FRAMEWORK FOR SIMULATION BASED SCHEDULING OF ASSEMBLY LINES

Falk Stefan Pappert
Evangelos Angelidis
Oliver Rose

Dresden University of Technology
Institute for Applied Computer Science
01062 Dresden, GERMANY

ABSTRACT

Planning and scheduling of assembly lines is a complex problem which is often very hard to solve with classical scheduling approaches. A promising way of dealing with problems of this domain is simulation based scheduling. We introduce the architecture of a framework which is designed to aid in the creation of solutions for assembly line workforce scheduling. The framework combines a meta model to describe production networks and facilities with a complete modular design. Through the combination and reuse of exchangeable modules the framework offers the opportunity to focus on the development of optimization. Besides necessary input and output components there is a generic KPI data structure which allows to manage and calculate new values without the need to adapt existing modules. Optimizer modules are considered to be implementations of solution. To prove our concept we implemented scheduling solutions from previous industry projects with our new framework.

1   INTRODUCTION

The optimization of the resource scheduling in assembly line production systems is a very complex task. Until now there still is demand for research in the creation of new and the adaption of common scheduling approaches. Currently most smaller and medium size companies still depend on scheduling by experienced staff which leads to unused capacities and due date violations. There is also no fully developed software support which satisfies the needs of the production companies. Besides the complexity of the production system itself, the reasons for this situation are mainly a large variety of performance objectives and solutions strategies. Our framework aims to offer the possibility to create various solution strategies for problems in this area while reducing the overhead of typical company specific software.

In Section 2 we will further discuss the production system and its characteristics. Section 3 provides a short overview of the literature in the field of project scheduling and will give a short introduction to simulation-based scheduling. Section 4 is devoted to the outline of our framework and we consider a simulation-based scheduling approach from previous projects. In Section 5 we draw some conclusions and discuss future research.

2   PROBLEM DESCRIPTION

The goal of our work is to create resource scheduling solutions for real world heavy machine assembly scenarios. Typical products are, for example, turbines, planes and industrial machines, which are produced in small series or even as just one of a kind. High customization varieties are usually offered in these markets making each product almost unique. Due to these circumstances, workers are still the major

production resource in these facilities. Other important resources are the construction floors and transportation equipment like shuttles or cranes which sometimes are also used to hold parts in place during construction due to the size and weight of the products. These resources have to be shared by several simultaneously manufactured jobs which consist of a number of activities. These activities do not represent atomic operations. They are considered as a set of work packages which are usually done together and need similar equipment. Since humans are the main resource, we need to consider their working schedules. Therefore we need to consider shift and break schedules. There are other resource restrictions which are considered to obtain a more realistic model. Especially in the construction of heavy machinery a construction site is usually used for several consecutive activities, sometimes even parallel activities. Further resource characteristics we consider enable the modeling of early release of resources if they are not needed throughout the whole activity or blocking due to unavailability of resources which are necessary for the following process steps.

Since the products are very huge there is often the possibility to work on an activity with different amounts of workers in different amounts of time.

Furthermore the goal of scheduling in this domain is not necessarily minimizing the makespan. There are good reasons, e.g., storage cost and capital commitment, to achieve completion of products right at their due dates.

The scale of the problems exceeds common academic problem instances, even smaller companies are planning several hundred activities for which several resource groups have to be scheduled.

In conclusion we face a problem which is similar to project scheduling problems. Using the classification scheme from the project scheduling literature, we consider an extended multi-mode resource-constrained multi-project scheduling problem (MMRCMPSP) with activity splitting.

3   LITERATURE AND SIMULATION-BASED SCHEDULING

The scheduling of assembly lines belongs to the large combinatorial optimization problems. It can be described as a resource-constrained project scheduling problem (RCPSP) or as the generalization, the multi-mode resource-constrained project scheduling problem (MRCPSP).

