Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, eds.

FAST HEURISTICS FOR MAKING QUALIFICATION MANAGEMENT DECISIONS IN WAFER FABS

Denny Kopp
Lars Mönch

Department of Mathematics and Computer Science
University of Hagen
Universitätsstraße 1
Hagen, 58097, GERMANY

ABSTRACT

We discuss qualification management problems arising in wafer fabs. Steppers need to be qualified to process lots of different families. A qualification time window is associated with each stepper and family. The time window can be reinitialized as needed and can be extended by on-time processing of lots from qualified families. Due to the NP-hardness of the qualification management problem, heuristic approaches are required to tackle large-sized problem instances arising in wafer fabs in a short amount of computing time. We propose fast heuristics for this problem. The binary qualification decisions are made by heuristics while the real-valued quantities for each family and stepper are determined by linear programming. We conduct computational experiments based on randomly generated problem instances. The results demonstrate that the proposed heuristics are able to compute high-quality solutions using short computing times.

1 INTRODUCTION

Semiconductor manufacturing deals with producing integrated circuits (ICs) on wafers, thin discs made from silicon or gallium arsenide. Wafer fabs belong to the most complex existing manufacturing systems. They have several hundreds often extremely expensive machines also known as tools. Machines that provide the same functionality are organized in tool groups. Tool groups form work areas. Among the machines are so-called batch processing machines that can process several lots at the same time (Mönch et al. 2013). A lot is a group of wafers that serves as a moving entity in a wafer fab. A diverse product mix that often changes over time is typical for wafer fabs. Frequent tool breakdowns due to the complicated machinery also occur in wafer fabs. Wafer fabs are highly reentrant job shops, i.e., lots revisit certain tool groups up to 40 times. The reentrant process flows are caused by the fact that wafers are processed layer-by-layer in a wafer fab.

Steppers are the most expensive tools in wafer fabs. Therefore, it is likely that the stepper tool group serves as the planned bottleneck of a wafer fab. Steppers belong to the photolithography work area of wafer fabs. The circuit pattern of a product layer is transferred on steppers from a mask onto the surface of a wafer using ultraviolet light exposure. The machines of a wafer fab must be qualified to process wafers. Two qualification types are differentiated. On the one hand, principal qualifications require that a program associated with a process step, a so-called recipe, is executed on a tool to qualify it for this process step. On the other hand, quality-driven qualifications are required to increase yield, the fraction of raw wafers released into a wafer fab that finishes production as salable devices at their original specification. Quality-driven qualification activities for tools are performed to improve yield, an important key performance indicator in wafer fabs. A tool without a principal qualification cannot be used by the corresponding process step. However, process steps can be performed on a tool with missing quality-driven qualification but this might lead to rework or even scrapped wafers. In the specific context of the present paper this means that certain parameters of the steppers have to be adjusted to obtain high-quality wafers from the steppers for each mask layer. Qualifications are expensive and time-consuming. Scarce bottleneck capacity is wasted when more steppers are qualified than needed. Therefore, qualification decisions are important. In the present paper, we reconsider the qualification management problem studied in a series of papers by Kopp et al. (2016), Kopp and Mönch (2018), and Kopp et al. (2019). While qualification management decisions are made by a Mixed Integer Linear Program (MILP) in these papers, we propose heuristics that are hybridized with linear programming (LP) in the present paper. These heuristics are much faster than MILP-based approaches, especially when the number of steppers is large, but provide high-quality solutions at the same time.

This paper is organized as follows. We will describe the problem at hand in the next section. This includes also a discussion of related work. We present the LP formulation in Section 3. The heuristics are described in Section 4. The computational results are presented, analyzed, and discussed in Section 5. Conclusions and future research directions are discussed in Section 6.

2 PROBLEM SETTING

2.1 Problem Statement

The qualification management problem from Kopp et al. (2016) is briefly recalled. It can be described as follows:

1. **Planning situation:** A finite planning horizon of length T divided into discrete periods of length Δ is given.
2. **Lot families:** We consider lot families where a family is formed by all lots of a product that require the same reticle for manufacturing them on a stepper. Therefore, we have a family for each product and mask layer
3. **Targets:** The number of wafers for each family that must be processed on the steppers in a period are called target quantities. The target quantity for family f in period t is D_{ft}.
4. **Stepper dedications:** There are m steppers. A stepper might be only able to run wafers of specific families, i.e., dedications occur.
5. **Stepper qualification:** Each stepper has to be qualified for a family before wafers of this family can be processed on the stepper. A qualification time window of length Δ_{fk} is associated with family f and stepper k. The quantity Δ_{fk} is an integer multiple of Δ.
6. **Expiration and extension of the current qualification:** If no wafers of family f are processed on a qualified stepper within the time window, the qualification of this stepper for the family will be expired. The qualification time window for family f on stepper k can be extended by on-time processing of wafers of family f on k. This means that stepper k will be qualified for family f until the end of period $t+\Delta_{fk}$ if at least a single wafer of family f is processed on stepper k in period t.
7. **Requalification:** Each stepper can be requalified for family f on stepper k by performing a qualification activity. Requalification activities are expensive and time-consuming since they require the processing of a send-ahead wafer (SAW) on k. A SAW is taken from a mother lot. An exposure step on the stepper and additional development and measurement steps are carried out for the SAW. The stepper is qualified for the family when the measurement step for the SAW is successful (Akçali et al. 2001; Mönch et al. 2001).

