Proceedings of the 2025 Winter Simulation Conference
E. Azar, A. Djanatliev, A. Harper, C. Kogler, V. Ramamohan, A. Anagnostou, and S. J. E. Taylor, eds.

ENERGY-EFFICIENT PARALLEL BATCHING OF JOBS WITH UNEQUAL AND UNDEFINED JOB SIZES ON PARALLEL BATCH PROCESSING MACHINES

Lisa C. Günther[1]

[1]Dept. of Factory and Process Design, Fraunhofer Institute for Manufacturing Engineering and Automation IPA, Stuttgart, GERMANY

ABSTRACT

This paper investigates a parallel batching problem with incompatible families, unequal job sizes and non-identical machine capacities. The objective is to minimize the total energy costs. Motivated by a real-word autoclave molding process in composite material manufacturing additional factors need to be considered: machine eligibility conditions, machine availability constraints, machine dependent energy consumption, and the job size and machine capacity may not be available in absolute terms. Mathematical models are formulated for both defined and undefined job and machine sizes. The latter approach creates batches based on a set of best-known batches derived from past production data to meet the machine capacity constraint. Finally, a heuristic is presented. Computational experiments are conducted based on a real case study. Energy savings of over 20 % can be achieved compared to the actual planning with sensible batch forming and machine allocation in a short amount of computing time.

1 INTRODUCTION

In parallel batching (p-batching) multiple jobs are grouped as a batch and processed simultaneously on a so-called batch processing machine (BPM). It is an important part of many industrial production systems and can be encountered in a variety of – often energy-intensive (Abedi et al. 2022) – industries, including semiconductor manufacturing (Mathirajan and Sivakumar 2006), aerospace parts manufacturing (Azami, Demirli, and Bhuiyan 2018), the textile industry (Huynh and Chien 2018), steel production (Tang et al. 2016), and coating/painting applications (Horn, Demirović, and Yorke-Smith 2023). If several parallel BPMs are available, the assignment of the batches to the machines must be solved along with batch formation. Problems typically also include a scheduling component to optimize makespan (C_{max}), cycle-time or due-date related performance measures. In the context of rising energy prices and growing environmental concerns, minimizing energy consumption (PC) and total energy costs (TEC) are becoming increasingly important.

The present paper considers an autoclave molding process found in composite material manufacturing. Specifically, it studies a new energy-aware p-batching problem with parallel identical machines with unequal capacity limits and energy consumption rates. Articles, respectively jobs can only be processed on a limited set of machines, belong to incompatible families and have non-identical sizes. In contrast to existing studies, it is taken into account that the determination of article size and machine capacity can be a challenge in practical applications. This information is essential to ensure autoclave capacity is not exceeded. However, it is not always easily available to support batching decisions. Furthermore, the spatial requirements of the equipment and nesting considerations would have to be taken into account for an exact calculation. Solution methods are presented for both, known and unknown job and machine size.

The remainder of this paper is structured as follows: Section 2 details the problem. Section 3 gives an overview of related work on p-batching and bin-packing. Section 4 presents a MILP formulation for the assumption that job sizes and machine capacities are known. Section 5 proposes a mathematical model and a dispatching rule for the case of undefined job sizes and machine capacities. The results of computational experiments are reported in Section 6. Section 7 gives a summary and suggestions for future work directions.

2 PROBLEM SETTING

The batching problem under study is based on the following assumptions:

1. There are A different article types.
2. Each article a belongs to family $f(a) \in \{1,...,F\}$ where F is the number of incompatible families. Only articles of the same family can be batched together.
3. Each article has specific dimensions. To simplify, each article is assigned a size s_a.
4. There are n jobs that have to be processed, labeled by $j = 1,\dots,n$. Each job j has an article type $a(j) \in \{1,...,A\}$.
5. Each job j requires a specific curing operation $p(j) \in \{1,...,P\}$ where P is the number of unique operations. All jobs in a batch must require the same curing operation.
6. There are m parallel BPMs, labeled by $i = 1,\dots,m$.
7. The machine capacity V_i is limited. The maximum batch size depends on the compilation of parts.
8. Articles (or jobs) can only be processed on a limited set of eligible machines M_j with $1 \leq |M_j| \leq m$.
9. The processing time t_p depends on the curing operation p and is considered to be independent of the composition of the processed articles in a batch and the batch size.
10. Preemption is not allowed, i.e., after a batch is started on a BPM it cannot be interrupted.
11. All jobs and machines are available at time $t = 0$. The machine availability is defined by T_i.
12. Energy costs c_{pi} are assumed to be constant over time and related to performing a curing operation p on machine i.

