Linear Algebra and its Applications 573 (2019) 37–53

Contents lists available at ScienceDirect

Linear Algebra and its Applications

www.elsevier.com/locate/laa

Approximating the nearest stable discrete-time system

Nicolas Gillis [a,1], Michael Karow [b], Punit Sharma [c,*,2]

[a] *Department of Mathematics and Operational Research, Faculté Polytechnique, Université de Mons, Rue de Houdain 9, 7000 Mons, Belgium*
[b] *TU Berlin, Institut für Mathematik, Straße des 17. Juni 136, 10623 Berlin, Germany*
[c] *Department of Mathematics, Indian Institute of Technology Delhi, Hauz Khas, New Delhi, 110016, India*

ARTICLE INFO

Article history:
Received 10 June 2018
Accepted 9 March 2019
Available online 22 March 2019
Submitted by T. Damm

MSC:
15B48
34K20
90C26
65F15

Keywords:
Stability radius
Linear discrete-time systems
Stability
Convex optimization

ABSTRACT

In this paper, we consider the problem of stabilizing discrete-time linear systems by computing a nearby stable matrix to an unstable one. To do so, we provide a new characterization for the set of stable matrices. We show that a matrix A is stable if and only if it can be written as $A = S^{-1}UBS$, where S is positive definite, U is orthogonal, and B is a positive semidefinite contraction (that is, the singular values of B are less or equal to 1). This characterization results in an equivalent non-convex optimization problem with a feasible set on which it is easy to project. We propose a very efficient fast projected gradient method to tackle the problem in variables (S, U, B) and generate locally optimal solutions. We show the effectiveness of the proposed method compared to other approaches.

[*] Corresponding author.
E-mail addresses: nicolas.gillis@umons.ac.be (N. Gillis), karow@math.tu-berlin.de (M. Karow), punit.sharma@maths.iitd.ac.in (P. Sharma).

[1] N. Gillis acknowledges the support of the ERC (starting grant n° 679515) and F.R.S. - FNRS (incentive grant for scientific research n° F.4501.16).

[2] P. Sharma acknowledges the support of the DST-Inspire Faculty Award (MI01807-G) by Government of India and Institute New Faculty SEED Grant by IIT Delhi (Budget Head, NPN5R).

https://doi.org/10.1016/j.laa.2019.03.014
0024-3795/© 2019 Elsevier Inc. All rights reserved.

1. Introduction

Consider a discrete-time linear system described by the following difference equation

$$x(t+1) = Ax(t), \quad t \in \mathbb{N}, \tag{1}$$

where $A \in \mathbb{R}^{n,n}$ and $\mathbb{N}$ is the set of nonnegative integers, $x(t)$ denotes the n-dimensional state vector. If $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of A, then such a system is called stable (resp. asymptotically stable) if $|\lambda_i| \leq 1$ (resp. $|\lambda_i| < 1$) for all $i = 1, \ldots, n$, and the eigenvalues with unit modulus are semisimple; otherwise, it is called unstable.

In this paper, we consider the *nearest stable matrix problem* in the discrete-time case. More precisely, for a given unstable matrix A, we consider the following optimization problem

$$\inf_{X \in \mathbb{S}_d^{n,n}} \|A - X\|_F^2, \tag{2}$$

where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix and $\mathbb{S}_d^{n,n}$ is the set of all stable matrices of size $n \times n$. We consider in this paper the Frobenius norm of the error as it is arguably the most widely used norm. However, our approach can be directly applied to any differentiable cost function (e.g., any component-wise ℓ_p norm with $p > 1$).

Notation Throughout the paper, X^T, $\operatorname{tr}(X)$, and $\|X\|$ stand for the transpose, the trace and the spectral norm of a real square matrix X, respectively. By $\Lambda(X)$, $\rho(X) := \max_{\lambda \in \Lambda(X)} |\lambda|$ and $\kappa(X) = \|X\| \, \|X^{-1}\|$ we denote the spectrum (set of eigenvalues), the spectral radius and the condition number. We write $X \succ 0$ and $X \succeq 0$ ($X \preceq 0$) if X is symmetric and positive definite or positive semidefinite (symmetric negative semidefinite), respectively. The positive semidefinite symmetric square root of a positive semidefinite symmetric matrix X is denoted by $X^{1/2}$.

Related work For a given unstable matrix A (in the discrete- or continuous-time case), the problem of computing the smallest perturbation that stabilizes A, also known as the nearest stable matrix problem, occurs in system identification where one needs to identify a stable system depending on observations [1]. To the best of our knowledge, the nearest stable matrix problem was first introduced and analyzed in the discrete- and continuous-time case in [1], where a nearby stable approximation X of a given unstable system A is constructed by means of successive convex approximations of the set of stable systems. For the continuous-time case, two methods were recently proposed:

1. In [2], the problem is reformulated into an equivalent problem with a simple convex feasible set. In fact, it is shown that A is stable if and only if it can be written as $A = (J - R)Q$ where $J^T = -J$, $R \succeq 0$, and $Q \succ 0$. This result was later generalized to solve various nearness problems for continuous-time linear systems, namely, the

problems of finding the nearest stable Metzler matrix [3], the nearest stable matrix pair [4] and the nearest positive real system [5].

2. In [6], the problem is tackled by solving low-rank matrix differential equations.

The nearest stable matrix problem in discrete-time case has received much less attention, and to the best our knowledge, only [1] considered this problem without any assumption on the entries of the matrix. For the class of positive systems of the form (1), where the matrix A is component-wise nonnegative, the problem of computing the nearest stable nonnegative matrix has been studied very recently in [7,8]. In [7], authors consider the nearest stable/unstable nonnegative matrix with respect to the max-norm $\|X\|_{\max} = \max_{i,j} |X_{i,j}|$, the ℓ_∞ operator norm $\|X\|_\infty = \sup_{u\neq 0} \frac{\|Xu\|_\infty}{\|u\|_\infty}$, and the ℓ_1 operator norm $\|X\|_1 = \sup_{u\neq 0} \frac{\|Xu\|_1}{\|u\|_1}$, where $\|x\|_\infty = \max_i |x_i|$ and $\|x\|_1 = \sum_i |x_i|$. For these norms, it turns out that, rather surprisingly, the problem can be solved in polynomial-time. In [8], authors propose a more efficient heuristic than in [1] for the Frobenius norm for which the problem is more difficult, with the existence of many local minima (up to 2^n in dimension n).