A MILP formulation is proposed for this problem by Kopp et al. (2016). The MILP has a cost-based objective function that considers qualification penalties and backlog and inventory holding cost for the target quantities. The main decision variables are the number of wafers to be processed on the individual steppers and the qualifications to be performed on the different steppers in the periods of the planning horizon. The former decision variables are continuous while the latter ones are binary. It is shown by Kopp et al. (2019) that the qualification management problem is NP-hard. Therefore, only small-sized MILP instances can be solved using a reasonable amount of computing time. Therefore, we look for efficient heuristics in the present paper. We are interested in designing fast algorithms that choose the binary decision variables in a heuristic manner while the wafer quantities are determined by linear programming after the qualification decisions are made.

2.2 Discussion of Related Work

We will discuss related work with respect to the qualification management problem studied in this paper and with respect to hybridizing heuristics with mathematical programming. A principal qualification management problem for steppers is discussed by Ignizio (2009). A MILP is applied to make qualification decisions. Substantial cycle time and qualification cost reductions are observed when the MILP is used for decision-making. A principal qualification management problem for semiconductor backend facilities is discussed by Fu et al. (2010). The qualification management problem is formulated as a MILP taking into account deterministic demand of the entire backend facility. The MILP is later extended by Fu et al. (2015) to a stochastic programming approach to deal with uncertain demand. A two-stage stochastic programming approach is proposed by Chang and Dong (2017) to tackle a qualification management problem for tool groups motivated by process conditions found in semiconductor manufacturing. The uncertainty of the offered capacity of a tool group is considered. Tool breakdowns or uncertainty in qualification times cause this uncertainty. Lagrangean relaxation is applied to tackle this problem. Overall, it seems mathematical programming-based solution techniques are predominant for qualification management. However, based on the computational experiments performed in (Kopp et al. 2016; Kopp et al. 2019) we know that a MILP approach is too time-consuming for large-sized problem instances of the quality-driven qualification management problem considered in the present paper.

While heuristics are appropriate to deal with combinatorial optimization problems, i.e. discrete problems, mathematical programming approaches offer some advantage to compute the values of continuous decision variables. Recently, iteratively working heuristics or metaheuristics are hybridized with mathematical programming approaches, cf. (Maniezzo et al. 2010; Talbi 2016; Fischetti and Fischetti 2018). The heuristic or metaheuristic chooses the values of the binary or integer-valued decision variables, while an LP solver is used to determine the objective function value for prescribed values of the integer-valued decision variables. In (Almeder 2010), for instance, a max-min ant system is proposed to make setup decisions for a multi-level capacitated lot-sizing problem. The max-min ant system is hybridized with a commercial solver. Another example is Kim and Shin (2015) where a time-based decomposition scheme is proposed that is hybridized with a local search technique. However, to the best of our knowledge, such hybrid approaches are not applied so far to the qualification management problem studied in this paper.

3 HEURISTIC APPROACHES

3.1 LP Formulation for Existing Qualification Decisions

We assume that the values of the binary decision variables that are used to model qualification decisions in the MILP approach proposed by Kopp et al. (2019) are already known. The remaining LP model determines which quantities for each family have to be processes in a period on a specific tool. The following indices and sets are used in the formulation:

$f = 1, \ldots, F$ family index
$k = 1, \ldots, m$ tool index
$t = 1, \ldots, T$ period index.

The following parameters will be used within the model:

