_{yx} &= \frac{1}{2}\left[-\sin 2\varphi + 2\varphi + K_{\mathrm{r}}\cos 2\varphi\right]_{\varphi_{\mathrm{st}}}^{\varphi_{\mathrm{ex}}} \\
\alpha_{yy} &= \frac{1}{2}\left[-\cos 2\varphi - 2K_{\mathrm{r}}\varphi - K_{\mathrm{r}}\sin 2\varphi\right]_{\varphi_{\mathrm{st}}}^{\varphi_{\mathrm{ex}}}.
\end{aligned} \tag{E.14}$$

Dynamic milling equation Eq. E.8 giving the cutting forces therefore reduces to

$$\mathbf{F}_{\mathrm{c}}(t) = \frac{1}{2}a_{\mathrm{p}}K_{\mathrm{t}}\mathbf{A}_0\boldsymbol{\Delta}(t), \tag{E.15}$$

where matrix $\mathbf{A}_0$ is time invariant but depends on the direction dynamic milling force coefficients.

In order to determine the stability limits $a_{\mathrm{p_{lim}}}$, the system transfer function matrix $\mathbf{H}(i\omega)$ between force and displacement, identified at the cutter-workpiece contact zone, is expressed as

$$\mathbf{H}(i\omega) = \begin{bmatrix} H_{xx}(i\omega) & H_{xy}(i\omega) \\ H_{yx}(i\omega) & H_{yy}(i\omega) \end{bmatrix}, \tag{E.16}$$

where $H_{xx}(i\omega)$ and $H_{yy}(i\omega)$ are the direct frequency response functions, and $H_{xy}(i\omega)$ and $H_{yx}(i\omega)$ are the cross transfer functions. Then, defining $\mathbf{r} = [x(t)\ y(t)]^T$ and $\mathbf{r}_0 = [x(t-T)\ y(t-T)]^T$, the vibrations at chatter frequency ω_{c} in the frequency domain using harmonic functions are expressed as

$$\begin{cases} \mathbf{r}(i\omega_{\mathrm{c}}) & = \mathbf{H}(i\omega)\mathbf{F}_{\mathrm{c}}e^{i\omega_{\mathrm{c}}t} \\ \mathbf{r}_T(i\omega_{\mathrm{c}}) & = e^{-i\omega_{\mathrm{c}}T}\mathbf{r}(i\omega_{\mathrm{c}}). \end{cases} \tag{E.17}$$

By substituting $\boldsymbol{\Delta} = [(x - x_T)\ (y - y_T)]^T$ in Eq. E.17, it gives

$$\begin{aligned}
\boldsymbol{\Delta} &= \mathbf{r}(i\omega_{\mathrm{c}}) - \mathbf{r}_T(i\omega_{\mathrm{c}}) \\
&= [1 - e^{-i\omega_{\mathrm{c}}T}]e^{i\omega_{\mathrm{c}}t}\mathbf{H}(i\omega)\mathbf{F}_{\mathrm{c}},
\end{aligned} \tag{E.18}$$

where $\omega_{\mathrm{c}}T$ is the phase delay between the vibrations at successive tooth period T. Replacing $\boldsymbol{\Delta}(i\omega_{\mathrm{c}})$ in dynamic milling equation E.15 gives

$$\mathbf{F}e^{i\omega_{\mathrm{c}}t} = \frac{1}{2}a_{\mathrm{p}}K_{\mathrm{t}}\left[1 - e^{-i\omega_{\mathrm{c}}T}\right]\mathbf{A}_0\mathbf{H}(i\omega)\mathbf{F}_{\mathrm{c}}e^{i\omega_{\mathrm{c}}t}, \tag{E.19}$$

which results in non-trivial solution if its determinant is null such as

$$\det\left[\mathbf{I}-\frac{1}{2}a_{\mathrm{p}}K_{\mathrm{t}}\left[1-e^{-i\omega_{\mathrm{c}}T}\right]\mathbf{A}_0\mathbf{H}(i\omega)\right]=0, \tag{E.20}$$

being the characteristic equation of the closed loop dynamic milling system. Recalling the expression of $\mathbf{A}_0$ and defining the oriented transfer function matrix as

$$\mathbf{H}_0(i\omega_{\mathrm{c}})=\left[\begin{array}{cc}\alpha_{xx}H_{xx}(i\omega_c)+\alpha_{xy}H_{yx}(i\omega_{\mathrm{c}}) & \alpha_{xx}H_{xy}(i\omega_c)+\alpha_{\mathrm{xy}}H_{yy}(i\omega_{\mathrm{c}})\\ \alpha_{yx}H_{xx}(i\omega_{\mathrm{c}})+\alpha_{yy}H_{yx}(i\omega_{\mathrm{c}} & \alpha_{yx}H_{xy}(i\omega_{\mathrm{c}})+\alpha_{yy}H_{yy}(i\omega_{\mathrm{c}})\end{array}\right], \tag{E.21}$$

and the eigenvalue of the characteristic equation as

$$\Lambda=-\frac{\mathrm{N_z}}{4\pi}a_{\mathrm{p}}K_{\mathrm{t}}(1-e^{-i\omega_{\mathrm{c}}T}), \tag{E.22}$$

the resulting characteristic equation becomes

$$\det\left[\mathbf{I}+\Lambda\mathbf{H}_0(i\omega_{\mathrm{c}})\right]=0. \tag{E.23}$$

If no cross-coupling along orthogonal directions x and y is assumed, i.e. $H_{xy}=H_{yx}=0$, the characteristic equation reduces to a quadratic function such as

$$a_0\Lambda^2+a_1\Lambda+1=0, \tag{E.24}$$

where

$$\begin{aligned}a_0&=H_{xx}(i\omega_{\mathrm{c}})H_{yy}(i\omega_{\mathrm{c}})(\alpha_{xx}\alpha_{yy}-\alpha_{xy}\alpha_{yx}),\\ a_1&=\alpha_{xx}H_{xx}(i\omega_{\mathrm{c}})+\alpha_{yy}H_{yy}(i\omega_{\mathrm{c}}).\end{aligned} \tag{E.25}$$

Eigenvalue has a real and a imaginary part since the transfer functions are complex. The critical axial depth of cut $a_{\mathrm{p_{lim}}}$ at chatter frequency ω_{c} is provided by substituting $e^{-i\omega_{\mathrm{c}}T}=\cos\omega_{\mathrm{c}}T-i\sin\omega_{\mathrm{c}}T$ in E.22 such as

$$a_{\mathrm{p_{lim}}}=\frac{-2\pi}{\mathrm{N_z}K_{\mathrm{t}}}\left[\frac{\Lambda_{\mathrm{R}}(1-\cos\omega_{\mathrm{c}}T)+\Lambda_{\mathrm{I}}\sin\omega_{\mathrm{c}}T}{1-\cos\omega_{\mathrm{c}}T}+i\frac{\Lambda_{\mathrm{I}}(1-\cos\omega_{\mathrm{c}}T)-\Lambda_{\mathrm{R}}\sin\omega_{\mathrm{c}}T}{1-\cos\omega_{\mathrm{c}}T}\right]. \tag{E.26}$$

Since the critical axial depth of cut $a_{\mathrm{p_{lim}}}$ is a real number, the imaginary part of Eq. E.26 must disappear. Variable κ is therefore introduced as the ratio of the imaginary and real parts of the eigenvalue such as

$$\begin{aligned}\Lambda_{\mathrm{I}}(1-\cos\omega_{\mathrm{c}}T)-\Lambda_{\mathrm{R}}\sin\omega_{\mathrm{c}}T&=0\\ \Rightarrow\kappa=\frac{\Lambda_{\mathrm{I}}}{\Lambda_{\mathrm{R}}}&=\frac{\sin\omega_{\mathrm{c}}T}{(1-\cos\omega_{\mathrm{c}}T)},\end{aligned} \tag{E.27}$$

with ω_{c} the chatter frequency which is assumed to be known since $H(\omega)$ is swept for all ω_{c}=ω.

The final expression of the chatter free axial depth of cut is reduced to

$$a_{\mathrm{p_{lim}}} = -\frac{2\pi\Lambda_{\mathrm{R}}}{\mathrm{N_z}K_{\mathrm{t}}}(1+\kappa^2). \tag{E.28}$$

Hence, if ϵ is the phase shift between inner and outer modulations computed as $\epsilon = \pi - 2\tan^{-1}\kappa$ $(0 \leq \epsilon < 2\pi)$ and $\mathrm{N_c}$ is the integer number of full vibration waves or lobes imprinted on the cutting arc, the tooth passing periods T and corresponding spindle speeds Ω are computed as

$$T = \frac{1}{\omega_{\mathrm{c}}}(\epsilon + 2\mathrm{N_c}\pi) \rightarrow \Omega = \frac{60}{\mathrm{N_z}T} \text{ rev/min.} \tag{E.29}$$

E.2 Stability in 2D milling with cross-coupling

If cross transfer functions in $\mathbf{H}(i\omega)$ i.e. H_{xy} and H_{yx}, are non negligible, coefficients of the characteristic equation (Eq. E.24) must be recomputed. Hence, expressions of coefficients a_0 and a_1 become

$$\begin{aligned} a_0 =& (\alpha_{xx}H_{xx} + \alpha_{xy}H_{yx})(\alpha_{yx}H_{xy} + \alpha_{yy}H_{yy}) - \\ & (\alpha_{xx}H_{xy} + \alpha_{xy}H_{yy})(\alpha_{yx}H_{xx} + \alpha_{yy}H_{yx}) \\ a_1 =& (\alpha_{xx}H_{xx} + \alpha_{xy}H_{yx}) + (\alpha_{yx}H_{xy} + \alpha_{yy}H_{yy}). \end{aligned} \tag{E.30}$$

The remaining of the derivation leading to the computation of the chatter free axial depth of cut is the same.

E.3 Stability in 3D milling with cross-coupling

In case of 3D milling, the Cartesian cutting forces directed along the X, Y and Z directions read

$$\underbrace{\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}}_{\mathbf{F_c}(\mathrm{t})} = \frac{1}{2}a_{\mathrm{p}}K_{\mathrm{t}} \underbrace{\begin{bmatrix} a_{xx} & a_{xy} & a_{xz} \\ a_{yx} & a_{yy} & a_{yz} \\ a_{zx} & a_{zy} & a_{zz} \end{bmatrix}}_{\mathbf{A}(\mathrm{t})} \underbrace{\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}}_{\boldsymbol{\Delta}(t)}, \tag{E.31}$$

with $\mathbf{A}(t)$ a (3×3) matrix formed by the directional dynamic milling force coefficients.

Considering immersion angle κ giving the axial positioning of tooth j (Figure E.3), the zero-order components of matrix $\mathbf{A}_0$ are provided by the following integrated functions [181]

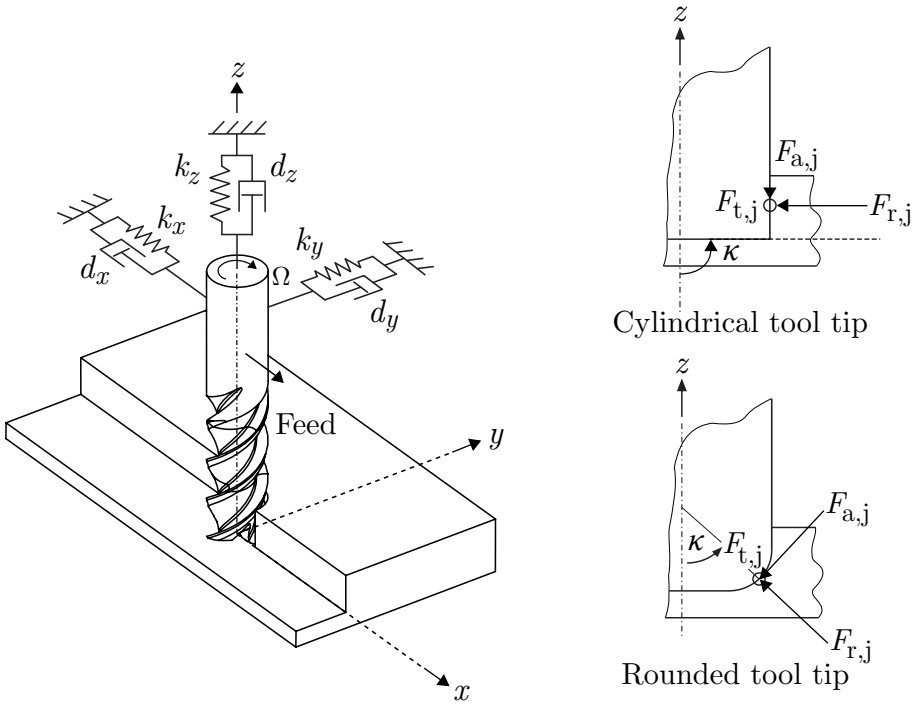


Figure E.3: Immersion angle κ in 3D milling

$$
\begin{aligned}
\alpha_{xx} =& \left[-K_\mathrm{a}\varphi\cos\kappa\sin\kappa + \frac{1}{2}\cos 2\varphi\sin\kappa - K_\mathrm{r}\varphi\sin^2\kappa + \right. \\
& \left. \frac{1}{2}K_\mathrm{a}\cos\kappa\sin\kappa\sin 2\varphi + \frac{1}{2}K_\mathrm{r}\sin^2\kappa\sin 2\varphi \right]_{\varphi_\mathrm{st}}^{\varphi_\mathrm{ex}} \\
\alpha_{xy} =& \left[-\varphi\sin\kappa + \frac{1}{2}K_\mathrm{a}\cos\kappa\cos 2\varphi\sin\kappa + \frac{1}{2}K_\mathrm{r}\cos 2\varphi\sin^2\kappa - \frac{1}{2}\sin\kappa\sin 2\varphi \right]_{\varphi_\mathrm{st}}^{\varphi_\mathrm{ex}} \\
\alpha_{xz} =& 2\left[-K_\mathrm{a}\cos^2\kappa\cos\varphi - K_\mathrm{r}\cos\kappa\cos\varphi\sin\kappa + \cos\kappa\sin\varphi \right]_{\varphi_\mathrm{st}}^{\varphi_\mathrm{ex}} \\
\alpha_{yx} =& \left[\varphi\sin\kappa + \frac{1}{2}K_\mathrm{a}\cos\kappa\cos 2\varphi\sin\kappa + \frac{1}{2}K_\mathrm{r}\cos 2\varphi\sin^2\kappa - \frac{1}{2}\sin\kappa\sin 2\varphi \right]_{\varphi_\mathrm{st}}^{\varphi_\mathrm{ex}} \\
\alpha_{yy} =& \left[-K_\mathrm{a}\varphi\cos\kappa\sin\kappa - \frac{1}{2}\cos 2\varphi\sin\kappa - K_\mathrm{r}\varphi\sin^2\kappa - \right. \\
& \left. \frac{1}{2}K_\mathrm{a}\cos\kappa\sin\kappa\sin 2\varphi - \frac{1}{2}K_\mathrm{r}\sin^2\kappa\sin 2\varphi \right]_{\varphi_\mathrm{st}}^{\varphi_\mathrm{ex}} \\
\alpha_{yz} =& 2\left[\cos\kappa\cos\varphi + K_\mathrm{a}\cos^2\kappa\sin\varphi + K_\mathrm{r}\cos\kappa\sin\kappa\sin\varphi \right]_{\varphi_\mathrm{st}}^{\varphi_\mathrm{ex}} \\
\alpha_{zx} =& 2\left[-K_\mathrm{r}\cos\kappa\cos\varphi\sin\kappa + K_\mathrm{a}\cos\varphi\sin^2\kappa \right]_{\varphi_\mathrm{st}}^{\varphi_\mathrm{ex}} \\
\alpha_{zy} =& 2\left[K_\mathrm{r}\cos\kappa\sin\kappa\sin\varphi - K_\mathrm{a}\sin^2\kappa\sin\varphi \right]_{\varphi_\mathrm{st}}^{\varphi_\mathrm{ex}} \\
\alpha_{zz} =& 2\left[\varphi\cos\kappa(-K_\mathrm{r}\cos\kappa + K_\mathrm{a}\sin\kappa) \right]_{\varphi_\mathrm{st}}^{\varphi_\mathrm{ex}},
\end{aligned}
\tag{E.32}
$$

with $K_\mathrm{a} = \dfrac{K_\mathrm{ac}}{K_\mathrm{t}}$ the axial cutting force coefficient and $K_\mathrm{r} = \dfrac{K_\mathrm{rc}}{K_\mathrm{t}}$ the radial cutting force coefficient.

Similarly, the transfer function matrix $\mathbf{H}(i\omega)$ is a (3×3) matrix. Hence, the characteristic equation becomes a cubic function such as

$$a_0\Lambda^3 + a_1\Lambda^2 + a_2\Lambda + 1 = 0. \tag{E.33}$$

If matrix $\mathbf{H}_0(i\omega)$ is expressed as

$$\mathbf{H}_0(i\omega_c) = \begin{bmatrix} \alpha_{xx} & \alpha_{xy} & \alpha_{xz} \\ \alpha_{yx} & \alpha_{yy} & \alpha_{yz} \\ \alpha_{zx} & \alpha_{zy} & \alpha_{zz} \end{bmatrix} \begin{bmatrix} H_{xx}(i\omega_c) & H_{xy}(i\omega_c) & H_{xz}(i\omega_c) \\ H_{yx}(i\omega_c) & H_{yy}(i\omega_c) & H_{yz}(i\omega_c) \\ H_{zx}(i\omega_c) & H_{zy}(i\omega_c) & H_{zz}(i\omega_c) \end{bmatrix}, \tag{E.34}$$

then, coefficients of the cubic function are expressed as follows [143]

$$\begin{aligned} a_0 =& (-H_{0,xx}H_{0,yy}H_{0,zz} + H_{0,xx}H_{0,yz}H_{0,zy} + H_{0,yx}H_{0,xy}H_{0,zz} \\ & - H_{0,yx}H_{0,xz}H_{0,zy} - H_{0,zx}H_{0,xy}H_{0,yz} + H_{0,zx}H_{0,xz}H_{0,yy}) \\ a_1 =& (H_{0,yy}H_{0,zz} - H_{0,yz}H_{0,zy} + H_{0,xx}H_{0,zz} + H_{0,xx}H_{0,yy} - H_{0,yx}H_{0,xy} - H_{0,zx}H_{0,xz}) \\ a_2 =& (-H_{0,zz} - H_{0,yy} - H_{0,xx}). \end{aligned} \tag{E.35}$$

Finally, the chatter free axial depth of cut is still given by equation Eq. E.28.

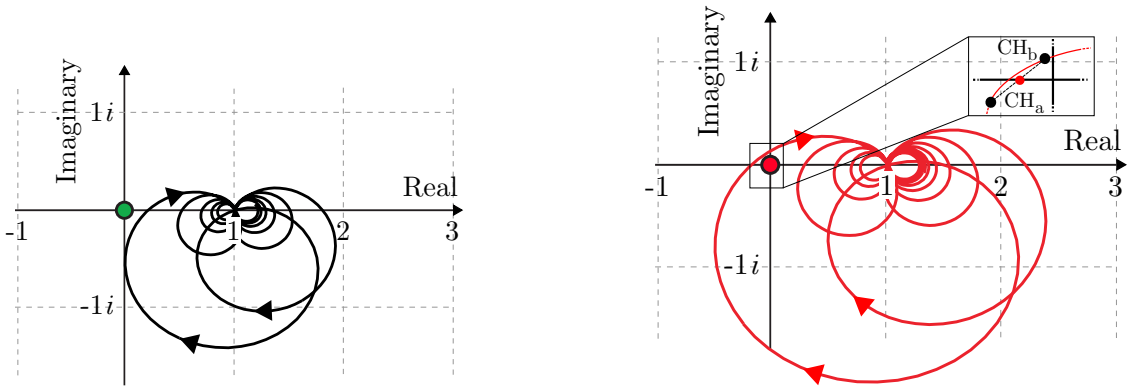
E.4 Alternative resolution with Nyquist contour

An alternative resolution of the zero-order method is explained in this appendix and follows the steps presented by Eynian in [144]. It relies on the Nyquist contour plot formed by the solutions of the characteristic equation. Characteristic equation to solve in milling was presented in Eq. E.23 and was expressed such as

$$\mathrm{CH}(i\omega_c) = \det\left[\mathbf{I} + \Lambda\mathbf{H}_0(i\omega_c)\right] = 0. \tag{E.36}$$

Poles of the characteristic equation $\mathrm{CH}(i\omega_c)$ are the poles of the structure $\mathbf{H}_0(i\omega_c)$ which are all stable. Any unstable zero of the characteristic equation $\mathrm{CH}(i\omega_c)$ creates a clockwise encirclement of the origin of the complex plane by a Nyquist mapping of the characteristic equation. In Eq. E.36, the highest order of s in Λ, if $s = (i\omega_c)$, is one. On the other hand, the lowest order of s in $\mathbf{H}_0(\omega)$ is $\frac{1}{s^2}$. Therefore, if $\mid s \mid \rightarrow \infty$, $\mathrm{CH}(\infty)$=det($\mathbf{I}$)=1. In other words, the semicircle part of the Nyquist contour is mapped to point +1 on the real axis. As a result, it is sufficient to count the encirclements for mapping of the positive imaginary axis. Nyquist plots of the same delay system i.e. same oriented transfer function matrix $\mathbf{H}_0$, are shown for stable and unstable cases in figure E.4. Hence, any encirclement of the origin i.e. a crossing of the negative real axis, represents instability.

While Nyquist contour is a continuous path, the curve is sampled at discrete frequencies when using a digital computer. It is also the case when measuring the frequency response



(a) Stable case (b) Unstable case (chatter vibrations)

Figure E.4: Nyquist contour [144]

function matrix of the milling machine $\mathbf{H}(\omega)$. Hence, the Nyquist contour is only evaluated at the measured frequencies of the FRFs. Consequently, in order to compute the stability lobe diagram, the stability of each point of Nyquist contour reflecting a pair of axial depth of cut a_p and spindle speed Ω is assessed. Considering that the stability is appraised from a stable state (right part of complex plane), any instability is detected when the Nyquist curve cross the negative real axis i.e. when there is a transition from point CH_a ($Re(CH_a) < 0$ and $Im(CH_a) < 0$) to point CH_b ($Re(CH_b) < 0$ and $Im(CH_b) \geq 0$) as illustrated in Figure E.4b. It is eventually possible to get a better estimation of the chatter frequency by using a linear interpolation between points CH_a and CH_b. Chatter frequency is determined when the imaginary part is null.