The nearest stable matrix problem (2) is the converse problem of stability radius problem in the discrete-time case, where a stable matrix A is given and one looks for the smallest perturbation that moves an eigenvalue outside the stability region. The converse problem has been introduced and studied extensively; see, e.g., [9–14] and the references therein.

The problem (2) is notoriously difficult due to properties of the spectral radius as a function of matrix: the set $\mathbb{S}_d^{n,n}$ of stable matrices is highly nonconvex [1], and neither open nor closed. For example, $B_\epsilon \notin \mathbb{S}_d^{2,2}$ for $\epsilon > 0$ but $B \in \mathbb{S}_d^{2,2}$, where

$$\underbrace{\begin{bmatrix} 1 & \epsilon \\ -\epsilon & 1 \end{bmatrix}}_{=:B_\epsilon} \to \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}_{=:B},$$

and $C_\delta \in \mathbb{S}_d^{2,2}$ for $0 \leq \delta < 1$, but $C \notin \mathbb{S}_d^{2,2}$, where

$$\underbrace{\begin{bmatrix} 1 & 1 \\ 0 & \delta \end{bmatrix}}_{=:C_\delta} \to \underbrace{\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}}_{=:C}.$$

Therefore it is in general difficult to obtain a global optimal solution to problem (2).

The aim of this paper is to derive counterparts of a number of results in [2] for the discrete-time case. These results require special constructions and show special features in the discrete-time case. Our principle strategy for computing a nearby stable approximation to a given unstable matrix is to reformulate the problem (2) into an equivalent problem with a simple feasible set onto which points can be projected relatively easily. We aim to provide in many cases a better approximation than the one obtained with the code from [1] at a lower computational cost.

The paper is organized as follows. In Section 2, we define the SUB form of a matrix: the matrix A admits a SUB form if there exists (S, U, B) with $S \succ 0$, U is orthogonal, $B \succeq 0$ and $\|B\| \leq 1$ such that $A = S^{-1}UBS$. We prove that a matrix is stable if and only if it admits a SUB form. In Section 3, we propose a fast gradient method (FGM) to solve the reformulated problem in variables (S, U, B), along with several initialization strategies. To illustrate the performance of FGM, we apply it on several examples of unstable matrices and compare the results with the algorithm from [1].

2. A new characterization for stable matrices

In this section, we derive a factorization of stable matrices into symmetric and orthogonal matrices. This will allow us to reformulate problem (2) into an equivalent problem with a simple feasible set for which standard optimization methods can be applied. In order to do this, we define the SUB form of a matrix.

Definition 1. A matrix $A \in \mathbb{R}^{n,n}$ is said to admit a *SUB form* if there exist $S, U, B \in \mathbb{R}^{n,n}$ such that $A = S^{-1}UBS$ where $S \succ 0$, U is orthogonal, $B \succeq 0$ and $\|B\| \leq 1$.

Theorem 1. *A matrix is stable (asymptotically stable) if and only if it admits a SUB form (SUB form with* $\|B\| < 1$*).*

Proof. The proof follows by the following two facts: 1) the Lyapunov criterion of the Schur stability [15]; 2) the polar decomposition. Indeed, by the Lyapunov theorem, A is stable (asymptotically stable) if and only if there exists an ellipsoid E such that $AE \subset E$ (respectively, $AE \subset \text{int}E$). This is equivalent to say that there exist matrices C and L such that $\|L\| \leq 1$ (respectively, $\|L\| < 1$) and $A = C^{-1}LC$. Now we write the polar decomposition $C = VS$, where V is orthogonal and $S \succ 0$. Thus, $A = S^{-1}V^{-1}LVS$. Denote $V^{-1}LV = M$. Clearly, $\|M\| = \|L\|$. Finally, write the polar decomposition: $M = UB$ with U orthogonal, $B \succ 0$, and $\|B\| = \|M\|$. We have $A = S^{-1}UBS$, which completes the proof. □

In view of Theorem 1, the set $\mathbb{S}_d^{n,n}$ of stable matrices can be characterized as the set of matrices that admit a SUB form, or equivalently, we can parameterize the set of stable matrices using a matrix triple (S, U, B) as follows

$$\mathbb{S}_d^{n,n} = \left\{ S^{-1}UBS \in \mathbb{R}^{n,n} \;\middle|\; S \succ 0,\ U \text{ orthogonal},\ B \succeq 0 \text{ with } \|B\| \leq 1 \right\}.$$

This characterization changes the feasible set and the objective function in the nearest stable matrix problem (2) as

$$\inf_{X \in \mathbb{S}_d^{n,n}} \|A - X\|_F^2 = \inf_{S \succ 0,\ U \text{ orthogonal},\ B \succeq 0,\ \|B\| \leq 1} \|A - S^{-1}UBS\|_F^2. \tag{3}$$

As we mentioned earlier, the set $\mathbb{S}_d^{n,n}$ of stable matrices is neither open nor closed and clearly the new parameterization of $\mathbb{S}_d^{n,n}$ in terms of matrix triple (S,U,B) does not change this, since $\mathbb{S}_d^{n,n}$ is not open because of the constraint $B \succeq 0$ and not closed due to the constraint $S \succ 0$. Therefore the infimum in the right hand side of (3) may not be attained.

In the next section, we will provide an algorithmic solution for the nearest stable matrix problem (2) by trying to solve the reformulated problem (3). Note that in view of Lyapunov's Theorem replacing S, U, and B in terms of the variable $P \succ 0$ leads to the formulation

$$\inf_{X,\, P \succ 0} \|A - X\|_F \quad \text{such that} \quad X^T P X - P \preceq 0,$$

which is difficult to solve numerically as it involves highly non-linear constraints [1]. Thus a key contribution of this paper is the reformulation (3). An advantage of this reformulation is that the feasible set is rather simple and therefore it is relatively easy to project onto it. As a result we propose a fast projected gradient method to solve the reformulated problem (3), see Algorithm 1. We close the section with some useful remarks on the matrices that admit a SUB form.

Remark 1. Let $\gamma \in \mathbb{R}$ with $0 < \gamma < 1$. Then A is called γ-stable if $\lambda \in \Lambda(A)$ satisfies $|\lambda| \leq \gamma$. Note that A is γ-stable if and only if $B = \frac{A}{\gamma}$ is stable, since for any nonzero $x \in \mathbb{C}^n$ we have $Ax = \mu x$ if and only if $\mu = \gamma\lambda$ for some $\lambda \in \Lambda(B)$ such that $Bx = \lambda x$. Thus from Theorem 1 A is γ-stable if and only if $\frac{A}{\gamma}$ admits a SUB form if and only if A admits a SUB form with $\|B\| \leq \gamma$. This observation can be used to find a nearby γ-stable matrix to a given unstable one.

