Enhancements to the Perfect Matching Approach for Graph Enumeration-based Engineering Challenges

IDETC2020-22774

Daniel R. Herber

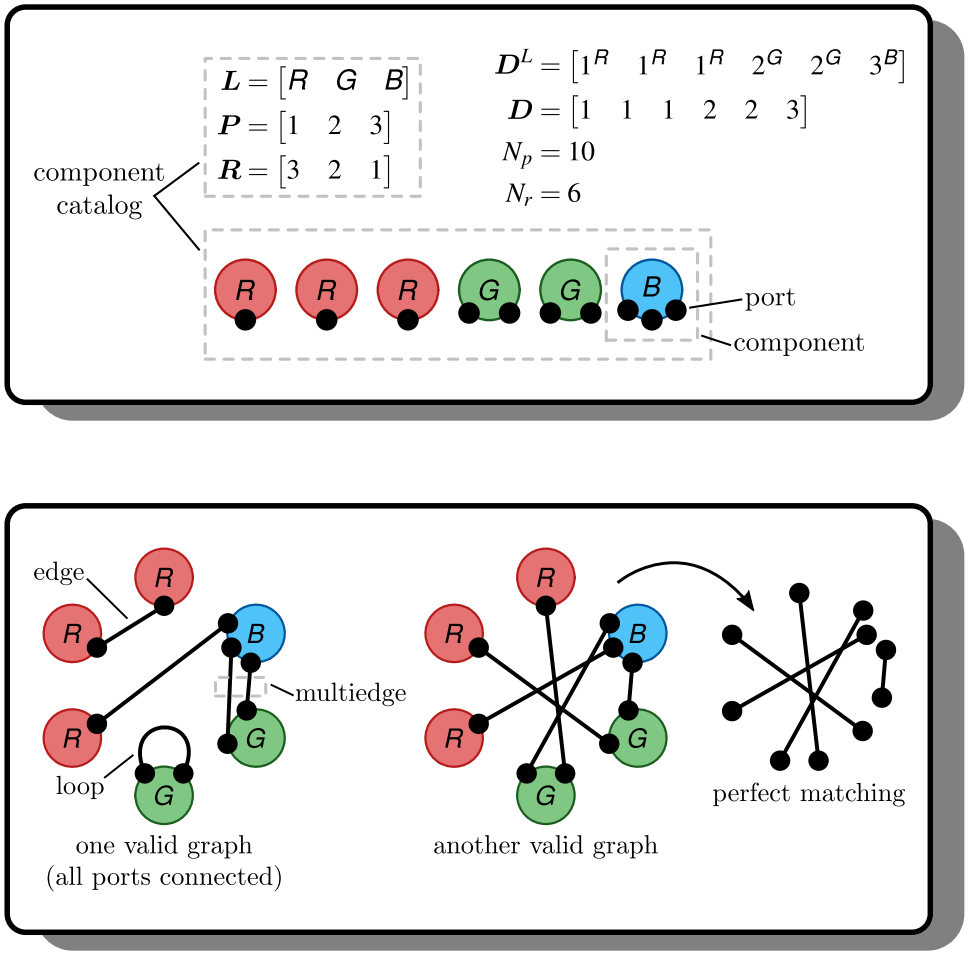
Colorado State University
Department of Systems Engineering
daniel.herber@colostate.edu

