Proceedings of the 2018 Winter Simulation Conference
M. Rabe, A.A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, eds.

ELECTRICITY POWER COST-AWARE SCHEDULING OF JOBS ON PARALLEL BATCH PROCESSING MACHINES

Jens Rocholl
Lars Mönch

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

John W. Fowler

Department of Supply Chain
Management
Arizona State University
Main Campus, PO BOX 874706
Tempe, AZ 85287-4706, USA

ABSTRACT

We discuss a bicriteria scheduling problem for parallel identical batch processing machines in semi-conductor wafer fabrication facilities (wafer fabs). Only jobs that belong to the same family can be batched together. The performance measures of interest are the total weighted completion time and the electricity power cost. Unequal release dates of the jobs are taken into account. The jobs can have non-identical sizes. We provide a Mixed Integer Linear Programming (MILP) formulation for the general setting. Moreover, we analyze the special case where all jobs have the same size, the maximum batch size is an integer multiple of this job size, and all jobs are available at time zero. We prove certain properties of Pareto-optimal schedules for this special case. These properties lead to a MILP formulation that is more tractable than the one for the general setting. We perform computational experiments with the ε-constraint method for both formulations.

1 INTRODUCTION

Semiconductor manufacturing requires one of the most complex manufacturing processes existing today (Mönch et al. 2013). Integrated circuits are produced in semiconductor wafer fabrication facilities (wafer fabs) layer by layer on thin discs made from silicon, so-called wafers. A single wafer may contain up to several thousand integrated circuits. The moving entities in wafer fabs are lots of wafers. We refer to lots in this paper as jobs to align with the scheduling literature. The production process inside a wafer fab can be modeled as a job shop with a number of unusual facets (cf. Mönch et al. 2011). Among them the following ones are the most important:

- A large number of different products with a product mix that changes over time
- Process flows, that might contain up to 800 process steps for the most advanced products
- Processing times ranging from several minutes to 24 hours per job
- Re-entrant process flows, i.e. a job might visit the same machine up to 40 times
- Machine groups, i.e., parallel machines
- A mix of different process types, for instance, single wafer processes and batch processes, i.e., a group of jobs that are processed at the same time on the same machine forms a batch
- Sequence-dependent setup times that can be considerably longer than the processing times

Up to one-third of all operations in a wafer fab are performed on batch processing machines. The corresponding processing times on batch processing machines are very long. Since batch processing machines typically process several jobs at the same time, these machines tend to off-load multiple jobs on machines that are able to process only single jobs. Long queues in front of these serial machines are the result. Hence, an appropriate scheduling of jobs on batch processing machine is crucial for the performance of the entire wafer fab.

The diffusion furnaces in wafer fabs are a typical example of batch processing machines. Diffusion furnaces consume considerably more electricity than most of the other machines in a wafer fab. This is caused by the fact that the diffusion process is a high temperature process that disperses material on the wafer surface. Leading semiconductor manufacturers strive for energy saving and carbon reduction initiatives because 70% of greenhouse gas emission comes from electricity consumption (cf. TSMC 2016; Yu et al. 2017). Therefore, it is desirable to schedule jobs on diffusion furnaces in such a way that a compromise between production-related objectives and energy consumption-aware objectives is reached. However, a recent survey paper shows that sustainability issues are a widely underresearched topic in semiconductor supply chains (Mönch et al. 2018). In the present paper, we are therefore interested in studying a model problem where the total weighted completion time, an important indicator for cycle time performance, and the electricity power cost are considered. We present MILP formulations and show that under some assumptions the ε-constraint can be applied to solve small- and medium-sized problem instances.

The paper is organized as follows. The problem is described and analyzed in the next section. This includes a discussion of related work. MILP formulations for the two situations are provided in Section 3. Moreover, the ε-constraint method is applied in this section too. The results of computational experiments are discussed and analyzed in Section 4. Conclusions and future research directions are presented in Section 5.

2 PROBLEM SETTING AND ANALYSIS

2.1 Problem Description and Structural Properties

The scheduling problem is based on the following assumptions:

1. There are F job families. Due to the different chemical nature of the processes, only jobs of the same family can be batched together.
2. All jobs that belong to family $1 \le f \le F$ have the same processing time $p_{s(j)} \equiv p_f$ where $s(j) := f$ is a mapping that assigns to each job its family.
3. There are $n_f, 1 \le f \le F$ jobs in family f. In total, we have to schedule $n = \sum_{f=1}^{F} n_f$ jobs. Jobs are labeled by $j = 1, \ldots, n$.
4. Each job has a weight w_j that is used to express the importance of the job.
5. The size of job j is s_j. It is measured in number of wafers.
6. There are m identical parallel machines. They are labeled by $k = 1, \ldots, m$. All the machines are available at time $t = 0$.
7. Each job j has a ready time $r_j \ge 0$.
8. All the machines have the same maximum batch size B.
9. Once a batch processing machine is started, it cannot be interrupted, i.e., no preemption of the machine is allowed.
10. We assume that a finite scheduling horizon is divided into periods of equal length, i.e. the periods are labeled by $t = 1, \ldots, T$. The electricity power cost is modeled as a piecewise constant function over the set of periods.