D_{ft}: target for family f wafers in period t (in wafers)
C_{kt}: capacity of stepper k in period t (in minutes)
p_{fk}: processing time of a single wafer from family f on stepper k (in minutes)
b_f: backlog cost for family f (per wafer)
h_f: inventory holding cost for family f (per wafer)
γ_{fk}: allowed deviation of the load on tool k from the average load on the steppers in period t (in minutes)
$\tilde{\Delta}_{fk}$: remaining number of periods at the beginning of the planning horizon until stepper k will lose the qualification for processing wafers of family f
θ_{fk}: length of the total qualification window for family f on stepper k (in periods), this quantity accounts for current qualifications and new qualifications

$$y_{fkt} = \begin{cases} 1, & \text{if tool } k \text{ is qualified for family } f \text{ in period } t \\ 0, & \text{otherwise} \end{cases}$$

$$xr_{fk} = \begin{cases} 1, & \text{if tool } k \text{ is qualified for family } f \text{ at the beginning of the planning horizon and } \tilde{\Delta}_{fk} \le T \\ 0, & \text{otherwise} \end{cases}$$

$$R_{fk} = \begin{cases} \tilde{\Delta}_{fk}, & \text{if } \tilde{\Delta}_{fk} \le T \\ 1, & \text{otherwise} \end{cases}$$

$$xq_{fk} = \begin{cases} 0, & \text{if } \theta_{fk} < T \\ 1, & \text{otherwise} \end{cases}$$

$$TQ_{fk} = \begin{cases} T - \theta_{fk}, & \text{if } \theta_{fk} < T \\ 1, & \text{otherwise.} \end{cases}$$

The following decision variables are used within the MILP:

x_{fkt}: number of processed wafers of family f on tool k in period t
B_{ft}. backlog quantity of family f in period t
I_{ft}: inventory quantity of family f in period t.

The qualification management problem can be stated as follows:

$$\min \sum_{t=1}^{T} \sum_{f=1}^{F} \left(b_f B_{ft} + h_f I_{ft} \right), \tag{1}$$

subject to

$$\sum_{k=1}^{m} x_{fkt} + I_{f,t-1} + B_{ft} = I_{ft} + D_{ft} + B_{f,t-1}, \qquad f = 1,\ldots,F,\ t = 1,\ldots,T, \tag{2}$$

$$\sum_{f=1}^{F} p_{fk} x_{fkt} \le C_{kt} \ , \ k = 1,\ldots,m \ , \ t = 1,\ldots,T \ , \tag{3}$$

$$\left| \frac{1}{m} \sum_{f=1}^{F} \sum_{l=1}^{m} p_{fl} x_{flt} - \sum_{f=1}^{F} p_{fk} x_{fkt} \right| \le \gamma_{kt} \, , k = 1,\ldots,m \ , \ t = 1,\ldots,T \ , \tag{4}$$

$$p_{fk} x_{fkt} \le C_{kt} y_{fkt} \, , \ f = 1,\ldots,F \ , \ k = 1,\ldots,m \ , t = 1,\ldots,T \ , \tag{5}$$

$$xr_{fk} \le \sum_{t=1}^{R_{fk}} x_{fkt} \, , \ f = 1,\ldots,F \ , k = 1,\ldots,m \ , \tag{6}$$

$$y_{ftk} - xq_{fk} \le \sum_{\tau=t}^{t+T-TQ_{fk}} x_{fk\tau} \, , \ f = 1,\ldots,F \ , k = 1,\ldots,m \ , \ t = 1,\ldots,TQ_{fk} \ , \tag{7}$$

$$x_{fkt}, Q_{fkt}, B_{ft}, I_{ft} \ge 0 \ , \ f = 1,\ldots,F \ , \ k = 1,\ldots,m \ , t = 1,\ldots,T \ . \tag{8}$$

The objective function (1) to be minimized is the sum of the backlog and inventory holding cost. Constraints (2) serve as inventory balance equations. A capacity restriction for each tool is given by constraint set (3). The absolute allowed deviation of the load on tool k from the average load on the tools in each period t is modeled by constraint set (4). Constraint set (5) ensures that a wafer of a certain family can only be processed on a stepper if the stepper is qualified for this family. Constraint set (6) enforces that a requalification occurs within the remaining planning window. Note that the definition of the parameter xr_{fk} makes sure that a processing of wafers of family f only takes place if the qualification expires before. At least a single wafer that belongs to family f must be produced in this situation to extend the time window. The constraint set (7) is required to initiate production and related extension of the qualification if the total qualification time window is smaller than the planning horizon T . We abbreviate the LP (1)-(8) for qualification management by LP-QM in the rest of the paper.

3.2 Heuristics

3.2.1 Overall Principles

The problem instances and the decision variables are the same for the MILP approach and the heuristics. However, the heuristics have to make the qualification decisions, i.e., when is on which stepper for which family a qualification planned. This means that the qualifications are fixed. As a result, it remains the allocation problem LP-QM which is easy to solve. To incorporate the time windows into the LP-QM formulation, the constraint sets (6) and (7) are added which require adjusted data that are based on the qualification decisions. It is ensured by these constraints that all qualifications will be maintained by on-time processing, i.e., no qualification can expire within the planning horizon if there is enough capacity. However, if a time window is larger than T, there is no need to consider this qualification.