It should be noted that the same article type may require different curing operations. Moreover, the concept of the family differs from the understanding that is common, for example, in the context of p-batching in semiconductor manufacturing. Here, families are formed based on material, part type, higher-level assembly group or customer specifications. This means that parts of the same family may require different operations and may be qualified for different machines. Belonging to the same family is therefore a necessary but not sufficient condition for manufacturing parts together. Using the three-field notation from deterministic scheduling theory of Graham et al. (1979), the problem can be described as

$$P|p-batch, incompatible, s_j, V_i, M_j|EC,$$

where P refers to identical parallel machines, *incompatible* to incompatible job families and s_j to non-identical job sizes, V_i indicates different maximum machine capacities and M_j the eligibility constraints. The total energy costs EC required to process all jobs are to be minimized.

Two interdependent problems must be solved: (1) forming batches and (2) assigning batches to machines. The scheduling of batches on the machines can be disregarded, since the performance measure is not time-related. Job characteristics such as unequal ready times, due dates, or sequence-dependent setup times are not considered. The task is therefore similar to the one-dimensional (1D) bin packing problem (BPP), in which a set of items must be packed into a minimum number of identical bins. In fact, we can consider instances of the problem where V_i and c_{ip} are identical for all machines, $|F| = 1$, $|P| = 1$ and $|M_j| = m$ for all j, and obtain instances of the BPP. Since the BPP is therefore a special case of the given problem and is NP-hard due to Garey and Johnson (1979), this problem is also NP-hard.

Note that, although a small number of batches or bins is generally associated with low energy costs, a minimum number of batches is not a necessary or sufficient condition for attaining optimality.

3 RELATED WORK

There is an extensive literature on p-batch scheduling and bin packing. Here, we (1) give an overview on energy-aware scheduling in batch processing environments with parallel machines, (2) refer to batch processing with uncertainties, and (3) provide references to related BPP literature. A detailed overview of papers on p-batching can be found in Fowler and Mönch (2022).

Despite the growing interest in energy-aware scheduling in general and also in the context of p-batching (Gahm et al. 2016; Gao et al. 2020; Fowler and Mönch 2022), there are still relatively few papers dealing with energy-aware batch processing in parallel machine environments. The majority of these studies address multi-objective problems (Table 1). Consequently, they use methods designed for such problems, like the non-dominated sorting genetic algorithm II (NSGA-II) approach (Rocholl, Mönch, and Fowler 2020) or a bi-criteria ant colony optimization (ACO) algorithm (Jia et al. 2017; Jia et al. 2019). Multiple papers present mixed-integer linear programming (MILP) models and use the ε-constraint method to obtain Pareto-optimal solutions (Rocholl, Mönch, and Fowler 2018; Rocholl, Mönch, and Fowler 2020; Cheng 2017; Cheng, Cheng, and Chu 2022). The problem structure also offers opportunities for decomposition approaches. Cheng (2017) proposes a two-stage heuristic. Batches are formed using a linear programming (LP) or a successive knapsack-based method; batch allocation is done with a LP based method. A three-stage heuristic developed by Feng et al. (2022) first assigns jobs to machines, then forms batches for each machine and finally assigns batches to periods. Stage one and two use rule-based approaches, stage three solves an integer programming model. Typically, a conventional performance measure, such as C_{max} or the total weighted tardiness (TWT) is optimized while also minimizing TEC or PC. Energy costs are usually calculated based on TOU tariffs, i.e., the energy price depends of the time during which the energy is consumed. Sometime a machine-dependent power rate (Jia et al. 2019; Cheng 2017; Cheng, Cheng, and Chu 2022; Feng et al. 2022; Tian and Zheng 2024) or a machine- and mode-dependent power rate (Jia et al. 2017; Zhou et al. 2018; Qian, Jia, and Li 2020; Abedi et al. 2022) is presumed.