The digital resolution of zero-order method is used in 3D milling to derive the stability lobe diagrams (method is later called 3D ZOA digital).

Appendix F

Semi-discretisation method detail

F.1 Stability in 2D milling without cross-coupling

In 2D milling, tangential (subscript t) and radial (subscript r) dynamic cutting forces applied by tooth j on the system are proportional to the axial depth of cut a_p and chip load such as

$$\begin{bmatrix} F_{\mathrm{t,j}}(t) \\ F_{\mathrm{r,j}}(t) \end{bmatrix} = -g_\mathrm{j}(t) a_\mathrm{p} K_\mathrm{t} \begin{bmatrix} 1 \\ K_\mathrm{r} \end{bmatrix} \begin{bmatrix} \sin\varphi_\mathrm{j}(t) \\ \cos\varphi_\mathrm{j}(t) \end{bmatrix}^T \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}, \tag{F.1}$$

with $g_\mathrm{j}(t)$ the unit step function that determines whether the tooth is in or out of cut (Eq. 4.8), $K_\mathrm{r} = \dfrac{K_\mathrm{rc}}{K_\mathrm{t}}$ the radial cutting force coefficient, a_p the axial depth of cut and φ_j the instantaneous angular immersion of tooth j. Then, the cutting forces in feed (x) and orthogonal (y) directions are given by the following geometric transformation:

$$\begin{bmatrix} F_{x,\mathrm{j}} \\ F_{y,\mathrm{j}} \end{bmatrix} = \begin{bmatrix} -\cos\varphi_\mathrm{j} & -\sin\varphi_\mathrm{j} \\ \sin\varphi_\mathrm{j} & -\cos\varphi_\mathrm{j} \end{bmatrix} \begin{bmatrix} F_{\mathrm{t,j}} \\ F_{\mathrm{r,j}} \end{bmatrix}. \tag{F.2}$$

Hence, similar expressions for the directional milling force coefficients as previously shown in Eq. E.9 are derived such as [150]

$$\begin{aligned} h_{xx}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_\mathrm{j}(\varphi_\mathrm{j}) \left[\sin\varphi_\mathrm{j}(K_\mathrm{t}\cos\varphi_\mathrm{j} + K_\mathrm{rc}\sin\varphi_\mathrm{j})\right] \\ h_{xy}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_\mathrm{j}(\varphi_\mathrm{j}) \left[\cos\varphi_\mathrm{j}(K_\mathrm{t}\cos\varphi_\mathrm{j} + K_\mathrm{rc}\sin\varphi_\mathrm{j})\right] \\ h_{yx}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_\mathrm{j}(\varphi_\mathrm{j}) \left[\sin\varphi_\mathrm{j}(-K_\mathrm{t}\sin\varphi_\mathrm{j} + K_\mathrm{rc}\cos\varphi_\mathrm{j})\right] \\ h_{yy}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_\mathrm{j}(\varphi_\mathrm{j}) \left[\cos\varphi_\mathrm{j}(-K_\mathrm{t}\sin\varphi_\mathrm{j} + K_\mathrm{rc}\cos\varphi_\mathrm{j})\right]. \end{aligned} \tag{F.3}$$

The governing equation of a 2DOF milling model with a symmetric tool reads

$$\begin{bmatrix} \ddot{x}(t) \\ \ddot{y}(t) \end{bmatrix} + \begin{bmatrix} 2\zeta\omega_n & 0 \\ 0 & 2\zeta\omega_n \end{bmatrix} \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \end{bmatrix} + \begin{bmatrix} \omega_n^2 + \frac{a_\mathrm{p} h_{xx}}{m} & \frac{a_\mathrm{p} h_{xy}}{m} \\ \frac{a_\mathrm{p} h_{yx}}{m} & \omega_n^2 + \frac{a_\mathrm{p} h_{yy}}{m} \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} \frac{a_\mathrm{p} h_{xx}}{m} & \frac{a_\mathrm{p} h_{xy}}{m} \\ \frac{a_\mathrm{p} h_{yx}}{m} & \frac{a_\mathrm{p} h_{yy}}{m} \end{bmatrix} \begin{bmatrix} x(t-T) \\ y(t-T) \end{bmatrix}, \tag{F.4}$$

with ω_n the angular natural frequency, ζ the damping ratio and m the modal mass (same in both direction as the tool is symmetric).

In the i^{th} semi-discretisation interval, Eq. F.4 can be approximated as

$$\begin{bmatrix} \ddot{x}(t) \\ \ddot{y}(t) \end{bmatrix} + \begin{bmatrix} 2\zeta\omega_n & 0 \\ 0 & 2\zeta\omega_n \end{bmatrix} \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \end{bmatrix} + \begin{bmatrix} \omega_n^2 + \frac{a_\mathrm{p} h_{xx,\mathrm{i}}}{m} & \frac{a_\mathrm{p} h_{xy,\mathrm{i}}}{m} \\ \frac{a_\mathrm{p} h_{yx,\mathrm{i}}}{m} & \omega_n^2 + \frac{a_\mathrm{p} h_{yy,\mathrm{i}}}{m} \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} \frac{a_\mathrm{p} h_{xx,\mathrm{i}}}{m} & \frac{a_\mathrm{p} h_{xy,\mathrm{i}}}{m} \\ \frac{a_\mathrm{p} h_{yx,\mathrm{i}}}{m} & \frac{a_\mathrm{p} h_{yy,\mathrm{i}}}{m} \end{bmatrix} \begin{bmatrix} x_{T,\mathrm{i}} \\ y_{T,\mathrm{i}} \end{bmatrix}. \tag{F.5}$$

By Cauchy transformation and linear approximation of the delayed term, Eq. F.5 can be written as $2n$ ordinary differential equations such as

$$\dot{\mathbf{u}}(t) = \mathbf{A}_\mathrm{i}\mathbf{u}(t) + w\mathbf{B}_\mathrm{i}\mathbf{u}_{\mathrm{i\text{-}k+1}} + w\mathbf{B}_\mathrm{i}\mathbf{u}_{\mathrm{i\text{-}k}}, \tag{F.6}$$

where

$$\mathbf{A}_\mathrm{i} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\omega_n^2 - \frac{a_\mathrm{p} h_{xx,\mathrm{i}}}{m} & -\frac{a_\mathrm{p} h_{xy,\mathrm{i}}}{m} & -2\zeta\omega_n & 0 \\ -\frac{a_\mathrm{p} h_{yx,\mathrm{i}}}{m} & -\omega_n^2 - \frac{a_\mathrm{p} h_{yy,\mathrm{i}}}{m} & 0 & -2\zeta\omega_n \end{bmatrix}, \tag{F.7}$$

$$\mathbf{B}_\mathrm{i} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{a_\mathrm{p} h_{xx,\mathrm{i}}}{m} & \frac{a_\mathrm{p} h_{xy,\mathrm{i}}}{m} & 0 & 0 \\ \frac{a_\mathrm{p} h_{yx,\mathrm{i}}}{m} & \frac{a_\mathrm{p} h_{yy,\mathrm{i}}}{m} & 0 & 0 \end{bmatrix}, \tag{F.8}$$

$$\mathbf{u}(t) = \begin{bmatrix} x(t) \\ y(t) \\ \dot{x}(t) \\ \dot{y}(t) \end{bmatrix}, \text{ and } \mathbf{u}_\mathrm{j} = \mathbf{u}(t_\mathrm{j}) = \begin{bmatrix} x(t_\mathrm{j}) \\ y(t_\mathrm{j}) \\ \dot{x}(t_\mathrm{j}) \\ \dot{y}(t_\mathrm{j}) \end{bmatrix} = \begin{bmatrix} x_\mathrm{j} \\ y_\mathrm{j} \\ \dot{x}_\mathrm{j} \\ \dot{y}_\mathrm{j} \end{bmatrix}. \tag{F.9}$$

for any integer j and $w = \frac{1}{2}$ since the time delay is equal to the time period in milling. Hence, for initial condition $\mathbf{u}(t_\mathrm{i}) = \mathbf{u}_\mathrm{i}$, $\mathbf{u}_{\mathrm{i}+1}$ is determined in a similar way as Eq. 4.30 with

$$\mathbf{u}_{\mathrm{i}+1} = \mathbf{P}_\mathrm{i}\mathbf{u}_\mathrm{i} + w\mathbf{R}_\mathrm{i}\mathbf{u}_{\mathrm{i-k}+1} + w\mathbf{R}\mathbf{u}_{\mathrm{i-k}}, \tag{F.10}$$

where

$$\mathbf{P}_\mathrm{i} = \exp(\mathbf{A}_\mathrm{i}\Delta t), \quad \text{and} \quad \mathbf{R}_\mathrm{i} = (\exp(\mathbf{A}_\mathrm{i}\Delta t) - \mathbf{I})\mathbf{A}_\mathrm{i}^{-1}\mathbf{B}_\mathrm{i}. \tag{F.11}$$

The resulting discrete map reads

$$\mathbf{y}_{\mathrm{i}+1} = \mathbf{C}_\mathrm{i}\mathbf{y}_\mathrm{i}, \tag{F.12}$$

in which transition matrix is given by $\mathbf{\Phi} = \mathbf{C}_{\mathrm{k-1}}\mathbf{C}_{\mathrm{k-2}}\dots\mathbf{C}_1\mathbf{C}_0$ to finally obtain the relationship between the initial state and the state one period later $T = \mathrm{k}\Delta t$ obtained as

$$\mathbf{y}_\mathrm{k} = \mathbf{\Phi}\mathbf{y}_0. \tag{F.13}$$

Expression of matrix $\mathbf{C}_\mathrm{i}$ is provided by Eq. 4.33 and depends on $\mathbf{P}_\mathrm{i}$, $\mathbf{R}_\mathrm{i}$ and w.

F.2 Stability in 3D milling without cross-coupling

In 3D milling, tangential (subscript t), radial (subscript r) and axial (subscript a) dynamic cutting forces applied by tooth j on the system at immersion angle κ are proportional to the axial depth of cut a_p and chip load such as

$$\begin{bmatrix} F_{\mathrm{t,j}}(t) \\ F_{\mathrm{r,j}}(t) \\ F_{\mathrm{a,j}}(t) \end{bmatrix} = -g_\mathrm{j}(t)a_\mathrm{p}K_\mathrm{t} \begin{bmatrix} 1 \\ K_\mathrm{r} \\ K_\mathrm{a} \end{bmatrix} \begin{bmatrix} \sin\varphi_\mathrm{j}(t)\sin\kappa \\ \cos\varphi_\mathrm{j}(t)\sin\kappa \\ -\cos\kappa \end{bmatrix}^T \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}, \tag{F.14}$$

with $g_\mathrm{j}(t)$ the unit step function that determines whether the tooth is in or out of cut (Eq. 4.8), $K_\mathrm{r} = \dfrac{K_\mathrm{rc}}{K_\mathrm{t}}$ the radial cutting force coefficient, $K_\mathrm{a} = \dfrac{K_\mathrm{ac}}{K_\mathrm{t}}$ the axial cutting force coefficient and φ_j the instantaneous angular immersion of tooth j. Then, the cutting forces in feed (x), orthogonal (y) and axial (z) directions are given by the following geometric transformation:

$$\begin{bmatrix} F_{x,\mathrm{j}} \\ F_{y,\mathrm{j}} \\ F_{z,\mathrm{j}} \end{bmatrix} = \begin{bmatrix} -\cos\varphi_\mathrm{j} & -\sin\varphi_\mathrm{j}\sin\kappa & -\sin\varphi_\mathrm{j}\cos\kappa \\ \sin\varphi_\mathrm{j} & -\cos\varphi_\mathrm{j}\sin\kappa & -\cos\varphi_\mathrm{j}\cos\kappa \\ 0 & \cos\kappa & -\sin\kappa \end{bmatrix} \begin{bmatrix} F_{\mathrm{t,j}} \\ F_{\mathrm{r,j}} \\ F_{\mathrm{a,j}} \end{bmatrix}. \tag{F.15}$$

Directional milling force coefficients previously shown in Eq. F.3 are extended to 3D milling such as

$$
\begin{aligned}
h_{xx}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[\sin\kappa \sin\varphi_{\mathrm{j}}(K_{\mathrm{t}} \cos\varphi_{\mathrm{j}} + K_{\mathrm{ac}} \cos\kappa \sin\varphi_{\mathrm{j}} + K_{\mathrm{rc}} \sin\kappa \sin\varphi_{\mathrm{j}})\right] \\
h_{xy}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[\sin\kappa \cos\varphi_{\mathrm{j}}(K_{\mathrm{t}} \cos\varphi_{\mathrm{j}} + K_{\mathrm{ac}} \cos\kappa \sin\varphi_{\mathrm{j}} + K_{\mathrm{rc}} \sin\kappa \sin\varphi_{\mathrm{j}})\right] \\
h_{xz}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[-\cos\kappa(K_{\mathrm{t}} \cos\varphi_{\mathrm{j}} + K_{\mathrm{ac}} \cos\kappa \sin\varphi_{\mathrm{j}} + K_{\mathrm{rc}} \sin\kappa \sin\varphi_{\mathrm{j}})\right] \\
h_{yx}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[\sin\kappa \sin\varphi_{\mathrm{j}}(K_{\mathrm{ac}} \cos\kappa \cos\varphi_{\mathrm{j}} - K_{\mathrm{t}} \sin\varphi_{\mathrm{j}} + K_{\mathrm{rc}} \sin\kappa \cos\varphi_{\mathrm{j}})\right] \\
h_{yy}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[\sin\kappa \cos\varphi_{\mathrm{j}}(K_{\mathrm{ac}} \cos\kappa \cos\varphi_{\mathrm{j}} - K_{\mathrm{t}} \sin\varphi_{\mathrm{j}} + K_{\mathrm{rc}} \sin\kappa \cos\varphi_{\mathrm{j}})\right] \\
h_{yz}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[-\cos\kappa(K_{\mathrm{ac}} \cos\kappa \cos\varphi_{\mathrm{j}} - K_{\mathrm{t}} \sin\varphi_{\mathrm{j}} + K_{\mathrm{rc}} \sin\kappa \cos\varphi_{\mathrm{j}})\right] \\
h_{zx}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[-\sin\kappa \sin\varphi_{\mathrm{j}}(K_{\mathrm{rc}} \cos\kappa - K_{\mathrm{ac}} \sin\kappa)\right] \\
h_{zy}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[-\sin\kappa \cos\varphi_{\mathrm{j}}(K_{\mathrm{rc}} \cos\kappa - K_{\mathrm{ac}} \sin\kappa)\right] \\
h_{zz}(t) &= \sum_{\mathrm{j}=1}^{\mathrm{N_z}} g_{\mathrm{j}}(\varphi_{\mathrm{j}}) \left[\cos\kappa(K_{\mathrm{rc}} \cos\kappa - K_{\mathrm{ac}} \sin\kappa)\right].
\end{aligned}
\tag{F.16}
$$

The projections of the cutting force coefficients are gathered in matrix $\mathbf{h}(t)$ such as

$$
\mathbf{h}(t) = \begin{bmatrix} h_{xx}(t) & h_{xy}(t) & h_{xz}(t) \\ h_{yx}(t) & h_{yy}(t) & h_{yz}(t) \\ h_{zx}(t) & h_{zy}(t) & h_{zz}(t) \end{bmatrix}. \tag{F.17}
$$

For a milling model with n_m modes along the X, Y and Z directions, the governing equation with a symmetric tool can be expressed as

$$\begin{bmatrix} m_{x,1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & m_{x,n_m} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_{y,1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & m_{y,n_m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & m_{z,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & m_{z,n_m} \end{bmatrix} \begin{bmatrix} \ddot{x}_1(t) \\ \vdots \\ \ddot{x}_{n_m}(t) \\ \ddot{y}_1(t) \\ \vdots \\ \ddot{y}_{n_m}(t) \\ \ddot{z}_1(t) \\ \vdots \\ \ddot{z}_{n_m}(t) \end{bmatrix} +$$

$$\begin{bmatrix} \mathbf{c}_x & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{c}_y & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{c}_z \end{bmatrix} \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{y}}(t) \\ \dot{\mathbf{z}}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{k}_x & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{k}_y & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{k}_z \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{y}(t) \\ \mathbf{z}(t) \end{bmatrix} = \mathbf{h}(t) \otimes \mathbf{I} \begin{bmatrix} \mathbf{x}(t-T) \\ \mathbf{y}(t-T) \\ \mathbf{z}(t-T) \end{bmatrix}, \tag{F.18}$$

with $\mathbf{x}(t)$, $\mathbf{y}(t)$ and $\mathbf{z}(t) \in \mathbb{R}^{n_m \times 1}$, $m_{x|y|z,\mathrm{j}}$ the modal mass of mode j along the X, Y or Z direction, $\mathbf{c}_x$, $\mathbf{c}_y$ and $\mathbf{c}_z \in \mathbb{R}^{n_m \times n_m}$ the diagonal matrices containing the modal damping for each mode along each direction, $\mathbf{k}_x$, $\mathbf{k}_y$ and $\mathbf{k}_z \in \mathbb{R}^{n_m \times n_m}$ the diagonal matrices containing the modal stiffness for each mode along each direction such as, for instance,

$$\mathbf{c}_x = \begin{bmatrix} 2m_{x,1}\zeta_{x,1}\omega_{x,1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 2m_{x,n_m}\zeta_{x,n_m}\omega_{x,n_m} \end{bmatrix}, \text{and} \, \mathbf{k}_x = \begin{bmatrix} m_{x,1}\omega_{x,1}^2 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & m_{x,n_m}\omega_{x,n_m}^2 \end{bmatrix} \tag{F.19}$$

and $\mathbf{I} \in \mathbb{R}^{n_m \times n_m}$ the identity matrix and $\otimes$ the Kronecker tensor product.

Eq. F.18 is written in a compact form as

$$\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{C}\dot{\mathbf{u}}(t) + \mathbf{K}\mathbf{u}(t) = \mathbf{H}(t)\mathbf{u}(t-T), \tag{F.20}$$

with $\mathbf{M}$, $\mathbf{C}$, $\mathbf{K} \in \mathbb{R}^{3n_m \times 3n_m}$ the modal mass, damping and stiffness matrices of the milling system and vector $\mathbf{u}(t)$ expressed as

$$\mathbf{u}(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_{n_m}(t) \\ y_1(t) \\ \vdots \\ y_{n_m}(t) \\ z_1(t) \\ \vdots \\ z_{n_m}(t) \end{bmatrix}. \tag{F.21}$$

Similarly to Eq. F.6, the Cauchy transformation and the linear approximation of the delayed term allows writing the milling model into a $(2 \times 3 \times n_m)$ system of equations (3 directions are considered i.e. X, Y and Z directions) such as

$$\dot{\mathbf{u}}(t) = \mathbf{A}_{\mathrm{i}}\mathbf{u}(t) + w\mathbf{B}_{\mathrm{i}}\mathbf{u}_{\mathrm{i\text{-}k+1}} + w\mathbf{B}_{\mathrm{i}}\mathbf{u}_{\mathrm{i\text{-}k}}, \tag{F.22}$$

where

$$\mathbf{A}_{\mathrm{i}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}(a_{\mathrm{p}}\mathbf{H}(t) + \mathbf{K}) & -\mathbf{M}^{-1}\mathbf{C} \end{bmatrix}, \tag{F.23}$$

and

$$\mathbf{B}_{\mathrm{i}} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{M}^{-1}a_{\mathrm{p}}\mathbf{H}(t) & \mathbf{0} \end{bmatrix}, \tag{F.24}$$

with $\mathbf{0}$ and $\mathbf{I} \in \mathbb{R}^{3n_m \times 3n_m}$, $w = \frac{1}{2}$ since the time delay is equal to the time period in milling. Determination of the transition matrix $\mathbf{\Phi}$ and the stability is carried out in a similar way as presented from Eq. F.10 to F.13.

Appendix G
Triple-dexel implementation

The modelling of the workpiece can be carried out using the triple-dexel representation. Triple-dexel modelling is a geometric representation method, which depicts the intersection of a solid with rays cast in three orthogonal directions. Due to its fast Boolean operations, simple data structure, and easy implementation, triple-dexel modelling is highly suitable for real-time graphics-based simulation applications such as numerical control machining verification and virtual sculpting [182]. The method was implemented but intersection module with the end mill is missing.

Figure G.1 presents the dexel concept for one single direction. The single-dexel representation of a solid, also called ray representation, is constructed via a process of computing intersections between the solid and rays cast in one direction. For a given solid, a set of parallel and equidistant rays are projected and intersected with the solid. For each ray the intersected points are stored in the following manner. First, a dexel is defined by two intersection points in a line segment that is completely inside the solid. Then the dexels on a ray are sorted and concatenated into a linked list structure.

