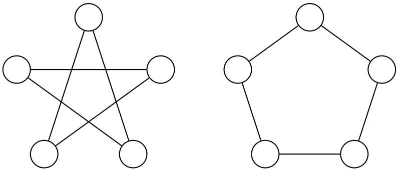
TransProof: Computer assisted graph transformations

Gauvain Devillez
Joint work with P. Hauweele, A. Hertz and H. Mélot

UMONS
Computer Science Department
Algorithm Lab

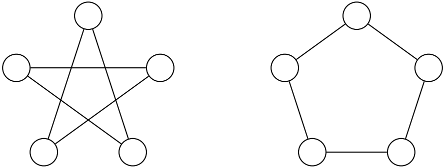
Isomorphism

- We work on simple undirected graphs.
- For a graph $G = (V, E)$, we denote
  - its order $|V|$ by n,
  - its size $|E|$ by m.
- We consider two graphs as equivalent if they are isomorphic.



Graph invariants

- A graph invariant is a function on graphs constant by isomorphism.
- Examples : average distance ($\overline{d}$), diameter (D), chromatic number (χ), planarity, ...



$$\overline{d} = 1.5, D = 2, \chi = 3, \textit{planarity} = \textit{true}, \ldots$$

This talk

- Context : Computer-assisted Proofs in Extremal Graph Theory.
- Objective of this talk :
  - Presentation of TransProof, a module of PHOEG
  - use of an illustrative problem (EMD).
- Remarks :
  - TransProof is currently a prototype
  - The problem about EMD is still open

Conjectures

- Extremal Graph Theory tries to define bounds on these invariants with respect to some constraints.
- The constraints are usually of two forms :
  - restricting to a class of graphs,
  - fixing or restricting some other invariant.
- Since tight bounds are even better, we search for graphs that realize these bounds : the extremal graphs.

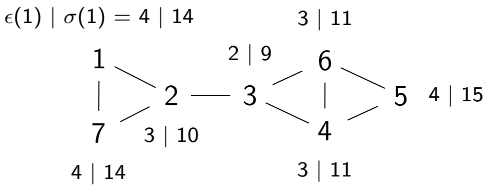
Distances

Definition

The eccentricity *of a vertex u ($\epsilon(u)$) is the maximal distance between u and any other vertex.*

Definition

The transmission *of a vertex u ($\sigma(u)$) is the sum of the distances between u and all the other vertices.*



Average eccentricity - average distance

- We denote by $(\overline{\epsilon} - \overline{d})(G)$ the difference between the average eccentricity and the average distance (EMD).

$$(\overline{\epsilon} - \overline{d})(G) = \frac{\sum\limits_{v \in V(G)} \epsilon(v)}{n} - \frac{\sum\limits_{v \in V(G)} \sigma(v)}{n \cdot (n-1)}$$

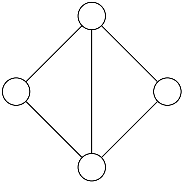
$$= \frac{1}{n \cdot (n-1)} \cdot \left(\sum_{v \in V(G)} (n-1) \cdot \epsilon(v) - \sigma(v) \right)$$

Conjecture (Aouchiche, 2006)

Let $\mathcal{G}$ be the set of connected graphs of order n,

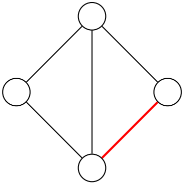
$$\forall G \in \mathcal{G}, (\overline{\epsilon} - \overline{d})(G) \leq (\overline{\epsilon} - \overline{d})(P_n)$$