Table 1: Reviewed literature on p-batch scheduling with energy related performance measures and parallel machine environments. Mathematical models are only listed as solution approaches, if the paper presents related computational experiments. The following abbreviations have not been declared in the text: Q: parallel uniform machines; R: parallel unrelated machines; r_j: job release times; ND: non-dominated; NEM: number of enabled machines; TWC: total weighted completion time.

Reference	**Problem**	**Solution approach**
Jia et al. (2017)	$P\|p-batch,r_j,s_j\|ND(C_{max},TEC)$	bi-criteria ACO
Jia et al. (2019)	$P\|p-batch,r_j,s_j\|ND(C_{max},PC)$	bi-criteria ACO
Rocholl, Mönch, and Fowler (2018)	$P\|p-batch,incompatible,r_j,s_j\|ND(TWC,TEC)$	ε-constraint
Rocholl, Mönch, and Fowler (2020)	$P\|p-batch,incompatible,r_j,s_j\|ND(TWT,TEC)$	ε-constraint, NSGA-II
Schorn and Mönch (2023)	$P\|p-batch,incompatible,r_j\|\lambda TWT+(1-\lambda)TEC$	genetic programming
Zhou et al. (2018)	$Q\|p-batch,r_j\|ND(C_{max},TEC)$	differential evolution
Qian, Jia, and Li (2020)	$Q\|p-batch,r_j,s_j\|ND(C_{max},TEC)$	evolutionary algorithm
Cheng (2017)	$Q\|p-batch,s_j\|ND(TEC,NEM)$	ε-constr., 2-stage heuristic
Cheng, Cheng, and Chu (2022)	$Q\|p-batch,s_j\|ND(TEC,NEM)$	ε-constraint
Abedi et al. (2022)	$Q\|p-batch,incompatible,r_j,s_j\|\lambda TWT+(1-\lambda)PC$	MILP, tabu search
Feng et al. (2022)	$R\|p-batch\|TEC$	MILP, 3-stage heuristic
Tian and Zheng (2024)	$R\|p-batch,s_j,C_{max}\leq C\|TEC$	branch and price

Very few papers address p-batch scheduling problems with uncertain data (Shahnaghi et al. 2016; Fowler and Mönch 2022). Uncertainty in optimization problems is generally understood as the presence of non-deterministic data. For instance, in the context of p-batching, the expected values for job size or processing time are known, yet the actual values vary within established limits (Shahnaghi et al. 2016; Shahmoradi-Moghaddam et al. 2016; Wang, Shao, and Yan 2022; Wu et al. 2023) or according to a known distribution function (Rocholl, Yang, and Mönch 2022). Sources of uncertainty are, e.g., a non-zero defect rate (Wu et al. 2023), tolerance in production (Shahmoradi-Moghaddam et al. 2016) or data quality issues (Wu et al. 2023; Rocholl, Yang, and Mönch 2022). The two methods commonly used for problems with uncertainty are stochastic or robust optimization approaches. This paper explores a different case. Information is theoretically available, e.g. the article size could be determined from CAD models. However, data extraction or incorporation of the exact data into the planning process is considered too time-consuming. I.e., the data is uncertain in the sense of being undefined or unknown during batching.

In order to differentiate between the general conception of uncertainty and the present case, the term "undefined" is used for the remainder of the text to refer to undetermined job sizes and machine capacity. To the best of our knowledge, there is no research on the subject of p-batching or bin packing in which data on a required parameter is not directly available.

A close bin-packing counterpart to the p-batching problem under consideration is the 1D variable-sized BPP with conflicts or (in)compatible categories. A well-known approach to solving the classical (offline) 1D BPP are approximation algorithms, most notably the offline variants of Next-(k-)fit, First-fit (decreasing) and Best-fit (decreasing) described in Johnson (1973) and Johnson et al. (1974). Items are iteratively fit into open bins; if no open bin has sufficient capacity, a new bin is opened. The heuristics differ in terms of how items are sorted, when bins are opened and closed and how they select an open bin for the current item. In addition to these rule-based approaches, various metaheuristics have been adapted to the 1D BPP. Munien and Ezugwu (2021) provide an overview. For lower bounds and heuristics for the BPP with conflicts we refer to Gendreau, Laporte, and Semet (2004) and Muritiba et al. (2010). For a literature review on packing and the related cutting problems under uncertainty readers can refer to Hadj Salem, Silva, and Oliveira (2023).