We are interested in the performance measure total weighted completion time (TWC) which is defined as follows:

$$TWC := \sum_{j=1}^{n} w_j C_j , \tag{1}$$

where C_j is the completion time of job j. Note that the TWC measure is a surrogate measure for the weighted cycle time, an important measure in most wafer fabs (Mönch et al. 2013). The second performance measure is the electricity power cost (EPC). When we denote the EPC value in period t by $e(t)$ then we can express the EPC value for schedule S as follows:

$$EPC(S) := \sum_{t=1}^{T} \sum_{k=1}^{m} e(t) z_t^k, \tag{2}$$

where z_t^k is an indicator value that is 1 if a batch is processed in period t on machine k in S and zero otherwise. Using the three-field notation from scheduling theory, the scheduling problem at hand can be represented as follows:

$$P \mid p-batch, incompatible, r_j, s_j \mid ND(TWC, EPC), \tag{3}$$

where P indicates identical parallel machines, *p-batch, incompatible* refers to batch processing machines with incompatible families, r_j to unequal ready times, and s_j to nonidentical sizes of the jobs. The notation $ND(TWC, EPC)$ refers to the set of all Pareto-optimal solutions, i.e., a schedule S is called non-dominated when no other feasible schedule S' exists with $TWC(S') \le TWC(S)$ and $EPC(S') \le EPC(S)$, and at least one of these inequalities is strict. The entire set of all non-dominated solutions for a problem instance is called the Pareto frontier.

Note that problem (3) is NP-hard since the problem $P \mid p-batch, incompatible \mid TWC$ is NP-hard due to Uzsoy (1995). Hence, we have to look for efficient heuristics if we want to tackle large-sized problem instances.

2.2 Discussion of Related Work

We discuss work related to energy-aware scheduling, especially with respect to scheduling models that include parallel batch processing machines and energy-aware objectives or constraints. Energy-aware scheduling currently receives a lot of attention, we refer to the recent survey papers Giret et al. (2015), Merkert et al. (2015), and Gahm et al. (2016).

While there are many papers that deal with single-criterion scheduling problems for batch processing machines, the number of publications for batch scheduling in a multicriteria setting is much smaller. We are only aware of the papers by Reichelt and Mönch (2006) and Mason et al. (2007) where multicriteria batch scheduling problems related to semiconductor manufacturing are studied. There are a few papers that consider scheduling problems with electricity power cost-related constraints for single batch processing machines. Cheng et al. (2014) discuss a scheduling problem for a single batch processing machine where all jobs are ready at time zero. All jobs have the same processing time. Time-of-use (TOU) electricity pricing is assumed. The makespan C_{max} and the total electricity costs are considered as criteria. The ε-constraint method is applied. A heuristic variant of the ε-constraint method for the same scheduling problem is proposed by Cheng et al. (2017). Wang et al. (2016) consider a similar scheduling problem. The problem setting is motivated by a real-world glass making facility. Again the makespan and the total energy cost are taken into account. The energy consumption depends on the furnace temperature choice. The ε-constraint method is applied for small-sized problem instances whereas constructive heuristics are applied to tackle large-sized instances. A case study using data from the real-world manufacturing environment is conducted.

We are also aware of the following two papers that deal with parallel machine environments. Liu (2014) studies the problem $P \mid p-batch, r_j \mid ND(TWT, CO_2)$ where p-batch refers to a situation where the processing time of a batch is determined by the longest processing time of the jobs that form the batch. Moreover, the total weighted tardiness (TWT) and the CO_2 performance measure are considered. A non-dominated sorting genetic algorithm (NSGA)-II approach is proposed for this problem. The later performance measure is derived from the total energy costs by multiplying them with a constant factor. Jia et al. (2017) consider the problem $P \mid p-batch, r_j \mid ND(C_{max}, EPC)$. They solve this problem by a bi-criteria ant colony optimization approach. Overall, we conclude that the problem discussed in this paper is not studied so far.

