

(20th ICPR) Modelling and Optimisation of Assembly Processes

Gerald Weigert, Thomas Henlich, Klemmt Andreas

► To cite this version:

Gerald Weigert, Thomas Henlich, Klemmt Andreas. (20th ICPR) Modelling and Optimisation of Assembly Processes. International Journal of Production Research, 2011, pp.1. 10.1080/00207543.2010.544338. hal-00680178

HAL Id: hal-00680178 https://hal.science/hal-00680178

Submitted on 18 Mar 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers. L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



(20th ICPR) Modelling and Optimisation of Assembly Processes

1	
Journal:	International Journal of Production Research
Manuscript ID:	TPRS-2010-IJPR-0416.R1
Manuscript Type:	Original Manuscript
Date Submitted by the Author:	24-Nov-2010
Complete List of Authors:	Weigert, Gerald; Technische Universität Dresden, Electronics Packaging Laboratory Henlich, Thomas; Technische Universität Dresden, Electronics Packaging Laboratory Andreas, Klemmt; Technische Universität Dresden, Electronics Packaging Laboratory
Keywords:	SCHEDULING, OPTIMIZATION, MODELLING, SIMULATION, ASSEMBLY PLANNING
Keywords (user):	

SCHOLARONE[™] Manuscripts

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy

Modelling and Optimization of Assembly Processes

G. WEIGERT*, T. HENLICH and A. KLEMMT

† Dresden University of Technology Electronics Packaging Laboratory D-01062 Dresden, Germany

The following paper is a case study for scheduling of assembly processes. Two optimization methods, mixed integer programming (MIP) solver and heuristic simulation-based optimization are compared to each other. It can be shown, up to which model complexity the solver methods have the advantage over the simulation-based methods concerning a possible online application. Both, the mathematical formulation as well as the discrete event simulation (DES) system bases on a meta-model - a special Petri net. It is shown that bipartite graphs, i.e. Petri nets, avoid ambiguities in the case of alternative assembly sequences and have advantages in comparison to the usual simple precedence graphs. The results of optimization experiments are described by the example of assembling a numeric controlled (NC) milling table. The model is capable of reflecting multiple parallel assembly processes, alternative assembly steps and can predict characteristics like utilisation, stock or adherence to schedules.

Keywords: scheduling, optimization, modelling, simulation, assembly planning

1 Introduction

In mechanical and plant engineering as well as in other industrial fields such as aircraft construction and shipbuilding, assembly processes are very common. In many cases they are the last part of the manufacturing chain and contribute a great deal to the economic value added. This is one of the reasons for the big interest in the optimization of such processes. The concept of optimization means, in this context to find an optimal schedule for the assembly process; it is not subject of this work to optimize other technological aspects of assembly, for instance motion sequences. In addition our goal is the operational planning and real-time control of the process. This implicates some further important requirements:

- 1. The control variables as well as the objective variables are determined by the industrial user.
- 2. The required data must be transferred to the planning system without manual intervention and remarkable delay. Finally this also requires an automatic generation of simulation models.
- 3. The optimization run itself must not take a long time period to complete, because of the intended online application.

Because of the great importance of assembly processes, their optimization is widely discussed in the literature. As long ago as in 1989 you can find a comprehensive literature-review and analysis of the design, balancing and scheduling of assembly systems (Ghosh and Gagnon 1989). Gosh and Gagnon distinguish between single or multi/mixed models respectively between stochastic or deterministic models. In this meaning our problem can be allocated as a multi/mixed deterministic model. They also stated that "...the criteria reported in the literature for designing and operating assembly line systems have typically been singular. ... Therefore, the further use of multi-criteria objective functions in modelling assembly systems holds the promise of added research interest and of more closely reflecting the set of real-world objectives.". From this it follows that an investigation of optimization methods should consider multi-objective functions today. Among the optimization algorithms the Genetic Algorithm (GA) has been an important issue for years. Chaudhry and Luo give a general literature overview about the application of GA in production and operations management (Chaudhry and Luo 2005). They highlight the suitability and performance of GA for solving such problems, but deplore a

G. Weigert et al.

lack of availability of user friendly commercial software packages. So the GA applications are still concentrated in only a few areas of production and operations management. Guo and Wong use indeed GA for solving an assembly line balancing problem with work-sharing and workstation revisiting (Guo and Wong 2008), but they do not compare this algorithm with other heuristics or MIP.