Blazewicz, Lenstra, and Rinnooy Kan (1983) proved that the RCPSP is a NP-hard problem and so is also its generalization (MRCPSP). In the paper of Kolisch and Drexl (1997) it is proven that the problem is NP-complete if there is more than one non-renewable resource. Furthermore the real world is more complex. A further generalization is the Multi-Mode Resource- Constrained Multi-Project Scheduling Problem (MMRCMPSP) or the "MMRCMPSP with activity splitting". Because of the generalization these are also NP-hard.

Although there are many studies and papers for RCPSP and MRCPSP, in the research area of MMRCMPSP and "MMRCMPSP with splitting activities" there are only very few. The topic is still important, but an applicable solution approach is difficult to find. Peteghem and Vanhoucke (2010) give an overview on papers considering these problems and present work about preemptive and non-preemptive MRCPSP. Tseng (2008) presented two approaches for MMRCMPSP and Buddhakulsomsiri and Kim (2006) investigated an "MMRCPSP with activity splitting". Research in this area is usually done by applying exact approaches on very small problem instances or by using genetic algorithms to tackle medium-sized problems.

Another promising approach to solve these problems seems to be a heuristic optimization algorithm based on simulation which we are going to investigate in further detail (see Figure 1).

The basic idea is to simulate a scenario, analyze it and create a new one according to the results of the current and of the previously simulated scenarios. The goal of these iterations is to generate a feasible solution within only a few steps (see Figure 2).

Since the approach seems to be promising there is a need to evaluate and compare it with other approaches and to test solution strategies on data sets of different companies. To provide a basis for further research we designed and implemented a framework.
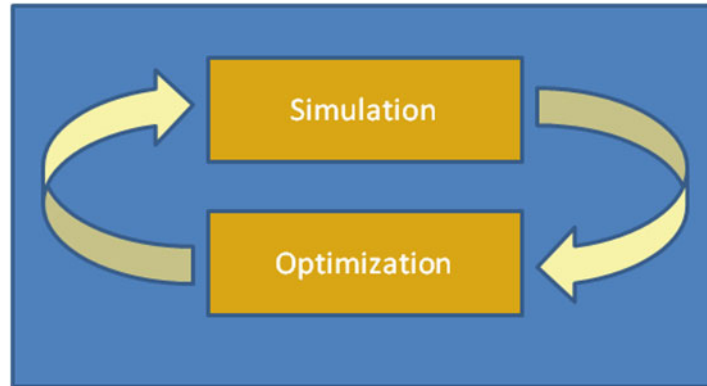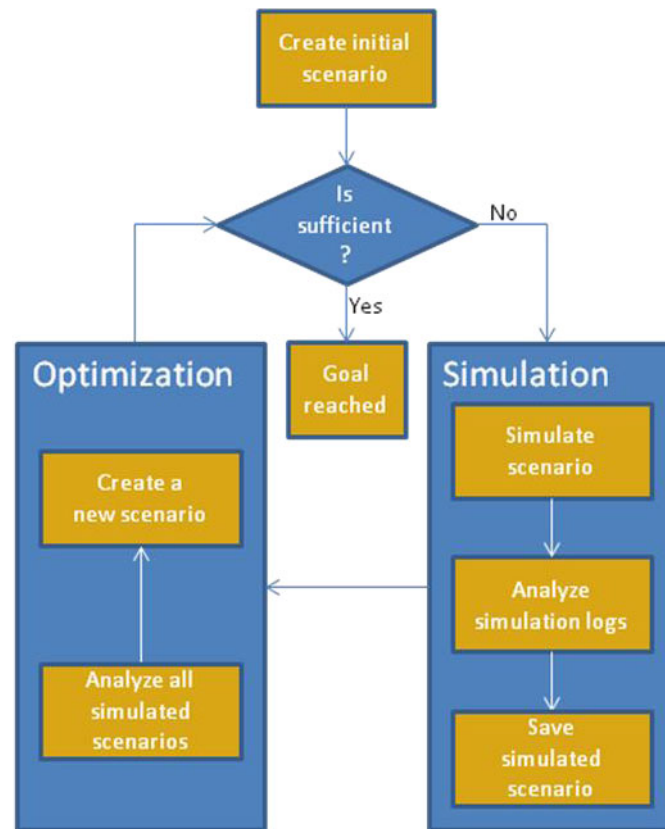
Figure 1: Simulation based scheduling