3 EXACT APPROACHES

3.1 MILP Formulation for the General Case

Next, we present a MILP formulation for scheduling problem (3). We use a time-indexed formulation for the scheduling horizon $t = 1,\ldots,T$. The following indices and sets are used in the model:

$j = 1,\ldots,n$: job indices
$i = 1,\ldots,b$: batch indices
$k = 1,\ldots,m$: machine indices
$t = 1,\ldots,T$: period indices.

The following parameters are included in the model:

w_j : weight of job j
r_j : ready time of job j
s_j : size of job j (in wafers)
B : maximum batch size B (in wafers)
$s(j)$: family of job j
$f(i)$: family of batch i
$p_{s(j)}$: processing time of job j
$p_{f(i)}$: processing time of batch i
$e(t)$: electricity power cost in period t
M : big number.

The following decision variables are part of the model:

$$x_{ij} \begin{cases} 1, \text{if job } j \text{ is assigned to batch } i \\ 0, \text{otherwise} \end{cases}$$

$$y_{itk} \begin{cases} 1, \text{if batch } i \text{ is started in period } t \text{ on machine } k \\ 0, \text{otherwise} \end{cases}$$

C_j : completion time of job j
C_i : completion time of batch i .

The model can be formulated as follows:

$$\min \left(\sum_{j=1}^{n} w_j C_j , \sum_{i=1}^{b} \sum_{t=1}^{T} \sum_{k=1}^{m} \sum_{\tau=t}^{t+p_{f(i)}-1} e(\tau) y_{itk} \right) \tag{4}$$

subject to

$$\sum_{i=1}^{b} x_{ij} = 1 \qquad j = 1,\ldots,n \tag{5}$$

$$\sum_{i=1}^{n} s_j x_{ij} \le B \qquad j = 1,\ldots,n,\ i = 1,\ldots,b \tag{6}$$

$$x_{ij}\left(f(i) - s(j)\right) = 0 \qquad j = 1,\ldots,n,\ i = 1,\ldots,b \tag{7}$$

$$x_{ij} \le \sum_{t=1}^{T}\sum_{k=1}^{m} y_{itk} \qquad j = 1,\ldots,n,\ i = 1,\ldots,b \tag{8}$$

$$C_i \le T + M\left(1 - x_{ij}\right) \qquad j = 1,\ldots,n,\ i = 1,\ldots,b \tag{9}$$

$$\sum_{i=1}^{b}\ \sum_{\tau=t-p_{f(i)}+1}^{t} y_{i\tau k} \le 1 \qquad k = 1,\ldots,m,\ t = 1,\ldots,T \tag{10}$$

$$r_j x_{ij} + p_{f(i)} \le C_i \qquad j = 1,\ldots,n,\ i = 1,\ldots,b \tag{11}$$

$$C_i = \sum_{k=1}^{m}\sum_{t=1}^{T} t\, y_{itk} + p_{f(i)} \qquad i = 1,\ldots,b \tag{12}$$

$$C_i \le C_j + M\left(1 - x_{ij}\right) \qquad j = 1,\ldots,n,\ i = 1,\ldots,b \tag{13}$$

$$C_j \le C_j + M\left(1 - x_{ij}\right) \qquad j = 1,\ldots,n,\ i = 1,\ldots,b \tag{14}$$

$$C_i, C_j \ge 0,\ x_{ij}, y_{itk} \in \{0,1\} \qquad j = 1,\ldots,n,\ i = 1,\ldots,b\ t = 1,\ldots,T,\ k = 1,\ldots,m. \tag{15}$$

We are interested in determining all Pareto-optimal solutions with respect to the TWC and EPC measures. This is represented by (4). The constraints (5) ensure that each job is assigned to exactly one batch. The maximum batch size is respected by constraints (6). Constraint set (7) models the fact that only jobs belonging to the same family can be used to form a batch. The family of a batch is determined by its jobs. If at least one job is assigned to a batch that this batch has to be started in some period on some machine. This is modeled by the constraints (8). The constraints (9) ensure that the completion time of a batch is not larger than the end of the scheduling horizon. It is modeled by constraints (10) that each batch starts at most once before the end of the scheduling horizon. The constraints (11) enforce that the ready time of the jobs are respected, i.e., a batch can only start if all jobs that belong to the batch are ready. The completion of a batch is calculated by equation (12). The constraint sets (13) and (14) ensure that the completion time of a batch and the completion time of the jobs that belong to this batch are the same. The domains of the decision variables are respected by constraints (15).

3.2 MILP Formulation for the Special Case

We analyze the special case where all jobs are ready at time $t = 0$, i.e. $r_j \equiv 0$. Moreover, we assume that all jobs have the same size and that B is an integer multiple of this job size. Note that this problem is still NP-hard. We prove a first property that generalizes an insight from Uzsoy (1995).