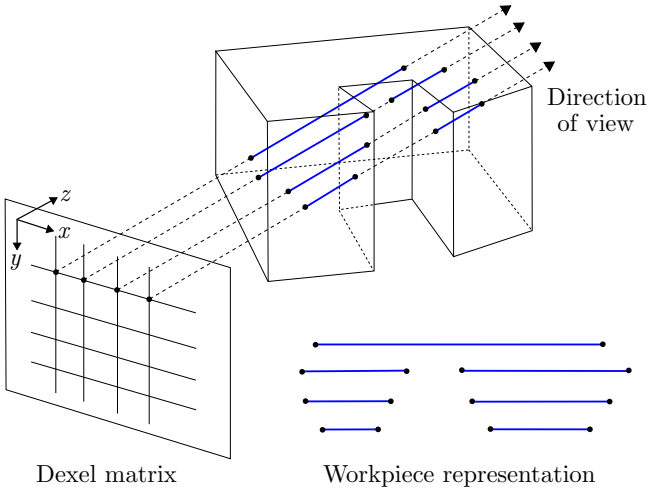
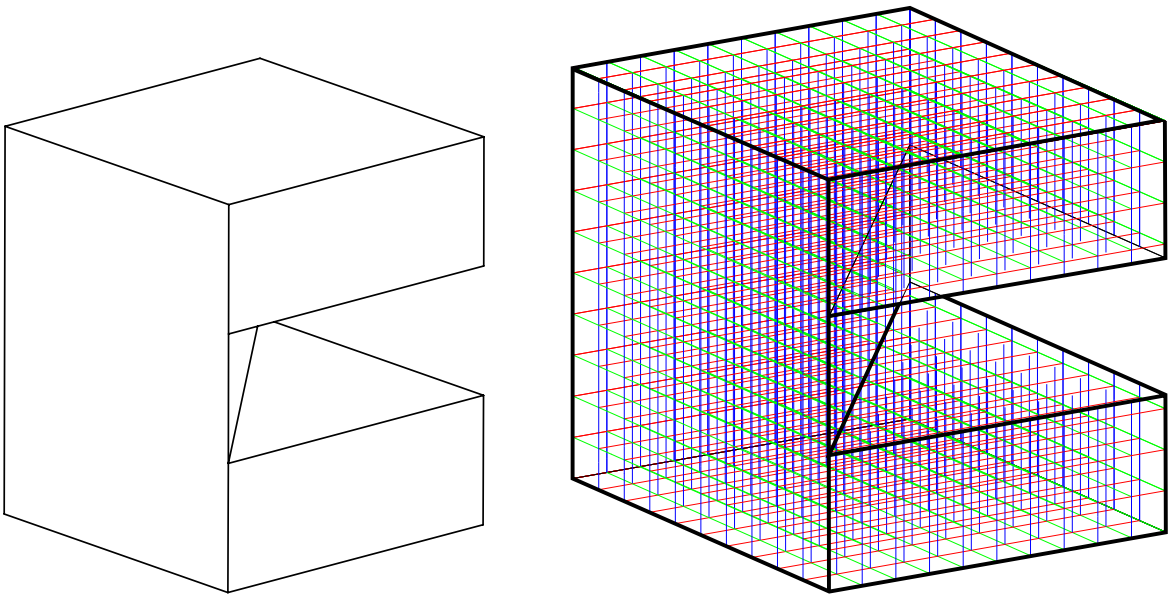


Figure G.1: 1D dexel concept [182]

However, in the single-dexel model, low sampling quality occurs in regions where the surface normals are nearly perpendicular to the ray direction. To address this problem, a triple-dexel model can be constructed by casting rays in three orthogonal directions normally in X, Y, and Z directions to discretise the model. For instance, for a cube with edges of 5 mm in length, 441 dexels are used to represent its geometry if a step of 0.5 mm between each ray is chosen (Figure G.2).



(a) Original part (b) Triple-dexel representation of the part

Figure G.2: Workpiece modelling using triple-dexel method

Appendix H
Base parameter derivation

Base parameters of the Stäubli TX200 robot are computed from the article of Mayeda et al. [162]. This section clarifies the derivation of the base parameters for the considered robot but limited to its first three axes n_θ=3. Hence, only the first three moving links of the robot are accounted for, namely, the robot shoulder, the robot arm and the third link is the assembly of the remaining links n_{Link}=3.

A base parameter set is a minimum collection of inertial parameters whose values can determine the dynamic model of a manipulator uniquely i.e. its equations of motion are defined uniquely. The base parameters are useful for an efficient and accurate identification of the manipulator dynamic model. Mayeda et al. [162] proposed a method to compute the expressions of the base parameters for parallel and perpendicular manipulators with revolute joints. Parallel and perpendicular manipulators refer to robots whose links are either parallel or perpendicular to each other. They describe the base parameters as linear combinations of elementary inertial parameters of the links. Their method is based on the partition of the manipulator in so-called *clusters*. A cluster is composed of a set of links which are parallel to each other. In Figure H.1, k clusters of parallel links are illustrated for an arbitrary manipulator, α_{k} designates the first joint axis of cluster k while β_{k} indicates the last joint axis of cluster k. The value assigned to α_{k} or β_{k} is related to the link number following the joint axis to which they are associated.

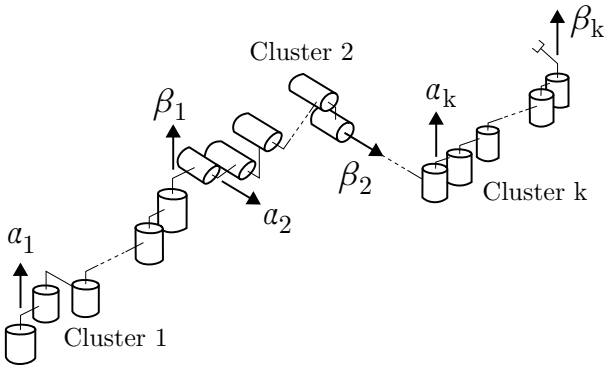


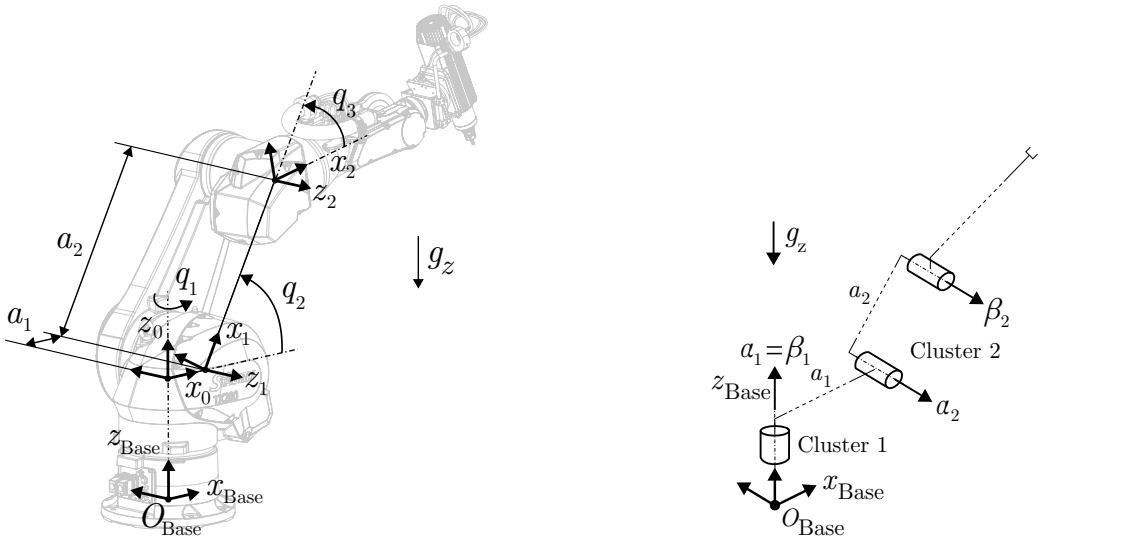
Figure H.1: Link cluster concept

According to Mayeda et al., the minimum number of base parameters n_{p} can be computed as

$$n_{\text{p}} = \begin{cases} 7n_\theta - 4\beta_1 - 2 & : \text{if gravity vector } \mathbf{g}_0 \text{ is parallel to the first joint axis } z_0, \\ 7n_\theta - 4\beta_1 & : \text{otherwise.} \end{cases} \tag{H.1}$$

H.1 Base parameters of the Stäubli TX200 robot

Base parameters of the Stäubli TX200 robot are derived in accordance with the augmented link model presented in Section 5.1.3 (Rigid body identification method). It is worth reminding that the considered inertia tensor $\mathbf{\Phi}_{\mathrm{i}}$ of augmented link i is defined in joint frame $O_{\mathrm{i\text{-}1}}$. Compared with the developed kinematic model of the Stäubli TX200 robot (Section 3.2.1), Mayeda et al. adopted the Craig's convention [115] which reverts the orientation of axes z_1 and z_2 (Figure H.2a). It implies that the positive orientations of q_2 and q_3 are also inverted. In addition, the null reference configuration sets the robot arm horizontally. These considerations affect the computation of the regressor matrix presented in Appendix I. In Figure H.2a, variables a_1 and a_2 still refer to the corresponding standard Denavit-Hartenberg parameters.



(a) Craig's convention (b) Clusters of the Stäubli TX200 robot

Figure H.2: Kinematic model for base parameter derivation

Following Mayeda et al., the Stäubli TX200 robot is partitioned into link clusters (Figure H.2b). Two link clusters are determined since the first joint is parallel to gravity vector g_z while the second and third joints are perpendicular to the former. Vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, containing their associated link number, therefore take the following values: $\boldsymbol{\alpha} = [\alpha_1\ \alpha_2]^T = [1\ 2]^T$ and $\boldsymbol{\beta} = [\beta_1\ \beta_2]^T = [1\ 3]^T$. As a result, the minimum number of base parameters is n_{p}=15. Additionally, vector $\mathbf{c}$ is defined and contains the cluster number associated to each joint. For the Stäubli TX200 robot, vector $\mathbf{c}$ is $\mathbf{c} = [1\ 2\ 2]^T$.

The symbolic expressions of the base parameters are computed in the following way. Vector $\mathbf{a}_{\mathrm{i}}$ is defined for each link i and gives the position of joint frame O_{i} with respect to $O_{\mathrm{i\text{-}1}}$. For the Stäubli TX200 robot, $\mathbf{a}_{\mathrm{i}}$ is expressed such as

$$\begin{aligned} \mathbf{a}_1 &= [a_1\ 0\ 0]^T, \\ \mathbf{a}_2 &= [a_2\ 0\ 0]^T, \\ \mathbf{a}_3 &= [0\ 0\ 0]^T. \end{aligned} \tag{H.2}$$

Following Mayeda et al. notations, they define

$$M_{\mathrm{i}} = \sum_{\mathrm{j=i}}^{n_{\mathrm{Link}}} m_{\mathrm{j}}, \tag{H.3}$$

$$\mathbf{RR}_{\mathrm{i}} = M_{\mathrm{i+1}}\mathbf{a}_{\mathrm{i}} + m_{\mathrm{i}}\mathbf{C}_{\mathrm{i}}, \tag{H.4}$$

$$\mathbf{JJ}_{\mathrm{i}} = \mathbf{\Phi}_{\mathrm{i}} + M_{\mathrm{i+1}}\left[(\mathbf{a}_{\mathrm{i}}\cdot\mathbf{a}_{\mathrm{i}})\mathbf{I} - \mathbf{a}_{\mathrm{i}}\mathbf{a}_{\mathrm{i}}^{T}\right], \tag{H.5}$$

with $\mathbf{I}$ a (3×3) unit matrix.

The following quantities are also defined

$$RZ_{\mathrm{i}} = \begin{cases} 0 & \text{if i} = \beta_{c_{\mathrm{i}}} \\ \sum_{\mathrm{j=i+1}}^{\beta_{c_{\mathrm{i}}}} \mathbf{RR}_{\mathrm{j}}\mathbf{u}_{z} & \text{otherwise,} \end{cases} \tag{H.6}$$

$$RZB_{\mathrm{i}} = \begin{cases} \sum_{\mathrm{j}=\alpha_{c_{\mathrm{i+1}}}}^{\beta_{c_{\mathrm{i+1}}}} \mathbf{RR}_{\mathrm{j}}\mathbf{u}_{z} & \text{if i} = \beta_{c_{\mathrm{i}}} \text{ and } c_{\mathrm{i}} \neq \mathrm{k} \\ 0 & \text{otherwise,} \end{cases} \tag{H.7}$$

$$JYB_{\mathrm{i}} = \begin{cases} \sum_{\mathrm{j}=\alpha_{c_{\mathrm{i+1}}}}^{\beta_{c_{\mathrm{i+1}}}} \left[JJ_{yy_{\mathrm{j}}} + 2a_{\mathrm{j}_z}RZ_{\mathrm{j}}\right] & \text{if i} = \beta_{c_{\mathrm{i}}} \text{ and } c_{\mathrm{i}} \neq \mathrm{k} \\ 0 & \text{otherwise.} \end{cases} \tag{H.8}$$

with $\mathbf{u}_{x|y|z}$ a unit vector along x, y or z e.g. $\mathbf{u}_x$=$[1\ 0\ 0]^T$. All relationships are demonstrated in [162].

Finally, the base parameters are computed using the following relationships in the general case. If the first joint axis z_0 is not parallel to gravity vector $\mathbf{g}_0$, the following inertial parameters constitute a base parameter such as

$$JJ_{zz_{\mathrm{i}}} + JYB_{\mathrm{i}}, \quad RR_{\mathrm{i}}\mathbf{u}_{xx}, \quad RR_{\mathrm{i}}\mathbf{u}_{yy} - RZB_{\mathrm{i}}, \tag{H.9}$$

for $1 \leq \mathrm{i} \leq n_{\mathrm{Link}}$, and

$$\begin{aligned} JJ_{xx_{\mathrm{i}}} - JJ_{yy_{\mathrm{i}}} + JYB_{\mathrm{i}}, \quad & JJ_{xz_{\mathrm{i}}} - a_{\mathrm{i}_x}RZ_{\mathrm{i}}, \\ JJ_{xy_{\mathrm{i}}} + a_{\mathrm{i}_x}RZB_{\mathrm{i}}, \quad & JJ_{yz_{\mathrm{i}}} + a_{\mathrm{i}_z}RZB_{\mathrm{i}}, \end{aligned} \tag{H.10}$$

for $\alpha_2 \leq \mathrm{i} \leq n_{\mathrm{Link}}$. For the case that the first joint axis z_0 is parallel to gravity vector $\mathbf{g}_0$, relationships $RR_{\mathrm{i}}\mathbf{u}_{xx}$ and $RR_{\mathrm{i}}\mathbf{u}_{yy} - RZB_{\mathrm{i}}$ are removed from the set.

The 15 base parameters of the Stäubli TX200 robot stemmed from the application of the method from Mayeda et al. are provided in Table H.1.

π_i	Physical meaning of base parameter i
1	$\phi_{yy_2} + \phi_{yy_3} + \phi_{zz1} + {a_1}^2\,(m_2 + m_3) + {a_2}^2 m_3$
2	$\phi_{zz_2} + m_3\,{a_2}^2$
3	$m_3\,a_2 + C_{x2}\,m_2$
4	$C_{y_2}\,m_2$
5	$\phi_{xx2} - \phi_{yy_2} - m_3\,{a_2}^2$
6	$\phi_{xz2} + C_{z3}\,m_3\,a_2$
7	ϕ_{xy_2}
8	ϕ_{yz_2}
9	ϕ_{zz3}
10	$C_{x3}\,m_3$
11	$C_{y_3}\,m_3$
12	$\phi_{xx3} - \phi_{yy3}$
13	ϕ_{xz3}
14	ϕ_{xy_3}
15	ϕ_{yz_3}

Table H.1: Expressions of the base parameters from Mayeda et al. for the Stäubli TX200 robot limited to its first three joints

The obtained expressions are somewhat different from those showcased in Table 5.6. As a matter of fact, the parameters obtained by the method of Mayeda et al. do not comprise the effects of the motors. By choosing, base parameters related to the motors as

$$\begin{aligned} \pi_{16} &= \phi_{\mathrm{m},zz_3}, \\ \pi_{17} &= \phi_{\mathrm{m},xx_2} - \phi_{\mathrm{m},yy_2}, \\ \pi_{18} &= \phi_{\mathrm{m},xx_3} - \phi_{\mathrm{m},yy_3}, \end{aligned} \tag{H.11}$$

it is possible to include their effect while sustaining the rank efficiency of the associated regressor matrix $\mathbf{Y}$. Since the consideration of motors modifies the inertia characteristics of the system, and therefore the equations of motion, base parameters found in Table H.1 are revised accordingly.

Appendix I

Regressor matrix detail

I.1 Regressor matrix of the Stäubli TX200 robot

The general form of the equations of motion for a manipulator is expressed as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_{\mathrm{v}}\dot{\mathbf{q}} + \mathbf{F}_{\mathrm{s}}\mathrm{sgn}(\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \mathbf{\Gamma}. \tag{I.1}$$

The expression of the equations of motion can be symbolically derived using expressions of the mass $\mathbf{M}(\mathbf{q})$, damping $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ and gravity $\mathbf{g}(\mathbf{q})$ matrices from Eqs. 3.37 to 3.40. Recalling that the equations of motion can be written in a linear form with

$$\mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\boldsymbol{\pi} = \mathbf{\Gamma}, \tag{I.2}$$

the expression of the regressor matrix $\mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$ is obtained using the selected base parameters $\boldsymbol{\pi}$. In Subsection 5.1.3, the expressions of the base parameters are provided in Table 5.6 for the Stäubli TX200 robot limited to its first three joints. Equations I.2 can therefore be developed such as

$$\begin{bmatrix} Y_{1,1} & Y_{1,2} & \cdots & Y_{1,27} \\ Y_{2,1} & Y_{2,2} & \cdots & Y_{2,27} \\ Y_{3,1} & Y_{3,2} & \cdots & Y_{3,27} \end{bmatrix} \begin{bmatrix} \pi_1 \\ \vdots \\ \pi_{27} \end{bmatrix} = \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \\ \Gamma_3 \end{bmatrix}. \tag{I.3}$$

Afterwards, each element of matrix $\mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$ is found by partial derivation of the symbolic expression of total torque Γ_{i} with respect to all base parameters π_{j} such as

$$Y_{\mathrm{i,j}} = \frac{\partial \Gamma_{\mathrm{i}}}{\partial \pi_{\mathrm{j}}}. \tag{I.4}$$

Hereafter, all elements of regressor matrix $\mathbf{Y}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})$ for the Stäubli TX200 robot limited to its first three joints are provided below with k_{i} the reduction ratio of joint i. Such regressor matrix is useful since the structure of the considered manipulator is commonly found in most industrial robots. Indeed, the first joint is often vertical while joints 2 and 3 are horizontal.

$$Y_{1,1} = \ddot{q}_1 \tag{I.5}$$

$$Y_{1,2} = Y_{1,9} = 0 \tag{I.6}$$

$$Y_{1,3} = 2\,a_1\ (\ddot{q}_1\ \cos(q_2) - \dot{q}_1\,\dot{q}_2\ \sin(q_2)) \tag{I.7}$$

$$Y_{1,4} = -2\,a_1\ (\ddot{q}_1\ \sin(q_2) + \dot{q}_1\,\dot{q}_2\ \cos(q_2)) \tag{I.8}$$

$$Y_{1,5} = \frac{\ddot{q}_1}{2} - \frac{\ddot{q}_1\ \cos(2\,q_2)}{2} + \dot{q}_1\,\dot{q}_2\ \sin(2\,q_2) \tag{I.9}$$

$$Y_{1,6} = -\cos(q_2)\ \dot{q}_2^2 - \ddot{q}_2\ \sin(q_2) \tag{I.10}$$

$$Y_{1,7} = -\ddot{q}_1\ \sin(2\,q_2) - 2\,\dot{q}_1\,\dot{q}_2\ \cos(2\,q_2) \tag{I.11}$$

$$Y_{1,8} = \dot{q}_2^2\ \sin(q_2) - \ddot{q}_2\ \cos(q_2) \tag{I.12}$$

$$\begin{aligned} Y_{1,10} =& a_2\,\ddot{q}_1\ \cos(q_3) + a_2\,\ddot{q}_1\ \cos(2\,q_2 + q_3) + 2\,a_1\,\ddot{q}_1\ \cos(q_2 + q_3) \\ & - 2\,a_1\,\dot{q}_1\,\dot{q}_2\ \sin(q_2 + q_3) - 2\,a_1\,\dot{q}_1\,\dot{q}_3\ \sin(q_2 + q_3) - a_2\,\dot{q}_1\,\dot{q}_3\ \sin(q_3) \\ & - 2\,a_2\,\dot{q}_1\,\dot{q}_2\ \sin(2\,q_2 + q_3) - a_2\,\dot{q}_1\,\dot{q}_3\ \sin(2\,q_2 + q_3) \end{aligned} \tag{I.13}$$

$$\begin{aligned} Y_{1,11} =& -a_2\,\ddot{q}_1\ \sin(q_3) - a_2\,\ddot{q}_1\ \sin(2\,q_2 + q_3) - 2\,a_1\,\ddot{q}_1\ \sin(q_2 + q_3) \\ & - 2\,a_1\,\dot{q}_1\,\dot{q}_2\ \cos(q_2 + q_3) - 2\,a_1\,\dot{q}_1\,\dot{q}_3\ \cos(q_2 + q_3) - a_2\,\dot{q}_1\,\dot{q}_3\ \cos(q_3) \\ & - 2\,a_2\,\dot{q}_1\,\dot{q}_2\ \cos(2\,q_2 + q_3) - a_2\,\dot{q}_1\,\dot{q}_3\ \cos(2\,q_2 + q_3) \end{aligned} \tag{I.14}$$

$$\begin{aligned} Y_{1,12} =& \frac{\ddot{q}_1}{2} - \frac{\ddot{q}_1\ \cos(2\,q_2 + 2\,q_3)}{2} + \dot{q}_1\,\dot{q}_2\ \sin(2\,q_2 + 2\,q_3) \\ & + \dot{q}_1\,\dot{q}_3\ \sin(2\,q_2 + 2\,q_3) \end{aligned} \tag{I.15}$$

$$\begin{aligned} Y_{1,13} =& -\cos(q_2 + q_3)\ \dot{q}_2^2 - 2\ \cos(q_2 + q_3)\ \dot{q}_2\,\dot{q}_3 \\ & - \cos(q_2 + q_3)\ \dot{q}_3^2 - \ddot{q}_2\ \sin(q_2 + q_3) - \ddot{q}_3\ \sin(q_2 + q_3) \end{aligned} \tag{I.16}$$

$$\begin{aligned} Y_{1,14} =& -\ddot{q}_1\ \sin(2\,q_2 + 2\,q_3) - 2\,\dot{q}_1\,\dot{q}_2\ \cos(2\,q_2 + 2\,q_3) \\ & - 2\,\dot{q}_1\,\dot{q}_3\ \cos(2\,q_2 + 2\,q_3) \end{aligned} \tag{I.17}$$