The following paper is a case study which investigates the possibility of an online optimization of assembling a numeric controlled (NC) milling table. Thus two main aspects are discussed in the following: on the one hand a suitable approach for modelling a general assembly process must be found and on the other hand efficient algorithms for optimization are needed. So the article is divided into the two parts, modelling (section 2) and optimization (section 3). In section 3 two different approaches are investigated and compared to each other:

- 1. Heuristic simulation-based optimization
- 2. Mixed integer programming (MIP)

While the first approach needs a classical simulation model, the second method is based on a mathematical formulation of the problem. The specific requirements, the advantages and disadvantages as well as the limits of both methods are discussed by means of the results. Both of these methods are fundamentally different and also based on a different approach of modelling. It should be interesting to see which method is suitable for an online application to schedule real processes. Because we are experienced in MIP and in simulation-based scheduling as well (see also Klemmt et al. 2009), an objective comparison should be possible as well as a connection of these methods should be possible. Both approaches are brought together in a common meta-model as a database, which is described in the next section.

The paper is structured as follows: Section 2 introduces the example of an assembly process and distinguishes the importance of assembly alternatives for controlling this process. The advantages of a bipartite graph model are explained there, it is used as a meta-model for describing this processes. In section 3 two basic kinds of optimization methods are explained: the simulation-based optimization and a mixed integer programming (MIP) approach. Section 4 includes an optimization study, which compares several heuristics with the mathematical approach. The comparison should consider the later online application. Section 5 gives a short summery and an outlook as well.

2 A Suitable meta-model for assembly processes

First of all an assembly process is described by a graph-oriented meta-model, where i.e. the several assembly steps are represented by edges and the results of these steps represented by nodes of this graph, as shown in Figure 1 and 2. The advantage of this meta-model is its clearness for the operator. Using a computer for i.e. simulation, optimization or both, this graph must be mapped into a database. If this database is a relational one, it consists of some tables (Figure 3). One table could contain all nodes, another table contains the edges etc. – this is the database model of the assembly process. The structure of such a database is in turn based on a meta-model. Requirement for every operational planning is a database model describing the assembly process in all its characteristics in a sufficiently exact way and in real-time. For many practical applications the widely-known precedence graph has proved itself (Prenting and Battaglin 1964), despite its obvious disadvantages.

Among all the meta-models which are able to describe the static as well as the dynamic characteristics of the assembly process in an intuitive and formally correct way, bipartite graphs (i.e. Petri nets) have shown their suitability (Weigert et al. 2008). Unlike the well-known precedence graphs, which are commonly used for description of assembly processes in practice, they are unambiguous (Figure 1).

In the following the example of the assembly process is described in detail before both graph models are compared to each other.

2.1 The example

The chosen process, the simplified assembling of the milling table, is a part of the assembly process of a NC milling machine. The final product, the milling table (ABCDE), consists of 5 pre-products: spur

International Journal of Production Research

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy

gear (A), table top (B), gearbox (C), oil pump (D) and measuring system (E). Assembly is possible in 5 alternatives (A1–A5) including the intermediate products resulting from the respective steps:

- Alternative A1: (((A AND C) AND D) AND B) AND E
- Alternative A2: ((A AND C AND D) AND B) AND E
- Alternative A3: ((A AND B) AND (C AND D)) AND E
- Alternative A4: ((A AND C) AND D) AND (B AND E)
- Alternative A5: (A AND C AND D) AND (B AND E)

So one can assemble the spur gear (A) and the gearbox (C) at first, subsequently one assemble the oil pump (D), then assemble the table top (B) and at last assemble the measuring system (E). This is the alternative A1 of the list. Another possibility is to start similarly to alternative A1, but assemble the table top (B) and the measuring system (E) separately. At last, both intermediate products, the gear with the oil pump (ACD) and the top table with the measuring system (BE) are assembled. This is the alternative A4 of the list. Both alternatives as well the other three lead to the same final product. The alternatives have a central importance here, because they are the control variables used for optimization. The duration of the alternatives and their respective steps is shown in Table 1. The left row of the table shows all the possible assembly steps and the right row shows the duration for this step, measured in hour and minute. Step A+B means that A and B are assembled in this step (duration 3 hour 20 minutes), AC+D means that D is assembled to AC (duration 2 hour 30 minutes) while A and C were assembled in an assembly step before (duration 3 hour 20 minutes). The latter assembly operation can also be done by one single step A+C+D (duration 4 hour 10 minutes). In the centre of the table the 5 possible alternatives are marked (look at the list ahead). The bottom line of the table finally shows the total duration for every alternative which is the sum of the durations of every involved single assembly step (i.e. A1: 18:20=3:20+2:30+6:40+5:50). The objective variables are freely selectable and will be discussed in the following sections. The durations for the assembly represent typical values from NC table assembly practice.

Table 1

The structure of the assembly process is shown as a set of precedence graphs. It must be pointed out that the pre-products A through E are delivered at specified dates (delayed supply). There are n_M mechanics available for assembly. (In the example $n_M = 2$ was used). They can complete assembly steps of the same order in parallel or work at different orders. Thus the organizational order of the assembly steps is open. (see Figure 1, A3: AB can be completed before, after, or in parallel with CD). Only the precedence relation (also known as technological order) of the respective assembly steps matter. In total there are six orders to be scheduled in this example, with the following objectives:

- 1. minimal total tardiness T in h
- 2. minimal average lead time (or cycle time), *L* in h
- 3. maximal capacity utilisation, U where $0 \le U \le 1$ (this objective function is only observed, but not included in optimization)

Depending on the alternative chosen there are 3 or 4 assembly steps needed, to get the complete result. The results of most of the assembly steps are intermediate products (e.g. ACD, BE, etc.).