Property 1: For each point in the criteria space there is a corresponding Pareto-optimal schedule where all batches except maybe the last scheduled batch of each family are fully loaded, i.e., the batch size is equal to the maximum batch size.

Proof: We assume that there is a Pareto-optimal schedule S that contains a batch ρ of family f that is not full and this batch does not have the largest start time among the batches of family f. We construct a new schedule S' from S by moving a job from a batch of family f with the largest start time to batch ρ. Let ρ' be the batch of family f with the largest start time. Since the completion time of ρ in S' is smaller than in S we obtain $TWC(S') < TWC(S)$. At the same time, we have $EPC(S') \le EPC(S)$ since the electricity power cost only depends on the start time of a batch which is unchanged for ρ and all batches except maybe for batch ρ'. If ρ' contains in S only a single job then ρ' is empty in S' and its EPC value is zero. Overall, we have $TWC(S') < TWC(S)$ and $EPC(S') \le EPC(S)$ which contradicts with the Pareto-optimality of schedule S. ■

It follows from Property 1 that a Pareto-optimal schedule exists for each point of the criterion space where the number of batches in a family with l jobs is $\lceil l/B \rceil$. We show a second property where the structure of batches in Pareto-optimal schedules is considered.

Property 2: For each point of the criteria space there is a Pareto-optimal schedule where for each pair of batches ρ_1 and ρ_2 of the same family with completion times $C_1 \le C_2$ the weight of each job belonging to ρ_1 is not smaller than any weight of a job of ρ_2.

Proof: We assume that there is a Pareto-optimal schedule S where batch ρ_1 contains a job j with a weight $w_j < w_i$ where job i belongs to batch ρ_2. We obtain a new schedule S' by exchanging job j and job i. We then have $TWC(S') < TWC(S)$ and $EPC(S') = EPC(S)$ which contradicts to the Pareto-optimality of schedule S. ■

Based on Property 2, we sort the jobs in each family with respect to non-increasing values of the job weights. These job sequences are used to form the appropriate number of batches according to Property 1 for each family. The total number of batches is denoted by b. The following MILP formulation is possible to solve instances of this special case of Problem (3). Due to space limitations, we only introduce additional notation compared to the indices, parameters, and decision variables of the MILP formulation (4)-(15):

w_i: sum of the weights of the jobs that belong to batch i.

The model can be formulated as follows:

$$\min\left(\sum_{i=1}^{b}\sum_{t=1}^{T}\sum_{k=1}^{m} w_i\left(t+p_{f(i)}\right)y_{itk}, \sum_{i=1}^{b}\sum_{t=1}^{T}\sum_{k=1}^{m}\sum_{\tau=t}^{t+p_{f(i)}-1} e(\tau)\, y_{itk}\right) \tag{16}$$

subject to

$$\sum_{t=1}^{T+1-p_{f(i)}}\sum_{k=1}^{m} y_{itk} = 1 \qquad i=1,\ldots,b \tag{17}$$

$$\sum_{i=1}^{b}\sum_{\tau=\left(t-p_{f(i)}+1\right)^{+}}^{t} y_{i\tau k} \le 1 \qquad k=1,\ldots,m,\ t=1,\ldots,T \tag{18}$$

$$y_{itk} \in \{0,1\} \qquad i=1,\ldots,b\ \ t=1,\ldots,T\ ,k=1,\ldots,m. \tag{19}$$

The constraints (17) ensure that each batch is used exactly once, whereas the constraints (18) enforce that at most one batch can start in each period on each machine. Here, the abbreviation $x^{+} := \max(x,0)$ is used. We see that the main difference between the formulations (4)-(15) and (16)-(19) is the fact that the batches are already formed in the latter formulation. This results in a compact formulation.

3.3 $\varepsilon-$Constraint Method

The ε-constraint method is an approach that allows one to obtain a Pareto frontier (cf. Ehrgott 2010). It is based on the idea that instead of combining the different objectives into an integrated objective function, a single objective is optimized in a certain step while the remaining objectives are transformed into constraints. An ε-constraint problem with K objectives $f_k, k=1,...,K$ to be minimized is given as:

$$\min f_k \tag{20}$$

subject to

$$f_j \le \varepsilon_j, j = 1,...,K, j \ne k \;, \tag{21}$$

where $\varepsilon \in IR^K$ is given. Next, the formulation for the ε -constraint method applied to the MILP (4)-(15) is shown.