Proofs by Transformation



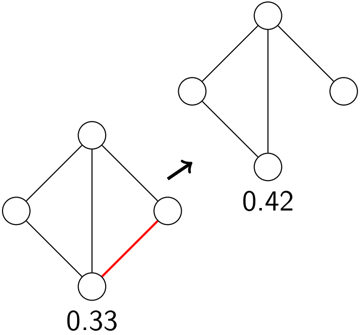
0.33

Proofs by Transformation

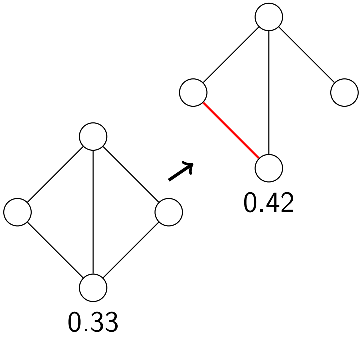


0.33

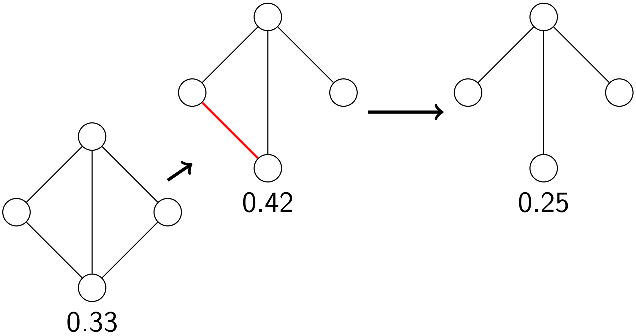
Proofs by Transformation



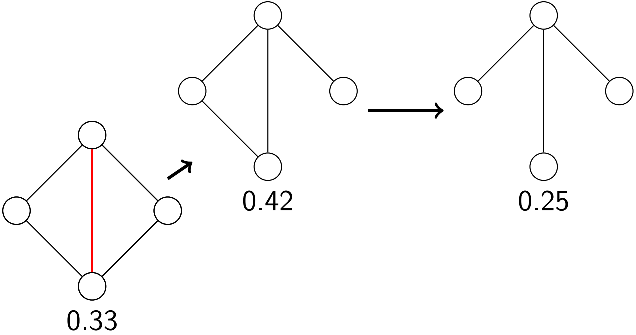
Proofs by Transformation



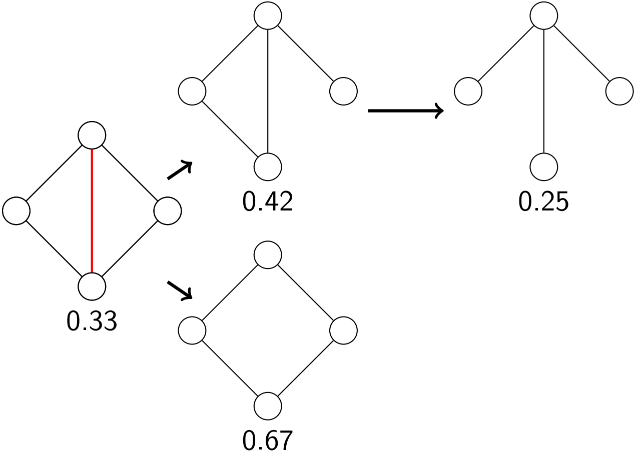
Proofs by Transformation



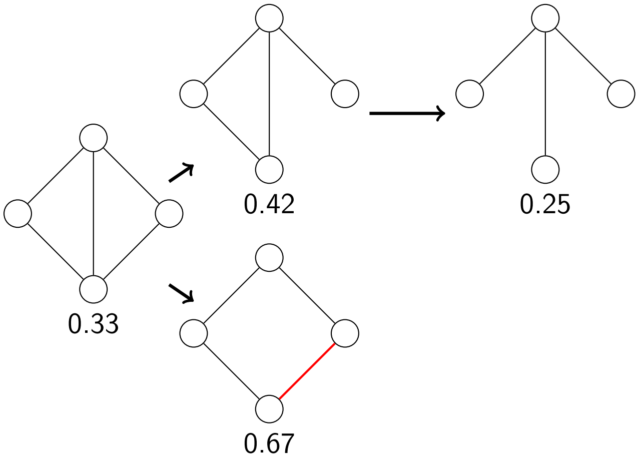
Proofs by Transformation



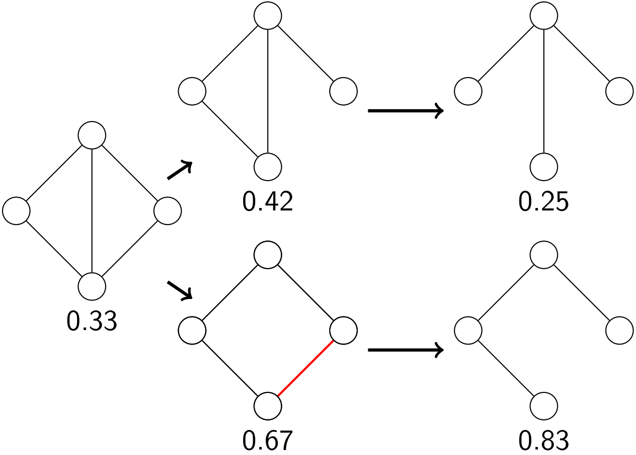
Proofs by Transformation



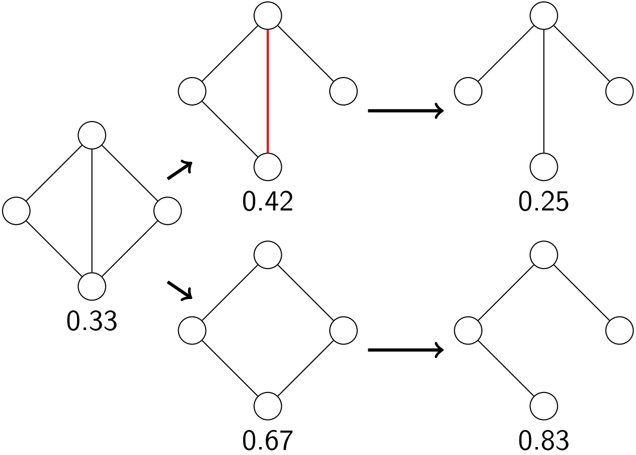
Proofs by Transformation



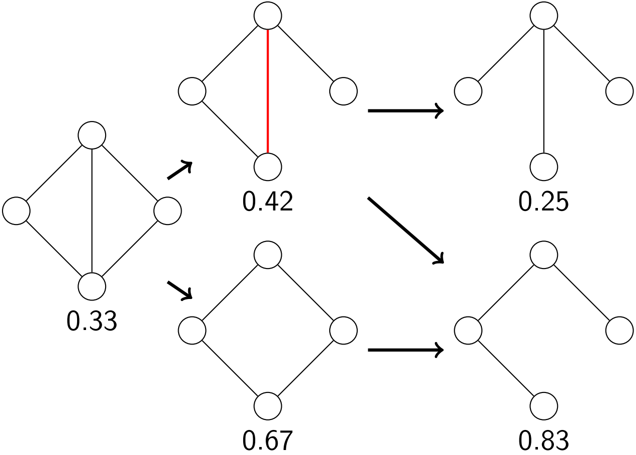
Proofs by Transformation



Proofs by Transformation

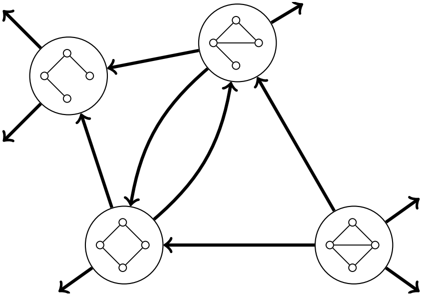


Proofs by Transformation



Metagraph of Transformations