①

Introduction

Introduction

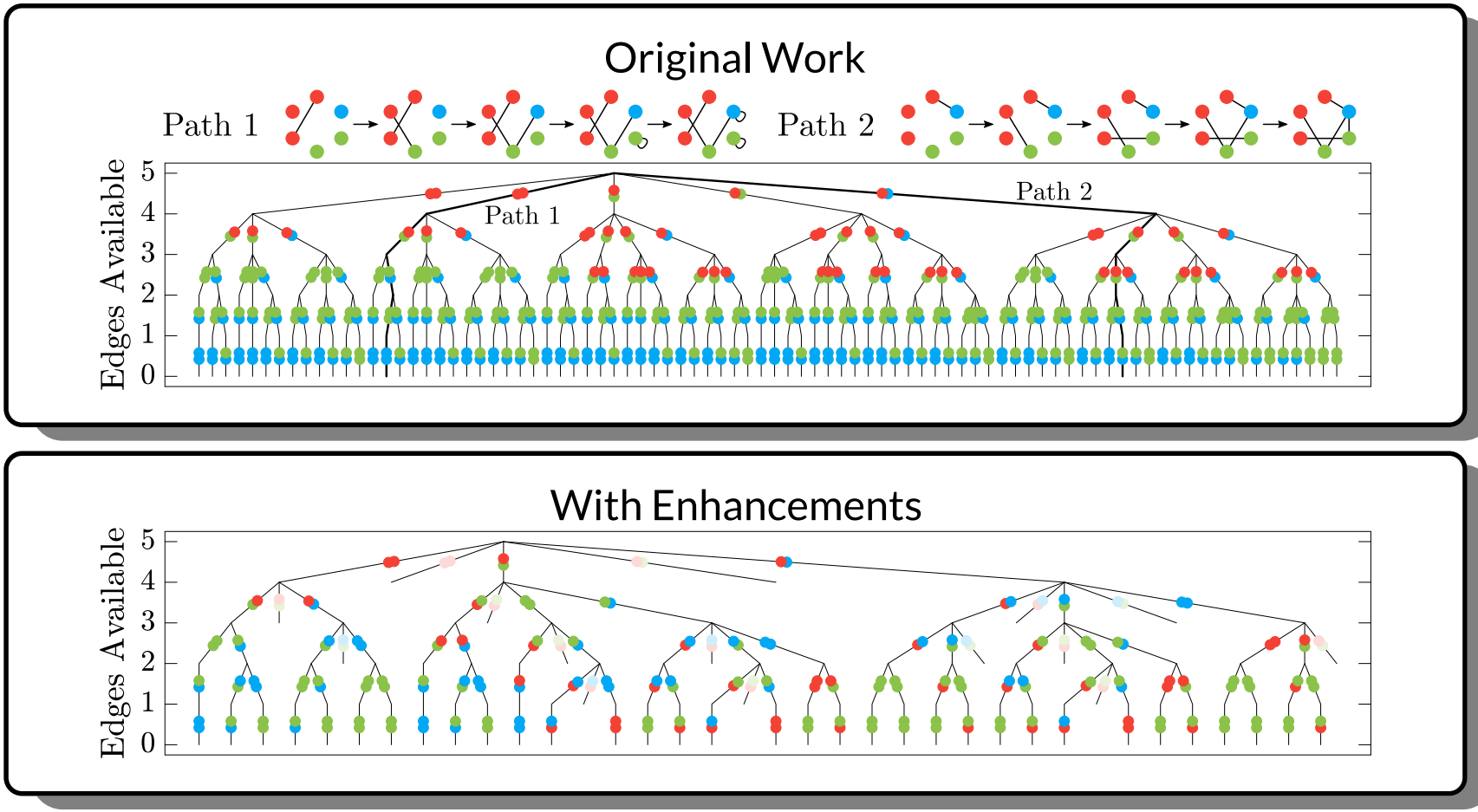
- **Graphs** can be used to represent many engineering systems and decisions because of their ability to capture **discrete compositional and relational information**
- Here, **improved methods are presented for effectively representing and generating all graphs** in a space defined by certain complex specifications
- Builds on previous work[1]
- Used in engineering design optimization of mechanical, electrical, and thermal systems[2]



[1] Herber, Guo, and Allison 2017 [2] Herber 2017; Herber, Allison, et al. 2020; Herber and Allison 2019; Peddada et al. 2019

Introduction (continued)

- Tree-like structure to the algorithms



Links to the code on GitHub:
github.com/danielrherber/pm-architectures-project
doi:10.5281/zenodo.3963751