Remark 2. We note that a remark for the non-uniqueness of the SUB decomposition of a stable matrix similar to [2, Remark 6] for continuous systems also holds in the discrete case. The SUB representation of a stable matrix A, that is, $A = S^{-1}UBS$ where $S \succ 0$, $U^TU = I_n$, $B \succeq 0$, and $\|B\| \leq 1$, is non-unique. As there is always a scaling degree of freedom: for any scalar $\alpha > 0$ we have $A = (\alpha S)^{-1}UB(\alpha S)$. This can partially be addressed by the fact that the ellipsoid E in the Lyapunov Theorem is non-unique and the matrices S and B in the SUB form depend on E, see the proof of Theorem 1. However, characterizing precisely the non-uniqueness of the SUB form (and possibly taking advantage of it in a numerical algorithm) is a direction for further research.

Remark 3. The results of this section are readily extended to handle complex matrices by substituting X^*, the conjugate transpose for X^T and unitary matrices for orthogonal matrices. In particular, we have that $A \in \mathbb{C}^{n,n}$ is stable if and only if there exist $S, U, B \in \mathbb{C}^{n,n}$ such that $A = S^{-1}UBS$ where $S \succ 0$, U is unitary, $B \succeq 0$ and $\|B\| \leq 1$. We note that a similar observation also holds for the characterization of complex stable matrices in the continuous-time case. In particular following the terminology in [2] we have that

$A \in \mathbb{C}^{n,n}$ is stable in the continuous-time case if and only if there exist $J, R, Q \in \mathbb{C}^{n,n}$ such that $A = (J - R)Q$ where $J^* = -J$, $R \succeq 0$, and $Q \succ 0$. This was not mentioned in [2].

3. Algorithmic solutions to the nearest stable matrix problem

As shown in Section 2, finding the nearest stable matrix to an unstable one is equivalent to solving (3). In this section, we propose a fast projected gradient method [16, p. 90] to tackle (3). Although fast gradient methods (FGM's) were initially designed for convex optimization problems, they have recently been shown to work well for non-convex problems as well; see, e.g., [17–19]. In particular, for the problem of finding the nearest stable matrix in the continuous-time case, they work significantly better than standard gradient schemes and coordinate descent methods [2], while being relatively simple to implement. We use a similar implementation as in [5]; see Algorithm 1 for the details. As for the standard projected gradient method, FGM requires the computation of the gradient of the objective function, and the projection onto the feasible set. The gradient of $f(S, U, B) = \|A - S^{-1}UBS\|_F^2$ with respect to S is given by

$$\nabla_S f(S, U, B) = 2\, S^{-T}[R^T(R - A) - (R - A)R^T],$$

where $R = S^{-1}UBS$. The details are given in Appendix A. For U and B, we have

$$\nabla_U f(S, U, B) = -2S^{-1}(A - R)SB^T$$

and

$$\nabla_B f(S, U, B) = -2U^T S^{-1}(A - R)S.$$

The projections of a solution (S, U, B) onto the feasible set of (3) are described in Section 3.1.

Convergence. Algorithm 1 is guaranteed to decrease the objective function at each step because of the line-search (steps 7-10). Hence, at every iteration, we have $\|A - S^{-1}UBS\|_F \leq f_0$ where f_0 is the initial objective function value. Since the objective function is bounded from below by zero, this implies that the objective function values converge to some value f^*. Moreover, the approximations $S^{-1}UBS$ generated at each step of the algorithm are in a compact set: in fact,

$$\|S^{-1}UBS\|_F - \|A\|_F \leq \|A - S^{-1}UBS\|_F \leq f_0 \quad \Longrightarrow \quad \|S^{-1}UBS\|_F \leq f_0 + \|A\|_F.$$

Therefore, there exists a subsequence of approximations $S^{-1}UBS$ generated by Algorithm 1 that converge to some limit point A_p^* with $||A - A_p^*||_F = f^*$. However, it is more difficult to prove convergence of the iterates (S, U, B) as S is not bounded (e.g., if $A = 0$, then $B = 0$ is optimal while S can be any invertible matrix). It is possible to add an

Algorithm 1 Fast Gradient Method (FGM) for (3) with restart from [5].

```
Require: An initialization $X = (S, U, B)$, a parameter $\alpha_1 \in (0, 1)$, a lower bound for the step length $\underline{\gamma}$, an
    initial step length $\gamma > \underline{\gamma}$.
Ensure: An approximate solution $X = (S, U, B)$ to (3).

 1: $X' = X$. % Create the second sequence of iterates of FGM.
 2: for $k = 1, 2, \ldots$ do
 3:     $\hat{X} = X$. % Keep the previous iterate in memory.
 4:     % Project the gradient step from $X'$ onto the feasible set.
 5:     $X = \mathcal{P}(X' - \gamma \nabla f(X'))$. % $\mathcal{P}$ is the projection onto the feasible set.
 6:     % Check if the objective function $f$ has decreased, otherwise decrease the step length.
 7:     while $f(X) > f(\hat{X})$ and $\gamma \geq \underline{\gamma}$ do
 8:         $\gamma = \frac{2}{3}\gamma$.
 9:         $X = \mathcal{P}(X' - \gamma \nabla f(X'))$.
10:     end while
11:     % If the step length has reached the lower bound ($f$ could not be decreased), reinitialize $X'$ (the
        next step will be a standard gradient descent step).
12:     if $\gamma < \underline{\gamma}$ then
13:         Restart fast gradient: $X' = X$; $\alpha_k = \alpha_1$.
14:         Reinitialize $\gamma$ at the last value for which it allowed decrease of $f$.
15:     else
16:         $\alpha_{k+1} = \frac{1}{2}\left(\sqrt{\alpha_k^4 + 4\alpha_k^2} - \alpha_k^2\right)$, $\beta_k = \frac{\alpha_k(1-\alpha_k)}{\alpha_k^2 + \alpha_{k+1}}$.
17:         $X' = X + \beta_k \left(X - \hat{X}\right)$.
18:     end if
19:     $\gamma = 2\gamma$.
20: end for
```

upper bound on the norm of S to guarantee a subsequence of iterates to converge, but we have not observed in practice that this was an issue. Providing a rigorous proof of convergence of the iterates of Algorithm 1 to a stationary point of (3) is a difficult problem which we leave as a question for further research. It has to be noted that only stationary points are fixed point of our method, since this is a projected gradient method.