Figure 2: Flow chart simulation based scheduling

4   FRAMEWORK

Our main goal while creating this software was to design a generic framework which is able to support solutions for most companies working in heavy machine assembly. But with different companies come different IT environments and different objectives in scheduling, while having similar production facilities and processes. This lead to the decision to create a framework core which represents a domain specific data meta model and interfaces to the major components needed to build a support system, which you can find in Figure 3.
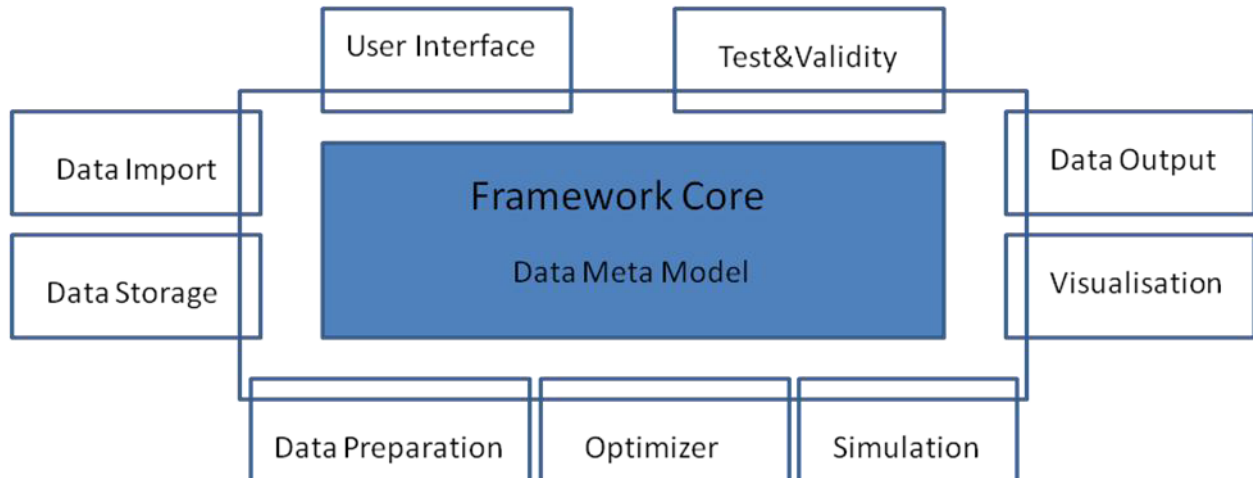
Figure 3: Framework architecture

These components can be divided in two categories: data management and solution strategy.

- Data management:
  – data input
  – data storage/access management
  – data output/visualization
  – test and validity

- Strategy:
  – optimizer
  – data preparation
  – simulation

The task of the data management components is basically to provide all data needed to schedule a scenario and to evaluate the performance of a schedule. Data input components are to read models from a hard drive or in the future will serve as direct connection to enterprise resource planning (ERP) or production planning systems (PPS). Currently we have plug-ins to support models in two proprietary formats and models in SysML. For further information about SysML for production systems see Schönherr and Rose (2009) and Huang, Ramamurthy and McGinnis (2007). Different input plug-ins are necessary since different companies use different data models even if they use the same software. Test and validity components support data integrity and search for problems in the model before any scheduling algorithm is applied.

An even larger problem is the simulation software. Simulation models are usually very specific to a simulation tool and if a company has models for one simulator the transition to another software is very expensive thus giving most companies no choice but to stay loyal to their simulation software provider. Besides the opportunity of choice there is in some cases the necessity of change in case the previous software vanishes from the market or due to the fact that in new software versions old models are no longer working. To tackle this very costly problem the model we are working on is not fitted to a specific simulator but our simulation plug-ins are considered translators that automatically generate a simulator-specific model from our universal data model. The strategy components are discussed in more detail in the following subsections.