- The idea of a proof by transformation can be represented as a directed multi graph.
- We call this graph the metagraph of transformations.
- It can be used to study transformations but also to help define proofs by transformation.



Graph Database

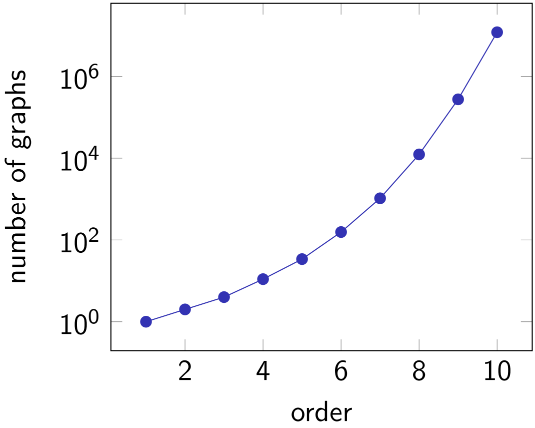
- TransProof can compute the metagraph and store it in a graph database.
- This database allows for easy queries about the transformations.

```
MATCH (n)-[r:edgeRemove]->(m) where n.invariant <
m.invariant return n,r,m;
```

```
MATCH (n id:42)-[r:edgeRemove*..]->(m id:65536) return r;
```

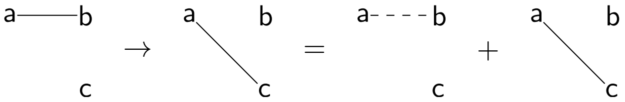
Limitations

- The number of graphs of some order n increases exponentially.
- The number of possible transformations is even bigger.