②

Brief Overview of the Enhancements

Enhancement: Subcatalogs

- The **set of all subcatalogs** $\mathcal{C}$ of $(\boldsymbol{L}, \boldsymbol{P}, \boldsymbol{R})$ is all possible valid combinations
- With no additional constraints, the number of subcatalogs is:

$$|\mathcal{C}| = \prod_{i=1}^{|\boldsymbol{L}|} \left[\sum_{j=\underline{R}_i}^{\bar{R}_i} \binom{j + \bar{P}_i - \underline{P}_i}{j} \right] \tag{1}$$

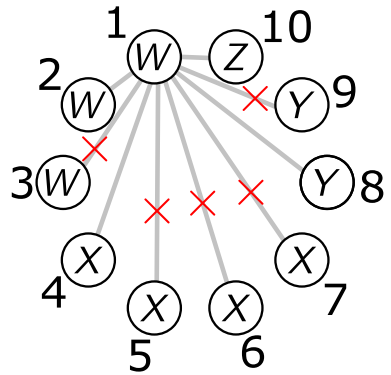
- Graph enumeration for each subcatalog is fully parallelizable
- We may also want to add **subcatalog constraints**
  - **Linear penalty constraints** where $\boldsymbol{\rho} \cdot \boldsymbol{R} \leq \alpha$ and $\boldsymbol{\rho}$ defines the penalty for each replicate. Some common penalty functions include a maximum system cost or mass.
  - Many other constraint forms defined including some based on well-known graph theory results[1]

[1] Choudum 1986; Brualdi and Ryser 1991

Enhancement: Enumeration of a Single Catalog

- Many enhancements for the **enumeration of a single catalog**
  - Alternative spanning tree traversals (DFS vs. BFS, sorting, touched vertex promotion)
  - Replicate ordering
  - Loops
  - Multiedges
  - Line-connectivity constraints
  - Connected saturated subgraphs
- **Capitalize on problem structure** and expand what classes of graphs can be efficiently enumerated
- Some inspired by techniques found in the literature[1]

Replicate ordering



[1] Faulon, Churchwell, and Visco 2003

→ Enhancement: Set-based Labeled Graph Isomorphism Checking

- The graph generation algorithms may produce graphs that are identical in the context of **labeled graph isomorphisms**
- Has been discussed in the context of engineering applications[1]
- Major improvements by using **graph invariants** to reduce isomorphism checking
  - Graph invariants are graph properties which are equivalent for isomorphic graphs (i.e., they are necessary conditions)[2]
  - **New graph invariant defined** combining L with the graph spectrum with perfect accuracy on the tested problems (but is generally limited by precision)

[1] Hartmann et al. 2018; Silvas et al. 2015; Bayrak, Ren, and Papalambros 2016; Wyatt, Wynn, and Clarkson 2014 [2] Diestel 2017

③

Graph Enumeration Examples

Case Studies from the Original Paper

- Major improvements to all the original case studies[1]
- With all the enhancements in the suspension case study:
  - 751× reduction # of candidate graphs
  - 1019× reduction # of isomorphism checks
  - **971× reduction computational cost**