$$\begin{aligned} Y_{1,15} =& \sin(q_2+q_3)\ \dot{q}_2^2 + 2\ \sin(q_2+q_3)\ \dot{q}_2\,\dot{q}_3 + \sin(q_2+q_3)\ \dot{q}_3^2 \\ & - \ddot{q}_2\ \cos(q_2+q_3) - \ddot{q}_3\ \cos(q_2+q_3) \end{aligned} \tag{I.18}$$

$$Y_{1,16} = 0 \tag{I.19}$$

$$Y_{1,17} = \frac{\ddot{q}_1}{2} - \frac{\ddot{q}_1\ \cos(2\,k_2\,q_2)}{2} + k_2\,\dot{q}_1\,\dot{q}_2\ \sin(2\,k_2\,q_2) \tag{I.20}$$

$$\begin{aligned} Y_{1,18} =& \frac{\ddot{q}_1}{2} - \frac{\ddot{q}_1\ \cos(2\,q_2+2\,k_3\,q_3)}{2} + \dot{q}_1\,\dot{q}_2\ \sin(2\,q_2+2\,k_3\,q_3) \\ & + k_3\,\dot{q}_1\,\dot{q}_3\ \sin(2\,q_2+2\,k_3\,q_3) \end{aligned} \tag{I.21}$$

$$Y_{1,19} = Y_{1,20} = Y_{1,21} = Y_{1,24} = Y_{1,25} = Y_{1,26} = Y_{1,27} = 0 \tag{I.22}$$

$$Y_{1,22} = \dot{q}_1 \tag{I.23}$$

$$Y_{1,23} = \mathrm{sgn}(\dot{q}_1) \tag{I.24}$$

$$Y_{2,1} = 0 \tag{I.25}$$

$$Y_{2,2} = \ddot{q}_2 \tag{I.26}$$

$$Y_{2,3} = a_1\,\dot{q}_1^2\ \sin(q_2) - g_z\cos(q_2) \tag{I.27}$$

$$Y_{2,4} = a_1\ \cos(q_2)\ \dot{q}_1^2 + g_z\sin(q_2) \tag{I.28}$$

$$Y_{2,5} = -\frac{\dot{q}_1^2\ \sin(2\,q_2)}{2} \tag{I.29}$$

$$Y_{2,6} = -\ddot{q}_1\ \sin(q_2) \tag{I.30}$$

$$Y_{2,7} = \dot{q}_1^2\ \cos(2\,q_2) \tag{I.31}$$

$$Y_{2,8} = -\ddot{q}_1\ \cos(q_2) \tag{I.32}$$

$$Y_{2,9} = \ddot{q}_2 + \ddot{q}_3 \tag{I.33}$$

$$\begin{aligned} Y_{2,10} =& 2\,a_2\,\ddot{q}_2\,\cos\left(q_3\right) - g_z\cos\left(q_2+q_3\right) + a_2\,\ddot{q}_3\,\cos\left(q_3\right) + a_1\,\dot{q}_1^2\,\sin\left(q_2+q_3\right) \\ & - a_2\,\dot{q}_3^2\,\sin\left(q_3\right) + a_2\,\dot{q}_1^2\,\sin\left(2\,q_2+q_3\right) - 2\,a_2\,\dot{q}_2\,\dot{q}_3\,\sin\left(q_3\right) \end{aligned} \tag{I.34}$$

$$\begin{aligned} Y_{2,11} =& g_z\sin\left(q_2+q_3\right) - 2\,a_2\,\ddot{q}_2\,\sin\left(q_3\right) - a_2\,\ddot{q}_3\,\sin\left(q_3\right) + a_1\,\dot{q}_1^2\,\cos\left(q_2+q_3\right) \\ & - a_2\,\dot{q}_3^2\,\cos\left(q_3\right) + a_2\,\dot{q}_1^2\,\cos\left(2\,q_2+q_3\right) - 2\,a_2\,\dot{q}_2\,\dot{q}_3\,\cos\left(q_3\right) \end{aligned} \tag{I.35}$$

$$Y_{2,12} = -\frac{\dot{q}_1^2\,\sin\left(2\,q_2+2\,q_3\right)}{2} \tag{I.36}$$

$$Y_{2,13} = -\ddot{q}_1\,\sin\left(q_2+q_3\right) \tag{I.37}$$

$$Y_{2,14} = \dot{q}_1^2\,\cos\left(2\,q_2+2\,q_3\right) \tag{I.38}$$

$$Y_{2,15} = -\ddot{q}_1\,\cos\left(q_2+q_3\right) \tag{I.39}$$

$$Y_{2,16} = \ddot{q}_2 + k_3\,\ddot{q}_3 \tag{I.40}$$

$$Y_{2,17} = -\frac{k_2\,\dot{q}_1^2\,\sin\left(2\,k_2\,q_2\right)}{2} \tag{I.41}$$

$$Y_{2,18} = -\frac{\dot{q}_1^2\,\sin\left(2\,q_2+2\,k_3\,q_3\right)}{2} \tag{I.42}$$

$$Y_{2,19} = \cos(q_2) \tag{I.43}$$

$$Y_{2,20} = \cos(2q_2) \tag{I.44}$$

$$Y_{2,21} = \cos(3q_2) \tag{I.45}$$

$$Y_{2,22} = Y_{2,23} = Y_{2,26} = Y_{2,27} = 0 \tag{I.46}$$

$$Y_{2,24} = \dot{q}_2 \tag{I.47}$$

$$Y_{2,25} = \mathrm{sgn}(\dot{q}_2) \tag{I.48}$$

$$Y_{3,1} = Y_{3,2} = Y_{3,3} = Y_{3,4} = Y_{3,5} = Y_{3,6} = Y_{3,7} = Y_{3,8} = 0 \tag{I.49}$$

$$Y_{3,9} = \ddot{q}_2 + \ddot{q}_3 \tag{I.50}$$

$$\begin{aligned} Y_{3,10} =& a_2\,\ddot{q}_2\,\cos(q_3) - g_z\cos(q_2+q_3) + a_1\,\dot{q}_1^2\,\sin(q_2+q_3) + \frac{a_2\,\dot{q}_1^2\,\sin(q_3)}{2} \\ &+ a_2\,\dot{q}_2^2\,\sin(q_3) + \frac{a_2\,\dot{q}_1^2\,\sin(2\,q_2+q_3)}{2} \end{aligned} \tag{I.51}$$

$$\begin{aligned} Y_{3,11} =& g_z\sin(q_2+q_3) - a_2\,\ddot{q}_2\,\sin(q_3) + a_1\,\dot{q}_1^2\,\cos(q_2+q_3) + \frac{a_2\,\dot{q}_1^2\,\cos(q_3)}{2} \\ &+ a_2\,\dot{q}_2^2\,\cos(q_3) + \frac{a_2\,\dot{q}_1^2\,\cos(2\,q_2+q_3)}{2} \end{aligned} \tag{I.52}$$

$$Y_{3,12} = -\frac{\dot{q}_1^2\,\sin(2\,q_2+2\,q_3)}{2} \tag{I.53}$$

$$Y_{3,13} = -\ddot{q}_1\,\sin(q_2+q_3) \tag{I.54}$$

$$Y_{3,14} = \dot{q}_1^2\,\cos(2\,q_2+2\,q_3) \tag{I.55}$$

$$Y_{3,15} = -\ddot{q}_1\,\cos(q_2+q_3) \tag{I.56}$$

$$Y_{3,16} = k_3\,(\ddot{q}_2 + k_3\,\ddot{q}_3) \tag{I.57}$$

$$Y_{3,17} = 0 \tag{I.58}$$

$$Y_{3,18} = -\frac{k_3\,\dot{q}_1^2\,\sin(2\,q_2+2\,k_3\,q_3)}{2} \tag{I.59}$$

$$Y_{3,19} = Y_{3,20} = Y_{3,21} = Y_{3,22} = Y_{3,23} = Y_{3,24} = Y_{3,25} = 0 \tag{I.60}$$

$$Y_{3,26} = \dot{q}_3 \tag{I.61}$$

$$Y_{3,27} = \mathrm{sgn}(\dot{q}_3) \tag{I.62}$$

Appendix J

Identification of the KUKA KR90 robot

Thanks to a scientific internship at the University of Victoria in Canada, the rigid body identification method was applied to the determination of the base parameters of the KUKA KR90 R3100 robotic arm (Figure J.1). KUKA KR90 robot is a six-axis robot with a similar architecture compared with the Stäubli TX200 robot. It also possesses a gravity compensator system between the first and second moving links. However, the mechanism is not composed of a spring enclosed in the robot arm but uses a pneumatic system. The nominal payload of the KUKA KR90 robot is 90 kg for a maximum reach of 3095 mm. Using the estimated inertial parameters, a fitting of the joint elastic parameters is also carried out.

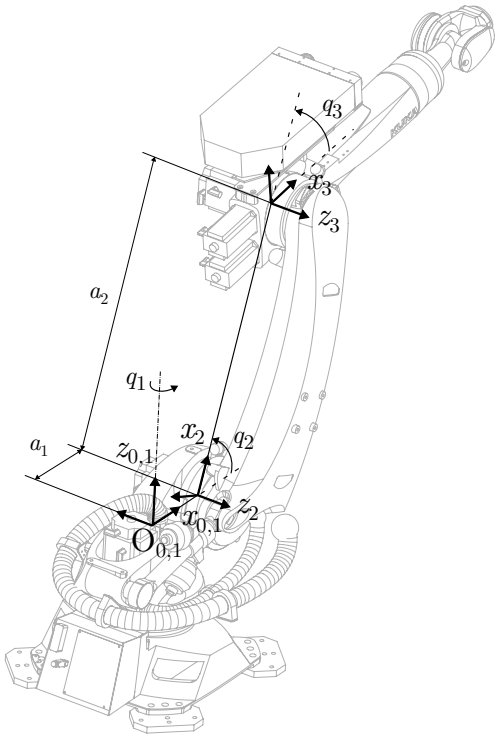
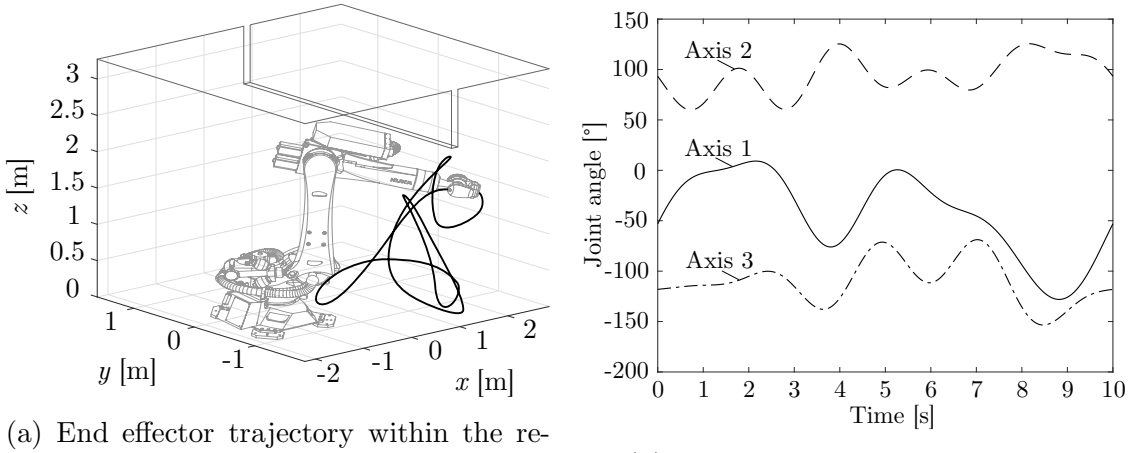


Figure J.1: Kinematic model of the KUKA KR90 R3100 HA robot using Craig's convention [183]

J.1 Base parameters of the KUKA KR90 robot

The identification of the base parameters of the KUKA KR90 robot is also limited to its first three joints. As for the Stäubli TX200 robot, the same base parameters along with the gravity and friction coefficients presented in Table 5.6 are identified using the same procedure presented in Subsection 5.1.3. For the KUKA KR90 robot, the length

of the first two moving links are $a_1 = 0.35$ m and $a_2 = 1.35$ m. As for the Stäubli TX200 robot, optimum excitation trajectories are determined by solving the optimisation problem defined in Eqs. 5.13 and 5.14. An example of excitation trajectory generated for the KUKA KR90 robot is presented in Figure J.2.



(a) End effector trajectory within the restrained workspace

(b) Corresponding joint trajectories

Figure J.2: Example of excitation trajectory for the KUKA KR90 robot

Five harmonic terms N_f=5 are also used in the Fourier series expansion defining the trajectory of each joint in Eqs. 5.10 to 5.12. The fundamental frequency of the excitation trajectory $\frac{\omega_F}{2\pi}$ was also set to 0.1 Hz. The joint constraints restricting their displacement, velocity and acceleration are shown in Table J.1.

Axis	q_{min} [rad]	q_{max} [rad]	$\dot{q}_{min}$ [rad/s]	$\dot{q}_{max}$ [rad/s]	$\ddot{q}_{min}$ [rad/s^2]	$\ddot{q}_{max}$ [rad/s^2]
1	-3.23	3.23	-1.83	1.83	-8	8
2	0.09	2.44	-1.76	1.76	-8	8
3	-2.7	2.1	-1.87	1.87	-8	8

Table J.1: Joint angular, velocity and acceleration limits for the KUKA KR90 robot

Workspace limits **S** for the KUKA KR90 robot are presented in Table J.2 in relation to the coordinate system displayed in Figure J.2a. Note that the workspace represents a smaller volume than the one available for the Stäubli TX200 robot which was a smaller manipulator than the KUKA KR90 robot.

x_{min} [m]	x_{max} [m]	y_{min} [m]	y_{max} [m]	z_{min} [m]	z_{max} [m]
-1.5	2.6	-1.6	1.4	0.5	2.5

Table J.2: Workspace limits **S** for the KUKA KR90 robot

Considering the identification of the parameters of the KUKA KR90 robot, only six different trajectories were generated by starting from various initial values for the optimisation parameter $\boldsymbol{\delta}$. In the trajectory optimisation problem, observation matrix **A** was built

using regressor matrix $\mathbf{Y} \in \mathbb{R}^{(3\times15)}$. Figure J.2b depicts the trajectory of each joint for one of the excitation trajectories. The corresponding end effector trajectory is shown in the Cartesian space in Figure J.2a.

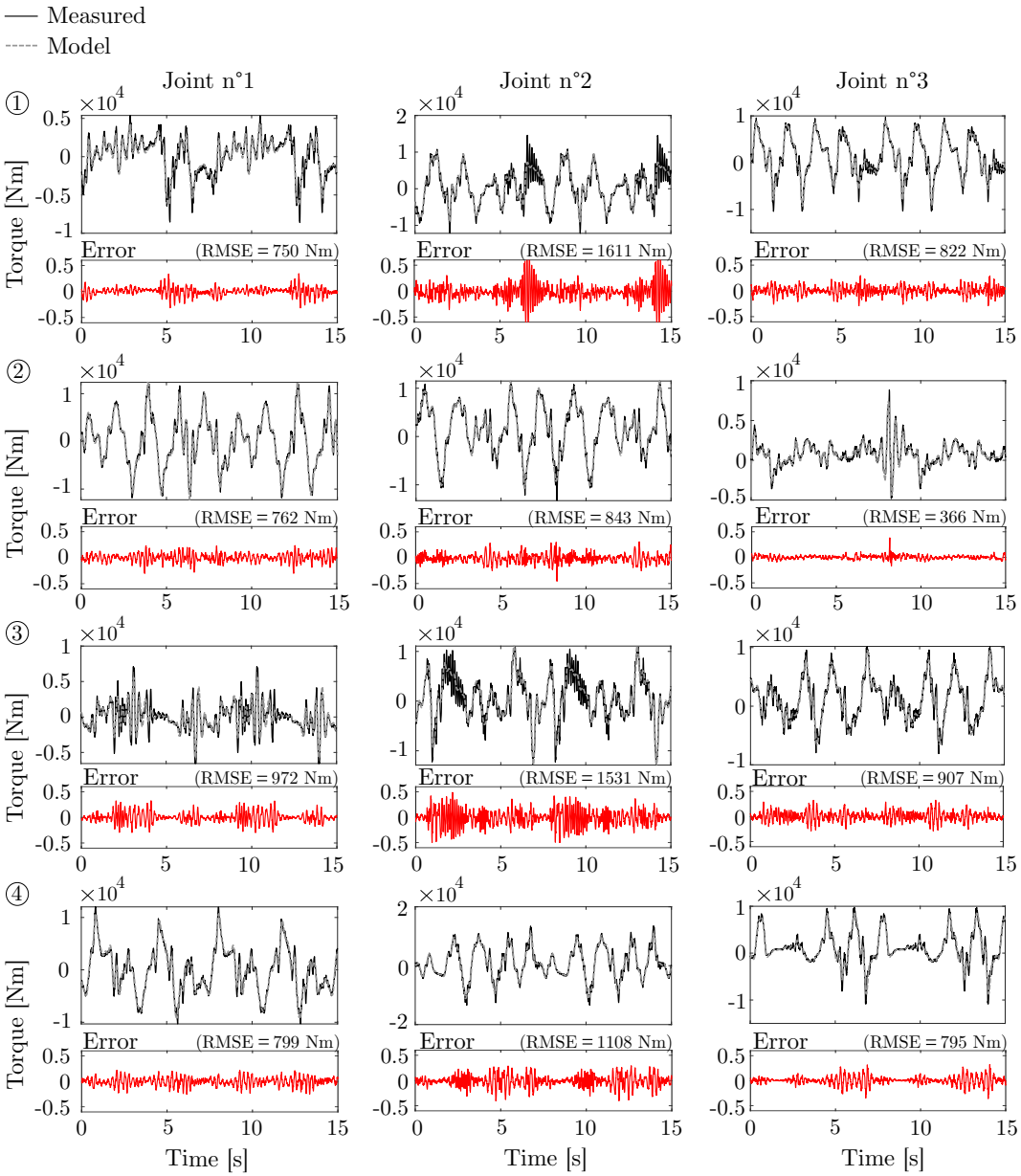


Figure J.3: Fitting of the measured torques over four excitation trajectories generated with $\mathbf{Y}^{(3\times15)}$ for the KUKA KR90 robot

The KUKA KR90 robot was programmed using the `point-to-point spline` command in order to produce smooth motions. Using the `Trace` feature of the KUKA teach pendant, motor torques and encoder positions were recorded using a sampling rate of 250 Hz on a Personal Computer, and the joint velocity and acceleration signals were obtained by numerical differentiation, with the same low-pass filter, of the encoder data using the same FFT filtering method as for the Stäubli TX200 robot. For the KUKA KR90 robot, reduction ratios of the first three joints are as follows: k_1=256.87, k_2=267.43 and k_3=252.33. The 18 base parameters along with the friction and gravity compensation

coefficients were thus estimated by solving the LSE problem for the KUKA KR90 robot. The measured and estimated torque signals corresponding to the four trajectories are shown in Figure J.3, as well as their error in Nm. The root mean square error has an average value of 939 Nm. Torque errors seem to reach higher values because higher torques are needed to actuate the robot. Nevertheless, measured torque signals are less noisy than the ones retrieved on the Stäubli TX200 robot, probably because of a different parametrisation of the internal low-pass filter.

As for the Stäubli TX200 robot, the fitting of the measured torques with the estimated ones indicates a good identification of the parameters of the KUKA robot model with rigid joints. The 27 identified parameters $\boldsymbol{\pi}$ along with the joint position measured during the last two trajectories are also used to predict the joint torque signals. Again, the predicted and measured torque signals shown in Figure J.4 are in close agreement. The root mean square error has an average value of 888 Nm.

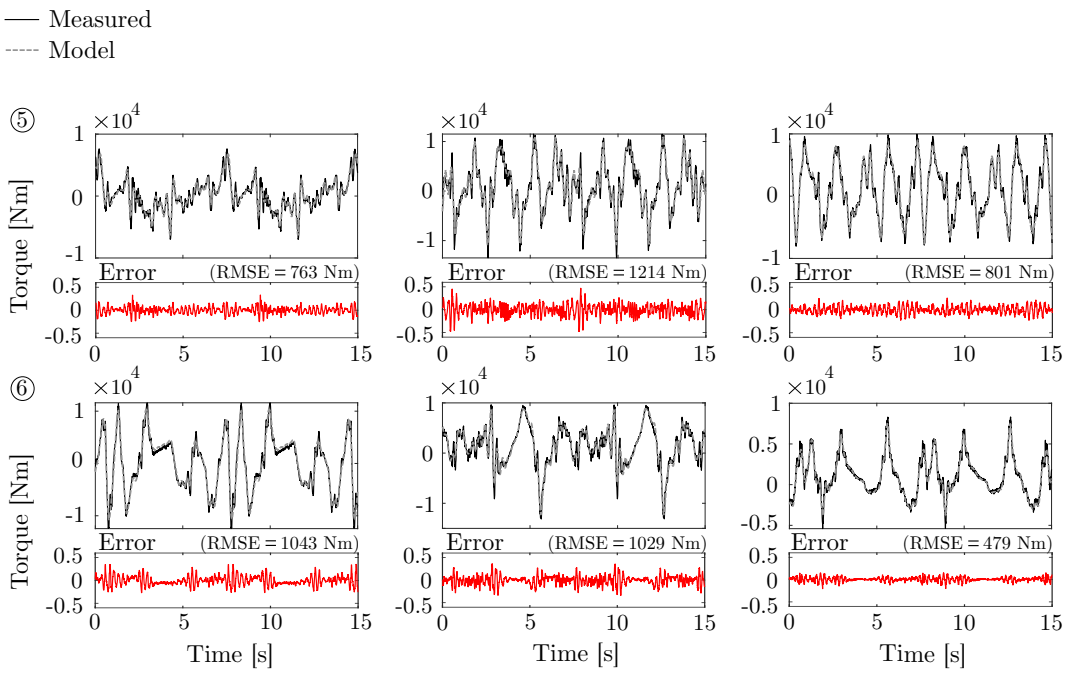


Figure J.4: Torque prediction over two other measured trajectories generated with $\mathbf{Y}^{(3\times15)}$ for the KUKA KR90 robot

The estimated base parameters along with the gravity compensation and friction coefficients are presented in Table J.3. Since KUKA did not provide the actual elementary inertial parameters of the robot, it is unfortunately not possible to derive an error estimate. Nevertheless, in view of the good results obtained with the Stäubli TX200 robot by applying the exact same method, it suggests an accurate identification of the parameters of the KUKA KR90 robot.