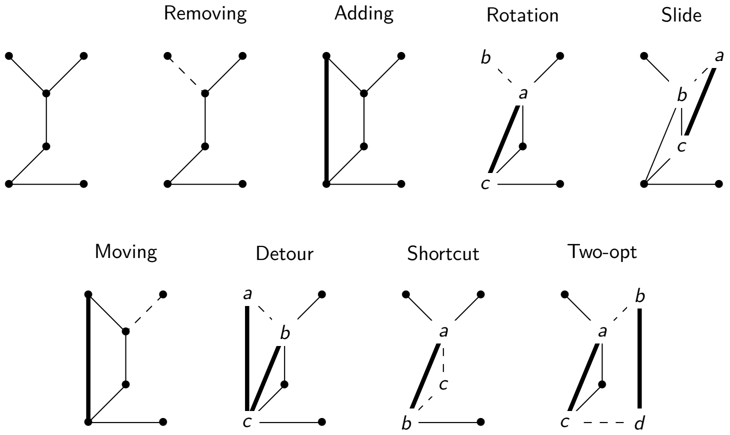


Basis of Transformations

- Most transformations can be described as combination of simpler transformations.
- Why not use a subset of simple transformation to generate more complex ones?



Commonly used transformations



Specific language

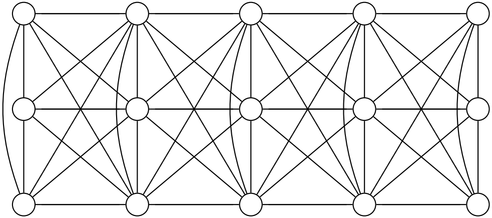
- To help with the writing of queries, we could define a specific language.
- This language could also provide some optimization.

```
MATCH (n)-[r:rotation]->(m) where
not edge(n.sig,r.b,r.c) with n,r,m
match (m)-[s:addEdge]->(o)
where s.a = r.b && and s.b = r.c
return n,r,m,s,o;
```

```
rotation(a,b,c):
  !edge(b,c);
addEdge(b,c);
```

Still Incomplete

- This language is not yet complete.
- It cannot describe transformations with unfixed number of edges
- As an example, consider a set of cliques joined to form a path.



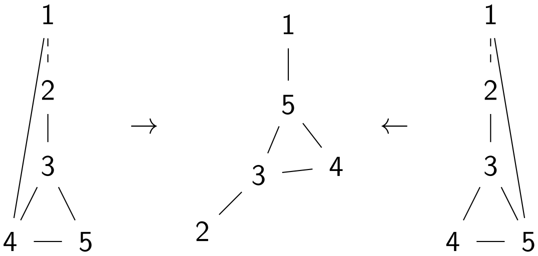
Some Numbers

- The problem is that only considering simple transformations still produces a huge amount.

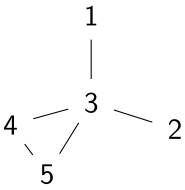
order	# graphs	# arcs
1	1	0
2	2	4
3	4	36
4	11	362
5	34	3 188
6	156	34 376
7	1 044	468 936
8	12 346	10 143 824
9	274 668	380 814 904

Symmetries

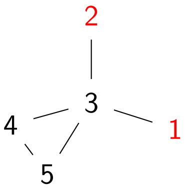
- Some of these transformations are symmetries.
- They actually come from automorphisms.



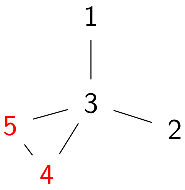
Automorphisms



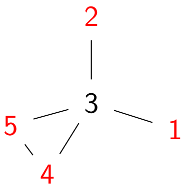
(1, 2, 3, 4, 5)



(2, 1, 3, 4, 5)



(1, 2, 3, 5, 4)

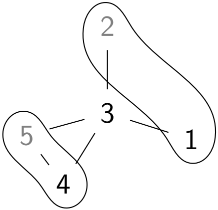


(2, 1, 3, 5, 4)

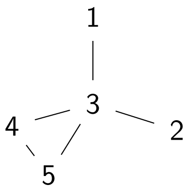
(1 2)(3)(4 5)

Using automorphisms

- To avoid these automorphisms, we have to :
  - Compute the orbits of the automorphism group.
  - Order them based on the lowest index.
  - Simply take one vertex from each.
  - When a vertex is fixed, the orbits can change.