4.1 Data meta model

Figure 4 shows a simplified version of the factory model part of the data meta model which is used to model an assembly facility in our framework. It is designed to support small series and single runs. It takes the aforementioned characteristics into account.

Resources are considered to be separated into resource groups with equal elements which share their data. Besides information about the classification of resources there is also data about their availability. Availability is modeled with shift and break schedules. Shift schedules contain data for the whole time frame in which the jobs need to be scheduled. Each day can be divided into a number of shifts with the amounts of resources available within this shift. Holidays and special occasions are modeled with shift plans, too. Breaks schedules represent a single week's break schedule. This schedule is later repeatedly mapped onto the shift schedule. This modeling approach provides a flexible model of the availability of resources while reducing the amount of data necessary considerably.

Due to the nature of the facilities and slight changes in each order of a product, each product is represented by an independent project network graph of activities. Arcs represent the activities while nodes model the logical connections between activities. There are parallel as well as alternative routes through a production network. Different combinations of resources and processing steps may lead to the same result. We differentiate two ways of modeling equivalent activities. The key differentiation characteristic is whether different ways of production use different types of resources or just different amounts of the same resource. Different amounts of resources are represented as modes within an activity. Different resource type usage is represented as alternative arcs in the project network. Nodes also contain resource selection rules, these rules offer an opportunity to define dispatching rules which may be used to decide which of following alternative routes should be taken.
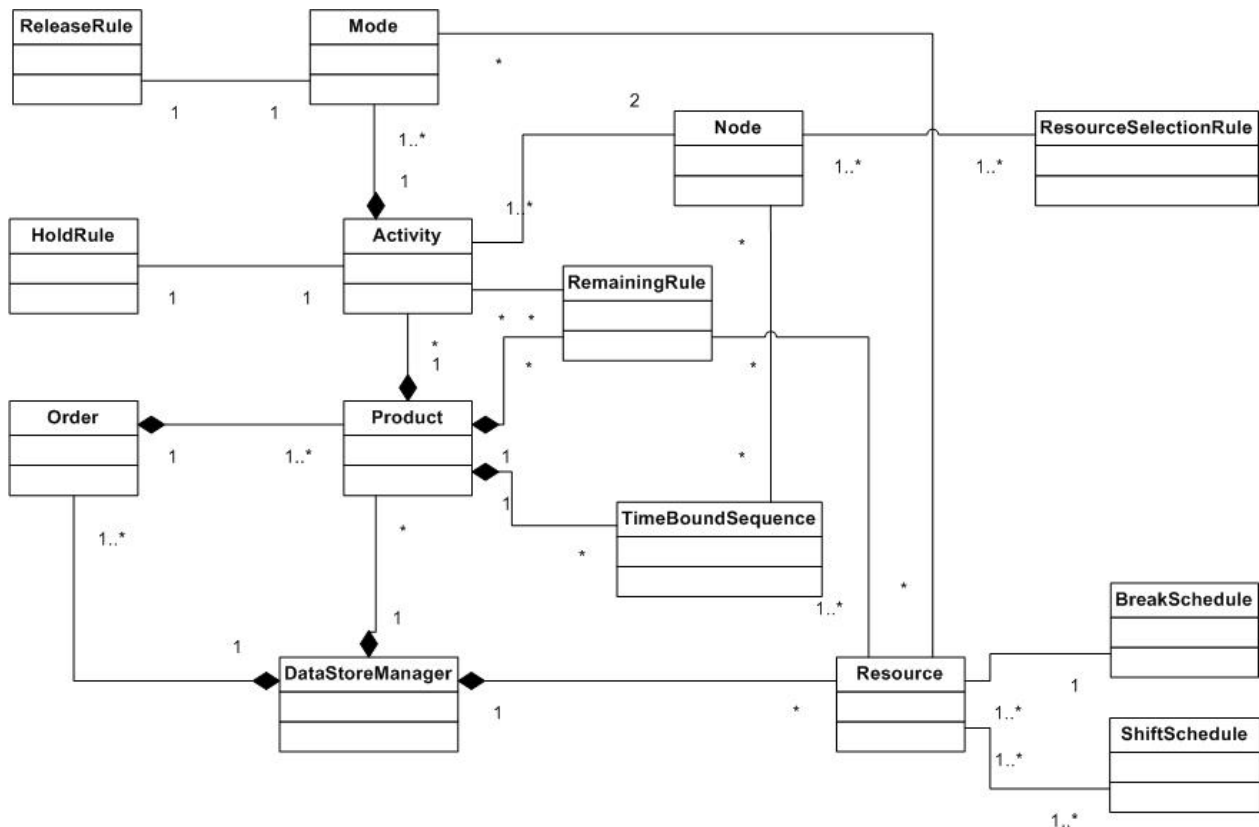