4 MATHEMATICAL FORMULATION FOR BATCHING WITH DEFINED JOB SIZES

In the following, the batching problem introduced in Section 2 is formulated as a MILP. Item size and machine capacity are assumed to be known. The article level is excluded, item volumes and families are directly assigned to the jobs. The indices, parameters, and decision variable used to model the problem are introduced in Table 2.

Table 2: Notation of the mathematical model for batching jobs with defined size.

Indices and sets	$i = 1,...,m$	:	set of machines
	$j = 1,...,n$	:	set of jobs
	$p = 1,...,P$	:	set of processes
	$f = 1,...,F$	:	set of families
	$b = 1,...,B$	:	set of batches
Parameters	c_{ip}	:	non-negative energy costs related to performing process p n machine i
	t_p	:	processing time of process p
	s_j	:	space (volume) requirements of job j
	T_i	:	maximum availability (time) of machine i
	V_i	:	maximum capacity (volume) of machine i
	e_{ij}	=	1 if job j is qualified to be processed on machine i, 0 otherwise
	r_{jp}	=	1 if job j requires operation p, 0 otherwise
	h_{jf}	=	1 if job j belongs to family f, 0 otherwise
Decision variables	x_{ijb}	=	1 if job j belongs to batch b on machine i, 0 otherwise
	y_{ipb}	=	1 if batch b on machine i requires operation p, 0 otherwise
	z_{ifb}	=	1 if batch b on machine i belongs to family f, 0 otherwise

The MILP can then be formulated as follows.

$$\min \sum_{b=1}^{B} \sum_{i=1}^{m} \sum_{p=1}^{P} c_{ip} y_{ipb} \tag{1}$$

subject to

$$\sum_{i=1}^{m}\sum_{b=1}^{B} x_{ijb} = 1 \qquad j = 1,\ldots,n \tag{2}$$

$$\sum_{p=1}^{P} y_{ipb} \le 1 \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B \tag{3}$$

$$r_{jp}x_{ijb} \le y_{ipb} \qquad i = 1,\ldots,m\ ,\ j = 1,\ldots,n\ ,\ p = 1,\ldots,P\ ,\ b = 1,\ldots,B \tag{4}$$

$$\sum_{f=1}^{F} z_{ifb} \le 1 \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B \tag{5}$$

$$h_{jf}x_{ijb} \le z_{ifb} \qquad i = 1,\ldots,m\ ,\ j = 1,\ldots,n\ ,\ f = 1,\ldots,F\ ,\ b = 1,\ldots,B \tag{6}$$

$$\sum_{j=1}^{n}\sum_{b=1}^{B} x_{ijb}e_{ij} = \sum_{j=1}^{n}\sum_{b=1}^{B} x_{ijb} \qquad i = 1,\ldots,m \tag{7}$$

$$\sum_{j=1}^{n} x_{ijb}s_j \le V_i \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B \tag{8}$$

$$\sum_{p=1}^{P}\sum_{b=1}^{B} y_{ipb}t_p \le T_i \qquad i = 1,\ldots,m \tag{9}$$

$$x_{ijb}, y_{ipb}, z_{ifb} \in \{0,1\} \qquad i = 1,\ldots,m\ ,\ j = 1,\ldots,n\ ,\ p = 1,\ldots,P\ ,\ f = 1,\ldots,F\ ,\ b = 1,\ldots,B. \tag{10}$$

The objective (1) is to minimize the sum of energy costs, calculated as the sum of the element-wise product of the matrices c_{ip} and y_{ipb} over all b. Equations (2) ensure that each job belongs to exactly one batch and all demand is satisfied. Constraints (3) make sure each batch requires at most one curing operation and constraints (4) ensure that all jobs in one batch require the same operation. Similarly, constraints (5) and (6) assure that each batch belongs to at most one family and all jobs in one batch belong to the same family. Equations (7) express that all jobs in a batch must be qualified to be processed on the respective machine. Finally, constraints (8) ensure that the total volume of a batch must not exceed machine's capacity and constraints (9) make sure that the time required to process all batches of a machine does not exceed the machine's availability. Constraints (10) indicate that the decision variables are binary.