π_{i}	Physical meaning of parameter i	Value
1	$\phi_{\mathrm{m},yy_2} + \phi_{\mathrm{m},yy_3} + \phi_{yy_2} + \phi_{yy_3} + \phi_{zz_1} + {a_1}^2\,(m_2 + m_3) + m_3\,{a_2}^2 + m_{\mathrm{m}2}\,{a_1}^2 + m_{\mathrm{m}3}\,{a_1}^2 + m_{\mathrm{m}3}\,{a_2}^2 + \phi_{\mathrm{m},zz_1}\,{k_1}^2$	1897.8
2	$\phi_{zz_2} + m_3\,{a_2}^2 + m_{\mathrm{m}3}\,{a_2}^2 + \phi_{\mathrm{m},zz_2}\,{k_2}^2$	1466.5
3	$m_3\,a_2 + m_{\mathrm{m}3}\,a_2 + C_{x2}\,m_2$	678.7
4	$C_{y_2}\,m_2$	48.3
5	$\phi_{xx_2} - \phi_{yy_2} - m_3\,{a_2}^2 - m_{\mathrm{m}3}\,{a_2}^2$	-662.2
6	$\phi_{xz_2} + C_{z3}\,m_3\,a_2$	34.4
7	ϕ_{xy_2}	91.9
8	ϕ_{yz_2}	-1.0
9	ϕ_{zz_3}	162.1
10	$C_{x3}\,m_3$	129.1
11	$C_{y_3}\,m_3$	-2.7
12	$\phi_{xx_3} - \phi_{yy_3}$	-155.7
13	ϕ_{xz_3}	1.4
14	ϕ_{xy_3}	3.1
15	ϕ_{yz_3}	-1.0
16	ϕ_{m,zz_3}	0.0113
17	$\phi_{\mathrm{m},xx_2} - \phi_{\mathrm{m},yy_2}$	0.0147
18	$\phi_{\mathrm{m},xx_3} - \phi_{\mathrm{m},yy_3}$	0.0112
19	B_1	-7938.1
20	B_2	-72.8
21	B_3	257.1
22	F_{v_1}	1387.1
23	F_{s_1}	308.2
24	F_{v_2}	1337.2
25	F_{s_2}	374.4
26	F_{v_3}	933.6
27	F_{s_3}	227.1

Table J.3: Base parameters of the Kuka KR90 R3100 robot and gravity compensation and friction coefficients

J.2 Identification of joint elastic parameters

Having identified the base parameters of the KUKA KR90 robot, one torsional spring and one torsional viscous damper are appended to each of its first three joints. Following the same model updating described in Subsection 5.3.2 for the Stäubli TX200 robot, the method is applied to the KUKA KR90 robot for the identification of the joint elastic parameters. A milling pose is also chosen to derive them such as q_1=0°, q_2=70° and q_3=-100° expressed in the Craig's convention. From the experimental modal analysis tests, four modes are measured below 30 Hz:

1. Mode one exhibits a deflection around the motion axis of the first joint at 10 Hz with a damping ratio of 1.4 %.

2. Mode two presents a deflection around the motion axis of the second joint at 11 Hz with a damping ratio of 1.0 %.

3. Mode three showcases a deflection perpendicular to the motion axis of the first joint at 19.2 Hz with a damping ratio of 0.5 % and is therefore not reproduced by the robot model with one flexible element around each motion axis.

4. Mode four shows a deflection around the motion axis of the third joint at 23.7 Hz with a damping ratio of 0.8 %.

Identified joint parameters are presented in Table J.4 in terms of joint stiffness and joint damping.

	Joint 1	Joint 2	Joint 3
Joint stiffness [Nm/rad]	3.12e6	3.75e6	2.98e6
Joint damping [Nm.s/rad]	1.45e3	1.10e3	0.29e3

Table J.4: Identified joint stiffness and damping parameters with the joint axial flexibility model for the KUKA KR90 robot

As shown in Figure J.5 comparing the measured and simulated FRFs at the end effector of the robot, all the modes involving a rotation around a motion axis (1, 2 and 4) are correctly fitted. It means that depending on the robot geometry and identified base parameters, the model updating of such robot model can lead to satisfactory results for some of the modes or all the modes. Of course, the third mode involving a deflection perpendicular to the first joint is not predicted since the joint axial flexibility model only allows deflections around motion axes.

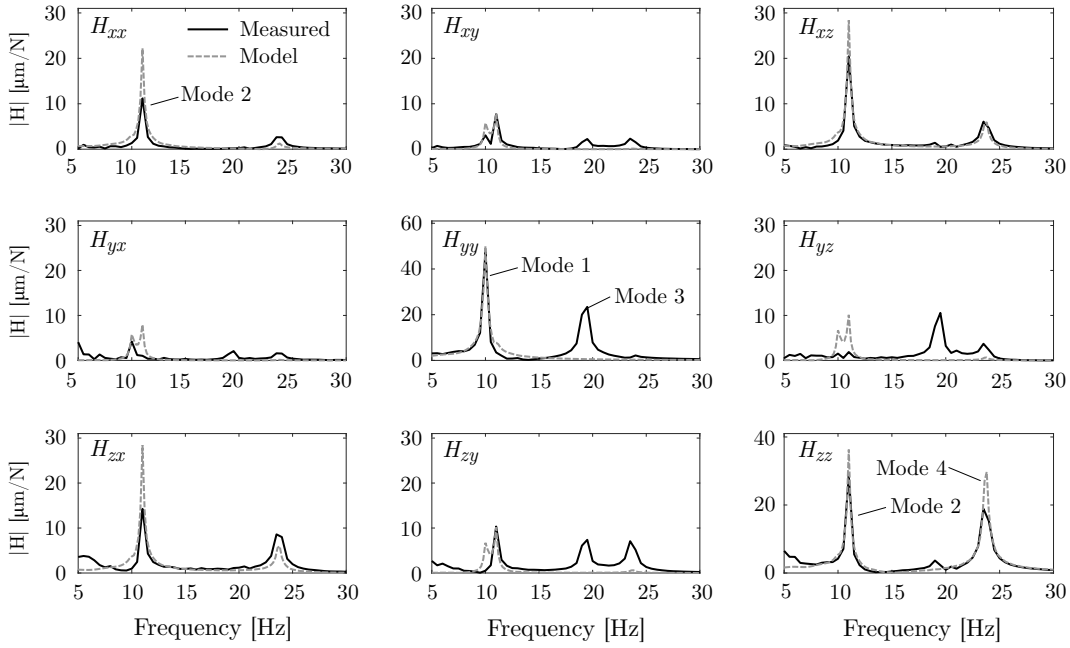


Figure J.5: Amplitude of measured and fitted frequency response function matrix $\mathbf{H}(\omega)$ at the end-effector obtained with the joint axial flexibility model for the milling posture of the KUKA KR90 robot

The robot model identified in the milling posture of the KUKA robot was used to predict the modal parameters in two other postures with a satisfactory accuracy.

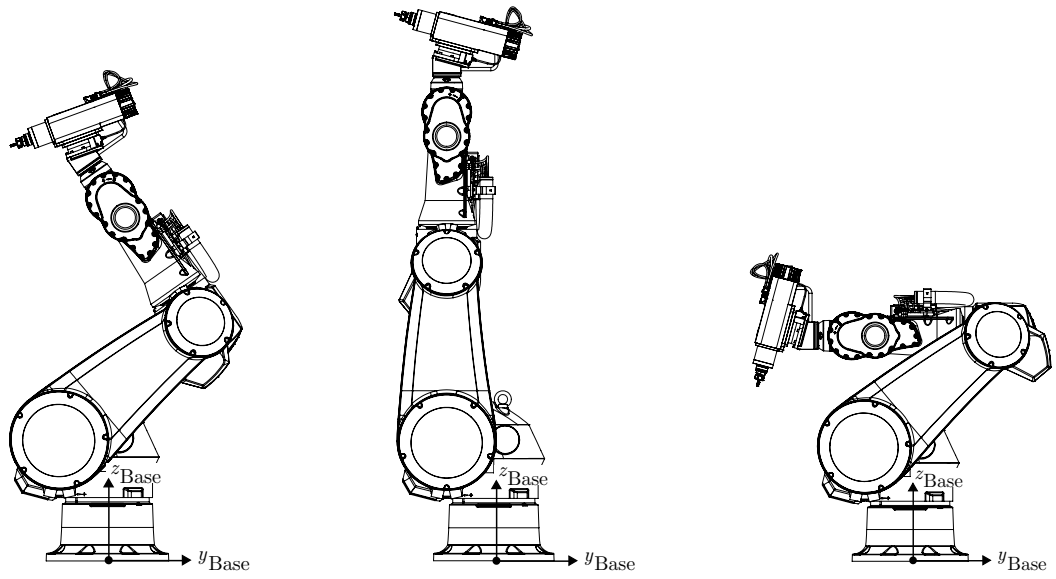
Appendix K

EMA in three uncommon postures

Three uncommon robot poses are examined through complete experimental modal analyses. The controller is in action and FRFs are measured over a bandwidth of 200 Hz. The three uncommon poses are later named the *upright* pose, the *vertic*al pose and the *rigid* pose (Figure K.1). Corresponding joint positions are provided in Table K.1.

	q_1 [°]	q_2 [°]	q_3 [°]	q_4 [°]	q_5 [°]	q_6 [°]
Upright pose	-90	-50	95	0	0	0
Vertical pose	-90	0	0	0	0	0
Rigid pose	-90	-50	140	0	0	0

Table K.1: Joint positions of the upright, vertical and rigid poses in the encoder reference



(a) Upright pose (b) Vertical pose (c) Rigid pose

Figure K.1: EMA in three uncommon postures of the Stäubli TX200 robot with a straight wrist

K.1 EMA results in uncommon postures

Experimental modal analysis results in the three uncommon postures are presented through the mode shape animations generated by LMS Test.Lab and the auto-MAC matrices. FRFs and resulting stabilisation diagrams are not reported.

Upright posture

In the upright posture, only five modes are detected below 35 Hz instead of six for all other analysed postures in this work. It is remarked that the first mode now exhibits a downward motion of the TCP resulting from the rotation of the second joint around its axis of motion ($q_{z,2}$) at 10.7 Hz as shown in Figure K.2. The mode involving a rotation of the first joint around its axis of motion ($q_{z,1}$) is now the second mode at 12 Hz. It is therefore clearly witnessed that a change in the robot configuration can modify the order of appearance of the robot modes. Although their frequencies and damping ratios vary, shapes of modes 3, 4 and 5 remain unchanged compared to the EMA results obtained in the milling posture presented in Chapter 5. Hence, modes 4 and 5 share high correlation values (78 %) in the auto-MAC matrix exposed in Figure K.5a.

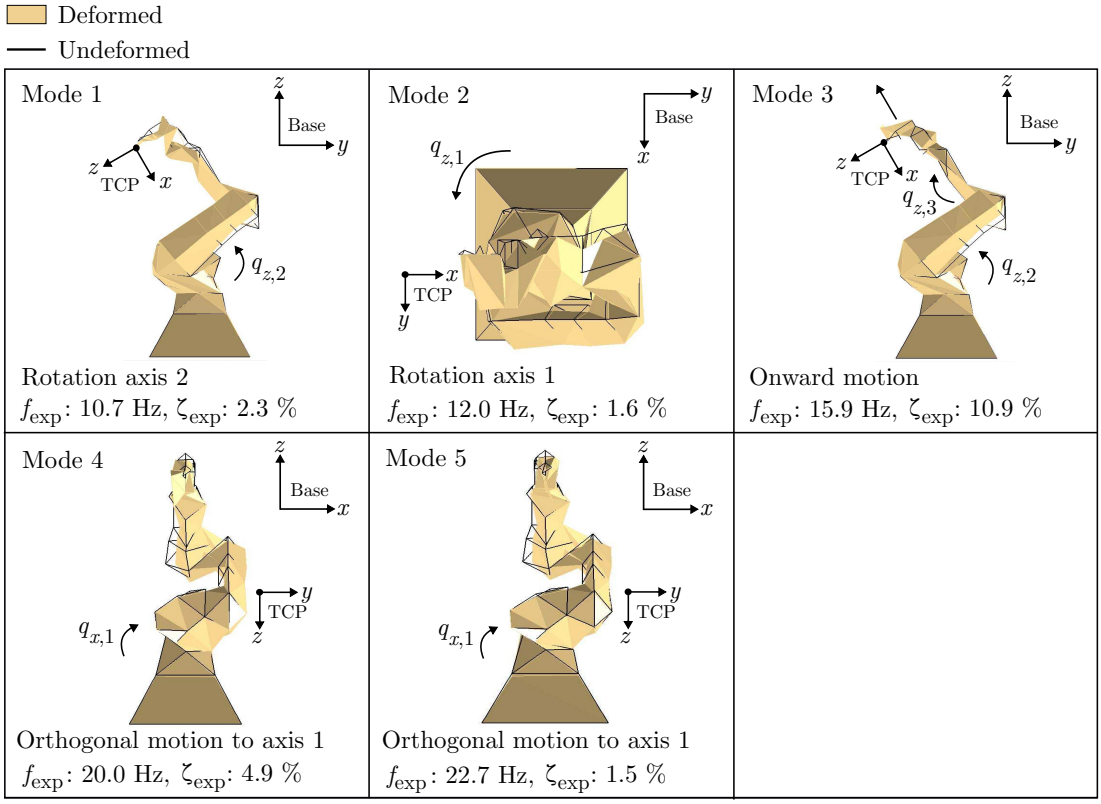


Figure K.2: Mode shapes in the controlled upright posture of the Stäubli TX200 robot

Vertical posture

When the robot is completely outstretched vertically, six modes are detected below 35 Hz. First mode also involves a rotation of the second joint around its motion axis ($q_{z,2}$) at 5.7 Hz as shown in Figure K.3. The motion of the second mode is solely generated by a deflection perpendicular to the motion axis of the first joint ($q_{x,1}$) at 9.2 Hz. Main component of the motion is therefore captured along y_{TCP}. For the third mode, the natural frequency of the onward motion elevates to 20.6 Hz. Then, the interpretation of shapes of modes 4, 5 and 6 is less obvious. It seems that mode four is created by the combination of perpendicular motions of the first and third joints. Motion of mode 3 is also generated

by the same joint combination which might explain the high correlation value of modes 3 and 4 in the auto-MAC matrix shown in Figure K.5b. In mode five, the robot base is animated by a vertical motion along z_{Base} and the wrist moves perpendicularly to the motion axis of the fifth joint ($q_{x,5}$) at 30.2 Hz. Mode six only shows a motion of the robot wrist originating from a rotation of the fifth joint around its motion axis ($q_{z,5}$).

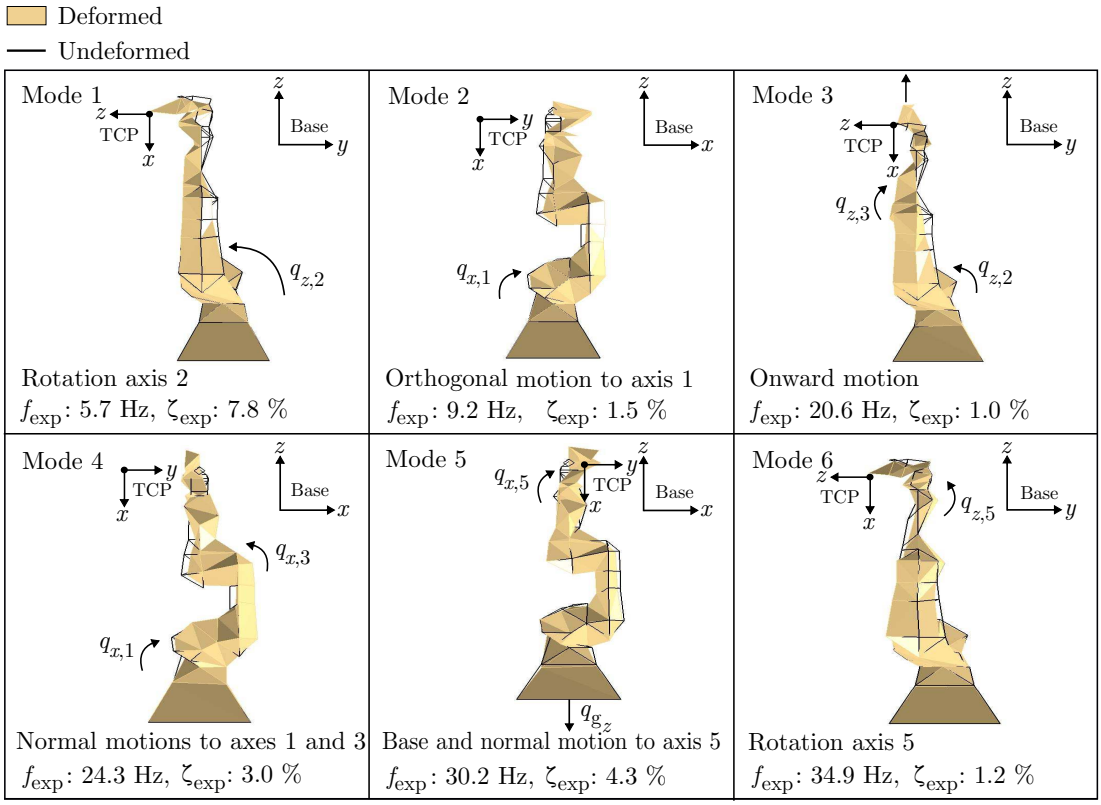


Figure K.3: Mode shapes in the controlled vertical posture of the Stäubli TX200 robot

Rigid posture

The rigid posture in named by the fact that the arm and forearm are folded near the robot base which increases the stiffness of the structure. Mode one exhibits a rotation of the first joint around its motion axis ($q_{z,1}$) at 12.4 Hz as seen in Figure K.4. The first natural frequency is indeed higher than the first one in the milling posture with the same mode shape at 9.4 Hz. In fact, mode shapes in the rigid posture are the same as the ones identified for the milling posture. Therefore, modes four and five again show high correlation values (83 %) in the auto-MAC matrix shown in Figure K.5c.

K.2 Evaluation of the simulated posture dependency

The Stäubli and UMons robot multibody models are used to estimate their ability to predict the mode shapes, frequencies and damping ratios in uncommon configurations. Both models comprise the tri-axial flexibility representation for each of the joints. The elastic parameters of both models were identified through the proposed identification method and fitted values of joint stiffness and damping were reported in Section 5.3.