Two heuristics are proposed in the rest of this subsection. Qualifications will be iteratively added in both heuristics while a first fit strategy is applied. The total cost for a given solution must be calculated. Qualification penalties for performing qualifications as in the MILP approach from Kopp et al. (2016) can be derived from the qualification decisions from the heuristic. Moreover, the objective function value of the LP-QM formulation are the backlog and inventory holding cost. The total cost (TC) is the sum of the backlog and inventory holding cost and the qualification cost (see Kopp et al. 2016). The heuristics terminate if no more improvement is obtained by adding additional qualifications.

3.2.2 Family-based Heuristic

The main idea of the family-based heuristic, abbreviated by FBH, is to ensure that at least one qualified stepper exists for each family. It is a two-phase heuristic. Within phase 1 each family will be covered by at least one qualified tool. In each iteration the tools and families are sorted in non-increasing order with respect to available capacity and backlog and inventory cost, respectively, using the solution from the corresponding LP-QM instance. Each qualification to be placed considers the first family without any qualified stepper and the first stepper which is able to process wafers from this family. It is not checked whether this additional qualification leads to an improvement or not. The qualifications are all planned for the first period, i.e., we have $y_{ftk} = 1$. The first phase of the FBH is summarized in the following procedure:

FBH – Phase 1 Procedure

1. **Initialize** the procedure by solving the LP-QM model taking into account initial qualifications.
2. **Repeat** the following steps until for all families exist at least one qualified tool.
3. **Determine capacity and cost** by computing the available capacity for each tool and the backlog and inventory holding cost for each family based on the last solved LP-QM instance.
4. **Sort** the families in non-increasing order with respect to cost and the tools in non-increasing order with respect to available capacity.
5. **Select** the first family from the sorted family list which contains only families that do not have any qualified tool.
6. **Iterate** over the sorted tool list.
7. **Set a qualification** by doing the following: If tool k can be qualified for family f then
    a. Set $y_{fk1} := 1$
    b. Do the preprocessing for LP-QM based on the problem instance
    c. Solve the resulting LP-QM instance,

    otherwise try to update the current tool. Continue the iteration in Step 6 if such a tool exists.
8. Go to Step 2.

In the second phase of the FBH, the families and tools are sorted in the same manner. First, a family is selected. A stepper which can be qualified is then chosen, i.e., dedications and already existing or planned qualifications are considered. An LP-QM instance is solved for each qualification to be placed for which the backlog and inventory holding cost are higher than the qualification penalty to check whether an improvement is achieved or not. If not, the next possible qualification is considered. The procedure terminates if no improvement can be found anymore. When an improvement occurs this qualification is added. The qualification is initially planned for the first period. However, it is not always favorable that a qualification has to be planned for the first period. It is desirable to avoid qualifying a large number of tools at the beginning of the planning horizon since some of them are only required later. For qualifications placed early in the horizon it is more likely that they will expire. Therefore, we iteratively check whether a qualification can be planned for a later period or not. To better support this behavior an increase of the TC value by β is allowed. Here, β is a given parameter. The previous period will be selected as soon as an improvement including β cannot be achieved. The second phase of the FBH can be summarized as follows:

FBH – Phase 2 Procedure

1. **Initialize** this phase by starting from the solution obtained by the first phase.
2. **Repeat** the following steps until no improvement of the objective function TC is found anymore.
3. **Capacity and cost calculation:** Compute the available capacity for each tool and the backlog and inventory holding cost for each family based on the last solved LP-QM instance that leads to a chosen qualification.

4. **Sort** the families in non-increasing order with respect to cost and the tools in non-increasing order with respect to available capacity.
5. **Iterate** over the sorted family list.
6. **Iterate** over the sorted tool list.
7. **Set a qualification** by doing the following: If tool k can be qualified for family f then
   a. Do the preprocessing for LP-QM based on the problem instance using $y_{fk1} = 1$
   b. Solve the resulting LP-QM instance
   c. If an improvement is found
      i. Repeat for each $t = 2,\ldots,T$
      ii. Do the preprocessing for LP-QM based on the problem instance using $y_{fkt} = 1$
      iii. Solve the resulting LP-QM instance
      iv. If no improvement based on the modified TC objective function is found then
      v. Set $y_{f,k,t-1} := 1$ and go to Step 3.
      vi. If an improvement based on the modified TC objective function is found and $t = T$ then set $y_{f,k,T} := 1$ and go to Step 3.
8. Go to Step 6 if there are uncovered tools in tool list, otherwise go to Step 5 if there are uncovered families in the family list.

3.2.2 Tool-based Heuristic

The second heuristic is tool-based. It is abbreviated by TBH. The heuristic is similar to phase 2 of the FBH except that first the tool and then the family are chosen. Hence, Step 5 and Step 6 are exchanged. This is motivated by tools with dedications that are typical for wafer fabs (Mönch et al. 2013). In this situation, it is desirable to prefer steppers with a smaller number of possible families. Apart from phase 1 of the FBH which will not necessarily executed if there are enough initial qualifications at the beginning of the first period, the qualification decisions made by the FBH and the TBH are eventually different especially if many dedications exist.