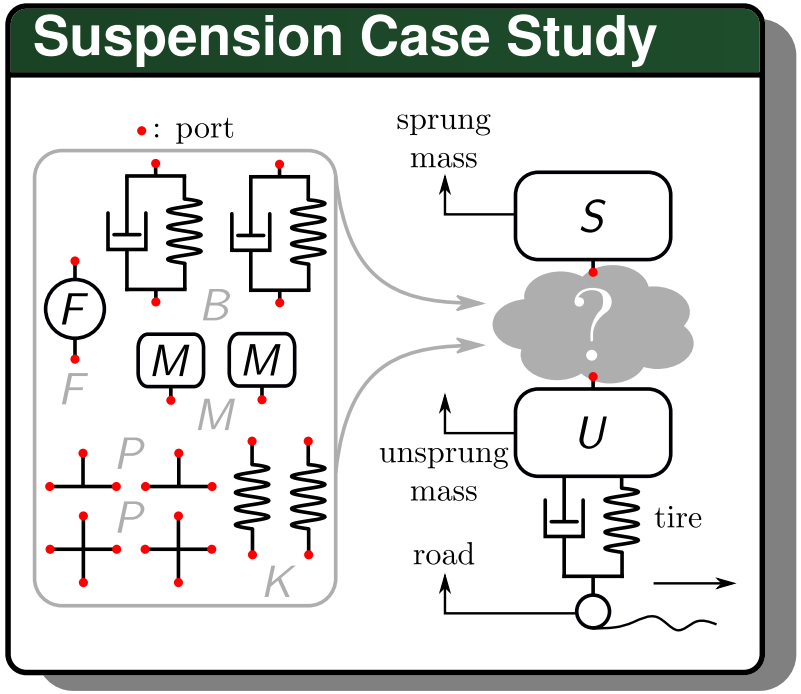


Table: Suspension case study comparisons.

	NST	CG	NIFG	UFG	LGIC	t (s)
original	1761015019	158154694	1943862	**12480**	28471024	17903
current	5230799	210637	40432	**12480**	27952	18

NST: nodes in spanning tree, CG: candidate graphs from enumeration algorithm, NIFG: non-trivially isomorphic and feasible graphs, UFG: unique feasible graphs, LGIC: labeled graph isomorphism checks