Figure 4: Simplified UML diagram

The objects ReleaseRule, HoldRule, Remaining Rule and TimeBoundSequence represent additional constraints during simulation. These constraints are stored as close to their actual application as possible. Time-bound sequences and remaining-rule constraints are therefore attached to the product network while local early releases of resources are attached to the mode because the release time depends on the process time of the activities which depends on the chosen mode. The HoldRule object represents the need to block resources in the case resources which are necessary for the following process step are not yet available.

The data shown in Figure 4 shows only the model part of the data meta model. The other part of the data model is concerned with keeping track of previous simulation runs which is important for approaches like simulation-based scheduling since new scenarios are created by evaluating previous ones. This part of the data consists of log objects. There are log objects for activities, resources, orders, products, nodes, and for each simulation to keep track of overall simulation data. All data necessary to recreate a scenario is stored as well as data generated during the simulation or while evaluating simulation results. This offers a data foundation for following scheduling cycles and provides the opportunity to evaluate the strategy. Besides the simulation results a variety of information can be stored for each simulation run. Key performance indicators, timelines and factory behavior can be stored in a generic data structure.

4.2 Key performance indicators

To offer a wide range of evaluation possibilities and to keep maintenance of the framework for future research as easy as possible we decided to use a generic data structure to store KPI information. This allows future developers to add data in case they are needed for their solution strategies. KPIs are split in two parts. On one hand there is a template-like description which is stored once for this KPI. These template objects contain information about type, unit and the objects they belong to. For identification we also store an ID. On the other hand there is a value part which can be identified with a corresponding ID in the template. The value objects are stored in the above mentioned log objects. Currently there are different types of KPIs supported. Besides basic numeric values there are types for strings and timelines. There is even the possibility to store the change in production state of a resource. Since basic types are not always sufficient to keep track of all data, there are also array and hash types.

4.3 Simulation plug-in

The basic function of a simulation plug-in is to simulate a given scenario which was generated by an optimizer plug-in. The necessity of this plug-in depends on the optimization strategy. Since we want to be able to compare various approaches, some might only use simulation as evaluation while others are able to work completely without it. Since our current focus is in researching simulation-based scheduling approaches such plug-ins have to be implemented.

Usually simulation plug-ins do not start automatically but are called by an optimizer plug-in. Practically the simulation plug-in controls an external simulation tool, which simulates the scenarios. As a general rule, every simulation plug-in is designed only for just one simulation tool, which increases the effectiveness of the plug-in.

Furthermore the architecture of this framework and new multi-core processor architectures allow the simulation of several scenarios at a time. Since most current commercial simulators only use a single core this decreases the run time of the optimization or allows a deeper search within the same time frame. The general steps that a simulation plug-in executes, are
- Reading the scenario information from the data model
- Generating a tool specific simulation model
- Executing the external simulation tool with the generated model
- Analyzing simulation run data
- Reporting important data back to the framework

The communication concept is based on the KPIs. Since the simulation plug-ins implement a generic interface they can easily be exchanged. So far we have successfully implemented simulation plug-ins for Simcron Modeler 3.0 and Anylogic 6.0. While working with different problem scenarios we recognized that even while generating equivalent simulation models, some simulators are better suited than others for different problems due to the necessary customization in the translation and different features offered by the simulation software.