4 COMPUTATIONAL EXPERIMENTS

4.1 Design of Experiments

We expect that the solution quality depends on the length of the planning horizon, the planned bottleneck utilization (BNU), and the unit qualification penalty term (UQPT). We generate a first set of problem instances based on the design of experiments summarized in Table 1.

Table 1: Design of experiments – set 1.

Factor	Level	Count
BNU	70%, 90%	2
Length of the planning horizon (in periods)	12, 24, 36	3
Qualification unit penalty term scenarios (QS)	low (QL): $q_{fk} = 100$ moderate (QM): $q_{fk} = 800$ high (QH), $q_{fk} = 4000$	3
Number of independent replications per factor combination	6	
Total number of problem instances		108

These 108 problem instances are generated using BNU-dependent D_{ft} values from the MIMAC I simulation model with six steppers and 17 families (cf. Kopp et al. 2016 for details). No initial qualifications are considered. We also consider a second set of 54 instances with initial qualifications and corresponding remaining time windows. This set is obtained by randomly choosing three instances per factor combination from Table 1. Initial qualifications and remaining time windows are randomly added using a number of qualifications that is appropriate for the considered qualification scenario (cf. Kopp et al. 2019). Moreover, four additional sets of large-sized instances are considered where $T=12$ is assumed. The 18 instances of set 3 are formed by doubling three of the instances of set 1 for each factor combination. Therefore, these instances have $m=12$ and $F=34$. Another 18 instances with initial qualifications at the begin of the first period are collected in set 4 which is created from instances of set 2. In addition, the instances of set 3 and set 4 are doubled in the same way for 70% BNU to obtain nine instances which form set 5 and set 6, respectively. Hence, the instances have $m=24$ and $F=68$.

To assess the quality of the solutions found by the MILP we report the relative MIP gap after a given maximum computing time. Therefore, a maximum computing time (MCT) of one hour for each instance of set 1 and set 2 is allowed. Six hours are applied for the sets 3 and 4, whereas 24 hours are used for the sets 5 and 6. Moreover, the average computing time (CT) is reported. The characteristics of the six instance sets are summarized in Table 2. The third column refers to initial qualifications.

Table 2: Summary of the features of the different instance sets.

Set	**(*m*,*F*)**	**initial**	**BNU**	**T**	**QS**	**MCT**	**#Instances**
1	(6,17)	no	70%, 90%	12, 24, 36	QL, QM, QH	1h	108
2	(6,17)	yes	70%, 90%	12, 24, 36	QL, QM, QH	1h	54
3	(12,34)	no	70%, 90%	12	QL, QM, QH	6h	18
4	(12,34)	yes	70%, 90%	12	QL, QM, QH	6h	18
5	(24, 68)	no	70%	12	QL, QM, QH	24h	9
6	(24, 68)	yes	70%	12	QL, QM, QH	24h	9

We are also interested in the cost and penalty term breakdowns. Therefore, we report the ratio of the number of qualifications and maximum possible number of qualifications denoted by Q%. The latter quantity depends on F, m and the dedications, i.e., each family can be processed on half of the steppers. Therefore, the maximum possible number of qualifications are 51, 204, and 816. We report the sum of the backlog quantities over the planning horizon relative to the sum of the target quantities where the backlog from the previous period represents additional demand. This measure is abbreviated by BL%. We also compute the ratio of the sum of inventory holding cost over the different periods and the sum of the target quantities corrected by the inventory from the previous period. This quantity is called INV%.

4.2 Parameter Setting and Implementation Details

The period length is set as $\Delta = 4h$. This leads to $T \in \{2,4,6\}$ days based on Table 1. Moreover, the settings $b_f = 2.5$, $h_f = 1.0$, and $\gamma_{fk} = 60$ minutes are applied. The length of the qualification time windows are $\Delta_{fk} \sim DU[6,18]$, where $DU[a,b]$ refers to a discrete uniform distribution over the interval $[a,b]$. The parameter β in the heuristics is chosen as $\beta := q_{fk}/(2T)$. The MILP model and the LP-QM model (1)-(8) are implemented using ILOG CPLEX 12.7.1.0. The two heuristics are coded in the C++ programming language. All the computational experiments are executed on a PC with a quad core Intel Core i7 3.60 GHz processor and 16GB RAM.

4.3 Computational Results

The computational results for the instances of set 1 are presented in Table 3. The results of all instances are not compared individually. Instead of this, they are grouped according to factor levels. For instance, the results in the second row are average values for all instances with BNU=70%, QL, and $T = 12$. Moreover, the average computing times in seconds and the average MIP gaps are reported.