The model requires a value B, that specifies the maximum number of batches allocated to each machine. Two simple upper bounds for B are the number of jobs to be processed and the maximum number of batches that can be processed on a machine within T_i. It is clear that a smaller value B is preferable, as it is associated with a smaller problem size so that $B = \min(n, \max_{i\in m}\{\lceil \frac{T_i}{\min\{o_{ip}(i)\circ t_p | o_{ip}(i)\circ t_p > 0\}} \rceil\}$ where $\lceil\cdot\rceil$ is the ceiling function and o_{ip} is a binary matrix indicating which process p can be performed on machine i. A better upper bound can be derived from the result of a heuristic such as the BKB dispatching rule (Section 5.3) or a BPP approximation algorithm.

A further approach to reduce the formulation size is to extend data preprocessing. If information about the required curing process and the job family is summarized in an item group g with $g = 1,\ldots,G$ where G is the number of unique combinations of process and family (Figure 1) then the number of constraints and decision variables is reduced, on the condition that $G < P + F$. r_{jp} and h_{jf} are then replaced by k_{jg}, so that $k_{jg} = 1$ if job j belongs to item group g and 0 otherwise. Similarly, the decision variables y_{ipb} and z_{ifb} are omitted and replaced by y_{igb} with $y_{igb} = 1$ if batch b on machine i belongs to group g and 0 otherwise. Constraints (3) – (6) are substitued by the analogous constraints

$$\sum_{g=1}^{G} y_{igb} \leq 1 \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B \tag{11}$$

$$k_{jg} x_{ijb} \leq y_{igb} \qquad i = 1,\ldots,m\ ,\ j = 1,\ldots,n\ ,\ g = 1,\ldots,G\ ,\ b = 1,\ldots,B. \tag{12}$$

Constraints (11) ensure that each batch is linked to a maximum of one item group and constraints (12) state that all jobs in one batch belong to the same group. Finally, the parameters c_{ip} and t_p have to be converted to c_{ig} and t_g, i.e., the costs and processing time associated with each respective group g. Constraints (1) and (9) must be adapted accordingly.

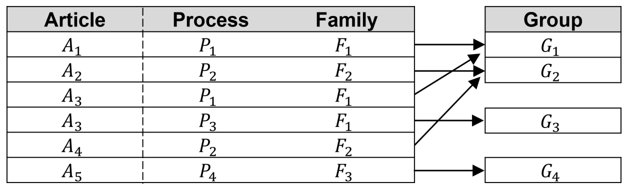

Figure 1: Formation of groups from unique process-family combinations.

5 BATCHING JOBS WITH UNDEFINED SIZE

5.1 Identification of Best-Known Batches

In real-world settings, data on article size and machine volume is not always available. In other cases, a calculation using only job and machine volumes may not be accurate enough. To address these concerns, the methods presented below, a MILP to determine an optimal solution (Section 5.2) and a computationally faster heuristic approach (Section 5.3), utilize a set of machine- and process-dependent best-known batches (BKBs). These are extracted from past production data. For each unique combination of articles, the batches containing the highest known number of parts per article are identified (Figure 2). For mathematical modeling the data is transformed into a three-dimensional matrix s_{ioa^*}. Analogous to the procedure described in Section 4, A^* new items are created that combine information about articles and processes. Batches are then represented machine-related by arrays of shape (O, A^*), where O is the maximum number of BKB options over all machines. Empty batches fill out the data structure for machines with less than O BKBs. The creation of batches based on historical data ensures the validity of the batches. The constraints regarding article volume and machine capacity, family compatibility, admissibility of material combinations and qualification of articles for machines are implicitly met. A disadvantage is that only known material combinations can be used, as well as the requirement that the number of parts per material must lie within a given range.