Parameters. Algorithm 1 is not too sensitive to the initial step length γ as it increases/decreases it to allow the objective function to decrease, and reinitialize the value to the previous value that allowed decrease when it is restarted (step 14). We chose the initial step length to be $\gamma = 1/L$ where $L = \left(\frac{\lambda_{\max}(S)}{\lambda_{\min}(S)}\right)^2 = \kappa(S)^2$. The reason for this choice is that L is the Lipschitz constant of the gradient of f with respect to B so that using the step length $1/L$ would guarantee the decrease of f if we would only optimize over B as the problem in variable B is convex [16]. For α_1, we use 0.5 as in [5]. Since (3) is a difficult non-convex optimization problem, any local optimization scheme such as our FGM approach will be sensitive to initialization; this is discussed in Section 3.2.

3.1. Projections

In this section we derive the relevant formulas to project S, U and B onto the feasible set of (3).

Projections for S and B. In order to calculate the projection of a square matrix onto the set of positive semidefinite contractions, we introduce some notation. For a symmetric matrix $H \in \mathbb{R}^{n,n}$ with eigenvalues λ_k $(1 \leq k \leq n)$ and orthogonal diagonalization $H = V\text{diag}(\lambda_1, \ldots, \lambda_n)V^T$, we set $f(H) = V\text{diag}(f(\lambda_1), \ldots, f(\lambda_n))V^T$, where f is any complex valued function defined on the spectrum of H. The matrix $f(H)$ does not depend on the particular orthogonal matrix V since it is easily verified that $f(H) = q(H)$, where q is any polynomial that maps each λ_k to its value $f(\lambda_k)$. For a general matrix $X \in \mathbb{R}^{n,n}$, we consider functions of its symmetric part, $f^s(X) := f((X + X^T)/2)$. For an interval $[a, b] \subset \mathbb{R} \cup \{\infty\}$ and $\lambda \in \mathbb{R}$ let

$$p_{a,b}(\lambda) := \max\{a, \min\{b, \lambda\}\} = \begin{cases} a & \text{if } \lambda < a, \\ \lambda & \text{if } \lambda \in [a, b], \\ b & \text{if } b < \lambda. \end{cases}$$

Then $p_{a,b}(\lambda)$ is the nearest point projection of λ onto $[a, b]$, that is, $|\lambda - p_{a,b}(\lambda)| = \operatorname{argmin}_{h\in[a,b]}|\lambda - h|$. The statement below extends [20, Lemma 10] to the case that X is nonsymmetric.

Proposition 1. *With respect to Frobenius norm the matrix $p^s_{a,b}(X)$ is the nearest point projection of $X \in \mathbb{R}^{n,n}$ onto the set $\mathcal{I}_{a,b} = \{ H \in \mathbb{R}^{n,n} \mid H = H^T,\ a\,I \preceq H \preceq b\,I\}$, that is,*

$$p^s_{a,b}(X) = \operatorname{argmin}_{H\in\mathcal{I}_{a,b}}\|X - H\|_F.$$

Proof. Let $(X + X^T)/2 = V\operatorname{diag}(\lambda_1, \ldots, \lambda_n)V^T$ with orthogonal V. Let $H \in \mathcal{I}_{a,b}$, and let $\tilde{H} = V^T HV = [\tilde{h}_{ij}]$. Then $\tilde{H} \in \mathcal{I}_{a,b}$ and therefore $\tilde{h}_{ii} \in [a, b]$ for all $i = 1, \ldots, n$. By orthogonality between symmetric and skew symmetric matrices and the orthogonal invariance of the Frobenius norm we have

$$\begin{aligned}\|X - H\|_F^2 &= \left\|\frac{X - X^T}{2}\right\|_F^2 + \left\|\frac{X + X^T}{2} - H\right\|_F^2 \\ &= \left\|\frac{X - X^T}{2}\right\|_F^2 + \|\operatorname{diag}(\lambda_1, \ldots, \lambda_n) - \tilde{H}\|_F^2 \\ &= \left\|\frac{X - X^T}{2}\right\|_F^2 + \sum_i(\lambda_i - \tilde{h}_{ii})^2 + \sum_{i\neq j}\tilde{h}_{ij}^2. \qquad (4)\end{aligned}$$

The sum is minimized by $\tilde{H} = \operatorname{diag}(p_{a,b}(\lambda_1), \ldots, p_{a,b}(\lambda_n))$. Thus, $H = p^s_{a,b}(X)$. □

Since for a positive semidefinite matrix the inequality $\|B\| \leq \alpha$ is equivalent to $B \preceq \alpha I_n$ we have the corollaries below.

Corollary 1. *With respect to Frobenius norm the nearest point projection of $X \in \mathbb{R}^{n,n}$ onto the set of positive semidefinite contractions is $p^s_{0,1}(X)$, that is,*

$$p^s_{0,1}(X) = \operatorname{argmin}_{B\succeq 0, \|B\|\leq 1}\|X - B\|_F.$$

Corollary 2. [21] *With respect to Frobenius norm the nearest point projection of $X \in \mathbb{R}^{n,n}$ onto the cone of $n \times n$ positive semidefinite matrices is $p^s_{0,\infty}(X)$, that is,*

$$p^s_{0,\infty}(X) = \operatorname{argmin}_{B \succeq 0} \|X - B\|_F.$$

Projections for U. Before we give the projection onto the set of orthogonal matrices, we provide another closely related projection that will be useful to obtain initializations in Section 3.2. Note that these results require the polar decomposition [22].

Proposition 2. *Let $X \in \mathbb{R}^{n,n}$ and let $X = VH$ be the polar decomposition of X, where $V \in \mathbb{R}^{n,n}$ is orthogonal and $H \in \mathbb{R}^{n,n}$ satisfies $H \succeq 0$. Then*

$$\mathit{argmin}_{(U,B), U^TU=I_n, B \succeq 0, \|B\| \le 1} \|X - UB\|_F^2 = (V, p_{0,1}(H)),$$

Proof. Let $H = Q \operatorname{diag}(\lambda_1, \dots, \lambda_n) Q^T$ be a diagonalization of H with orthogonal Q. Let $U, B \in \mathbb{R}^{n,n}$ be such that $U^TU = I_n$ and $B \succeq 0$ with $\|B\| \le 1$. Then

$$\begin{aligned} \|X - UB\|_F^2 &= \|VH - UB\|_F^2 = \|H - V^TUB\|_F^2 \\ &= \|Q \operatorname{diag}(\lambda_1, \dots, \lambda_n) Q^T - V^TUB\|_F^2 \\ &= \|\operatorname{diag}(\lambda_1, \dots, \lambda_n) - Q^TV^TUBQ\|_F^2 \\ &\ge \sum_i (\lambda_i - p_{-1,1}(\lambda_i))^2 \qquad (5) \\ &= \sum_i (\lambda_i - p_{0,1}(\lambda_i))^2. \end{aligned}$$