$$\min \left[E_{TWC} \sum_{j=1}^{n} w_j C_j + E_{EPC} \left(\sum_{i=1}^{b} \sum_{t=1}^{T} \sum_{k=1}^{m} \sum_{\tau=t}^{t+p_{f(i)}-1} e(\tau) y_{itk} \right) \right] \tag{22}$$

subject to

(5)-(15)

$$(1 - E_{TWC}) \sum_{j=1}^{n} w_j C_j \le \varepsilon_{TWC} \tag{23}$$

$$(1 - E_{EPC}) \sum_{i=1}^{b} \sum_{t=1}^{T} \sum_{k=1}^{m} \sum_{\tau=t}^{t+p_{f(i)}-1} e(\tau) y_{itk} \le \varepsilon_{TWC}. \tag{24}$$

The quantities $E_{TWC}, E_{EPC} \in \{0,1\}$, $E_{TWC} + E_{EPC} = 1,$ and $\varepsilon_{TWC}, \varepsilon_{EPC} \in IR$ are parameters of the model. For $E_{TWC} = 1$ and $E_{EPC} = 0$ the model pursues a TWC minimization whereas the EPC value is restricted to ε_{EPC}. In the case of $E_{TWC} = 0$ and $E_{EPC} = 1$, the model aims for a EPC minimization where the TWC value of the schedule is restricted to ε_{TWC}.

The model (22)-(24) and (5)-(15) is iteratively solved to obtain the set of Pareto optimal schedules. Here, we assume that the processing times, the weights and the electricity power cost values are integer values. The first iteration is started with $E_{TWC} = 1, E_{EPC} = 0, \varepsilon_{TWC} = 0,$ and $\varepsilon_{EPC} = M$ where M is set using the piecewise constant electricity power cost function. The solution is a schedule S with objective function value $TWC(S)$ where the EPC value is restricted to M. Afterwards, the MILP is solved a second time with $E_{TWC} = 0, E_{EPC} = 1, \varepsilon_{TWC} = TWT(S)$ leading to a Pareto optimal schedule.

The next iteration is started by $E_{TWC} = 0, E_{ECPI} = 1, \varepsilon_{ECP} = 0$, and $\varepsilon_{TWC} = TWC(S) - 1$. Note that the TWC and EPC values are always integer values due to our choice of the parameters of the instances. Therefore, choosing $\varepsilon_{TWC} = TWC(S) - 1$ is reasonable. This procedure is repeated until the MILP becomes infeasible for the parameters ε_{EPC} and ε_{TWC}. We obtain the set of all Pareto-optimal schedules for instances with parameter values choosing as described before. The ε -constraint method can be applied to the special-case MILP (16)-(19) in the same way. However, due to space limitation we do not present the details.

4 COMPUTATIONAL EXPERIMENTS

4.1 Design of Experiments

We are interested in assessing the computing time requirements for the ε -constraint method and how the electricity power cost setting influences the number of Pareto-optimal solutions, i.e., how many points are included in the criteria space. Moreover, we expect that the number of jobs, the number of families, the maximum batch size, and the number of machines also influence the results. Therefore, we use the design of experiments shown in Table 1 similar to (Mönch et al. 2005) for the special case. Here, $DU[a,b]$ refers to a discrete uniform distribution over the integer set $\{a,\ldots,b\}$.

Table 1: Design of experiments.

Factor	Level	Count
Number of families F	3, 5	2
Number of jobs per family	$90/F$	1
Maximum batch size B	4, 8	2
Number of machines m	2, 3	2
Family processing time p_f	2 with probability 0.2 4 with probability 0.2 10 with probability 0.3 16 with probability 0.2 20 with probability 0.1	1
Job weight w_j	$w_j \sim DU[1,\ldots,5]$	1
Electricity power cost $e(t)$	winter rate, summer rate	2
Number of independent replications	5 per factor combination	5
Total number of problem instances		80

The winter and summer electricity power cost settings are chosen as follows:

$$e_W(t) := \begin{cases} 10, 1 \le t < \frac{1}{2}T \\ 8,\ \text{otherwise} \end{cases}, \quad e_S(t) := \begin{cases} 10,\ 1 \le t < \frac{1}{3}T \\ 9, \frac{1}{3}T \le t < \frac{1}{2}T \\ 8,\ \frac{1}{2}T \le t < \frac{5}{6}T \\ 9, \text{otherwise} \end{cases}. \tag{25}$$

Note that the design in Table 1 ensures the applicability of the ε-constraint method as sketched in Subsection 3.3. For the general problem (3), we generate small-sized instances with ten jobs that belong to two families and two machines. We choose $B \in \{3,4\}$. Moreover, ready times of the jobs are generated according to $r_j \sim DU\left[0, \left\lfloor \frac{\alpha}{B} \sum_{j=1}^{n} p_j \right\rfloor\right]$ where $\alpha \in \{0.25, 0.5\}$ is a parameter that controls the range of the ready times. The remaining factor levels are chosen as in Table 1. This leads to 20 small-sized problem instances for the general case.