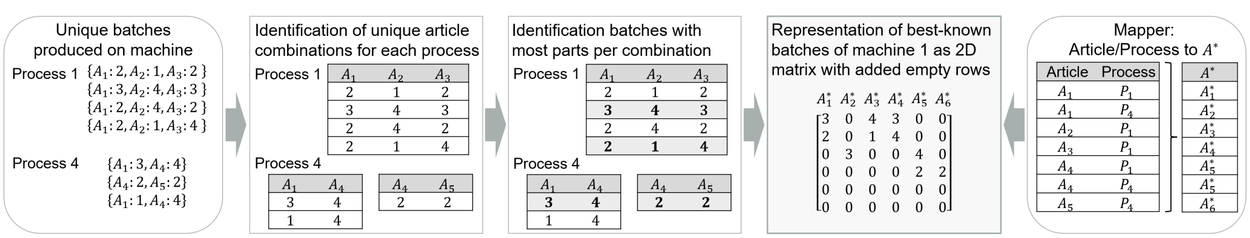

Figure 2: Example of extracting BKBs from set of batches produced on a specific machine.

5.2 Mixed Integer Linear-Programming Formulation

The set of BKBs extracted from historical data is stored in a 3D matrix s_{ioa^*}. Costs and processing time are now related to a batch option o on a machine i. Demand is specified as an integer vector d_{a*}. The indices, parameters, and decision variables of the model are summarized in Table 3.

Table 3: Notation of the mathematical model for batching jobs with undefined size.

Indices and sets	$a^* = 1,...,A*$	:	set of derived articles (including information on required processes)
	$i = 1,...,m$	:	set of machines
	$o = 1,...,O$	:	set of batch options (BKBs)
	$b = 1,...,B$	:	set of batches to be produced
Parameters	d_{a^*}	:	demand of article a^*
	s_{ioa^*}	:	quantity of article a^* in batch option o on machine i
	c_{io}	:	non-negative energy costs related to performing batch option o n machine i
	t_{io}	:	processing time associated to batch option o on machine i
	T_i	:	maximum availability (time) of machine i
Decision variables	x_{iba^*}	:	quantity of article a^* in batch b on machine i
	y_{ibo}	=	1 if batch b on machine i belongs to batch option o, 0 otherwise

The MILP can then be formulated as follows.

$$\min \sum_{b=1}^{B} \sum_{i=1}^{m} \sum_{o=1}^{O} c_{io} y_{ibo} \tag{13}$$

subject to

$$\sum_{i=1}^{m} \sum_{b=1}^{B} x_{iba^*} = d_{a*} \qquad a^* = 1,\ldots,A^* \tag{14}$$

$$\sum_{b=1}^{B} \sum_{o=1}^{O} y_{ibo} t_{io} \leq T_i \qquad i = 1,\ldots,m \tag{15}$$

$$\sum_{o=1}^{O} y_{ibo} \leq 1 \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B \tag{16}$$

$$\sum_{o=1}^{O} s_{ioa^*} y_{ibo} - x_{iba^*} \geq 0 \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B\ ,\ a^* = 1,\ldots,A^* \tag{17}$$

$$x_{iba^*} \in \mathbb{Z}_{\geq 0} \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B\ ,\ a^* = 1,\ldots,A^* \tag{18}$$

$$y_{ibo} \in \{0,1\} \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B\ ,\ o = 1,\ldots,O. \tag{19}$$

Again, the objective (13) is to minimize the sum of energy costs. Constraints (14) make sure that all demand is met. Constraints (15) ensure that the time required to process all batches of a machine does not exceed the machine's availability. Equations (16) indicate that each batch must be associated with at most one BKB option. If the batch is empty, i.e. not required, $\sum_{o=1}^{O} y_{ibo} = 0$. Constraints (17) ensure that all batches are valid batches, i.e. the quantity of each article a^* is equal or below the article quantity in the associated BKB. Constraints (18) indicate that x_{iba^*} is a non-negative integer for all values of i, b and a^* and constraints (19) express that y_{ibo} is a binary variable.