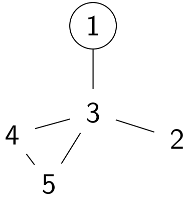


Avoiding Symmetries



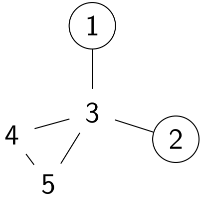
(1 2)(3)(4 5)

Avoiding Symmetries



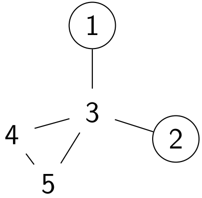
(1)(2)(3)(4 5)

Avoiding Symmetries

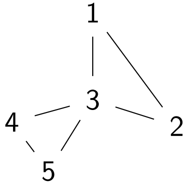


(1)(2)(3)(4 5)

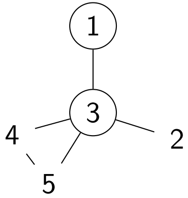
Avoiding Symmetries



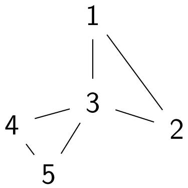
(1)(2)(3)(4 5)



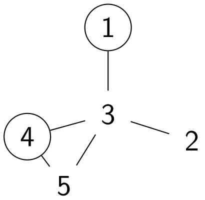
Avoiding Symmetries



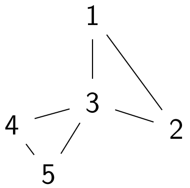
(1)(2)(3)(4 5)



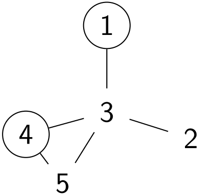
Avoiding Symmetries



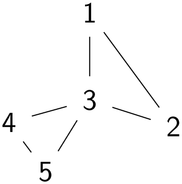
(1)(2)(3)(4 5)

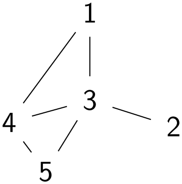


Avoiding Symmetries

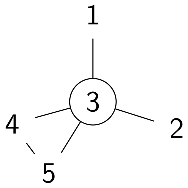


(1)(2)(3)(4 5)

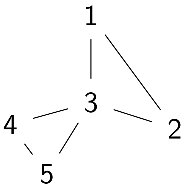


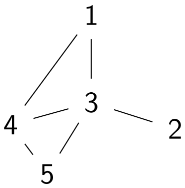


Avoiding Symmetries

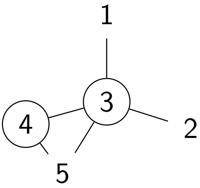


(1 2)(3)(4 5)

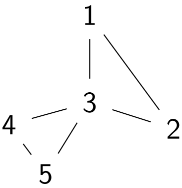


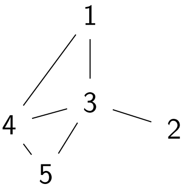


Avoiding Symmetries

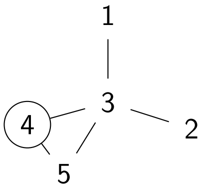


$(1\ 2)(3)(4\ 5)$

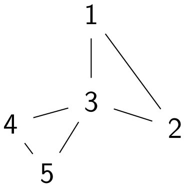


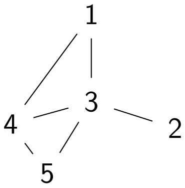


Avoiding Symmetries

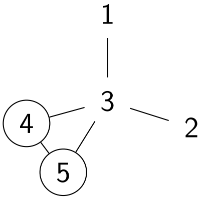


(1 2)(3)(4)(5)

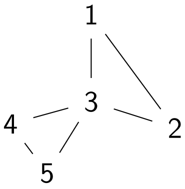


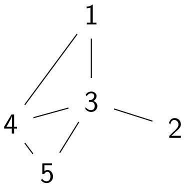


Avoiding Symmetries

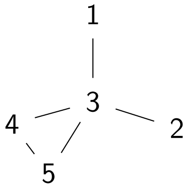


(1 2)(3)(4)(5)