Table 3: Computational results for the MILP approach – set 1.

BNU	QS	T	Q%	BL%	INV%	CT – MILP (in s)	MILP Gap (%)
70%	QL	12	47	0.9	7	3600	10
	QL	24	60	0.4	4	3600	14
	QL	36	67	0.3	3	3600	16
	QM	12	33	7	16	3600	18
	QM	24	39	3	16	3600	20
	QM	36	49	2	11	3600	28
	QH	12	22	59	18	3600	33
	QH	24	33	7	24	3600	37
	QH	36	35	7	27	3600	37
90%	QL	12	49	4	8	3600	16
	QL	24	60	2	9	3600	24
	QL	36	67	2	8	3600	19
	QM	12	35	8	18	3600	32
	QM	24	42	5	19	3600	34
	QM	36	50	3	16	3600	26
	QH	12	27	41	19	3600	9
	QH	24	34	11	27	3600	9
	QH	36	36	10	29	3600	20

Next, we present the computational results of all small-sized instances from sets 1 and 2 in Table 4 where aggregated results for both BNU levels are shown. All reported computing times are in seconds. Best results for the two heuristics are always marked in bold. In the case of initial qualifications the optimal solution are found for 48 out of the 54 instances when a maximum computing time of one hour per instance is allowed. In addition, the performance deviation of both heuristics from the MILP are shown. The deviation is the ratio of the difference of the objective function value of the heuristic and the MILP and the objective function value of the heuristic. The average computing times are also shown in Table 4. The results for the large-sized instances of the sets 3-6 are shown in Table 5. Only aggregated results for both BNU levels are presented for the sets 3 and 4. The cost and penalty term breakdowns for the instances of set 3 are shown in Figure 1.

4.4 Analysis and Discussion of the Results

We see from Tables 3, 4, and 5 that the computing times of the MILP are large since the problem is NP-hard. As expected, the MIP gaps are large for instances with more families and steppers. With increasing q_{fk} values the MIP gap also increases. The same can be observed for increasing T values because there is more room for improvements in this situation. The MIP gaps decrease if initial qualifications exist. For almost all small-sized instances, the optimality proof is available within the given maximum computing time. However, even with initial qualifications larger problem instances lead to larger computing times or larger MIP gaps. We can see from the computational results that the number of qualifications depends on the applied qualification scenario and on the T value. We see from Figure 3 and Table 3 that a small q_{fk} value results in a large number of qualifications. Hence, in this situation backlog and inventory holding cost are fairly small.

In the case of initial qualifications a larger number of qualifications can be observed. This is expected since the given qualifications are independent from the targets. This means that the targets can require additional qualifications of families of a certain product. At the same time it is possible that more initial qualifications of families exist than needed. Furthermore, we see from the Tables 4 and 5 that the Q% values decrease for problem instances of larger size.

Table 4: Computational results with and without initial qualifications.

Set	QS	T	Q%	BL%	INV%	CT MILP	MILP Gap(%)	Deviation FBH(%)	CT FBH	Deviation TBH(%)	CT TBH
1	QL	12	48	2	8	3600	13	**16.0**	2.5	19.7	3.8
	QL	24	60	1.3	6	3600	19	**13.6**	8.1	37.1	11.4
	QL	36	67	1.1	6	3600	18	**10.4**	16.6	11.6	18.2
	QM	12	34	8	17	3600	25	**10.0**	0.6	11.2	1.7
	QM	24	40	4	17	3600	27	**9.0**	2.1	20.9	5.5
	QM	36	50	2	13	3600	27	**4.5**	5.5	6.3	8.5
	QH	12	24	50	18	3600	21	10.6	0.5	**2.9**	1.3
	QH	24	34	9	25	3600	23	**3.1**	1.0	8.7	4.0
	QH	36	35	8	28	3600	28	**1.6**	1.6	2.9	5.7
2	QL	12	68	1	8	16	0	13.8	0.7	**9.1**	0.8
	QL	24	72	1	6	124	0	**9.5**	2.8	10.6	3.1
	QL	36	75	0	6	391	0	13.3	7.3	**11.8**	8.3
	QM	12	52	5	15	22	0	**3.3**	0.2	5.8	0.3
	QM	24	54	3	14	726	0	**5.2**	0.6	5.9	1.0
	QM	36	56	2	13	1369	2	9.5	1.1	**9.2**	1.7
	QH	12	44	15	21	108	0	9.3	0.2	**5.6**	0.4
	QH	24	45	7	24	1752	0	**4.9**	0.4	5.6	1.0
	QH	36	46	6	24	2453	5	**5.3**	0.6	7.5	1.5

Table 5: Computational results for large-sized problem instances.