To speed up computation, the implementation introduces an additional variable z_{iba^*} with non-negative integers for all values of i, b and a^*. It stores the BKB corresponding to the selected batch option (20). Therefore, recalculation is not necessary when the quantity of each individual article is compared to the maximum article quantity (21). Thus, equations (17) are replaced by the following constraints:

$$\sum_{o=1}^{O} s_{ioa} y_{ibo} = z_{iba*}^{T} \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B \tag{20}$$

$$z_{iba^*} - x_{iba^*} \geq 0 \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B\ ,\ a^* = 1,\ldots,A^*. \tag{21}$$

Following the proposal in Santos et al. (2019) additional constraints were implemented.

$$\sum_{o=1}^{O} y_{ibo} \geq \sum_{o=1}^{O} y_{i(b+1)o} \qquad i = 1,\ldots,m\ ,\ b = 1,\ldots,B-1. \tag{22}$$

The equations (22) reduce the problem's solutions space by ensuring that higher indexes are assigned to unused batches in comparison to used ones. However, tests have shown that, in the present case, the costs of the additional constraints outweigh the potential performance benefits. The equations were ultimately removed.

5.3 Dispatching Rule

In order to effectively solve large problem instances, an iterative BKB-Batching rule is introduced below. The previously presented approach based on BKBs is transferred to an iterative procedure, in which batches are simultaneously formed and assigned to machines until there is no more open demand (Figure 3).

The dispatching rule takes the demand per process and article d_{pa} of the specific time frame, the machine availability t_i, the costs c_{pi} related to process p on machine i and a collection of BKBs $bkb(p,i)$ per process and machine as input. The variable B_i stores the batches created for each machine. E^* contains the evaluation of production options. There are various possibilities for implementation of E^*. Here, two approaches are proposed: (1) a greedy approach *BKB-BC* trying to optimize costs and (2) an approach *BKB-MA* that focuses on completing the demand within the specified machine availability.

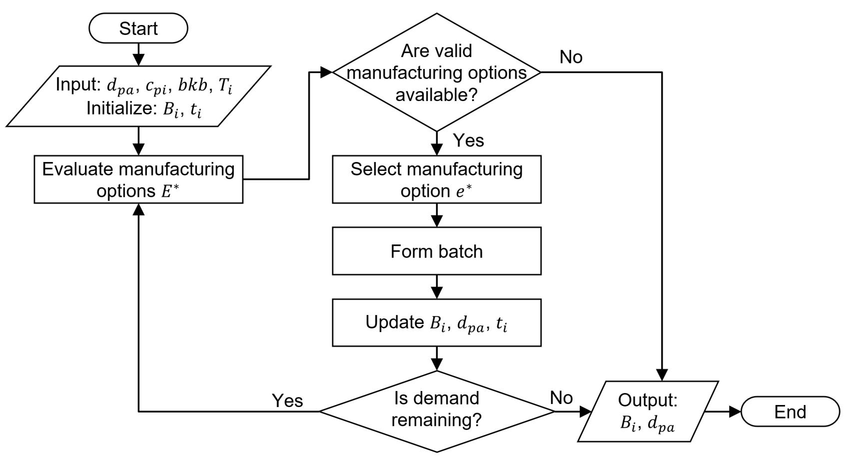

Figure 3: Illustration of the general procedure of the BKB dispatching rule over one period of time.

- *BKB-BC* evaluates all available process-machine combinations for cost-benefit ratio. Without a volume description, the evaluation of a batch composition must be based on the number of parts produced; differentiation according to total volume/filling level is not possible. Thus, E^* is a 2-dimensional array of shape (P,m). Each value of E_{pi} represents the minimum cost-part count-ratio for the respective process and machine, where the part count is the maximum number of parts produced in a batch considering machine availability, process duration, current demand and BKBs. All elements with invalid process-machine-combinations are set to zero.

- *BKB-MA* counts how many machines are available to process a specific article considering machine availability, process duration and machine capability. I.e., E^* becomes E_{ai}. Articles with few options are produced first.

The manufacturing option with the highest priority $\min\{e^*\}$ is selected. If several options are rated equally well, a random selection is made. A batch is then formed for the designated machine and the process and, if relevant, the selected article. Again, the batch evaluation can only be based on part count. Thus, the new batch should contain as many parts as possible. The new batch is added to B_i. The variables d_{pa} and t_i are updated accordingly. The algorithm stops if all demand has been met or when no feasible manufacturing option remains. The remaining demand is carried over to the next time period.