Figure 1

2.2 Precedence graphs

Precedence graphs describe the predecessor-successor relationships between the respective assembly steps and are thus in mathematical terms the representation of an order relation on the set of assembly steps. In event-node form every node of the precedence diagram stands for an assembly step or an article, the directed edges always point from the directly preceding step to the one immediately following it. In this way the AND relations which are typical for assembly processes can be described.



 By assigning weights to edges in addition, also bill-of-materials relationships (as in a Gozinto graph) can be displayd.

Precedence graphs are still widely used in practice (Niu, Ding and Xiong 2003), even though they have some inherent disadvantages. One of them is the inability to represent alternatives. Alternative assembly paths directly modify the order relation of the assembly process. The example of a machine tool producer described in the following shows this. It describes a subset of the assembly problem introduced in section 2.1.

Each of the technological alternatives has its own respective order relation (O_1-O_5) and thus its own precedence graph, as shown in Figure 1 for two of the alternatives. A "superposition" of the precedence graphs leads to ambiguity because AND and OR operations can not be distinguished. This ambiguity would have to be resolved in a yet to be determined manner, e. g. by introducing special labels for the assembly steps. Such a model extension would make the formal handling of the meta-model more difficult, especially in a computing context. That's why other means of description must be investigated

2.3 Bipartite graphs

Assembly alternatives can not be ignored because they make a great contribution to the flexibility of the assembly process and offer a substantial potential for optimization which needs to be exploited. So, the precedence graph can give an overview of the structure of the process but it is unsuitable as a metamodel for operational planning because of the missing OR relation. The solution is to use bipartite graphs containing two disjoint sets, or types, of nodes. A graph is called bipartite if and only if no two nodes of the same type are connected to each other by edges. By defining separate node types for AND and OR relations one obtains the so-called AND/OR graph, which is in principle capable of representing technological alternatives (Homem de Mello and Sanderson 1990). The practicality of AND/OR graphs in optimization was proven (Lambert 2006).

Another type of bipartite graph is the Petri net which has places and transitions for nodes. Petri nets provide all requirements to model assembly processes in a structurally unambiguous way (Zhou and Venkatesh 1999, Mînzu, Cernega and Henrioud 2001). This even includes disassembly processes which can be interspersed with assembly steps (e.g. In the case, where an auxiliary fixture is temporarily used) as well as synchronous or asynchronous parallel completion of steps.

As an extension, Timed Petri nets make it possible to describe assembly durations in the metamodel (König and Quäck 1988). The type of meta-model used in this work is a Petri net with timed transitions and forward reservation of places (Weigert and Henlich 2008). It was chosen because it is best fitted to represent the assembly situation in a general manner. For instance it allows us to model some restraints of the assembly process in a consistent way, e.g. the maximum capacity of a station.

Figure 2

The Petri net in Figure 2 shows the simplified assembly process of the NC table including all of the 5 alternatives. In contrast to the precedence graph, there are obviously no ambiguities, because the AND and OR operations can be distinguished by the node type (transition or place) they are pointing to. So the Petri net was modelled in a database from which both, the simulation model for heuristic optimization, as well as the mathematical formulation for the CPLEX solver (the well-known state of the art library, developed by ILOG) were directly derived in a fully automated way. Petri nets are used as the foundation for the relational database so that the entity-relationship diagram in Figure 3 is just another shape of it. The transitions (work steps) and places (articles) have a table each, and the two types of arcs (input and output arcs) also have a table (requirements and results), which makes a total of four tables. The relationship diagram of the database. Transformation into simulation models is done via the simulator's database interface (e. g. ODBC or similar SQL based APIs).

Figure 3

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy

3 Optimization of assembly processes

For both optimization methods described below, basically the same set of objective functions and control variables was used. Differences exist only in method-specific control variables or restrictions. Because many assembly processes are purely order-based, the compliance with agreed delivery dates is especially important. Thus one objective function used here is the total tardiness of all orders which needs to be minimized. The other objective function is the lead time which is also important in assembly processes.

The primary control variables are alternative assembly processes which can be classified as a technological control variable. Restrictions are also taken into account in the respective models (e.g. stock level, availability of resources, setup time etc.)

The problems to be solved are usually NP hard, and their solution complexity is further increased by adding assembly alternatives: in a problem with *n* orders where each can be assembled in *m* alternative ways there exist m^n assembly variants. So it can be shown, for instance, that some of these problems include the problem $J2||\sum C_i$, which is NP hard (Brucker 2007).