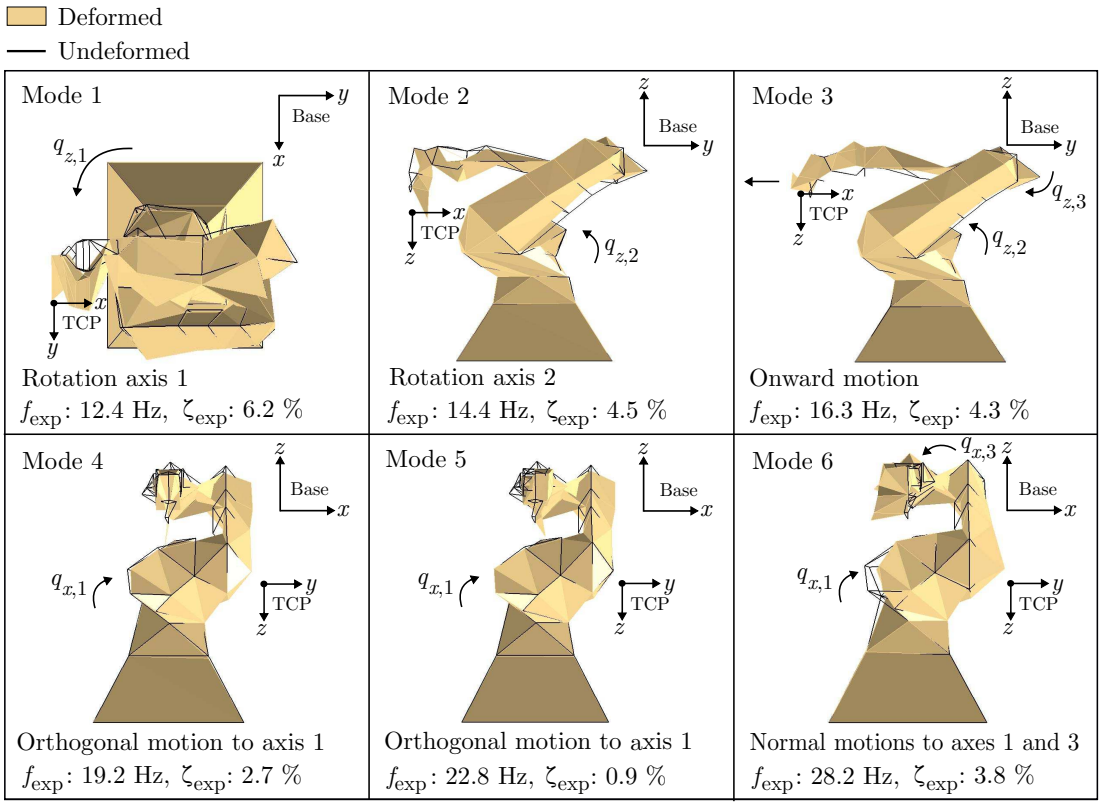


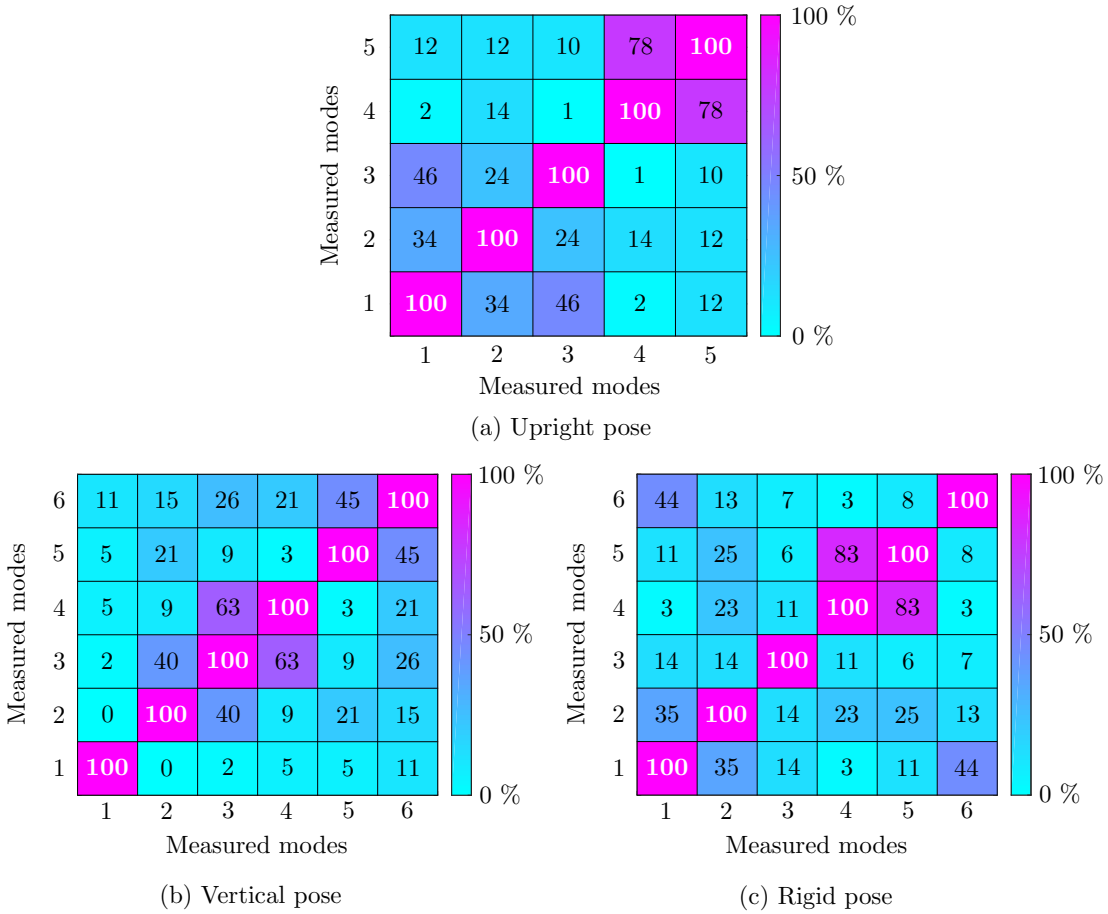
Figure K.4: Mode shapes in the controlled rigid posture of the Stäubli TX200 robot

Note that elastic parameters are only identified in the milling posture and then retained to predict the mode shapes, frequencies and damping ratios in other postures.

	Upright		Vertical		Rigid	
Mode	f_n [Hz]	ζ_n [%]	f_n [Hz]	ζ_n [%]	f_n [Hz]	ζ_n [%]
1	11.3 (10.7)	1.4 (2.3)	9.2 (5.7)	1.2 (7.8)	13.3 (12.4)	2.1 (6.2)
2	12.4 (12.0)	0.4 (1.6)	11.0 (9.2)	0.5 (1.5)	13.9 (14.4)	2.0 (4.5)
3	19.5 (15.9)	5.4 (10.9)	21.2 (20.6)	1.0 (1.0)	17.5 (16.3)	3.2 (4.3)
4	20.6 (20.0)	5.7 (4.9)	22.5 (24.3)	5.4 (3.0)	20.7 (19.2)	2.8 (2.7)
5	21.2 (22.7)	1.5(1.5)	24.5 (30.2)	4.1 (4.3)	21.5 (22.8)	2.8 (0.9)
6	29.4 (/)	2.2 (/)	31.0 (34.9)	3.9 (1.2)	29.1 (28.2)	1.9 (3.8)
$\text{Error}_{\text{Average}}$ [%]	7.5	120.0	17.7	159.9	6.1	85.2

Table K.2: Prediction of the natural frequencies and damping ratios in the upright, vertical and rigid postures using the joint tri-axial flexibility model with the manufacturer's data for the link inertia

Simulated frequencies and damping ratios are reported for the first six modes in the three uncommon postures in Table K.2, only for the Stäubli model. In terms of frequency prediction, largest average error (17.7 %) is observed for the vertical posture. It is much more accurate than the prediction of the damping ratios for which the largest average error is 159.9 % in the vertical posture.



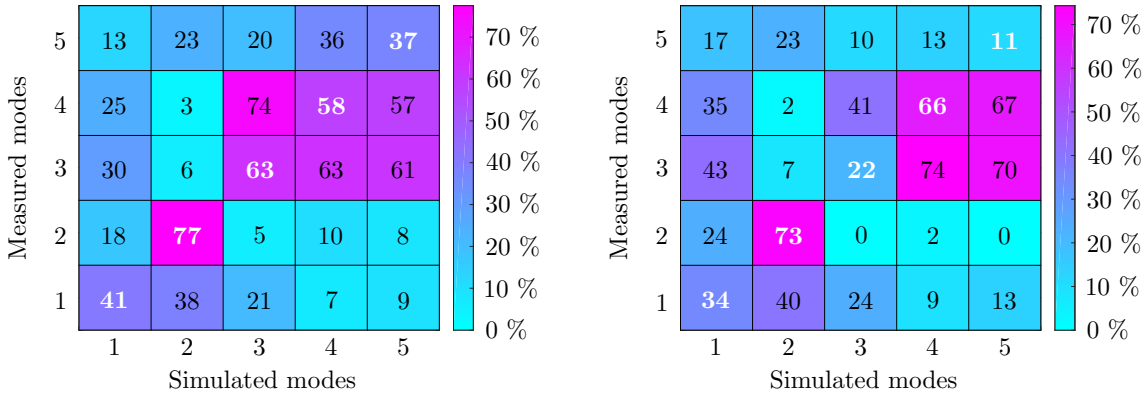
(a) Upright pose

(b) Vertical pose

(c) Rigid pose

Figure K.5: Auto-MAC matrices of the measured modes of the Stäubli TX200 robot in three other postures

In terms of mode shapes, MAC matrices comparing the measured and simulated mode shapes are computed for the Stäubli and the UMons models. Overall, correlation values are clearly worse than the ones reported in Section 5.3 for the milling pose with straight wrist and the square and flexible poses. Only some modes are correctly reproduced for the considered extreme poses. Note that similar mode shapes are obtained from both models in the considered postures but at different frequencies and damping ratios.

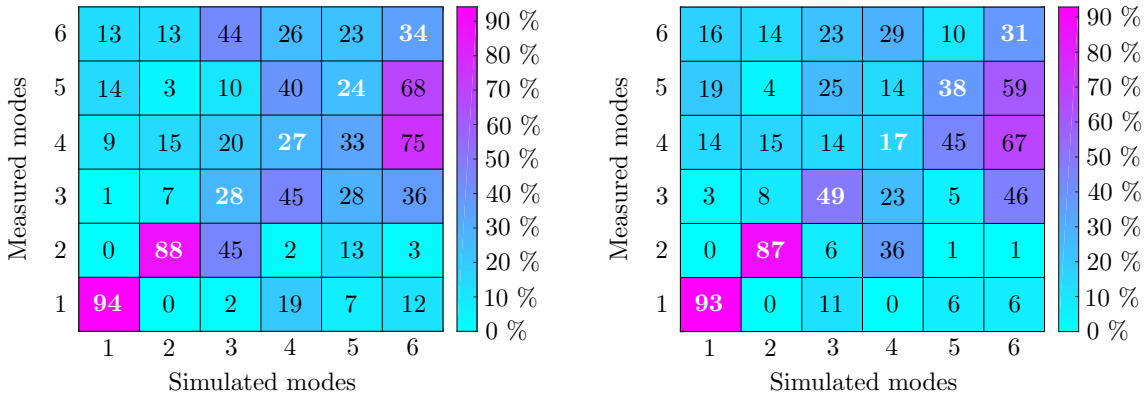


(a) Stäubli model

(b) UMons model

Figure K.6: MAC matrices in the upright posture using the first 5 modes

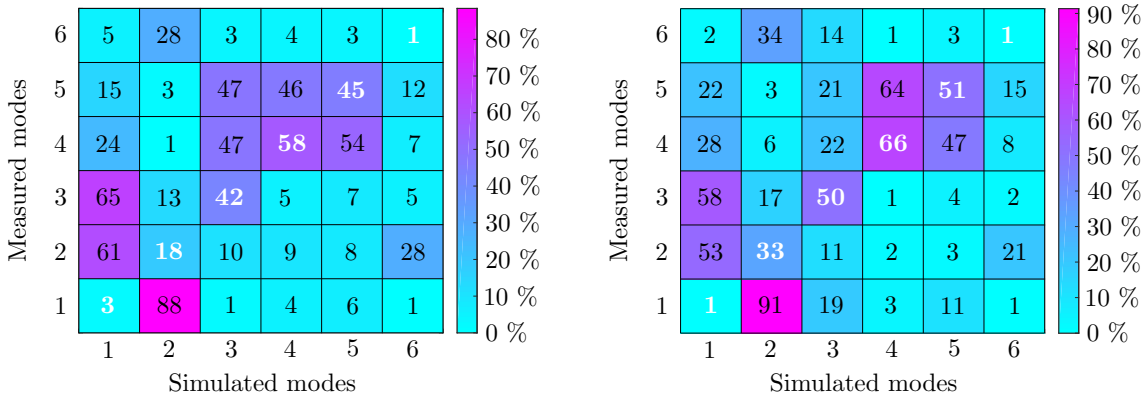
The MAC matrices comparing mode shapes in the upright posture are presented in Figure K.6. Even if the animation of the simulated mode shape of the first mode is comparable to the one provided by the LMS Test.Lab software, a correlation value less than 50 % is obtained. Second mode shape presents the highest correlation value which is above 70 %. Then, mode three is quite well approximated with the Stäubli model with 63 % but it is not the case for the UMons model. An opposite trend is observed for mode four. The last simulated mode shape has the lowest correlation value.



(a) Stäubli model (b) UMons model

Figure K.7: MAC matrices in the vertical posture using the first 6 modes

In the MAC matrices related to the vertical posture, the first two mode shapes are very well approximated with both models with correlation values over 80 % (Figure K.7). For higher modes, correlation values drastically drop under 50 %. Even with the animations of the simulated mode shapes, motion is still difficult to interpret for modes 3, 4, 5 and 6 in the vertical posture.



(a) Stäubli model (b) UMons model

Figure K.8: MAC matrices in the rigid posture using the first 6 modes

The MAC matrices of the rigid posture are eventually shown in Figure K.8. It appears that simulated shapes of modes 1 and 2 are reversed compared with the measured mode shapes (high correlation values for cells (1,2) and (2,1) in Figure K.8). Then, mode 3, represented with an onward robot motion, and modes 4 and 5, exhibiting the

perpendicular deflections to the motion axis of the first joint, are reproduced in the simulated animations but correlation values in the MAC matrices are only around 50 %. The last mode is not well reproduced.

Overall, the Stäubli and UMons models roughly produce the same trends in the results. Some modes can be captured in the presented extreme postures. Shown results suggest a finer modelling of the joint to better handle the posture dependency of the mode shapes. Nevertheless, as presented in Section 5.3, the proposed joint model with three rotational elastic elements per joint might be sufficient for large milling workspace if the robot keeps the configuration in which it was identified.

Appendix L
Jerk trajectory implementation

A symmetric jerk trajectory is selected to execute the milling trajectory and it is entirely defined by four quantities: $\mathrm{v_d}$, x_i, x_f and t_Total. Time duration of the blending motion is computed such as

$$|t_\mathrm{b}| = \left| \frac{(x_\mathrm{i} - x_\mathrm{f} + \mathrm{v_d} t_\mathrm{Total})}{\mathrm{v_d}} \right|. \tag{L.1}$$

Prior the determination of duration t_b, desired velocity must be ensured not too large ($|\mathrm{v_d}| > \frac{2|x_\mathrm{f}-x_\mathrm{i}|}{t_\mathrm{Total}}$) nor not too small ($|\mathrm{v_d}| < \frac{|x_\mathrm{f}-x_\mathrm{i}|}{t_\mathrm{Total}}$). The jerk trajectory is described by the following general equations

$$\begin{aligned}
j(t) =& j_{0,\mathrm{Phase_p}} \\
a(t) =& a_{0,\mathrm{Phase_p}} + j_{0,\mathrm{Phase_p}} t_\mathrm{Phase}(t) \\
v(t) =& v_{0,\mathrm{Phase_p}} + a_{0,\mathrm{Phase_p}} t_\mathrm{Phase}(t) + \frac{1}{2} j_{0,\mathrm{Phase_p}} t_\mathrm{Phase}(t)^2 \\
x(t) =& x_{0,\mathrm{Phase_p}} + v_{0,\mathrm{Phase_p}} t_\mathrm{Phase}(t) + \frac{1}{2} a_{0,\mathrm{Phase_p}} t_\mathrm{Phase}(t)^2 + \frac{1}{6} j_{0,\mathrm{Phase_p}} t_\mathrm{Phase}(t)^3,
\end{aligned} \tag{L.2}$$

with $j(t)$ the jerk, $a(t)$ the acceleration, $v(t)$ the velocity and $x(t)$ the position of the TCP at current time t. The quantities either represent a motion along x_Base (transversal pass) or y_Base (longitudinal pass). For each of the five phases, jerk ($j_{0,\mathrm{Phase_p}}$), acceleration ($a_{0,\mathrm{Phase_p}}$), velocity ($v_{0,\mathrm{Phase_p}}$) and position ($x_{0,\mathrm{Phase_p}}$) constants must be defined. Time $t_\mathrm{Phase}(t)$ is the current time of phase p. Subscript p refers to the index of the phase varying from 1 to 5. The constants are defined phase by phase below.

Phase 1: time interval $[t_1 - t_2]$

In order to select jerk constant $j_{0,\mathrm{Phase_1}}$, it is decided to impose that half of desired velocity $\mathrm{v_d}$ must be reached at instant $\frac{t_\mathrm{b}}{2}$. Since the jerk trajectory is chosen symmetric, absolute value of $j_{0,\mathrm{Phase_1}}$ is retained during the whole jerk trajectory. As a result, constants in phase 1 [t_1-t_2], which is the first part of the acceleration segment, are expressed such as

$$
\begin{aligned}
j_{0,\mathrm{Phase}_1} =& \frac{\frac{1}{2}\mathrm{v_d}}{\frac{1}{2}\left(\frac{t_\mathrm{b}}{2}\right)^2} \\
a_{0,\mathrm{Phase}_1} =& 0 \\
v_{0,\mathrm{Phase}_1} =& 0 \\
x_{0,\mathrm{Phase}_1} =& x_\mathrm{i}.
\end{aligned}
\tag{L.3}
$$

In Eq. L.2, current time of phase 1 is set to $t_{\mathrm{Phase}}(t)=t-t_1$, being known that current time $t > t_1$.

Phase 2: time interval $[t_2 - t_3]$

The constants of the second phase are computed from the values of jerk, acceleration, velocity and position attained at the end of the first phase. Therefore, in $t=\frac{t_\mathrm{b}}{2}$ i.e. at the beginning of phase 2 which is the second part of the acceleration segment, the constants read

$$
\begin{aligned}
j_{0,\mathrm{Phase}_2} =& -j_{0,\mathrm{Phase}_1} \\
a_{0,\mathrm{Phase}_2} =& j_{0,\mathrm{Phase}_1}\frac{t_\mathrm{b}}{2} \\
v_{0,\mathrm{Phase}_2} =& \frac{1}{2}j_{0,\mathrm{Phase}_1}\left(\frac{t_\mathrm{b}}{2}\right)^2 \\
x_{0,\mathrm{Phase}_2} =& x_{0,\mathrm{Phase}_1} + \frac{1}{6}j_{0,\mathrm{Phase}_1}\left(\frac{t_\mathrm{b}}{2}\right)^3.
\end{aligned}
\tag{L.4}
$$

In Eq. L.2, current time of phase 2 is set to $t_{\mathrm{Phase}}(t)=t-\frac{t_\mathrm{b}}{2}-t_1$.

Phase 3: time interval $[t_3 - t_4]$

The third phase is the time interval in which the velocity of the TCP is maintained constant to perform the milling operation. Therefore, both the jerk $j(t)$ and the acceleration $a(t)$ are null. The desired velocity is kept constant using the following constants

$$
\begin{aligned}
j_{0,\mathrm{Phase}_3} =& 0 \\
a_{0,\mathrm{Phase}_3} =& 0 \\
v_{0,\mathrm{Phase}_3} =& \mathrm{v_d} \\
x_{0,\mathrm{Phase}_3} =& x_{0,\mathrm{Phase}_2} + v_{0,\mathrm{Phase}_2}\frac{t_\mathrm{b}}{2} + \frac{1}{2}a_{0,\mathrm{Phase}_2}\left(\frac{t_\mathrm{b}}{2}\right)^2 + \frac{1}{6}j_{0,\mathrm{Phase}_2}\left(\frac{t_\mathrm{b}}{2}\right)^3.
\end{aligned}
\tag{L.5}
$$

In Eq. L.2, current time of phase 3 is set to $t_{\mathrm{Phase}}(t)=t-t_\mathrm{b}-t_1$.

Phase 4: time interval $[t_4 - t_5]$

The deceleration segment starts in the fourth phase and its constants are expressed as follows

$$\begin{aligned} j_{0,\mathrm{Phase}_4} &= -\, j_{0,\mathrm{Phase}_1} \\ a_{0,\mathrm{Phase}_4} &= 0 \\ v_{0,\mathrm{Phase}_4} &= \mathrm{v_d} \\ x_{0,\mathrm{Phase}_4} &= x_{0,\mathrm{Phase}_3} + \mathrm{v_d}\left(t_{\mathrm{Total}} - 2t_{\mathrm{b}}\right). \end{aligned} \tag{L.6}$$

In Eq. L.2, current time of phase 4 is set to $t_{\mathrm{Phase}}(t)=t-(t_{\mathrm{Total}}-t_{\mathrm{b}})-t_1$.

Phase 5: time interval $[t_5 - t_6]$

Second part of the deceleration segment starts in phase five. Jerk, acceleration, velocity and position constants depend on the values attained during the previous interval and are computed as follows

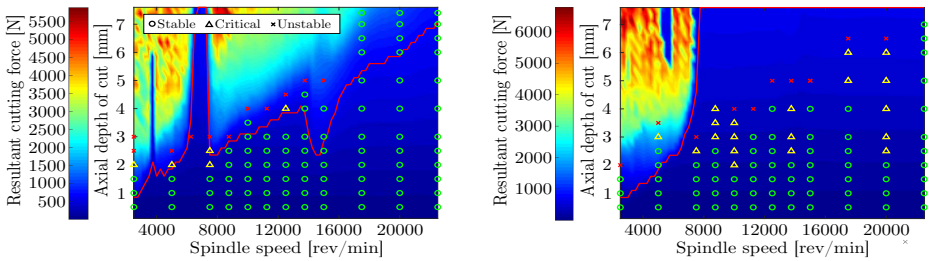
$$\begin{aligned} j_{0,\mathrm{Phase}_5} &= j_{0,\mathrm{Phase}_1} \\ a_{0,\mathrm{Phase}_5} &= -\, j_{0,\mathrm{Phase}_1}\frac{t_{\mathrm{b}}}{2} \\ v_{0,\mathrm{Phase}_5} &= \frac{1}{2} j_{0,\mathrm{Phase}_1}\left(\frac{t_{\mathrm{b}}}{2}\right)^2 \\ x_{0,\mathrm{Phase}_5} &= x_{0,\mathrm{Phase}_4} + v_{0,\mathrm{Phase}_4}\frac{t_{\mathrm{b}}}{2} + \frac{1}{2} a_{0,\mathrm{Phase}_4}\left(\frac{t_{\mathrm{b}}}{2}\right)^2 + \frac{1}{6} j_{0,\mathrm{Phase}_4}\left(\frac{t_{\mathrm{b}}}{2}\right)^3. \end{aligned} \tag{L.7}$$

In Eq. L.2, current time of phase 5 is set to $t_{\mathrm{Phase}}(t)=t-(t_{\mathrm{Total}}-\frac{t_{\mathrm{b}}}{2})-t_1$.

Appendix M
Stability lobe diagram detail

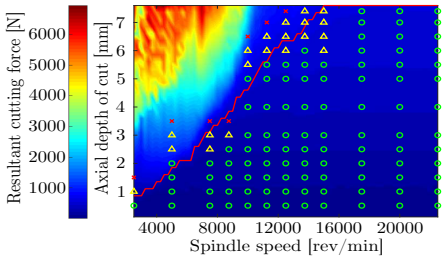
This appendix gathers all the individual stability lobe diagrams obtained from all the implemented methods for conventional machine tool and the various versions of the multibody model of the Stäubli TX200 robot, depending on the assessed flexibility.