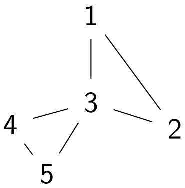


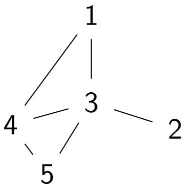


Avoiding Symmetries



(1 2)(3)(4 5)





Effect of symmetries

- Without symmetries, we greatly reduce the size of the metagraph.

order	# graphs	# arcs	now
1	1	0	0
2	2	4	2
3	4	36	12
4	11	362	78
5	34	3 188	617
6	156	34 376	6 717
7	1 044	468 936	108 022
8	12 346	10 143 824	2 776 023
9	274 668	380 814 904	119 430 801

Using TransProof

- Now, we can try to use this metagraph for the EMD conjecture.

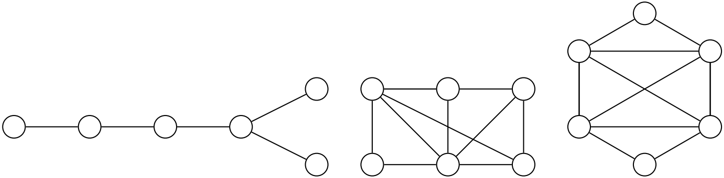
Conjecture (EMD)

Let $\mathcal{G}$ be the set of connected graphs of order n,

$$\forall G \in \mathcal{G}, (\overline{\epsilon} - \overline{d})(G) \leq (\overline{\epsilon} - \overline{d})(P_n)$$

Example of use of TransProof

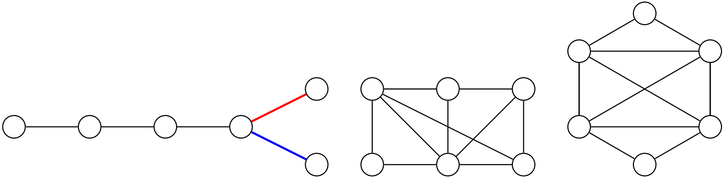
- For our problem, we could try removing an edge.



- Some problematic graphs appear because of some "symmetry".
- These graphs have (almost) twin nodes.

Example of use of TransProof

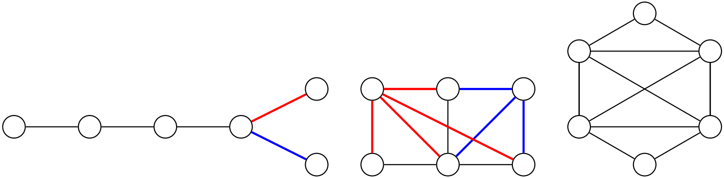
- For our problem, we could try removing an edge.



- Some problematic graphs appear because of some "symmetry".
- These graphs have (almost) twin nodes.

Example of use of TransProof

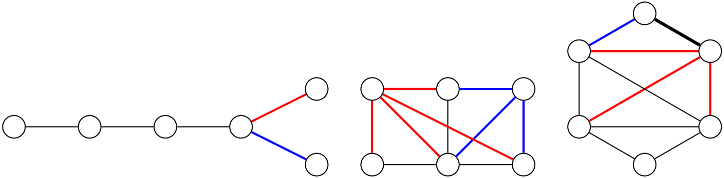
- For our problem, we could try removing an edge.



- Some problematic graphs appear because of some "symmetry".
- These graphs have (almost) twin nodes.

Example of use of TransProof

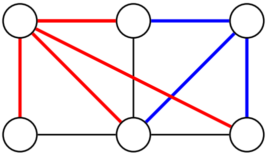
- For our problem, we could try removing an edge.



- Some problematic graphs appear because of some "symmetry".
- These graphs have (almost) twin nodes.

A more complex transformation

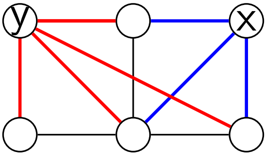
- Choose two vertices x and y such that y has at least the same neighbors as x.
- Choose a vertex z with maximal transmission.
- Make x a pending vertex to z.



$$(\overline{\epsilon} - \overline{d})(G) = 0.5$$

A more complex transformation

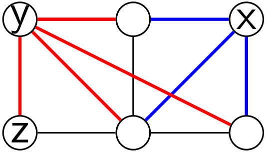
- Choose two vertices x and y such that y has at least the same neighbors as x.
- Choose a vertex z with maximal transmission.
- Make x a pending vertex to z.