3.1 Simulation-based optimization

For optimization it is first necessary to create a simulation model in a discrete event simulation (DES) system. This simulation model includes both, control variables on the one side and target variables on the other side. A complete simulation run is started with fixed values for the control variables. It results in values for the determined target variables. A special algorithm evaluates the target variables and looks for new values of control variables. So we get a closed optimization cycle as shown in Figure 5. The advantage of this simulation-based optimization method is that the simulation model can be arbitrarily detailed created, inclusive diverse dispatch rules. So, also very complex manufacturing systems can be principally optimized. The model generating process described here was designed from the start with online and real-time capabilities in mind, so it runs fully automated. For storage a relational data-base with SQL interface is used, from which all simulation data is read (Petri nets, orders, availability of resources) and to which the results are written back.

Modelling of places and transitions of the Petri net and of the mechanics is done by using the basis building blocks of the DES system (stations) and the corresponding connections (technological chains, branches). In this manner arbitrary structures of Petri nets can be modelled (Weigert and Henlich 2008). The orders and their pre-products, intermediate and final products are represented by the simulator's element "job". Figure 4 shows the described model for the simcron MODELLER. Although it is not a Petri net simulator original, you can recognize the structure of the Petri net in Figure 2. The 5 pre-products (A, B, C, D and E) as well as the 6 possible intermediate products (CD, AC, AB, ACD, BE, ABCD) and the final product (ABCDE) – the finished NC milling table – are represented as round elements. The square elements on the bottom of the model are queues, which include the several orders. The elements below the queues connect these elements to each other for an executable simulation model. The figure is a screen shot of one instant state, where several colours (grayscale) represent different states of stations (busy, waiting or empty).

Figure 4

A special characteristic of the model presented here is the use of alternatives (branches in terms of the DES system) as control variables for optimization: Whereas branches have traditionally been used for dispatching purposes when using them as control variables the pass and therefore the choice of a certain alternative is fixed during a single simulation run and will be reset only by the optimization algorithm for the next run.

In order to increase the portability of the model generating method the use of script code to be executed during the simulation run was kept to an absolute minimum. This also results (at least theoretically) in a performance gain, because all operations run in the (usually much faster) simulation core and not in the (slower) script language interpreter. However, it turned out that this performance gain is smaller than

G. Weigert et al.

12

3

4

5 6

7

8

9

10

11

12 13

14

15

16

17

18 19

20

21

22

23

originally estimated, because it is partially compensated by the increased model complexity caused by avoiding script code.

The optimization was done with 4 different algorithms: Blind Search, Old Bachelor II (Hu, Kahng and Tsao 1995, Beier 2007), Greedy Search, and a Genetic Algorithm. The algorithm determines the value of the control variables (the alternatives for each order) and sends it to the simulation model. Depending on the target value of the complete simulation run, a new control value is chosen randomly. The three algorithms Blind Search, Old Bachelor and Greedy Search distinguish in the probability of acceptance. The Blind Search accepts every target value and Greedy Search accepts only target values, which are better than all values before. The Old Bachelor algorithm is a special method of threshold acceptance. This means, the acceptance of a target value depends on a dynamic threshold, which becomes lower with an increasing cycle number, but can become higher again, if the last cycle was not accepted. The Genetic Algorithm deals with a number of simulation models with different values of their control variables (individuals of a population). New individuals are created by crossover and mutation. The best solutions of each population will survive, but all of the other solutions are discarded. The performance of these algorithms does not differ essentially, so a more detailed description of them is not necessary. Also the influences of the used population size, mutation type and crossover parameters as well as the threshold values of the other heuristics should be neglected here, because the given time limit is very tight. In addition, the goal of these investigations is not a comparison between different simulation-based heuristics but a comparison between simulation-based heuristics and several methods of mixed integer programming.

Figure 5

3.2 MIP-Solver

The goal is the creation of a general mathematical model in form of a Mixed Integer Program (MIP) for optimization of assembly graphs with alternatives and limited resources (resources include articles and mechanics). The problem will be solved by using the CPLEX solver. In order to be able to formulate the problem in a mathematical way, the following data are defined or extracted from the Petri net: Let S be the set of all places of the Petri net and n_s their number. Let T be the set of all pre-products (articles) to be assembled into a final product. n_T defines their number. B_{op} specifies the supply date of the p-th pre-product of type $o \in T$. In total there are n_J orders with due dates d_i $(i = 1, ..., n_J)$ to be assembled by n_M mechanics. The set of all assembly alternatives described by the Petri net is defined by A (see example: $A = \{A1, A2, ..., A5\}$). Let SA_l be the set of all places in alternative $l \in A$. P_{lm} describes the necessary assembling time of place $m \in S$ in alternative $l \in A$. $Z \in S$ defines the destination node (final assembly). Let the order relation of the places in alternative $l \in A$ be specified by O_l (see example: $O_{A4} = \{ \{AC, ACD\}, \{ACD, ABCDE\}, \{BE, ABCDE\} \}$). This results in n_{Ol} , the number of order relations in alternative $l \in A$. Furthermore the set of places which directly access the pre-product $o \in T$, identified as T_O , can be derived. R_{1kl} and R_{2kl} identify the first and second place of the k-th order relation $(k = 1, ..., n_{Ol})$ in alternative $l \in A$ (example: $R1_{2,A4} = ACD$ and $R2_{2,A4} = ABCDE$). For creating a mathematical model the following unknowns are to be defined:

 s_{im} ... Start date of article *i* at place *m*

 $(i = 1, ..., n_J; m \in S)$

 t_i Tardiness of order i ($i = 1, ..., n_J$)

 v_{io} ... Position of order *i* for article o ($i = 1, ..., n_J; o \in T$)

 w_{iop} ... Order *i* requires the *p*-th delivery of article *o*

 $(i = 1, ..., n_J; o \in T; p = 1, ..., n_J)$

International Journal of Production Research

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy

 x_{il} ... Order *i* is produced in alternative *l*

$$(i = 1, ..., n_J; l \in A)$$

 y_{imq} ... Order *i* is assembled at place *m* by mechanic q ($i = 1, ..., n_J$; $m \in S$; $q = 1, ..., n_M$)

 z_{imjn} ...Order *i* (place *m*) is assembled before order *j* (place *n*) (*i*, *j* = 1, ..., *n_J*; *i* ≤ *j*; *i* ≠ *j* ∨ *m* ≠ *n*;

 $m, n \in S$;

 $s_{im} \in \mathbb{R}_+; t_i \in \mathbb{R}_+; v_{io} \in \mathbb{Z}_+; w_{iop} \in \{0, 1\}$

 $x_{il} \in \{0, 1\}$; $y_{imq} \in \{0, 1\}$; $z_{imjn} \in \{0, 1\}$

The described optimization problem can be formulated as follows: Let *K* be a sufficiently large constant. Equation (1) describes the objective function to be minimized. Restriction (2) makes sure that every order can be assembled in exactly one alternative. Equation (3) assures compliance with the order relation of assembly. Equation (4) means that if order *i* is assembled on route *l* no mechanic can carry out assembly at a place which is not part of that alternative. Restrictions (5) and (6) request the assignment of exactly one mechanic to one assembly. Equations (7) and (8) make sure that one mechanic will carry out one assembly step at a time. Restrictions (9) and (10) as well as (15) and (16) define which pre-product is used by order *i*. The equations (11), (12) and (14) convert this information unambiguously into the Boolean unknown w_{iop} , which is used in equation (13) to implement the delayed supply of the individual pre-products. Equation (17) constricts the objective function to be minimized.

G. Weigert et al.

$$\sum_{i=1}^{n_j} t_i \to \min \quad \text{with} \tag{1}$$

$$\sum_{l \in A} x_{il} = 1 \qquad i = 1, ..., n_J \qquad (2)$$

$$s_{i,R1_{kl}} + P_{l,R1_{kl}} - K(1 - x_{il}) \leq s_{i,R2_{kl}} \qquad i \in 1, ..., n_j; l \in A; k = 1, ...$$
(3)

$$Kx_{il} + y_{imq} \leq K \qquad i = 1, \dots, n_J; q = 1, \dots, n_M \\ l \in A; m \in S \setminus SA_l \qquad (4)$$

$$Kx_{il} - \sum_{q=1}^{n_M} y_{imq} \leq K - 1$$

$$i = 1, \dots, n_J; l \in A$$

$$m \in SA_l$$
(5)

$$Kx_{il} + \sum_{q=1}^{n_M} y_{imq} \leq K+1$$
(6)

$$K\left(3 - y_{imq} - y_{jnq} - z_{imjn}\right) + s_{jn} - s_{im} \geq \sum_{l \in A} P_{lm} x_{il}$$

$$i, j = 1, \dots, n_J; i \leq j$$

$$i \neq i \lor m \neq n$$

$$(7)$$

$$K(2 - y_{imq} - y_{jnq} + z_{imjn}) + s_{in} - s_{jm} \ge \sum_{l \in A} P_{ln} x_{jl} \qquad \int m, n \in S; q = 1, ..., n_M$$
(8)

$$K z_{imjn} + v_{io} - v_{jo} \ge 1$$

$$K (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$k (1 - z_{imjn}) + v_{jo} - v_{io} \ge 1$$

$$K(1-w_{iop})+v_{io} \geq p$$

$$K(1-w_{iop})+p \geq v_{io}$$

$$i=1,...,n_{J}; o \in T$$

$$p=1,...,n_{J}$$

$$(11)$$

$$(12)$$

$$K(1-w_{iop}) + s_{im} \geq B_{op} \qquad i = 1, \dots, n_J; o \in T \\ p = 1, \dots, n_J; m \in T_O \qquad (13)$$

$$\left.\begin{array}{cccc}
\sum_{p=1}^{n_{j}} w_{iop} &= 1 \\
v_{io} &\geq 1
\end{array}\right\} \quad i = 1, \dots, n_{j}; o \in T \quad (15)$$