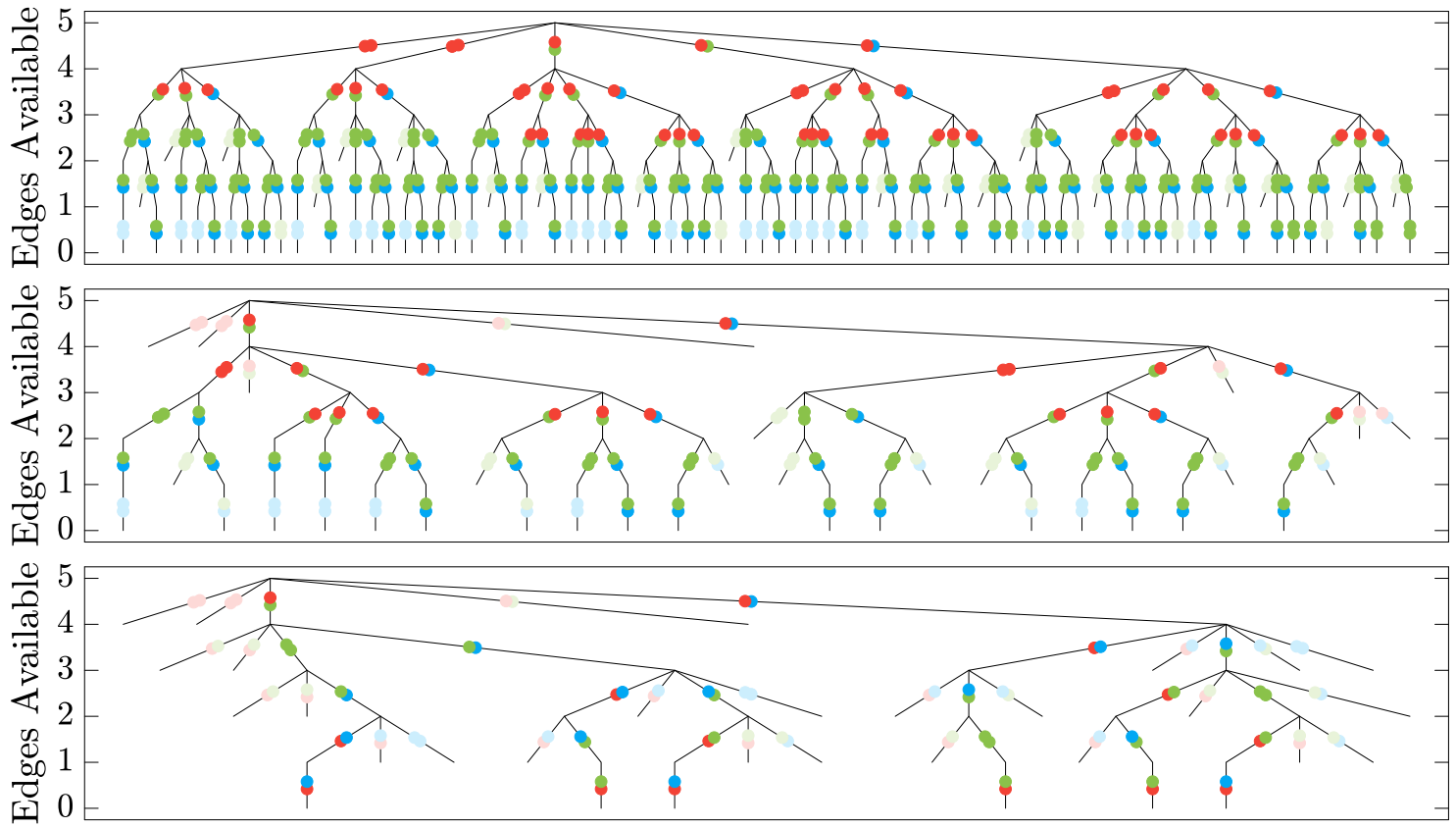
[1] Herber, Guo, and Allison 2017

→ Case Studies from the Original Paper (continued)

- Case study 1 component catalog:

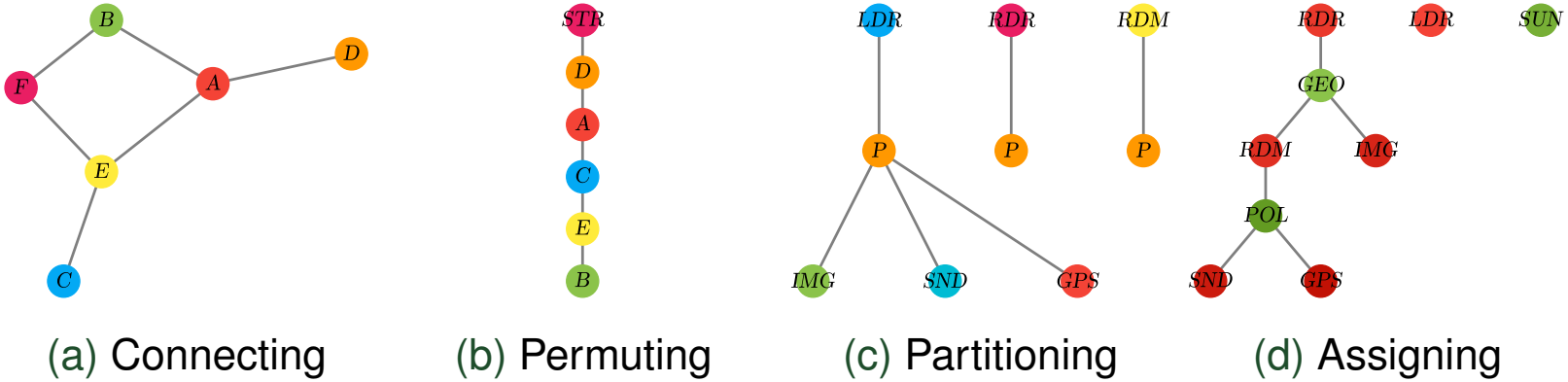
$$\boldsymbol{L} = [R\ G\ B], \quad \underline{\boldsymbol{P}} = \bar{\boldsymbol{P}} = [1\ 2\ 3], \quad \underline{\boldsymbol{R}} = \bar{\boldsymbol{R}} = [3\ 2\ 1] \tag{2}$$

Spanning Trees for Different Implementations



Other Examples

- (Six) Patterns in System Architecture Decisions[1] with NSCs



(a) Connecting (b) Permuting (c) Partitioning (d) Assigning

- The On-Line Encyclopedia of Integer Sequences (OEIS) representing known graph enumeration problems[2]
- Useful for validation, benchmarking performance, and discussing the pros and cons of the methods in this paper

[1] Selva, Cameron, and Crawley 2017 [2] *The on-line encyclopedia of integer sequences (A000041, A000079, A000110, A000142, A000262, A001147, A005177, A006125, A056156, A289158)* n.d.