Set	QS	T	Q%	BL%	INV%	CT MILP	MILP Gap(%)	Deviation FBH	CT FBH	Deviation TBH	CT TBH
***m* = 12, *F* = 34**											
3	QL	12	26	1	8	21600	22	**14.6**	27.2	19.5	33.7
	QM	12	18	8	18	21600	30	**4.6**	4.3	9.5	13.0
	QH	12	13	47	19	21600	47	8.8	3.1	**3.5**	8.8
4	QL	12	33	1	7	18120	4	21.2	8.6	**13.3**	8.1
	QM	12	27	3	14	21600	12	**8.3**	1.4	11.1	2.9
	QH	12	21	18	24	21600	23	**10.7**	1.2	22.3	2.6
***m* = 24, *F* = 68**											
5	QL	12	14	1	6	86400	30	**8.0**	358.4	32.4	498.6
	QM	12	10	8	17	86400	36	**-3.7**	35.9	4.9	180.4
	QH	12	6	44	23	86400	51	7.7	32.5	**-1.1**	101.4
6	QL	12	17	1	4	86400	12	23.6	248.4	**17.5**	123.2
	QM	12	13	4	16	86400	22	-4.1	8.0	**-5.4**	24.7
	QH	12	10	18	21	86400	36	**5.8**	10.7	19.1	30.1

Considering the absolute number of qualifications per family, we see that there is no relation. The ratio Q% also represents the average number of qualified tools per family compared to the possible number of tools with respect to dedications. For instance, comparing the results of instances with QL, and $T=12$ for set 3 (Q% = 26) and set 5 (Q% = 14), there are 1.56 and 1.68 qualified steppers per family. Besides this, the different BNU levels have no major impact on the number of qualifications apart from, as expected, more qualifications at 90%, QH, and $T=12$ due to the larger target values (see Table 3). Indeed, for a BNU level of 90% more backlog and inventory holding cost occur since it is harder to fulfill the larger targets quantities in the required periods due to the finite tool capacity.

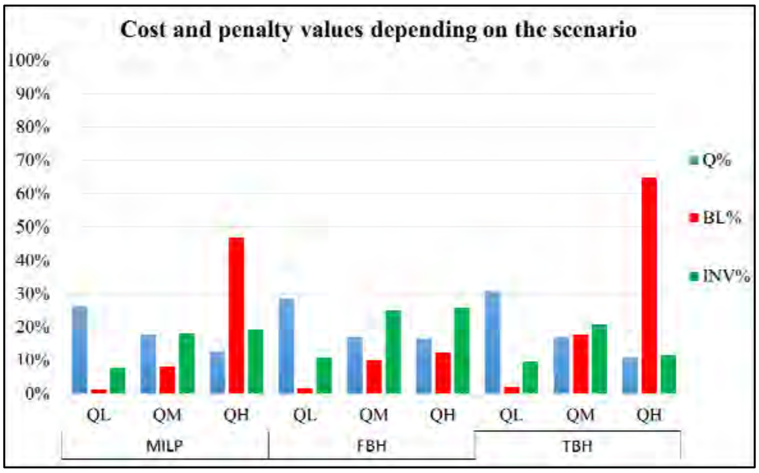

Figure 1: Cost and penalty term breakdowns for instance set 3 depending on the qualification scenario.

We can see from the different tables that the two heuristics outperform the MILP with respect to the average computing times. As expected, the MILP can provide better solutions, but this is only true after long computing times. In contrast, the solution quality of the FBH is under almost all experimental conditions less than 10% worse than the solution of the MILP. Overall, the FBH often performs better than the TBH. However, for the TBH, there are outliers which lead to a worse average performance. Ignoring these outliers, its solution quality is very similar to the one of the FBH. Because of enforcing at least one qualification for all families due to phase 1, the results of the family-based heuristic are worse for large q_{fk} and small T values.

5 CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

In the present paper, we studied a quality-driven qualification management problem in wafer fabs. A qualification time window is associated with each tool and family. On the one hand, it is possible to reinitialize the window after it is expired, but this is expensive and time-consuming. On the other hand, the window can be extended by on-time processing of lots from qualified families. Since the qualification management problem is NP-hard, MILP approaches are too time-consuming for large-sized problem instances. Therefore, we designed and computationally assessed fast heuristics for the qualification management problem. The computational experiments demonstrated that the heuristics are able to make high-quality qualification management decisions using a short amount of computing time.

There are several directions of future research. First of all it seems desirable to extend the proposed heuristics towards a matheuristics, for instance, by designing a greedy randomized adaptive search procedure (GRASP). Second, it would be also interesting to assess the heuristics in a rolling horizon setting based on the approach from Kopp et al. (2019) to study nervousness issues of the different approaches. It would be interesting to use the large-sized wafer fab model proposed by Hassoun et al. (2019).