(12)

$$v_{io} \leq n_J$$
(10)
(16)

$$s_{iZ} + \sum_{l \in A} x_{il} \cdot p_{lZ} - d_i < t_i \qquad i = 1, ..., n_J \qquad (17)$$

 International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy

Due to the large number of unknowns and restrictions the computing time for the exact solution is very high even with small problems. It has to be marked, however, that in praxis there are at least some restrictions which considerably simplify the optimization problem under investigation. These include for instance:

- A mechanic finishes an order alone (no order is handed from one mechanic to another).
- A mechanic can only start a new order after having finished the previous one (because the workspace is limited).
- There is a fixed allocation of pre-products to final products.

In these cases an optimal solution can be found even for larger problems. Another approach for reducing problem complexity, without simplifications in model constraints, is the implementation of a time-window approach, identified as "MIP cyclic" in the following. Here, a decomposition of the optimization problem into smaller sub-problems (grouping of orders) and their sequential solution is performed. Then the following objective function is used for every sub-problem:

$$K\sum_{i=1}^{n_J} t_i + \sum_{i=1}^{n_J} \left(s_{iZ} + \sum_{l \in A} x_{il} \cdot P_{lZ} \right) \to \min$$
(18)

instead of equation (1). Apart from minimization of tardiness this delivers at the same time the earliest possible completion of all orders. This leads to an initial situation for the respective next step of the time-window approach, which is in each case beneficial. So, this method represents a type of hybrid between simulation and solver.

For further complexity reduction, some restrictions and variables, e. g. equations (7) and (8), and z_{imjn} respectively, could be made redundant by defining suitable heuristics. Also, the omission of restriction (4) can result in a performance gain as well.

4 Results

In this section the results of the different optimization approaches to the described assembly problem are presented for two different manufacturing objectives. The first objective is the already introduced reduction of tardiness costs, which play an important role for all kinds of due-date dominated manufacturing problems (like supply chains). The second objective is the reduction of average lead time. Here the delay times of jobs in the manufacturing line are minimized which is important in cases where, due to the large size of the products, assembly space is limited, e.g. in plant engineering or the airspace industry.

Tardiness optimization:

The exact MIP approach was able to find the optimal solution for the NC table problem (for $n_J = 6$) for objective total tardiness (1). This allows a comparison to the quality of the results from the simulation-based optimization. On average it was able to find the exact optimum after approximately 5 minutes of computing time (on an ordinary PC, 2 GHz, 2 GB RAM). A Gantt chart representing the optimal assembly plan is shown in Figure 6. It contains all relevant information such as supply dates of the respective pre-products and delivery dates of the orders. The assembly steps of one particular order are printed in the same colour (gray-scale) of the box identifies the chosen assembly alternative. As can be seen, in the optimal solution four different alternatives (A2, A3, 3 times A4, and A5) are used. This shows clearly what big potential for optimization the flexibility of assembly processes with alternatives has to offer.

Figure 6

Now in order to further compare simulation-based and solver-based methods, the problem dimension (number of orders) was increased gradually. As criterion for online capability an upper limit of 1 minute for the optimization time was chosen. Table 2 compares the results of the different methods with the problem size. As can be seen, the exact MIP approach finds the optimum for 5 and 6 orders. However, for $n_J > 6$ the quality of the solution found after 1 minute decreases rapidly. The reason for this is the enormously fast growing number of precedence variables. Already at $n_J = 10$ finding a valid solution becomes difficult. Here the simulation-based optimization shows its advantages because every simulation run yields a valid solution. Although this method exceeds the chosen limit of 1 minute too, the results for bigger problem dimensions are still clearly better as with the exact MIP approach.

Table 2

A combination of simulation and solver in the form of the cyclic MIP approach consistently yielded the best results. The orders were grouped into "batches" of up to 3 orders per optimization run. Thus the sub-problems remained always exactly solvable independent of the total number of orders to be scheduled.

Average lead time optimization:

To adapt the MIP model concerning the objective average lead time equation (1) has to be substituted by (19). This also holds for the cyclic MIP approach. Thereby r_i described the start of the first possible processing of order *i*. The definition of variable t_i as well as the constraints set (17) can be omitted.

$$\frac{1}{n_J} \sum_{i=1}^{n_J} \left(s_{iZ} + \sum_{l \in A} x_{il} \cdot P_{lZ} - r_i \right) \to \min$$
(19)

As in the case of tardiness optimization the exact MIP approach was able to find the optimal solution for the given problem up to 6 orders ($n_J = 6$). In the Table 3 the results for objective average lead time concerning the best MIP approach (MIP-cyclic) and the best simulation-based optimization approach (Genetic Algorithm) are displayed.