References

- A. E. Bayrak, Y. Ren, and P. Y. Papalambros (2016). “Topology generation for hybrid electric vehicle architecture design”. *J. Mech. Des.* 138.8. DOI: 10.1115/1.4033656
- R. A. Brualdi and H. J. Ryser (1991). *Combinatorial Matrix Theory*. Cambridge University Press
- S. A. Choudum (1986). “A simple proof of the Erdos-Gallai theorem on graph sequences”. *B. Aust. Math. Soc.* 33.1. DOI: 10.1017/s0004972700002872
- R. Diestel (2017). *Graph Theory*. 5th. Springer. DOI: 10.1007/978-3-662-53622-3
- J.-L. Faulon, C. J. Churchwell, and D. P. Visco (2003). “The signature molecular descriptor. 2. Enumerating molecules from their extended valence sequences”. *J. Chem. Inf. Comput. Sci.* 43.3. DOI: 10.1021/ci020346o
- C. Hartmann et al. (2018). “A framework for automatic architectural synthesis in conceptual design phase”. *J. Eng. Des.* 29.11. DOI: 10.1080/09544828.2018.1532494

References (continued)

- D. R. Herber (2017). “Advances in combined architecture, plant, and control design”. Ph.D. Dissertation. University of Illinois at Urbana-Champaign. URL: http://hdl.handle.net/2142/99394
- D. R. Herber and J. T. Allison (2019). “A problem class with combined architecture, plant, and control design applied to vehicle suspensions”. *J. Mech. Des.* 141.10. DOI: 10.1115/1.4043312
- D. R. Herber, J. T. Allison, et al. (2020). “Architecture generation and performance evaluation of aircraft thermal management systems through graph-based techniques”. *Science and Technology Forum and Exposition*. AIAA 2020-0159. DOI: 10.2514/6.2020-0159
- D. R. Herber, T. Guo, and J. T. Allison (2017). “Enumeration of architectures with perfect matchings”. *J. Mech. Des.* 139.5. DOI: 10.1115/1.4036132
- S. R. T. Peddada et al. (2019). “Optimal flow control and single split architecture exploration for fluid-based thermal management”. *J. Mech. Des.* 141.8. DOI: 10.1115/1.4043203
- D. Selva, B. Cameron, and E. Crawley (2017). “Patterns in system architecture decisions”. *Syst. Eng.* 19.6. DOI: 10.1002/sys.21370

References (continued)

- E. Silvas et al. (2015). “Functional and cost-based automatic generator for hybrid vehicles topologies”. *IEEE/ASME T. Mech.* 20.4. DOI: 10.1109/tmech.2015.2405473
- *The on-line encyclopedia of integer sequences (A000041, A000079, A000110, A000142, A000262, A001147, A005177, A006125, A056156, A289158)* (n.d.). Online. url: https://oeis.org
- D. F. Wyatt, D. C. Wynn, and P. J. Clarkson (2014). “A scheme for numerical representation of graph structures in engineering design”. *J. Mech. Des.* 136.1. DOI: 10.1115/1.4025961

Questions?

Enhancements to the Perfect Matching Approach for Graph Enumeration-based Engineering Challenges

IDETC2020-22774

Daniel R. Herber

Colorado State University

daniel.herber@colostate.edu

www.engr.colostate.edu/~drherber

Links to the code on GitHub:

github.com/danielrherber/pm-architectures-project

doi: 10.5281/zenodo.3963751