The last equation holds since all λ_i's are nonnegative. The inequality (5) follows from the fact that all diagonal entries of Q^TV^TUBQ are contained in $[-1, 1]$ since $\|Q^TV^TUBQ\| = \|B\| \le 1$. Equality holds in (5) if and only if $Q^TV^TUBQ = \operatorname{diag}(p_{0,1}(\lambda_1), \dots, p_{0,1}(\lambda_n))$. The latter is equivalent to $UB = Vp_{0,1}(H)$. □

Proposition 3. *Denoting $\mathcal{P}_\perp(X)$ the projection of X onto the set of $n \times n$ orthogonal matrices, we have $\mathcal{P}_\perp(X) = \mathit{argmin}_{U^TU=I_n} \|X - U\|_F = V$, where $X = VH$ is the polar decomposition of X.*

The proof is analogous to the proofs of the other propositions in this section and therefore omitted.

3.2. Initializations

In this section, we propose three initializations.

Standard initialization We use $S = I_n$, for which the optimal values of U and B can be computed using the polar decomposition of A, see Proposition 2:

$$\operatorname{argmin}_{(U,B),U^TU=I_n,B\succeq 0,\|B\|\leq 1}\|A-UB\|_F = (V, p_{0,1}(H)),$$

where $A = VH$ is the polar decomposition of A. Since in the polar decomposition, we have $\lambda_i(H) = \sigma_i(A)$ where $\lambda_i(H)$ is the ith eigenvalue of H and $\sigma_i(A)$ is the ith singular value of A, the standard initialization provides an initial error of

$$\|A - V\,p_{0,1}(H)\|_F^2 = \sum_{i,\sigma_i(A)>1} (\sigma_i(A) - 1)^2. \tag{6}$$

LMI-based initialization Let $\mu = \max(1, \rho(A))$ so that $A' = \frac{A}{\mu}$ is stable. Then, we use the Lyapunov solution $P \succ 0$ to the system $A'^T PA' - P \preceq 0$ (we used the Matlab function `dlyap(A,eye(n))`) and define $S = P^{1/2}$, $R = SA'S^{-1}$, and (U, B) is the polar decomposition of $R = UB$ so that $A' = S^{-1}UBS$; see the proof of Theorem 1. This initialization provides a solution with initial error:

$$\|A - A'\|_F^2 = \|A - A/\mu\|_F^2 = \|A\|_F^2(1 - 1/\mu)^2. \tag{7}$$

Remark 4 *(Comparing* (6) *and* (7)*).* None of the two solutions from (6) and (7) dominate the other one. It depends on the singular- and eigen-values of A. For example, A may be stable so that $|\rho(A)| < 1$ while $\sigma_{\max}(A)$ (the largest singular value of A) is greater than one in which case (7) provides an optimal solution (with error zero) while (6) has a positive error. On the other hand, if A is symmetric so that $\sigma_{\max}(A) = \rho(A)$ and A has a single singular value larger than 1, then the solution (6) has smaller error than (7). In fact,

$$\begin{aligned}\|A\|_F^2(1 - 1/\mu)^2 &= \sum_{i=1}^{n} (\sigma_i(A))^2 \left(\frac{\sigma_{\max}(A) - 1}{\sigma_{\max}(A)}\right)^2 \\ &\geq (\sigma_{\max}(A))^2 (\sigma_{\max}(A) - 1)^2 \\ &> (\sigma_{\max}(A) - 1)^2.\end{aligned}$$

Random initialization We generate each entry of S using the normal distribution (in Matlab, `randn(n)`). Then, we replace S with $SS^T + I_n$ which is positive definite. Ideally, we then would like to compute the corresponding optimal (U, B), that is, minimize $\|A - S^{-1}UBS\|_F$. However, we do not know how to do this efficiently, and instead we take U and B as the optimal solution of

$$\min_{U \text{ orthogonal}, B \succeq 0, \|B\| \leq 1} \|SAS^{-1} - UB\|_F,$$

that is, (U, B) is the polar decomposition of SAS^{-1} and B is replaced with $p_{0,1}(H)$; see Proposition 2. The motivation is that if $SAS^{-1} \approx UB$ then $A \approx S^{-1}UBS$. In fact, $\|SXS^{-1}\|_F \leq \kappa(S)\|X\|_F$ for any X.

In general, using a single random initialization provides a poor solution compared to the two previously proposed initializations. However, we have developed a simple multi-start heuristic that works as follows. Given a total allotted time $t_{\max}$ to the algorithm, we spend $t_{\max}/2$ generating 100 random initializations and refine them using Algorithm 1 (which therefore runs for only $t_{\max}/200$ for each random initialization). Then, we keep the best solution obtained among the 100 random initializations and refine it for $t_{\max}/2$.

4. Numerical experiments

In this section, we compare our algorithm, which we refer to as FGM, with the only other known method for solving (2), namely the successive convex approximation approach [1], kindly made available to us by François-Xavier Orban de Xivry, that we refer to as SuccConv.

For both methods, we will use the standard and the LMI-based initializations. FGM initialized with the standard (resp. LMI-based) initialization is denoted Stand-FGM (resp. LMI-FGM), and similarly for SuccConv. We will use the multi-start heuristic only for FGM, which we will refer to as mRand-FGM, because it is not well suited for SuccConv that required much more time per iteration, and more iterations to converge.

Our code is available from https://sites.google.com/site/nicolasgillis/ and the numerical examples presented below can be directly run from this online code (there are also more numerical results in particular on randomly generated matrices). All tests are preformed using Matlab R2015a on a laptop Intel CORE i5-3210M CPU @2.5 GHz 6Go RAM. FGM runs in $O(n^3)$ operations per iteration, including projections on the set of positive semidefinite matrices, orthogonal matrices, and inversion of the matrix S and all necessary matrix-matrix products. Hence FGM can be applied on a standard laptop with n up to a thousand. SuccConv is a second-order method and cannot be applied to matrices with n much larger than 50 (one iteration of the algorithm requires about 30 seconds for $n = 50$).