The scheduling horizon T is set based on a makespan estimate that is multiplied with a safety factor larger than one to ensure feasibility. We are interested in determining the number of solutions that belong to the Pareto frontier of a problem instance. We also report the total computing time required to obtain the Pareto frontier and the average time necessary for computing a single point of the frontier to assess the effectiveness of the presented optimization models. The MILP approaches are coded using the C++ programming language and the IBM ILOG CPLEX 12.1 libraries. All the computational experiments are conducted on a computer with an Intel Core i7-2600 CPU 3.40 GHz and 16 GB RAM.

4.2 Results

A first experiment is conducted to investigate the sizes of the Pareto frontiers for instances of the special case that are generated according to Table 1. Instead of presenting the results individually for each problem instance, we show the average over the five independent replications for each factor combination. The average computing time for obtaining the Pareto frontier and the average time required for computing a single solution measured in seconds is reported. The latter values are put into brackets. The corresponding results are shown in Table 2.

Table 2: Number of solutions and computing times for problem instances of the special case.

			# Solutions		Computing time (in sec)	
m	*F*	*B*	winter rate	summer rate	winter rate	summer rate
2	**3**	**4**	37	39	77 (1.9)	126 (3.0)
		8	37	50	15 (0.3)	25 (0.4)
	5	**4**	46	54	113 (2.1)	212 (3.2)
		8	57	81	37 (0.6)	78 (0.9)
3	**3**	**4**	67	91	151 (2.0)	298 (3.1)
		8	64	93	40 (0.5)	108 (0.9)
	5	**4**	64	69	234 (3.0)	412 (5.6)
		8	71	103	37 (0.5)	90 (0.8)
Overall			55	72	88 (1.4)	169 (2.3)

From the results of the runs described above, we choose two problem instances with $m=3, F=3$, and $B=8$ that only differentiate by the fact that the first is based on the winter rate and the second on the summer rate. The obtained Pareto frontiers for these instances are visualized in Figure 1. The x and the y axis refer to the TWC and EPC values, respectively. Each point drawn in the criteria space represents a single Pareto-optimal solution.

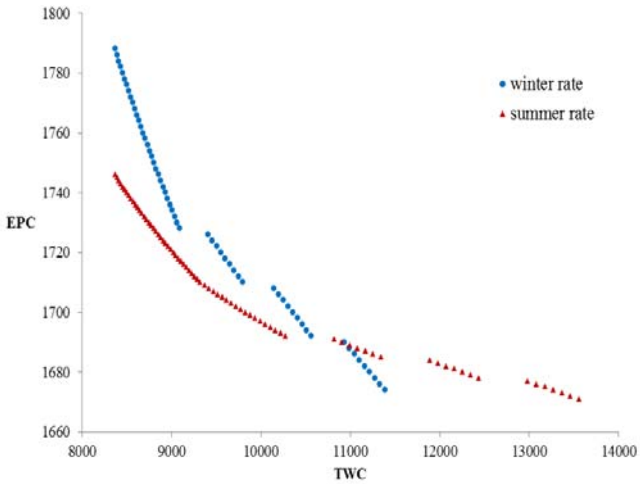

Figure 1: Pareto frontiers for two problem instances of the special case with different rates.

In addition, we conduct computational experiments for the 20 small-sized problem instances of problem (3). Again, we report the average performance measure values for the five independent replications of each factor combination. Table 3 shows the results grouped by the batch size and the range of the ready times represented by the α value.

4.3 Analysis and Discussion of the Results

Table 2 shows a range of 37 to 103 Pareto-optimal solutions for problem instances representing the special case. More Pareto-optimal solutions exist on average for problem instances with the three-level summer rate. The number of parallel machines has a large impact. When all the remaining factor levels are the same, the average number of solutions is between 25% and 130% higher if there are three machines available instead of two. The number of incompatible families seems to be more crucial if only two machines are available. More Pareto-optimal solutions exist in this situation for problem instances with five families. A larger batch size leads to a higher number of solutions under almost all experimental conditions.

Table 3: Number of solutions and computing times for small-sized problem instances.