With joint and controller flexibilities

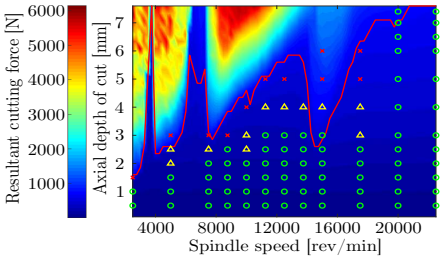


(a) Longitudinal down milling (aluminium)

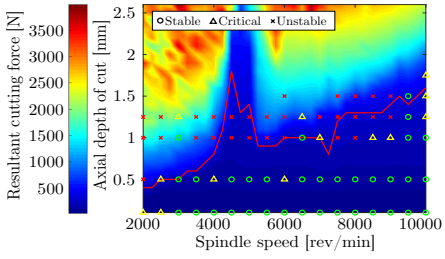
(b) Longitudinal up milling (aluminium)



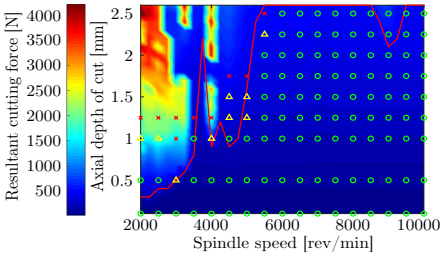
(c) Transversal down milling (aluminium)



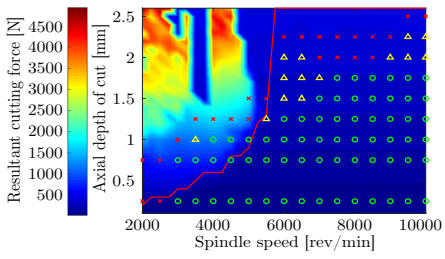
(d) Transversal up milling (aluminium)



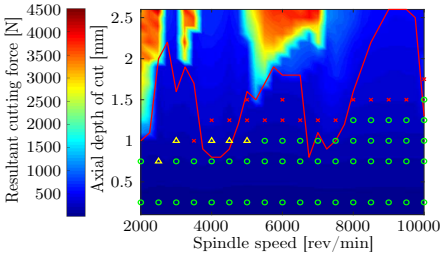
(e) Longitudinal down milling (steel)



(f) Longitudinal up milling (steel)



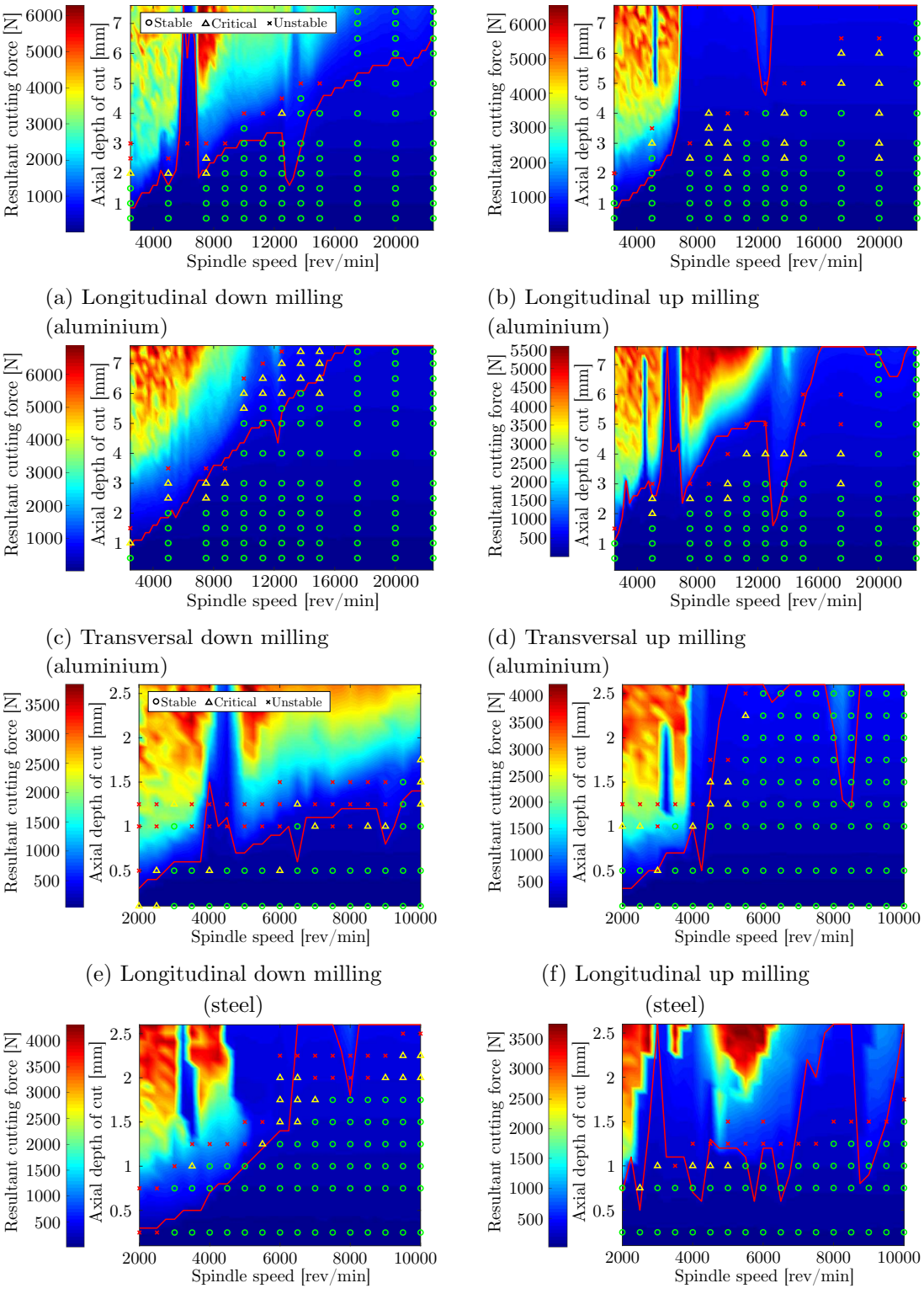
(g) Transversal down milling (steel)



(h) Transversal up milling (steel)

Figure M.1: Experimental and simulated stability charts using the Stäubli TX200 model with the joint and controller flexibilities with a radial depth of cut a_e= 4 mm and standard cutting coefficients

With joint and link flexibilities



(a) Longitudinal down milling (aluminium)

(b) Longitudinal up milling (aluminium)

(c) Transversal down milling (aluminium)

(d) Transversal up milling (aluminium)

(e) Longitudinal down milling (steel)

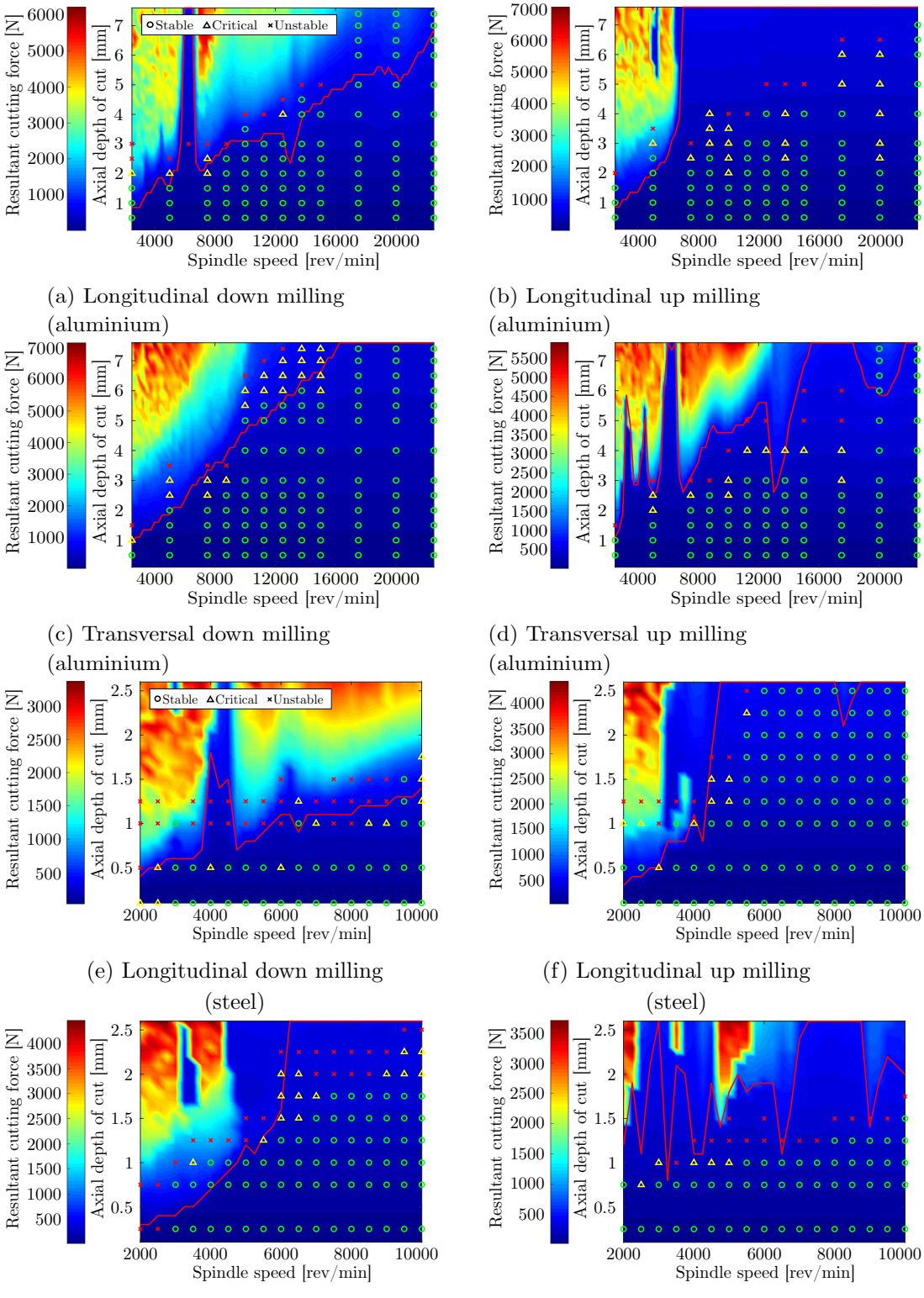
(f) Longitudinal up milling (steel)

(g) Transversal down milling (steel)

(h) Transversal up milling (steel)

Figure M.2: Experimental and simulated stability charts using the Stäubli TX200 model with the joint and link flexibilities with a radial depth of cut a_e= 4 mm and standard cutting coefficients

With joint, link and controller flexibilities



(a) Longitudinal down milling (aluminium)

(b) Longitudinal up milling (aluminium)

(c) Transversal down milling (aluminium)

(d) Transversal up milling (aluminium)

(e) Longitudinal down milling (steel)

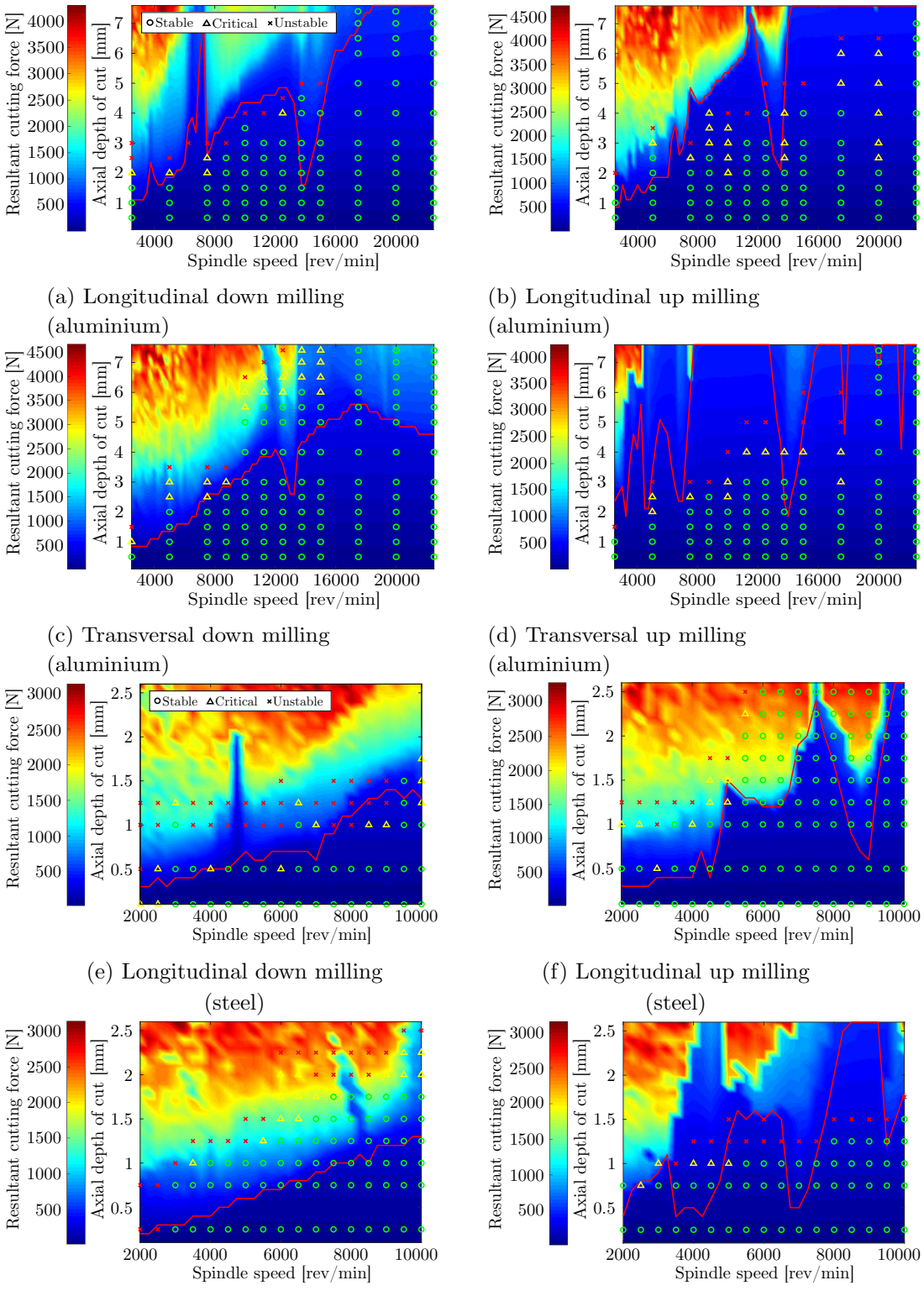
(f) Longitudinal up milling (steel)

(g) Transversal down milling (steel)

(h) Transversal up milling (steel)

Figure M.3: Experimental and simulated stability charts using the Stäubli TX200 model with all the flexibilities with a radial depth of cut a_e= 4 mm and standard cutting coefficients

With joint tri-axial flexibility and edge force coeffcients



(a) Longitudinal down milling (aluminium)

(b) Longitudinal up milling (aluminium)

(c) Transversal down milling (aluminium)

(d) Transversal up milling (aluminium)

(e) Longitudinal down milling (steel)

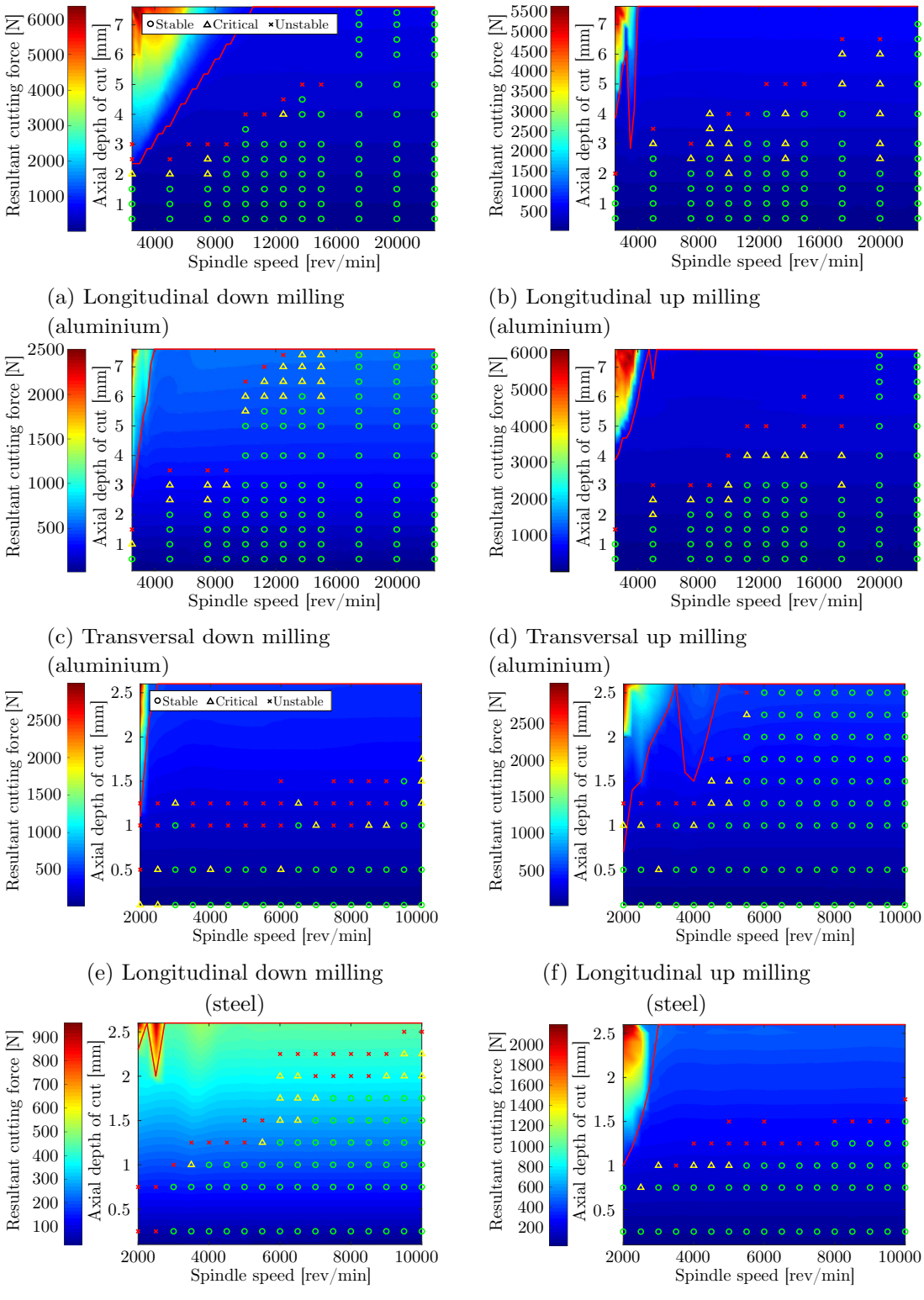
(f) Longitudinal up milling (steel)

(g) Transversal down milling (steel)

(h) Transversal up milling (steel)

Figure M.4: Experimental and simulated stability charts using the Stäubli TX200 model with joint flexibility with a radial depth of cut a_e= 4 mm and cutting coefficients comprising the edge forces

With joint tri-axial flexibility and fitting over 35 Hz



(a) Longitudinal down milling (aluminium)

(b) Longitudinal up milling (aluminium)

(c) Transversal down milling (aluminium)

(d) Transversal up milling (aluminium)

(e) Longitudinal down milling (steel)

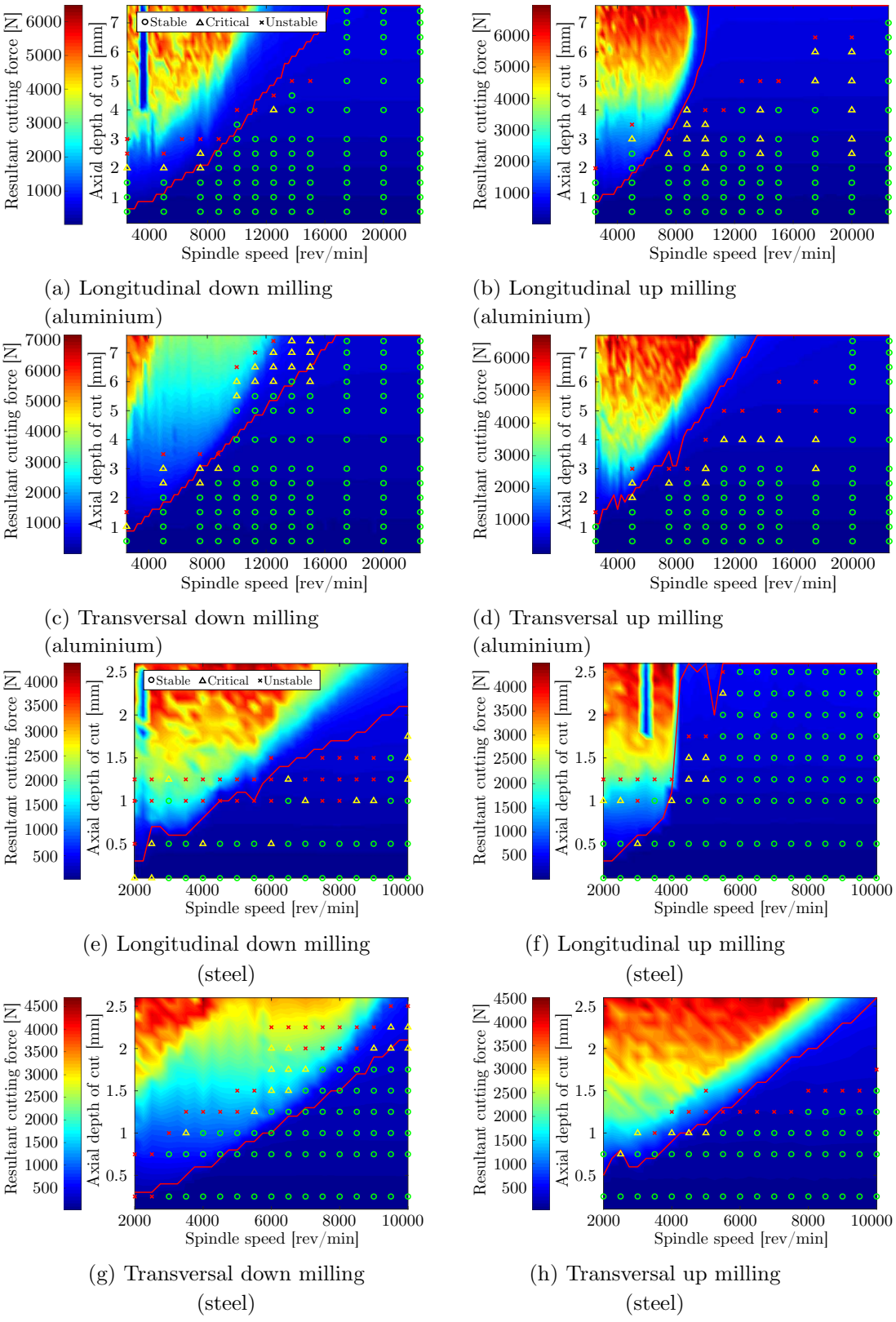
(f) Longitudinal up milling (steel)