4.1. Examples from [8]

We start with some examples from the paper [8]. In [8], authors study the nearest stable matrix problem (2) with component-wise nonnegativity constraint on the stable matrix X to be found. For these small examples, we set the time limit of the different algorithms to 30 seconds.

4.1.1. Example 2: 3-by-3 matrix

We consider

$$A = \begin{pmatrix} 0.6 & 0.4 & 0.1 \\ 0.5 & 0.5 & 0.3 \\ 0.1 & 0.1 & 0.7 \end{pmatrix}$$

with $\rho(A) = 1.096$, for which [8] shows that the nearest stable nonnegative matrix is

$$X = \begin{pmatrix} 0.5640 & 0.3599 & 0.0850 \\ 0.4716 & 0.4684 & 0.2881 \\ 0.0643 & 0.0602 & 0.6851 \end{pmatrix}.$$

FGM and SuccConv for any initialization strategy converge to the same solution. This is because, as shown in [8] for nonnegative matrices, if a local minimum to problem (2) is component wise positive, then it is a global minimizer.

4.1.2. Example 3: scaled all-one matrix

We now consider the matrix $A = \alpha E$ where $\alpha \geq 0$ and E is the matrix whose entries are all equal to one. Note that the matrix is of rank-one, and stable for $\alpha \leq \frac{1}{n}$. Authors [8] show that for any $\alpha \in \left[\frac{1}{n}, \frac{2}{n}\right]$, the nearest stable matrix is given by $\frac{1}{n}E$. We run FGM and SuccConv on this example for $n = 10$ and $\alpha = \frac{2}{n}$ and, as for the previous example, they converge to the solution $\frac{1}{n}E$ for any of the initializations – note that LMI-FGM is initialized with the optimal solution since $\frac{1}{n}E = \frac{A}{\rho(A)}$. (The same observation holds for $n = 20$.) For $\alpha > \frac{2}{n}$, $\frac{1}{n}E$ is not optimal anymore, and the optimal solution is not positive anymore [8]. For example, for $n = 2$ and any $\alpha > 1$, there are two nonnegative optimal solutions given by $\begin{pmatrix} 1 & \alpha \\ 0 & 1 \end{pmatrix}$, and $\begin{pmatrix} 1 & 0 \\ \alpha & 1 \end{pmatrix}$, with error $\|A - X\|_F^2 = 6$. Taking $\alpha = 3$, LMI-FGM is not able to recover an optimal solution: it recovers $\frac{1}{n}E$ with error 9 since it is initialized with this solution and it is a stationary point of the problem [8]. Stand-FGM recovers a slightly better solution with error 8. Stand-SuccConv and LMI-SuccConv obtain better but non-optimal solutions with error 6.27 and 6.24, respectively. Only mRand-FGM is able to recover one of the above optimal solutions: rather surprisingly, it seems the unconstrained solution coincides with the nonnegative one (although we do not have a proof for this fact) – as we will see in the next examples, this is usually not the case. For $n = 3$ and $\alpha = 2$, the algorithms converge to different stationary points. The triangular matrix with ones on the diagonal and α above or below has error 15. As before, LMI-FGM converges to the stationary point $\frac{1}{n}E$ with error 25, and Stand-FGM to a better solution with error 17. Stand-SuccConv and LMI-SuccConv converge to two rather different solutions with errors 15.2548 and 15.2558 respectively, while mRand-FGM provides the suboptimal solution

$$X = \begin{pmatrix} 0.9969 & 1.4010 & 0.7688 \\ 0.5544 & 0.9878 & -0.6507 \\ 1.2476 & 2.6740 & 1.0112 \end{pmatrix}$$

with error 15.02. This illustrates the fact that (2) is a difficult problem with many local minimizers.

4.1.3. Example in Section 4.4

We consider

$$A = \begin{pmatrix} 0.7 & 0.2 & 0.1 & 0.5 & 1 \\ 0.3 & 0.6 & 0.2 & 0.8 & 0.3 \\ 0.5 & 0.7 & 0.9 & 1 & 0.5 \\ 0.1 & 0.1 & 0.3 & 0.8 & 0.3 \\ 0.8 & 0.2 & 0.9 & 0.3 & 0.2 \end{pmatrix}$$

with $\rho(A) = 2.4$. The nonnegative solution provided by the authors with their algorithm is

$$X_+ = \begin{pmatrix} 0.3796 & 0.1797 & 0 & 0.5 & 0.7343 \\ 0 & 0.5791 & 0.0069 & 0.8 & 0.0274 \\ 0.0580 & 0.6719 & 0.6403 & 1 & 0.1334 \\ 0 & 0 & 0 & 0.8 & 0 \\ 0.4204 & 0.1759 & 0.6770 & 0.3 & 0 \end{pmatrix}$$

with error $\|A - X_+\|_F^2 = 1.2181$ (which is not necessarily optimal). Stand-SuccConv and LMI-SuccConv converge to the same solution

$$\begin{pmatrix} 0.5999 & 0.1317 & -0.0882 & 0.5337 & 0.8834 \\ 0.2582 & 0.5864 & 0.0967 & 0.8512 & 0.2089 \\ 0.4469 & 0.6904 & 0.8242 & 1.0419 & 0.4257 \\ -0.0828 & -0.1243 & -0.2132 & 0.8209 & 0.0595 \\ 0.7076 & 0.1273 & 0.7126 & 0.3255 & 0.0923 \end{pmatrix}$$

with error 0.5709. Stand-FGM, LMI-FGM and mRand-FGM converge to three different solutions with errors 0.6053, 0.5808 and 0.5759, respectively.

4.2. Grcar matrices

Grcar matrices of order k are a banded Toeplitz matrix with its subdiagonal set to -1 and both its main and k superdiagonals set to 1. For example, when $n = 5$ and $k = 3$, we have the following Grcar matrix

Table 1
Comparison of the algorithms for Grcar matrices A of order 3: final relative error in percent, that is, $\frac{\|A-X\|_F}{\|A\|_F}$ where X is the stable approximation of A, and, in brackets, the number of iterations performed. The best solution is indicated in bold. The second column reports the relative error in percent of the nearest nonnegative stable matrix (that is, $\sqrt{n-1}/\|A\|_F$).

n	$\max(A,0)$	Stand-FGM	LMI-FGM	mRand-FGM	Stand-SuccConv	LMI-SuccConv
5	47.14	**31.23** (5078)	**31.23** (5599)	31.24 (13029)	31.63 (9092)	31.64 (9104)
10	45.75	**30.02** (112539)	33.08 (115262)	33.18 (55136)	30.88 (5188)	31.33 (5163)
20	45.20	41.64 (49225)	45.34 (45417)	46.51 (24539)	40.07 (419)	**39.41** (421)
50	44.91	**53.25** (34054)	55.98 (35473)	49.70 (16596)	60.26 (6)	54.28 (6)