Table 3

The reason why the Old Bachelor, Greedy Search, and Genetic Algorithm are not significantly better than Blind Search is the absence of a meaningful distance measure for the control variable "choice of alternatives" (value 1 meaning A1 was chosen, etc.). For that reason in the second optimization series (average lead time) only the Genetic Algorithm was used. Table 4 shows the (order specific) correlation coefficients for the objective functions of total tardiness and lead time for the 6 job problem. As can be seen, they are significantly smaller than 1, thus eliminating the advantages of the three algorithms over Blind Search.

Table 4

This is visually confirmed by a plot of the objective function (total tardiness, z-axis) over the control variables of jobs 1 and 2 in Figure 7 which shows a relatively scattered distribution.

Figure 8 shows the average lead time over the same control variables, while Figure 9 shows all three investigated objective functions in one diagram.

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy

Figure 7,8,9

5 Outlook

The main objective of the presented work was the evaluation of the suitability of different approaches for planning and controlling assembly processes. In addition, it should be shown that assembly alternatives include a big optimization potential. A practical application on NC table assembly of milling machines is currently in preparation. It requires the expansion of the algorithms and of the underlying meta-model, which permits the processing of more than 16 steps per alternative and more than 20 pre-products. This problem dimension excludes the use of MIP to reach an exact solution. First results have shown that both the cyclic MIP approach as well as the simulation-based optimization is suitable for practical adoption. It remains to be explored if and how a combination of these two methods is possible. The potential of this approach exists. One possibility could be to use a MIP solver for subsets of the model and to transfer the solution into heuristic simulation-based systems. Also the usage of improved "intelligent" search methods as well as multicriterial optimization will be continued to be investigated.

References

Beier, Eik, Entwicklung und Bewertung simulationsgestützter Optimierungsverfahren für die

Halbleiterfertigung. Technische Universität Dresden, Masterthesis, Fachrichtung Mathematik, 2007.

Brucker, Peter, Scheduling Algorithms. Berlin, Springer Verlag, 2007.

Chaudhry, S. and Luo, W., Application of genetic algorithms in production and operations management: A review (International Journal of Production Research 43(19), pages 4083-4101), 2005.

Ghosh, Soumen and Gagnon, Roger J., A comprehensive literature-review and analysis of the design, balancing and scheduling of assembly systems (International Journal of Production Research 27(4), pages 637-670), 1989.

Guo, Z. X., Wong, W. K., Leung, S.Y.S., Fan, J.T. and Chan, S.F., A genetic algorithm based optimization model for solving the flexible assembly line balancing problem with work-sharing and workstation Revisiting (IEEE Transactions on Systems Man and Cybernetics Part C - Applications and Reviews 38(2), pages 218-228), 2008.

Homem de Mello, L.S. and Sanderson, A.C., AND/OR graph representation of assembly plans. In: Robotics and Automation, IEEE Transactions on Publication, Apr 1990 Volume: 6, Issue: 2, pp: 188–199.

Hu, T.C., Kahng, A.B. and Tsao, C.A., Old Bachelor Acceptance: A New Class of Non-Monotone Threshold Accepting Methods, ORSA Journal on Computing 7, No.4, 417–425, 1995.

Klemmt, A., Horn, S., Weigert, G. and Wolter, K.-J., Simulation-based optimization vs. mathematical programming: A hybrid approach for optimizing scheduling problems (International Journal of Robotics and Computer-Integrated Manufacturing 25(6), pages 917-925), 2009.

König, Rainer and Quäck, Lothar, Petri-Netze in der Steuerungstechnik. Berlin, Verlag Technik, 1988.

Lambert, Alfred J.D., Generation of assembly graphs by systematic analysis of assembly structures. European Journal of Operational Research 168 (2006) 93.

Mînzu, V., Cernega, D. and Henrioud, J. M., Linguistic Model and a Control Problem for Assembly Workstation. In: Proceedings of the 4th IEEE International Symposium on Assembly and Task Planning, Fukuoka, Japan. May 28–29 2001.

Niu, Xinwen, Ding, Han and Xiong, Youlun, A hierarchical approach to generating precedence graphs for assembly planning. International Journal of Machine Tools & Manufacture 43 (2003) 1473–1486.

Prenting, T.O. and R.M. Battaglin, R.M., The precedence diagram: a tool for analysis in assembly line balancing. The Journal of Industrial Engineering 15 (4), 208–213, 1964.

Weigert, Gerald and Henlich, Thomas, Using Petri Nets for DES modeling and optimization of assembly processes In: 18th International Conference FAIM'08, Flexible Automation & Intelligent Manufacturing. Skövde, Sweden, June 30–July 2 2008, Proceedings pp. 1228-1235.