(g) Transversal down milling (steel)

(h) Transversal up milling (steel)

Figure M.5: Experimental and simulated stability charts using the Stäubli TX200 model with joint flexibility fitted over 35 Hz with a radial depth of cut a_e= 4 mm and standard cutting coefficients

With joint tri-axial flexibility for the UMons TX200 model



(a) Longitudinal down milling (aluminium)

(b) Longitudinal up milling (aluminium)

(c) Transversal down milling (aluminium)

(d) Transversal up milling (aluminium)

(e) Longitudinal down milling (steel)

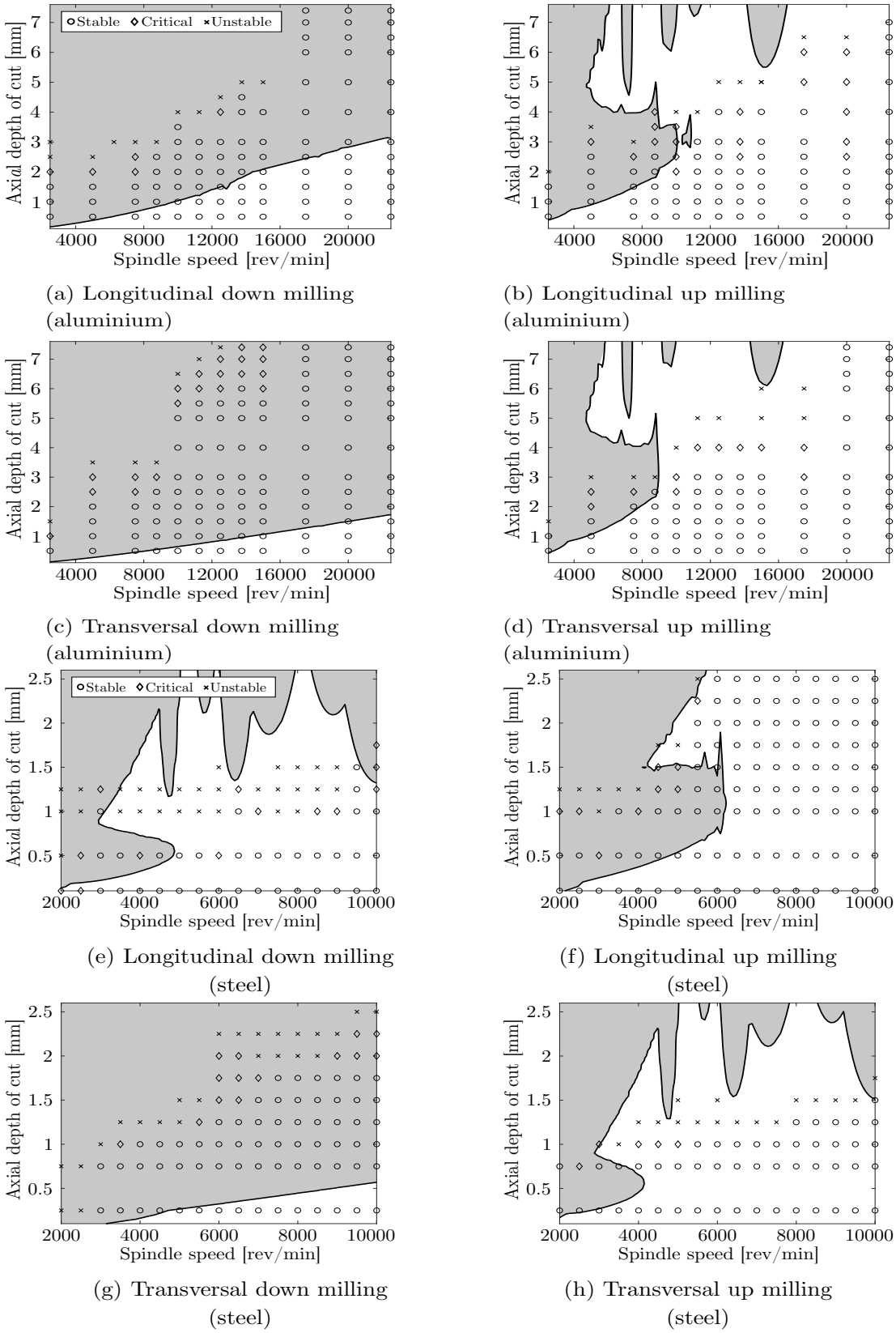
(f) Longitudinal up milling (steel)

(g) Transversal down milling (steel)

(h) Transversal up milling (steel)

Figure M.6: Experimental and simulated stability charts using the UMons TX200 model with only the joint flexibility with a radial depth of cut a_e= 4 mm and standard cutting coefficients

Semi-discretisation method (SDm)



(a) Longitudinal down milling (aluminium)

(b) Longitudinal up milling (aluminium)

(c) Transversal down milling (aluminium)

(d) Transversal up milling (aluminium)

(e) Longitudinal down milling (steel)

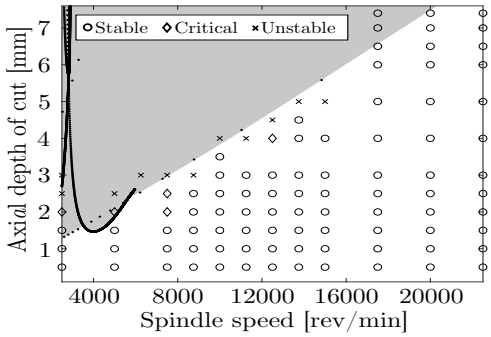
(f) Longitudinal up milling (steel)

(g) Transversal down milling (steel)

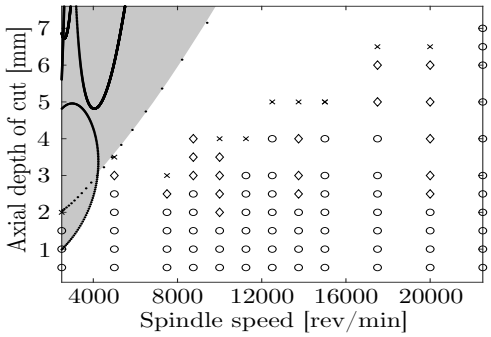
(h) Transversal up milling (steel)

Figure M.7: Experimental and simulated stability charts using SDm relying on identified modal parameters with a radial depth of cut a_e= 4 mm and standard cutting coefficients

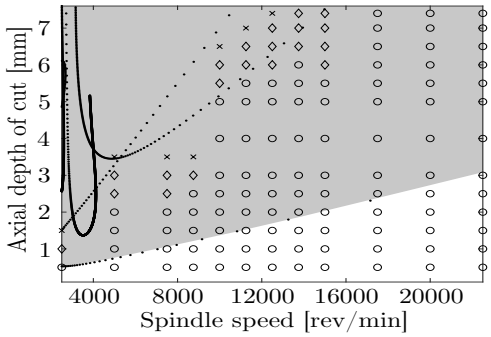
2D analytical zero-order approximation (ZOA)



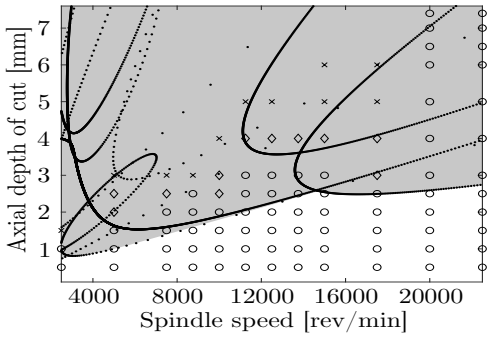
(a) Longitudinal down milling (aluminium)



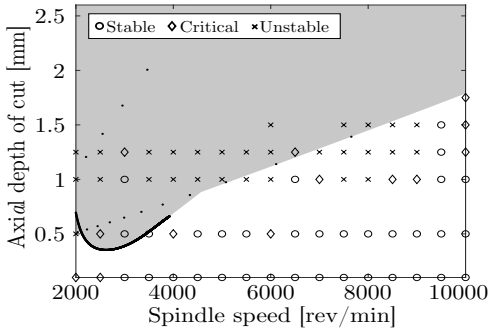
(b) Longitudinal up milling (aluminium)



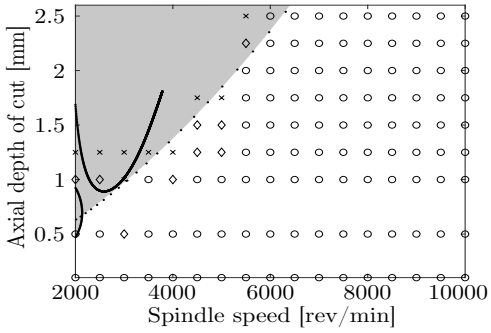
(c) Transversal down milling (aluminium)



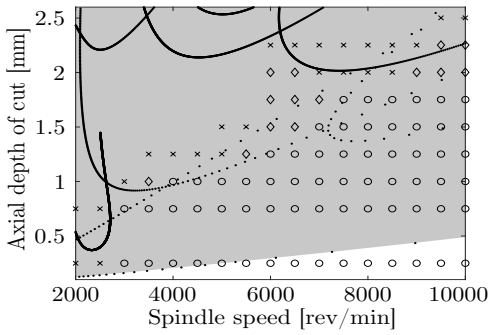
(d) Transversal up milling (aluminium)



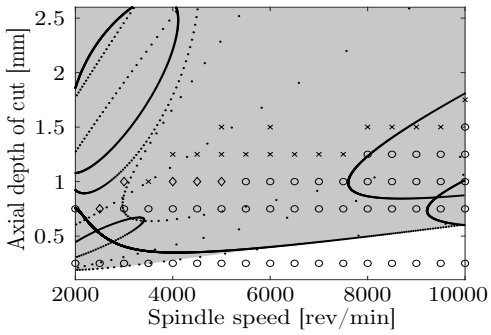
(e) Longitudinal down milling (steel)



(f) Longitudinal up milling (steel)



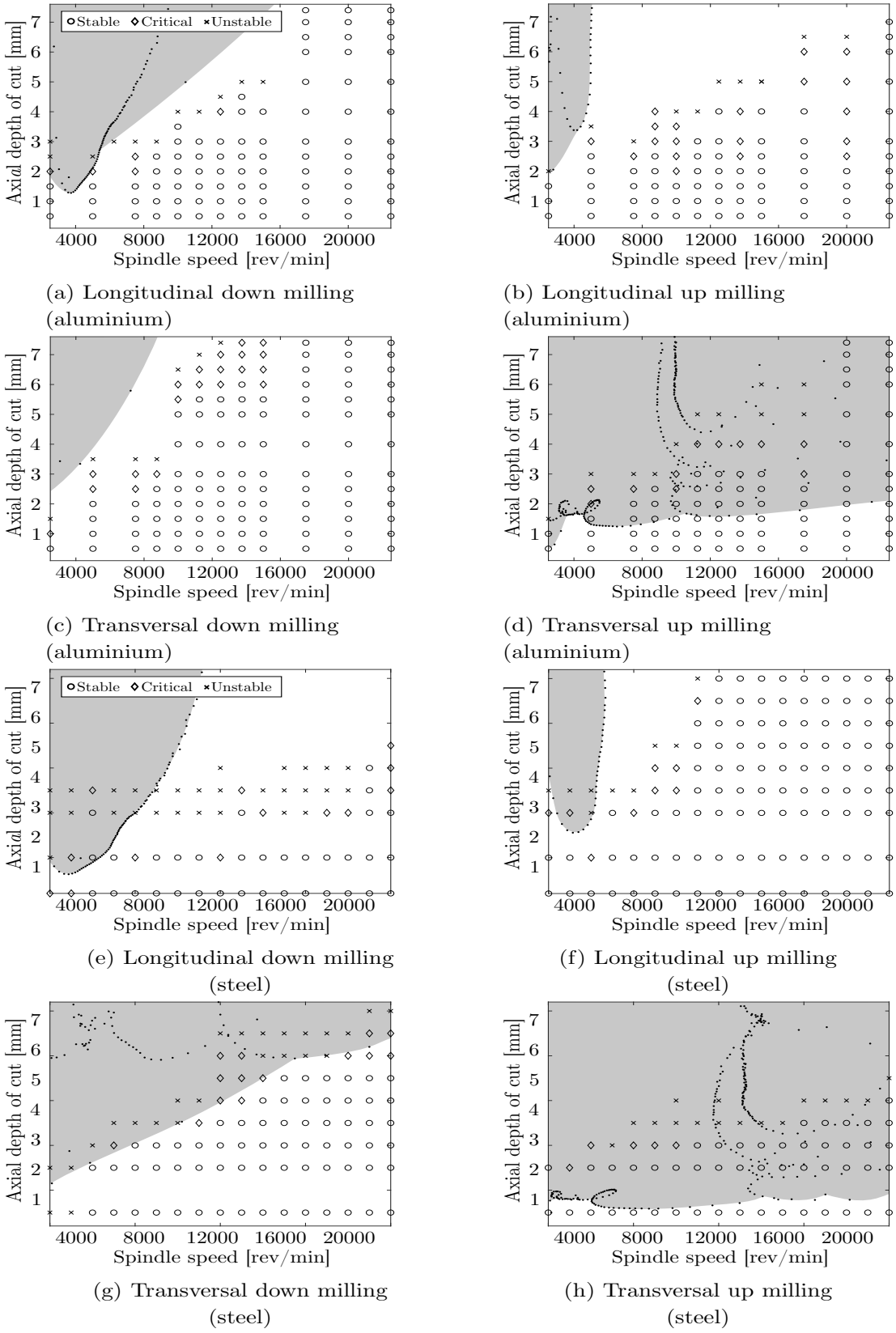
(g) Transversal down milling (steel)



(h) Transversal up milling (steel)

Figure M.8: Experimental and simulated stability charts using the analytical ZOA in 2D for N_c=0 relying on the synthesized tool tip FRFs with a radial depth of cut a_e= 4 mm and standard cutting coefficients

3D analytical zero-order approximation (ZOA)



(a) Longitudinal down milling (aluminium)

(b) Longitudinal up milling (aluminium)

(c) Transversal down milling (aluminium)

(d) Transversal up milling (aluminium)

(e) Longitudinal down milling (steel)

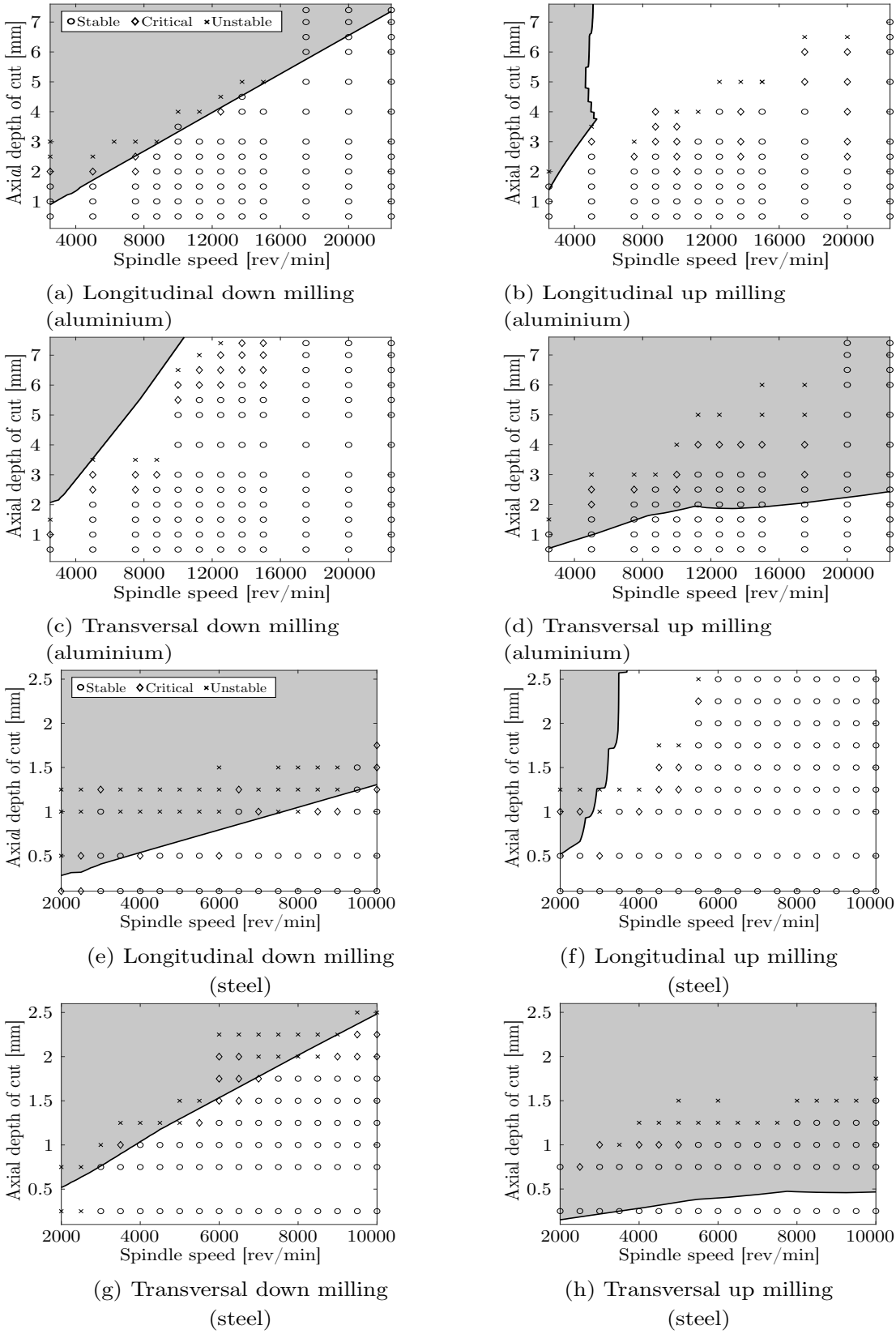
(f) Longitudinal up milling (steel)

(g) Transversal down milling (steel)

(h) Transversal up milling (steel)

Figure M.9: Experimental and simulated stability charts using the analytical ZOA in 3D for N_c=0 relying on the measured tool tip FRFs with a_e= 4 mm and standard cutting coefficients

3D digital zero-order approximation (ZOA)



(a) Longitudinal down milling (aluminium)

(b) Longitudinal up milling (aluminium)

(c) Transversal down milling (aluminium)

(d) Transversal up milling (aluminium)

(e) Longitudinal down milling (steel)

(f) Longitudinal up milling (steel)

(g) Transversal down milling (steel)

(h) Transversal up milling (steel)

Figure M.10: Experimental and simulated stability charts using the digital ZOA in 3D relying on the measured tool tip FRFs with a radial depth of cut a_e= 4 mm and standard cutting coefficients

ROBOTIC MACHINING

Development and validation of a numerical model of robotic milling in order to optimise the cutting parameters

Using industrial robots as machine tools is targeted by many industrials for their lo- wer cost and larger workspace. Nevertheless, performance of industrial robots is limited due to their serial mechanical structure involving rotational joints with a lower stiffness. As a consequence, vibration instabilities, known as chatter, are more likely to appear in industrial robots than in conventional machine tools. Commonly, chatter is avoided by using stability lobe diagrams which determine the stable combinations of axial depth of cut and spindle speed. Although the computation of stability lobes in conventional machine tools is a well-studied subject, developing them in robotic milling is challenging because of the lack of accurate dynamic multibody models involving joint compliance to predict the posture-dependent dynamics of the robot.

In this work, the stability lobe diagrams of milling operations are computed in the time domain using a dynamic multibody model of the Sta¨ubli TX200 robot, which is a six-axis serial robot. Since past studies revealed that the flexibility of industrial robots mainly originated from its joints, the multibody model of the Sta¨ubli TX200 robot comprises joints with torsional and transversal compliances which represent the transmission and bearing flexibilities, respectively. Variants of the multibody model are developed in order to assess the influence of the link and controller flexibilities. A dynamic milling model is coupled to the simulated robot allowing the computation of the cutting forces and a representation of the virtual machined surface.

Once the inertia characteristics of the robot are identified, its elastic parameters are fit- ted to modal measurements obtained via experimental modal analysis. In particular, a straightforward identification method, relying on the curve fitting of the tool tip frequency response functions, is proposed to determine the elastic parameters of the multibody mo- del. Parameters pertaining to the flexible links are determined by matching finite element models while the control parameters are settled based on the modal measurements. Cut- ting force coefficients are classically identified through milling experiments.

Using the robotic milling simulator validated in stable cutting conditions, stability lobe diagrams are simulated and experimentally validated. The influence of the feed direction and the aforementioned flexibilities is appraised on the system stability. The robotic milling simulator could reasonably capture the overall stability limits in all feed directions in aluminium and in steel. Appending the flexibility of the controller or the links did not lead to significant modification in the prediction of the stability charts. However, their consideration is important when dealing with static deflections. It was indeed shown that the consideration of flexible links induced 20 to 30 % of additional deflections while the ones introduced by the controller were almost negligible. Since robot structural modes depend on its posture, it appeared that stability limits were affected by the feed direction. It was also observed that in low-speed milling, mostly the low-frequency pose-dependent robot modes chatter while in high-speed milling robot wrist modes trigger the instability.

Université de Mons

20, Place du Parc, B7000 Mons - Belgique
Tél: +32(0)65 373111
Courriel: info.mons@umons.ac.be
www.umons.ac.be

UMONS Université de Mons | POLYTECH MONS