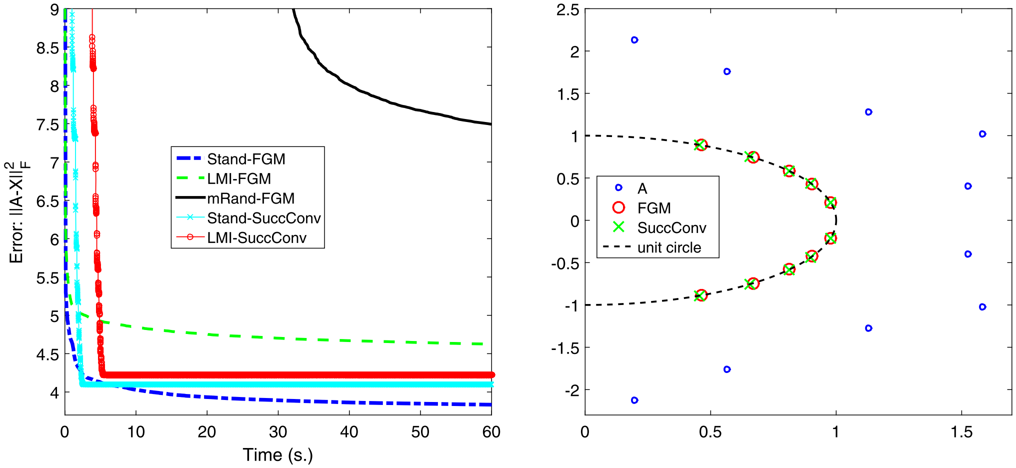

Fig. 1. (Left) Evolution of the error $\|A-X\|_F^2$ for the different algorithms for the Grcar matrix of dimension 10 and order 3. (Note that mRand-FGM only starts around 30 seconds as the multi-start heuristic spend half the time identifying the best solution among 100 randomly generated matrices.) (Right) Location of the eigenvalues of A and of the solutions obtained by FGM and SuccConv with the standard initialization.

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ -1 & 1 & 1 & 1 & 0 \\ 0 & -1 & 1 & 1 & 1 \\ 0 & 0 & -1 & 1 & 1 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix}.$$

Grcar matrices have all their eigenvalues outside the unit ball. Notice that the nearest nonnegative stable matrix is given by $\max(A,0)$ with error $\|A-\max(A,0)\|_F^2 = n-1$. Table 1 reports the results for $k = 3$ and $n = 5, 10, 20, 50$ with time limit of $t_{\max} = 30, 60, 120, 300$ seconds, respectively. We observe that Stand-FGM performs the best for $n = 5, 10, 50$ and LMI-SuccConv for $n = 20$. In most cases, the algorithms initialized with different initial points converge to different stationary points.

Fig. 1 shows the evolution of the objective function for the different algorithms for $n = 10$ (on the left), and the location of the eigenvalues of A, and of the solutions of Stand-FGM and Stand-SuccConv, the best solution found by the two algorithms (on the right). Although the eigenvalues of the solutions X_{fgm} of Stand-FGM and X_{sc} of

Stand-SuccConv are close to one another, they actually correspond to very different matrices, since $\frac{\|X_{\text{fgm}}-X_{\text{sc}}\|_F}{\|A\|_F} = 22.1\%$.

5. Conclusion

In this paper, we have proposed a new characterization of the set of stable matrices in the discrete-time case: We have shown that a matrix A is stable if and only if it admits a SUB form, that is, if there exists $S \succ 0$, U orthogonal and $B \succeq 0$ with $\|B\| \leq 1$ such that

$$A \;=\; S^{-1}UBS.$$

We have then used this characterization to provide a new algorithmic framework for the nearest stable matrix problem, that is, given an unstable matrix A, find the nearest stable matrix X. In fact, the SUB form is particularly useful as it is easy to project onto this set of matrices. We showed on several examples that our proposed algorithm that uses a fast gradient method (FGM) competes favorably with the method from [1]. In fact, in most cases, it provides better solutions while converging much faster.

Further research on the nearest stable matrix problem include the design of (1) other algorithms, (2) other initializations strategies, and (3) other heuristics to identify good solutions. Further research also includes the use of the SUB form in defining the structure of linear port-Hamiltonian systems at the discrete level analogous to the continuous-time linear port-Hamiltonian systems, see, e.g., [23–25], and to obtain the counterparts of the results in [26,27] for the discrete-time case.

Conflict of interest statement

There is no conflict of interest.

Acknowledgements

The authors would like to thank the reviewers for their insightful comments which helped improve the paper significantly.

Appendix A. Gradient with respect to *S*

The standard inner product on $\mathbb{R}^{n,n}$ is defined by $\langle A|B\rangle := \operatorname{tr}(A^TB) = \sum_{i,j} a_{ij}b_{ij}$. The associated norm is the Frobenius norm, $\|A\|_F = \sqrt{\langle A|A\rangle} = \sqrt{\sum_{i,j} a_{ij}^2}$. The relations $(AB)^T = B^TA^T$ and $\operatorname{tr}(AB) = \operatorname{tr}(BA)$ imply that

$$\langle A|BC\rangle = \langle B^TA|C\rangle = \langle AC^T|B\rangle. \tag{A.1}$$

Let $\mathcal{D}$ be a nonempty open subset of $\mathbb{R}^{n\times n}$. A matrix $G \in \mathbb{R}^{n,n}$ is said to be the gradient of a function $\mathcal{D} \ni S \mapsto f(S) \in \mathbb{R}$ at $S_0 \in \mathcal{D}$ if

$$\frac{d}{dt} f(S(t))|_{t=0} = \langle G, \dot{S}(0)\rangle \tag{A.2}$$

for all differentiable curves $\mathbb{R} \ni t \mapsto S(t) \in \mathcal{D}$ with $S(0) = S_0$ and derivative $\dot{S}(t)$. It easily seen that there is at most one matrix G with this property. Notation: $G = \nabla f(S_0)$. In the derivation below we omit the argument t and the index 0. Furthermore we simply write $\dot{f}$ for the left hand side of (A.2).