4.4   Data preparation plug-in

The primary purpose of a data preparation plug-in is the transformation of measurement data into KPIs. Data preparations are therefore considered libraries of computation procedures. We differentiate two kinds of performance indicators: static and dynamic. On one hand, static indicators are values which are only dependent on the structure of the facility and which are not influenced by a scheduling strategy. On the other hand there are values which can be calculated after each simulation or for each scenario. We consider this information as dynamic indicators.

Currently we have two implementations for data preparation plug-ins, one for static and one for dynamic key values. They support the generation of about 30 key values.

The framework is designed to work with several data preparation plug-ins at a time to maximize modularity and reusability of components.

4.5   Optimization plug-in

The optimizer plug-in represents the core of a solution strategy, it contains the central algorithms of the concept. The developer can implement an optimizer plug-in to fit to a general problem, a very specific problem or even to solve a class of similar problems. Regardless of the structure of the algorithm, the basic steps in an optimizer plug-in for simulation based-scheduling usually are:

- Generation of a new scenario (based on previous simulation results and the strategy)
- Start of simulation plug-ins to calculate a scenario
- Start of data preparation plug-in to evaluate and calculate KPI

In simulation-based scheduling these steps are usually performed in a loop which terminates if the quality of a solution is good enough or based on time restrictions.

As a test we implemented the concept of Majohr and Rose (2008) depicted in Figure 5 to show the functionality of our framework. The suggested algorithm has three steps with different goals. The second and the third step consist of several scenarios.
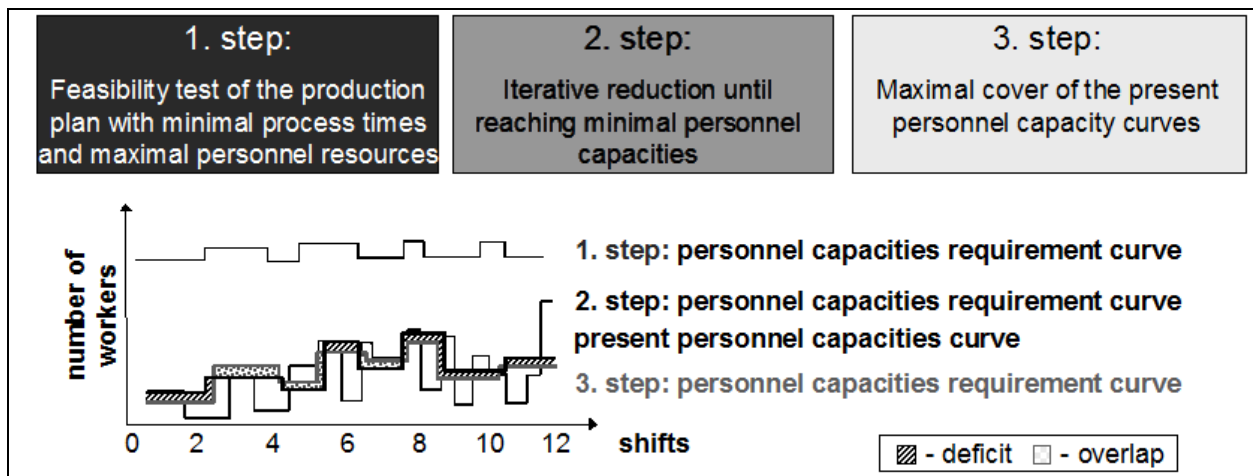

Figure 5: Solution strategy as presented in Majohr and Rose (2008)

4.6   Visual output plug-in

According to Fayyad, Grinstein, and Wierse (2002), visualization is a way to let people understand complex data sets through graphical interfaces. Furthermore visualization should not just be a presentation of the results (just a picture) but allow the user to search and find information in the results. Generally the visualization should help the user to have a deep look into simulated scenario, explore randomly the results, confirm hypotheses, and discover hidden information. The users have different mental models to understand results. As a consequence the framework should have a taxonomy of visualization levels and user-system interactions. To facilitate the qualitative visual presentation of the scenario, the user should trust and understand the visual output plug-in.