$$(\overline{\epsilon} - \overline{d})(G) = 0.5$$

A more complex transformation

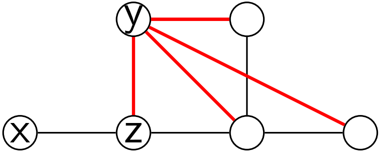
- Choose two vertices x and y such that y has at least the same neighbors as x.
- Choose a vertex z with maximal transmission.
- Make x a pending vertex to z.



$$(\overline{\epsilon} - \overline{d})(G) = 0.5$$

A more complex transformation

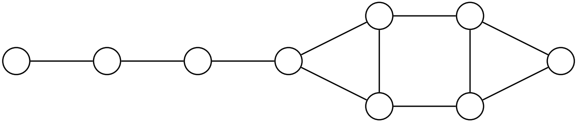
- Choose two vertices x and y such that y has at least the same neighbors as x.
- Choose a vertex z with maximal transmission.
- Make x a pending vertex to z.



$$(\bar{\epsilon} - \bar{d})(G) = 0.9$$

Remaining graphs

- There are still situations where this transformation does not apply.

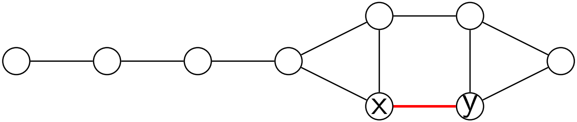


$$(\bar{\epsilon} - \bar{d})(G) = 2.1666\ldots$$

- Choose an edge xy and remove all vertices between x and all other vertices that are parts of a cycle.
- These two transformations are sufficient for all connected graphs up to order 10.

Remaining graphs

- There are still situations where this transformation does not apply.

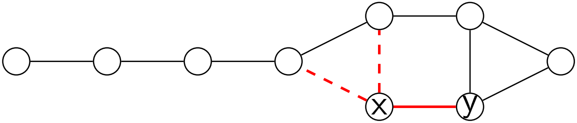


$$(\bar{\epsilon} - \overline{d})(G) = 2.1666\ldots$$

- Choose an edge xy and remove all vertices between x and all other vertices that are parts of a cycle.
- These two transformations are sufficient for all connected graphs up to order 10.

Remaining graphs

- There are still situations where this transformation does not apply.

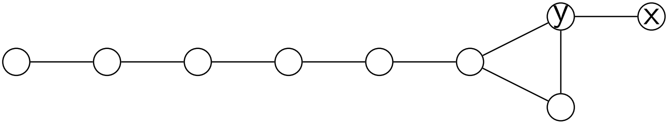


$$(\overline{\epsilon} - \overline{d})(G) = 2.1666\ldots$$

- Choose an edge xy and remove all vertices between x and all other vertices that are parts of a cycle.
- These two transformations are sufficient for all connected graphs up to order 10.

Remaining graphs

- There are still situations where this transformation does not apply.

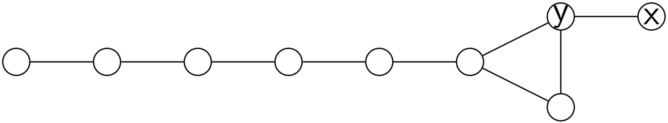


$$(\bar{\epsilon} - \overline{d})(G) = 2.1666\ldots$$

- Choose an edge xy and remove all vertices between x and all other vertices that are parts of a cycle.
- These two transformations are sufficient for all connected graphs up to order 10.

Remaining graphs

- There are still situations where this transformation does not apply.



$$(\bar{\epsilon} - \overline{d})(G) = 2.5555\ldots$$

- Choose an edge xy and remove all vertices between x and all other vertices that are parts of a cycle.
- These two transformations are sufficient for all connected graphs up to order 10.

Adding Constraints

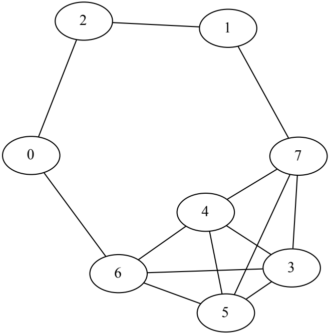
- While these transformations works on more than a million of graphs, there is no formal proof.
- There are still some cases where they do not increase the invariant.