For fixed square matrices A, C we are going to determine the gradient of the function

$$f(S) = \|A - S^{-1}CS\|_F^2 = \langle R - A | R - A\rangle,$$

where $R := S^{-1}CS$. The derivative of R along a differentiable curve is

$$\dot{R} = S^{-1}C\dot{S} - (S^{-1}\dot{S}S^{-1})CS = R\,S^{-1}\dot{S} - S^{-1}\dot{S}\,R,$$

where the left equation follows from the product rule and fact that the derivative of the function $S \mapsto S^{-1}$ along a differentiable curve is $-S^{-1}\dot{S}S^{-1}$ (this is obtained by differentiating the relation $S^{-1}S = I$). Now, the derivative of f along a differentiable curve can be calculated as

$$\begin{aligned}
\dot{f} &= \langle \dot{R} | R - A\rangle + \langle R - A | \,\dot{R}\,\rangle \\
&= 2\,\langle R - A \,|\, \dot{R}\,\rangle \\
&= 2\,\langle R - A \,|\, R\,S^{-1}\dot{S} - S^{-1}\dot{S}\,R\,\rangle \\
&= 2\,\langle (R\,S^{-1})^T(R - A) - S^{-T}(R - A)R^T \,|\, \dot{S}\,\rangle \\
&\qquad\qquad\qquad\qquad\qquad\qquad \text{(by (A.1))} \\
&= 2\,\langle S^{-T}[R^T(R - A) - (R - A)R^T] \,|\, \dot{S}\,\rangle.
\end{aligned}$$

Thus, the gradient of f at S is

$$\nabla f(S) = 2\,S^{-T}[R^T(R - A) - (R - A)R^T].$$

References

[1] F.-X. Orbandexivry, Y. Nesterov, P. Van Dooren, Nearest stable system using successive convex approximations, Automatica 49 (5) (2013) 1195–1203.

[2] N. Gillis, P. Sharma, On computing the distance to stability for matrices using linear dissipative Hamiltonian systems, Automatica 85 (2017) 113–121.

[3] J.T. Anderson, Distance to the nearest stable Metzler matrix, in: 2017 IEEE 56th Annual Conference on Decision and Control (CDC), 2017, pp. 6567–6572.

[4] N. Gillis, V. Mehrmann, P. Sharma, Computing nearest stable matrix pairs, Numer. Linear Algebra Appl. (2018) e2153, https://doi.org/10.1002/nla.2153.

[5] N. Gillis, P. Sharma, Finding the nearest positive-real system, SIAM J. Numer. Anal. 56 (2) (2018) 1022–1047.

[6] N. Guglielmi, C. Lubich, Matrix stabilization using differential equations, SIAM J. Numer. Anal. 55 (6) (2017) 3097–3119.

[7] Y. Nesterov, V.Y. Protasov, Computing closest stable non-negative matrices, URL http://www.optimization-online.org/DB_HTML/2017/08/6178.html.

[8] N. Guglielmi, V. Protasov, On the closest stable/unstable nonnegative matrix and related stability radii, arXiv:1802.03054.

[9] Y. Genin, R. Ştefan, P. Van Dooren, Real and complex stability radii of polynomial matrices, Linear Algebra Appl. 351–352 (2002) 381–410.

[10] D. Hinrichsen, A. Pritchard, Real and complex stability radii: a survey, in: D. Hinrichsen, B. Mårtensson (Eds.), Control of Uncertain Systems, in: Progress in Systems and Control Theory, vol. 6, Birkhäuser, Boston, MA, 1990, pp. 119–162.

[11] D. Hinrichsen, N.K. Son, Stability radii of positive discrete-time systems under affine parameter perturbations, Internat. J. Robust Nonlinear Control 8 (13) (1998) 1169–1188.

[12] D. Hinrichsen, N.K. Son, P.H.A. Ngoc, Stability radii of higher order positive difference systems, Systems Control Lett. 49 (5) (2003) 377–388.

[13] P.H.A. Ngoc, N.K. Son, Stability radii of positive linear difference equations under affine parameter perturbations, Appl. Math. Comput. 134 (2) (2003) 577–594.

[14] P.H.A. Ngoc, N.K. Son, Stability radii of positive linear functional differential equations under multi-perturbations, SIAM J. Control Optim. 43 (6) (2005) 2278–2295.

[15] F. Gantmacher, The Theory of Matrices I, Chelsea Publishing Company, New York, NY, 1959.

[16] Y. Nesterov, Introductory Lectures on Convex Optimization: A Basic Course, vol. 87, Springer Science & Business Media, 2004.

[17] S. Ghadimi, G. Lan, Accelerated gradient methods for nonconvex nonlinear and stochastic programming, Math. Program. 156 (1–2) (2016) 59–99.

[18] N. Agarwal, Z. Allen-Zhu, B. Bullins, E. Hazan, T. Ma, Finding approximate local minima faster than gradient descent, in: Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, ACM, 2017, pp. 1195–1199.

[19] M. O'Neill, S.J. Wright, Behavior of accelerated gradient methods near critical points of nonconvex problems, arXiv preprint, arXiv:1706.07993.

[20] P.-A. Absil, J. Malick, Projection-like retractions on matrix manifolds, SIAM J. Optim. 22 (1) (2012) 135–158.

[21] N. Higham, Computing a nearest symmetric positive semidefinite matrix, Linear Algebra Appl. 103 (1988) 103–118.

[22] R. Horn, C. Johnson, Matrix Analysis, Cambridge University Press, Cambridge, 1985.

[23] G. Golo, A. van der Schaft, P. Breedveld, B. Maschke, Hamiltonian formulation of bond graphs, in: A.R.R. Johansson (Ed.), Nonlinear and Hybrid Systems in Automotive Control, Springer-Verlag, Heidelberg, Germany, 2003, pp. 351–372.

[24] A. van der Schaft, Port-Hamiltonian systems: an introductory survey, in: J.V.M. Sanz-Sole, J. Verdura (Eds.), Proc. of the International Congress of Mathematicians, Madrid, Spain, in: Invited Lectures, vol. III, 2006, pp. 1339–1365.

[25] A. van der Schaft, B. Maschke, Port-Hamiltonian systems on graphs, SIAM J. Control Optim. 51 (2013) 906–937.

[26] C. Mehl, V. Mehrmann, P. Sharma, Stability radii for linear Hamiltonian systems with dissipation under structure-preserving perturbations, SIAM J. Matrix Anal. Appl. 37 (4) (2016) 1625–1654.

[27] C. Mehl, V. Mehrmann, P. Sharma, Stability radii for real linear Hamiltonian systems with perturbed dissipation, BIT Numer. Math. 57 (3) (2017) 811–843.