REFERENCES

Almeder, C. 2010. "A Hybrid Optimization Approach for Multi-level Capacitated Lot-sizing Problems". *European Journal of Operational Research* 200:599–606.

Akçali, E., K. Nemoto, and R. Uzsoy. 2001. "Cycle-time Improvements for Photolithography Process in Semiconductor Manufacturing". *IEEE Transactions on Semiconductor Manufacturing* 14(1):48-56.

Chang, X. and M. Dong. 2017. "Stochastic Programming for Qualification Management of Parallel Machines in Semiconductor Manufacturing". *Computers & Operations Research* 79:49-59.

Fischetti, M., M. Fischetti. 2018. "Matheuristics". In *Handbook of Heuristics*, edited by R. Martí, R. Panos and M. G. Resende, 1-33, Cham: Springer.

Fu, M., M. Haghnevis, R. Askin, J. W. Fowler, and M. Zhang. 2010. "Machine Qualification Management for a Semiconductor Back-end Facility". In *Proceedings of the 2010 Winter Simulation Conference*, edited by L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, 2486-2492. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Fu, M., R. Askin, J. W. Fowler, and M. Zhang. 2015. "Stochastic Optimization of Product–machine Qualification in a Semiconductor Back-end Facility". *IIE Transactions* 47(7):739-750.

Ignizio, J. P. 2009. "Cycle Time Reduction via Machine-to-operation Qualification". *International Journal of Production Research* 47(24):6899-6906.

Hassoun, M., D. Kopp, A. Kalir, and L. Mönch. 2019. A New High-Volume/Low-Mix Simulation Testbed for Semiconductor Manufacturing. In *Proceedings of the 2019 Winter Simulation Conference*, edited by N. Mustafee, K.-H. G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son. In press. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Kim, D. and H. J. Shin. 2015. "A Hybrid Heuristic Approach for Production Planning in Supply Chain Networks". *International Journal of Advanced Manufacturing Technology*, 78, 395-406.

Kopp, D. and L. Mönch. 2018. "A Hierarchical Approach to Qualification Management in Wafer Fabs". In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 3514-3525. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Kopp, D., L. Mönch, D. Pabst, and M. Stehli. 2016. "An Optimization Model for Qualification Management in Wafer Fabs". In *Proceedings of the 2016 Winter Simulation Conference*, edited by T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick, 2610-2620. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Kopp, D., L. Mönch, D. Pabst, and M. Stehli. 2019. "Qualification Management in Wafer Fabs: Optimization Approach and Simulation-based Performance Assessment". *IEEE Transactions on Automation Science and Engineering*. In press.

Maniezzo, V., T. Stützle, and S. Voß (eds.), 2010. "Matheuristics: Hybridizing Metaheuristics and Mathematical Programming". *Annals of Information Systems*, Springer, New York.

Mönch, L., J. W. Fowler, and S. J. Mason. 2013. *Production Planning and Control for Semiconductor Wafer Fabrication Facilities: Modeling, Analysis, and Systems*. New York: Springer.

Mönch, L., M. Prause, and V. Schmalfuss. 2001. "Simulation-based Solution of Load-balancing Problems in the Photolithography Area of a Semiconductor Wafer Fabrication Facility". In *Proceedings of the 2001 Winter Simulation Conference*, edited by B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 1170-1177. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Talbi, E.-G. 2016. "Combining Metaheuristics with Mathematical Programming, Constraint Programming and Machine Learning". *Annals of Operations Research* 240:171–215.

AUTHOR BIOGRAPHIES

DENNY KOPP is is a Ph.D. student at the Chair of Enterprise-wide Software Systems, University of Hagen. He received a MS degree in Business Economics from the Otto-von-Guericke-University Magdeburg, Germany. His research interests include applied optimization and simulation-based production control. His email address is Denny.Kopp@fernuni-hagen.de.

LARS MÖNCH is Professor in the Department of Mathematics and Computer Science at the University of Hagen, Germany. He received a master's degree in applied mathematics and a Ph.D. in the same subject from the University of Göttingen, Germany. His current research interests are in simulation-based production control of semiconductor wafer fabrication facilities, applied optimization and artificial intelligence applications in manufacturing, logistics, and service operations. He is a member of GI (German Chapter of the ACM), GOR (German Operations Research Society), and INFORMS. He is an Associate Editor for the European Journal of Industrial Engineering, Journal of Simulation, Business & Information Systems Engineering, IEEE Robotics and Automation Letters, IEEE Transactions on Semiconductor Manufacturing, and IEEE Transactions on Automation Science and Engineering. His email address is lars.moench@fernuni-hagen.de.