We implemented as visualization component for the framework a visual output plug-in. We used the libraries JFreeChart and Nebula. JFreeChart is a library to display professional quality charts and Nebula is a library of SWT widgets that, among other characteristics, is able to display Gantt charts. The user may choose a graphical view of the KPI and the scenario results including beginner and expert modes. The way to analyze the data depends only on the user and this makes it easier for him/her to understand the visual data.

5   CONCLUSIONS AND FURTHER RESEARCH

In this paper we presented a framework for simulation-based scheduling with its key components. Interchangeable modules work based on a meta model and a generic KPI structure. In future research we will implement different approaches for solving workforce scheduling problems to evaluate their results and advantages. Our goal is to create a decision support system which is able to generate useful schedules in less than 10 minutes. Our test sets are provided by several partner companies producing a variety of different products from industrial printing machines to aircrafts.

ACKNOWLEDGMENTS

REFERENCES

Blazewicz, J., J. Lenstra, A. Rinnooy Kan. 1983. Scheduling subject to resource constraints: classification and complexity. *Discrete Applied Mathematics* 5:11–24.

Buddhakulsomsiri, J., and D. Kim. 2006. Properties of multi-mode resource-constrained project scheduling problems with resource vacations and activity splitting. *European Journal of Operational Research* 175:279–295.

Fayyad, U. M., G.G. Grinstein, and A. Wierse. 2002. *Information visualization in data mining and knowledge discovery*. Morgan Kaufmann Publishers.

Huang, E., R. Ramamurthy, and L.F. McGinnis. 2007. System and simulation modeling using SysML. In *Proceedings of the 2007 Winter Simulation Conference*, ed. S. G. Henderson, B. Biller, M.-H Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, 796-803. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Kolisch, R., and A. Drexl. 1997. Local search for nonpreemptive multi-mode resource constrained project scheduling. *IIE Transactions* 29:987–999.

Majohr, M. F., and O. Rose. 2008. A Simulation-based workforce scheduling heuristic for complex assembly lines. In *Proceedings of the 2008 Operational Research Society Simulation Workshop.*

Peteghem, V.V., and M. Vanhoucke. 2010. A genetic algorithm for the preemptive and non-preemptive multi-mode resource-constrained project scheduling problem. *European Journal of Operational Research* 201:409–418.

Schönherr, O., and O. Rose. 2009. First steps towards a general SysML model for discrete processes in production systems. In *Proceedings of the 2009 Winter Simulation Conference*, eds. M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin and R. G. Ingalls, 1711-1718. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Tseng, C.-C. 2008. Two heuristic algorithms for a multi-mode resource-constrained multi-project scheduling problem. *Journal of Science and Engineering Technology* 4(2):63-74.

AUTHOR BIOGRAPHIES

FALK STEFAN PAPPERT is a student at Dresden University of Technology. He is a member of the scientific staff of Prof. Dr. Oliver Rose at the Chair of Modeling and Simulation. He is currently writing his master thesis in Computer Science. He is a member of GI. His e-mail address is <falk.pappert@mailbox.tu-dresden.de>.

EVANGELOS ANGELIDIS is a PhD student at Dresden University of Technology. He is a member of the scientific staff of Prof. Dr. Oliver Rose at the Chair of Modeling and Simulation. He received his M.S. degree in Computer Science from Dresden University of Technology. His e-mail address is <evangelos.angelidis@mailbox.tu-dresden.de>.

OLIVER ROSE holds the Chair for Modeling and Simulation at the Institute of Applied Computer Science of the Dresden University of Technology, Germany. He received an M.S. degree in applied mathematics and a Ph.D. degree in Computer Science from Würzburg University, Germany. His research focuses on the operational modeling, analysis and material flow control of complex manufacturing facilities, in particular, semiconductor factories. He is a member of IEEE, INFORMS Simulation Society, ASIM, and GI. His web address is <www.simulation-dresden.com> and his e-mail address is <oliver.rose@tu-dresden.de>.