		# Solutions		Computing time (in sec)	
B	α	**winter rate**	**summer rate**	**winter rate**	**summer rate**
3	**0.25**	31	43	207 (6.7)	310 (7.2)
	0.50	33	44	337 (10.2)	427 (9.7)
4	**0.25**	30	47	177 (5.9)	304 (6.5)
	0.50	36	55	264 (7.3)	402 (7.3)
Overall		33	47	246 (7.5)	361 (7.7)

The total computing time increases to a large extent linearly with the number of solutions. However, we observe a distinctively larger average computing time for determining a single solution in the case of a batch size of $B = 4$. From Property 1 in Subsection 3.2 it is clear that in a Pareto-optimal schedule all batches except maybe one per family have the maximum batch size. A smaller batch size thus leads to a higher number of batches for a given number of jobs. This results in more decision variables and constraints in the corresponding MILP model and therefore in the observed longer computing times per solution. Note that we know from some preliminary experimentation with a larger number of jobs that the MILPs cannot be solved to optimality within a reasonable amount of computing time when the number of jobs is larger than 90. This is expected because of the NP-hardness.

Results for problem instances of the general case presented in Table 3 show again a larger number of solutions if the summer rate is applied. Moreover, there are more Pareto-optimal solutions for problem instances with a larger batch size under almost all experimental conditions. Also, the number of solutions increases if the α value increases. Computing times ranging from 177 and 427 seconds are observed to obtain the Pareto front of a single instance. A larger amount of time is required for problem instances with the electricity power cost function with more levels. This behavior is caused by the larger number of solutions in this situation since the average computing per single solution is very similar. Note that we know again from some preliminary experiments that more than ten jobs or more than two machines lead to instances for which CPLEX is not able to prove that the obtained solutions are optimal within a reasonable amount of computing time. This is again expected due to the NP-hardness.

Solving problem instances with a smaller maximum batch size and a larger range of the ready times tend to be more time-consuming compared to the remaining instances. The computing time for these problem instances is larger than the average time over all instances indicating that solving the related MILPs is more difficult. We can see from the Figure 1 that, on the one hand, high-quality solutions in terms of a low TWC value can be achieved at a lower EPC level if the summer rate is applied. On the other hand, solutions with very low values of EPC result in higher TWC values in this situation.

5 CONCLUSIONS AND FUTURE RESEARCH

In this paper, we discussed a scheduling model for parallel batch processing machines. The jobs belong to incompatible families. Only jobs of the same family can be used to form a batch. The performance measures TWC and EPC were considered. The electricity power costs are piecewise constant functions of the period number. The jobs can have unequal ready times and nonidentical sizes. We proposed a MILP formulation for this scheduling problem. Moreover, we analyzed the special case where all jobs are ready at time $t = 0$, all jobs have the same size, and the maximum batch size is an integer multiple of this size. Two structural properties of Pareto-optimal schedules are shown in this situation. A second MILP formulation is provided that exploits these structural properties of the special case. The ε-constraint method was applied to both formulations. Designed computational experiments were performed to assess the behavior of the two MILP models with respect to computing time. Moreover, we looked at the number of Pareto-optimal solutions when the electricity power costs change. We found that the electricity power costs, the number of machines, the maximum batch size, and ready time range have an impact on the hardness of the problem instances.

There are several directions for future research. First of all, we are interested in designing metaheuristic approaches based on the NSGA-II approach to tackle large-sized problem instances for the general problem. As a second research avenue, we are interested in enriching the model problem by considering standby times, family-dependent electricity consumptions, and unrelated parallel machines. We believe also that it is interesting to use the TWT measure instead of the TWC measure. Since the TWT value and the TWC value are the same for instances where all jobs have zero due dates, the proposed ε-constraint method for medium-sized instances of the special case can be used to assess the quality of the NSGA-II approach for the more general problem with the TWT measure.

ACKNOWLEDGMENTS

The research of the first and second author is funded in parts by a research grant from the University of Hagen within the MaXFab project. The authors gratefully acknowledge this financial support. Parts of this research were carried out whilst the third author was visiting the Chair of Enterprise-wide Software Systems at University of Hagen.

REFERENCES

Cheng, J., F. Chu, W. Xia, J. Ding, and X. Ling. 2014. "Bi-objective Optimization for Single-machine Batch Scheduling Considering Energy Cost." In *Proceedings of the 2014 International Conference on Control, Decision and Information Technologies (CoDIT),* edited by I. Kacem et al., 236-241. Piscataway, New Jersey: IEEE.

Cheng, J., F. Chu, M. Liu, P. Wu, and W. Xia. 2017. "Bi-criteria Single-machine Batch Scheduling with Machine on/off Switching Under Time-of-use Tariffs." *Computers & Industrial Engineering* 112:721–734.

Ehrgott, M. 2010. *Multicriteria Optimization*. 2nd ed., New York: Heidelberg.