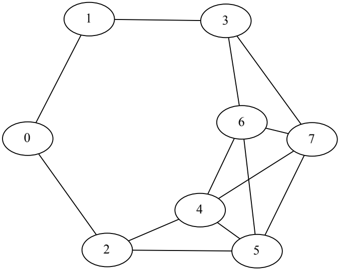
Problem

Among all connected graphs of order n and diameter D, what are the graphs maximizing the difference between the average eccentricity and average distance ?

New extremal graphs (when $2 \cdot D(G) \leq n$)

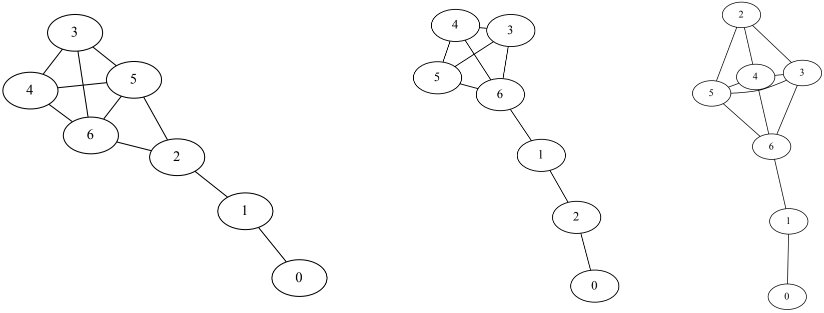
- When there is enough vertices, build a cycle of size $2 \cdot D(G)$.
- Add remaining nodes as twins to adjacent nodes of the cycle.





New extremal graphs (when $2 \cdot D(G) > n$)

- If there is not enough nodes, simply build a diametral path.
- Again, add remaining nodes as twins but to the node of index $p = \frac{2 \cdot d - n + 1}{4}$.



Notes

- If the diameter is 2, extremal graphs are complement of matchings.
- The cube is also extremal but not the square or other hypercubes.
- For a given n, the invariant strictly increases between extremal graphs when diameter increases.
- If $D(G) = n - 1$, the extremal graph is P_n.

Conclusion

- We are developing a system able to compute the metagraph
- It allows exploring the metagraph and testing proofs by transformations
- We use some ideas to tackle the problem of the amount of data.
- It could be used to speed up metaheuristics
- It is helpfull to define transformations.
- Sometimes, adding constraints can give some insight.

Conclusion

- We are developing a system able to compute the metagraph
- It allows exploring the metagraph and testing proofs by transformations
- We use some ideas to tackle the problem of the amount of data.
- It could be used to speed up metaheuristics
- It is helpfull to define transformations.
- Sometimes, adding constraints can give some insight.

Questions ?

Vertex index when $2 \cdot D(G) > n$

- Let G be a graph with n vertices and diameter d composed of a path $P = p_0, p_1, \ldots, p_d$ and $n - d$ vertices $(R = r_0, r_1, \ldots, r_{n-d-1})$ each twin to a vertex of the path.
- We want to find the vertex p_p which, if used as a twin to the vertices r_i, would maximise the invariant.
- We can suppose that vertices r_i form a clique (or we could increase the invariant).

$$(\overline{\epsilon} - \overline{d})(G) = \frac{1}{n \cdot (n-1)} \cdot \left(\sum_{v \in V(G)} (n-1) \cdot \epsilon(v) - \sigma(v) \right)$$

$$= \frac{1}{n \cdot (n-1)} \cdot \sum_{v \in V} W(v)$$

Vertex index when $2 \cdot D(G) > n$

- For the vertex of index p in the path :

$$W_P(p) = (n-1) \cdot \epsilon(p) - \sum_{u \in V} dpu$$

$$= (n-1) \cdot \max(d-p, p) - \left(\frac{(d-p) \cdot (d-p+1)}{2} + \frac{p \cdot (p+1)}{2} \right)$$

- For a twin node to the vertex of index p :

$$W_R(p) = (n-1) \cdot \max(d-p, p)$$

$$-2 \cdot \left(\frac{(d-p) \cdot (d-p+1)}{2} + \frac{p \cdot (p+1)}{2} \right) - 2 - (n-d-1)$$

$$= -2 \cdot p^2 + (2 \cdot d - n + 1) \cdot p - d^2 + (n-2) \cdot d$$

- This polynomial of degree 2 is maximal when $p = \frac{2 \cdot d - n + 1}{4}$