Page 13 of 27

International Journal of Production Research

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy

Zhou, MengChu and Venkatesh, Kurapati, Modeling, Simulation, and Control of Flexible Manufacturing Systems: A Petri Net Approach (Series in Intelligent Control and Intelligent Automation, Vol. 6). World Scientific, 1999.

<text><text><text>



International Journal of Production Research

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy



G. Weigert et al.



Figure 3: Entity-relationship diagram of assembly meta-model database

International Journal of Production Research

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy



Figure 4: Simulation model of assembly of NC milling tables for the simcron MODELLER (cut-out)



International Journal of Production Research



Jure 5: Optimization .

Page 19 of 27

International Journal of Production Research

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy



Figure 6: Optimal assembly plan for NC table assembly.



International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy







Page 23 of 27

International Journal of Production Research

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy

Assembly step	<i>A</i> 1	A2	A3	<i>A</i> 4	A5	duration (h:m)
A+B			\checkmark			3:20
A+C	\checkmark			\checkmark		3:20
B+E				\checkmark	\checkmark	2:30
C+D			\checkmark			5:00
AC+D	\checkmark			\checkmark		2:30
A+C+D		\checkmark			\checkmark	4:10
ACD+B	~	✓				6:40
AB+CD			\checkmark			4:10
ABCD+E	\checkmark	✓	\checkmark			5:50
ACD+BE				\checkmark	\checkmark	8:20
Total (h:m)	18:20	16:40	18:20	16:40	15:00	

Table 1: Duration of assembly steps and of alternatives A1-A5

	5	(7	0	0	10
Total tardiness/h レ	$n_J=5$	0	/	8	9	10
Blind Search	5	7.5	14.16	18.3	20.83	23.33
Old Bachelor 2	5	7.75	13.92	18.67	22.67	25.83
Greedy Search	5.83	8.33	13.33	17.5	20	22.5
Genetic Algorithm	5	7.58	12.5	16.75	19.5	22
Exact MIP	<u>3.33</u>	<u>5.83</u>	11.66	63.33	135.8	259.1
Cyclic MIP	3.33	5.83	10.83	15	17.5	20

Table 2: Comparison of different optimization methods with respect to problem size (number of orders) when optimization time is limited to 60 seconds, optimum

Page 25 of 27

International Journal of Production Research

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy

Table 3: Comparison of two different optimization methods with respect to problem size (number of orders) when optimization time is limited to 60 seconds, optimum

Number of orders \rightarrow	5	(7	0	0	10
Average lead time/h ע	$n_J=5$	0	/	8	9	10
Genetic Algorithm	17.8	19.0	20.1	20.0	20.6	21.0
Cyclic MIP	17.5	18.6	19.5	19.4	20.1	20.5

http://mc.manuscriptcentral.com/tprs Email: ijpr@lboro.ac.uk

Table 4: Correlation coefficients for total tardiness and average lead time				
Control variable for	Correlation coefficient	Correlation coefficient		
order	(tardiness)	(lead time)		
1	-0.02	-0.15		
2	-0.02	-0.30		
3	-0.002	0.16		
4	-0.009	-0.13		
5	0.04	-0.15		
6	0.07	-0.33		

International Journal of Production Research

International Journal of Production Research, Vol. X, No. Y, December 2010, xxx-yyy

2	
3	Figure 1: Ambiguity in modelling assembly alternatives
4	rigure 1.7 moleculty in modeling assembly alternatives
5 6 7	Figure 2: Assembly process with 5 assembly alternatives as a Petri net
7 8 9	Figure 3: Entity-relationship diagram of assembly meta-model database
10 11	Figure 4: Simulation model of assembly of NC milling tables for the simcron MODELLER (cut-out)
12 13	Figure 5: Optimization cycle of simulation-based optimization
14 15 16	Figure 6: Optimal assembly plan for NC table assembly.
17	Figure 7: Scatter plot of objective function total tardiness
18	Figure 8: Scatter plot of objective function average lead time
19 20	
21	Figure 9: Combined scatter plot of objective functions
22	
23 24	Table 1: Duration of assembly steps and of alternatives A1–A5
25	Table Q. Comparison of different estimization methods with respect to problem size (number of
26 27	Table 2. Compansion of different optimization methods with respect to problem size (number of
28	orders) when optimization time is limited to 60 seconds, <u>optimum</u>
29 20	Table 3: Comparison of two different optimization methods with respect to problem size (number of
31	orders) when optimization time is limited to 60 seconds, optimum
32	Table 4: Correlation coefficients for total tardiness and average load time
33 34	Table 4. Correlation coefficients for total tardiness and average lead time
35	
36 37	
38	
39	
40 41	
42	
43	
44 45	
46	
47 48	
49	
50	
51 52	
53	
54	
55 56	
57	
58 50	
60	
	http://mc.manuscriptcentral.com/tprs_Email: iipr@lboro.ac.uk