Gahm, C., F. Denz, M. Dirr, and A. Tuma. 2016. "Energy-efficient Scheduling in Manufacturing Companies: A Review and Research Framework." *European Journal of Operational Research* 248(3):744–757.

Giret, A., D. Trentesaux, and V. Prabhu. 2015. "Sustainability in Manufacturing Operations Scheduling: A State of the Art Review." *Journal of Manufacturing Systems* 37(1):126–140.

Jia, Z.-H., Y.-I. Zhang, J. Y.-T. Leung, and K. Li. 2017. "Bi-criteria Ant Colony Optimization Algorithm for Minimizing Makespan and Energy Consumption on Parallel Batch Machines." *Applied Soft Computing* 55:226-237.

Liu, C.-H. 2014. "Approximate Trade-off Between Minimisation of Total Weighted Tardiness and Minimisation of Carbon Dioxide (CO_2) Emissions in Bi-criteria Batch Scheduling Problem." *International Journal of Computer Integrated Manufacturing* 27(8):579-771.

Merkert, L., I. Harjunkoski, A. Isaksson, S. Säynevirta, A. Saarela, A., and G. Sand. 2015. "Scheduling and Energy – Industrial Challenges and Opportunities." *Computers & Chemical Engineering* 72(2): 183-198.

Mason, S. J., M. Kurz, L. M. Pohl, J. W. Fowler, and M. E. Pfund. 2007. "Random Keys Implementation of NSGA-II for Semiconductor Manufacturing Scheduling." *International Journal of Information Technology and Intelligent Computing* 2(3).

Mönch, L., H. Balasubramanian, J. W. Fowler, and M. E. Pfund. 2005. "Heuristic Scheduling of Jobs on Parallel Batch Machines with Incompatible Job Families and Unequal Ready Times". *Computers & Operations Research* 32(11):2731-2750.

Mönch, L., J. W. Fowler, S. Dauzère-Pérès, S. J. Mason, and O. Rose. 2011. "A Survey of Problems, Solution Techniques, and Future Challenges in Scheduling Semiconductor Manufacturing Operations." *Journal of Scheduling* 14(6):583-599.

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., R. Uzsoy, and J. W. Fowler. 2018. "A Survey of Semiconductor Supply Chain Models Part I: Semiconductor Supply Chains and Strategic Network Design." *International Journal of Production Research* https://doi.org/10.1080/00207543.2017.1401233.

Reichelt, D. and L. Mönch. 2006. "Multiobjective Scheduling of Jobs with Incompatible Families on Parallel Batch Machines." In *Proceedings EvoCop 2006*, LNCS 3906, edited by J. Gottlieb and G. R. Raidl, 209-221. Berlin: Springer.

TSMC. 2016. *Taiwan Semiconductor Manufacturing Company (TSMC) Annual Report 2016*. http://www.tsmc.com/download/ir/annualReports/2016/english/e_7_2.html, accessed May, 5th, 2018.

Uzsoy, R. 1995. "Scheduling Batch Processing Machines with Incompatible Families." *International Journal of Production Research* 33(10):2685-2708.

Wang, S., M. Liu, F. Chu, and C. Chu. 2016. "Bi-objective Optimization of a Single Machine Batch Scheduling Problem with Energy Cost Consideration." *Journal of Cleaner Production* 137:1205-1215.

Yu, C.-M., C.-F. Chien, and C.-J. Kuo. 2017. "Exploit the Value of Production Data to Discover Opportunities for Saving Power Consumption of Production Tools." *IEEE Transactions on Semiconductor Manufacturing* 30(4):345 – 350.

AUTHOR BIOGRAPHIES

JENS ROCHOLL is a Ph.D. student at the Chair of Enterprise-wide Software Systems, University of Hagen. He received a MS degree in Information Systems from the University of Hagen, Germany. His research interests include scheduling for complex manufacturing systems and metaheuristics. His email address is Jens.Rocholl@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.

JOHN W. FOWLER is the Motorola Professor of Supply Chain Management at the Arizona State University. His research interests include discrete event simulation, deterministic scheduling, and multi-criteria decision making. He was the Program Chair for the 2002 and 2008 Industrial Engineering Research Conferences and the 2008 Winter Simulation Conference (WSC). He recently served as Editor-in-Chief for *IIE Transactions on Healthcare Systems Engineering*. He is also an Editor of the *Journal of Simulation* and Associate Editor of *IEEE Transactions on Semiconductor Manufacturing* and *Journal of Scheduling*. He is a Fellow of the Institute of Industrial Engineers (IIE) and recently served as the IIE Vice President for Continuing Education, is a former INFORMS Vice President, and was on the WSC Board of Directors from 2005-2017. His email address is john.fowler@asu.edu.