

Model reduction for supervisor synthesis

Citation for published version (APA):

Moormann, L., van de Mortel-Fronczak, J. M., Fokkink, W. J., & Rooda, J. E. (2020). Model reduction for supervisor synthesis. In R. Carloni, B. Jayawardhana, & E. Lefeber (Eds.), *39th Benelux Meeting on Systems and Control : Book of Abstracts* (pp. 82-82). University of Groningen.
<https://www.beneluxmeeting.nl/2020/uploads/papers/boa.pdf>

Document status and date:

Published: 01/03/2020

Document Version:

Typeset version in publisher's lay-out, without final page, issue and volume numbers

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Model reduction for supervisor synthesis

L. Moormann¹, J.M. van de Mortel - Fronczak¹, W.J. Fokkink², J.E. Rooda¹

¹Eindhoven University of Technology and ²Vrije Universiteit Amsterdam
l.moormann@tue.nl

1 Introduction

The design of supervisory controllers for cyber-physical systems is an increasingly complex task due to the rising demand for functionality and safety. Through the use of formal methods, such as synthesis-based engineering, correct by construction supervisory controllers can be synthesized. To perform synthesis, first a discrete-event model is created that specifies what the system *can* do. This model is called the plant. Second, requirement models are created to specify what the system *may* do.

In [1], it is shown that under certain conditions the models can be reduced before synthesis, or that synthesis can be skipped entirely. In this work, the method presented there is extended to further reduce the plant and the requirement models, subsequently the method is applied in a case study.

2 Method

The method of [1] uses directed graphs, such as the one shown in Figure 1. In this graph, the vertices indicate components of the plant models and edges indicate dependencies between these components. The edges are based on the requirement models: there is an edge from the vertex of component A to the vertex of component B if there exists a requirement for component B that depends on component A. Cycles in the graph are indicated with red arrows. In [1], it is shown that synthesis can be skipped if the directed graph of a system contains no overlapping extended strongly connected components. Furthermore, if synthesis is needed, then all vertices without outgoing edges can be omitted during synthesis.

Models that contain dependency cycles can be reduced further by using the theory from [2]. In [2], symmetry is formally defined for discrete-event systems for the purpose of model reduction. This definition can be applied to the directed graphs of [1]. If the component models and requirement models of two vertices are symmetric, only one vertex needs to be included during synthesis. If synthesis for the reduced model does not add any new restrictions, then synthesis can be skipped entirely for the complete system.

3 Results

The proposed method has been applied to a case study in which a supervisory controller is designed for the Eerste Heinevoorttunnel (EHT), a roadway tunnel in the Netherlands. Figure 1 shows the initial graph of this model. This initial graph contains 369 vertices and 413 edges.

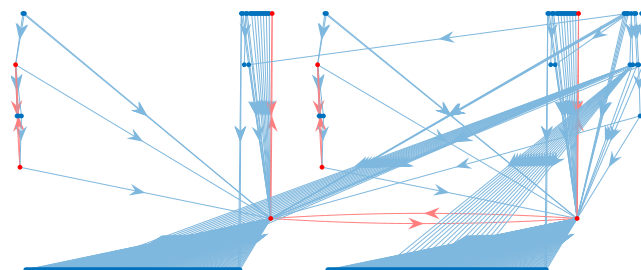


Figure 1: Directed graph of the initial EHT model.

First, the method from [1] is applied to reduce the model to 51 vertices and 73 edges. Second, the addition to this method is applied to reduce the model further to 21 vertices and 28 edges. Figure 2 shows this reduced graph.

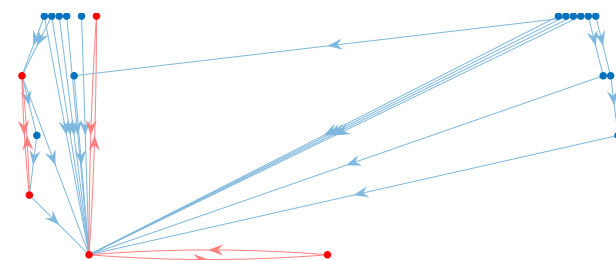


Figure 2: Directed graph of the reduced EHT model.

An attempt has been made to synthesize supervisors for the three models. Due to memory issues, no supervisor could be synthesized for the initial model. After model reduction, a supervisor has been synthesized for the two reduced models.

4 Conclusions and future work

A method has been proposed to extend the model reduction method of [1] for the purpose of supervisor synthesis. This method has been applied successfully to a case study. Future research will focus on possibilities to further reduce the number of vertices and edges in the model.

Acknowledgments

The authors like to thank Rijkswaterstaat for funding this research. Specifically we thank Patrick Maessen and Pascal Etman for their support in this project.

References

- [1] M.A. Goorden et al., “Model properties for nonblocking modular supervisors”, Submitted to *IEEE Transactions on Control Systems Technology*
- [2] L. Moormann et al., “Efficient validation of supervisory controllers using symmetry reduction”, Submitted to *Workshop on Discrete Event Systems 2020*