6 COMPUTATIONAL EXPERIMENTS

6.1 Problem Instances and Notes on Implementation

The solution approaches are validated and evaluated based on a real case study. As the data are confidential, instances and results are presented that are based on the original data but do not equal it. The following assumptions are made: A production with five BPMs is mapped over the course of one year. The production of the autoclave area is scheduled on a daily basis. All parts reaching the area from the upstream processes must be processed on the same day. The number of problem instances is 325. In total, over 2000 batches are completed, resulting in the production of around 200 different articles and over 25,000 parts. The number of parts produced per day can vary significantly, ranging from a minimum of one to a maximum of several hundred parts. The 25th and 75th percentiles of this distribution are 30 and 80 parts, respectively.

The MILP approaches are implemented in Julia (Bezanson et al. 2017) using the JuMP package (Lubin et al. 2023) and solved with HiGHS (Huangfu and Hall 2018). Depending on the problem structure, the computing time ranges from milliseconds to several hours in individual instances on a computer with an Intel Core i9 processor with 2.3 GHz and 32 GB of RAM. The latter is particularly the case when the demand of two consecutive days is bundled. To set a realistic time frame, the results presented below were calculated with a time limit of 30 minutes. The heuristic approaches are implemented in Python and run with 10 repetitions for each configuration. The calculation time is in the millisecond range when batching the daily demand and less than ten seconds when computing the aggregated annual demand. In the following only the mean values out of these runs are presented. The influence of randomness is limited; the coefficient of variation is below 0.01 for all configurations considered.

6.2 Results

Given the unavailability of article sizes in the present case, s_j is approximated by the bounding box volume of the devices. The information is accessible for a subset of the articles. The calculation is therefore conducted on a reduced data set of approximately half the batches produced. All instances can be solved with the MILPs for both defined and undefined job sizes within the time limit. The total cost, assuming known job sizes, is 20 % lower than that of the optimization with BKBs. However, it remains uncertain whether these savings can be realized in practice. The data preparation showed a large variation in the total volume of BKBs per machine. This could indicate that the estimation of the article sizes is not sufficiently accurate and/or that additional influencing factors need to be considered.

Table 4 summarizes the comparative results for the BKB methods with undefined job size. The MILP approach *BKB-MILP* reduces the total costs by 20 % compared to the actual production planning, respectively 24.5 % when the demand of each two consecutive days is bundled. The total number of batches decreases only by 2.3 %, 5.8 % respectively, indicating that a large part of the savings comes from allocating batches to machines with lower energy consumption. All instances of daily demand are solved within seconds, suggesting that the optimization approach can be used in the real production environment. If demand of two days is aggregated, 94 % of the instances are solved within, or well below, the timelimit.

The *BKB-BC* heuristic also reduces costs. At the same time, however, the number of batches increases, so that in ∼30 % of cases, part of the demand has to be carried over to the next time period. *BKB-MA* avoids carryover demand, yet ultimately attains less favorable outcomes.

Table 4: Comparison of results for undefined job sizes. Costs and batch count are relative to the as-is state.

	1 Day			2 Days		
	BKB-MILP	BKB-BC	BKB-MA	BKB-MILP	BKB-BC	BKB-MA
Costs /%	-20.21	-6.86	+31.50	- 24.51	-14.56	+25.24
# Batches /%	-2.30	+7.73	+38.76	-5.80	+6.20	+32.18
Solved instance /%	100	100	100	94.32	100	100
# Carry over /%	0.0	37.23	2.71	0.0	22.29	0.0

Figure 4 shows increasing potential savings under the assumption that demand can be aggregated and optimized over different periods of time. In order to evaluate the influence of both batch forming and machine assignment on the result, additional solutions are generated using two further approaches: (1) *Max. Batch*: Batches are rearranged for each autoclave, with the objective of optimizing machine utilization and minimizing the number of runs. (2) *Min. Cost Machine*: Batch combinations remain the same compared to the actual data, but are always executed on the machine with the lowest process costs. *Min. Cost Machine* achieves good results even with short planning periods. The advantages of improved batch forming only arise when the demand of several weeks is combined and optimized. Possible reasons are that the batch forming is already almost optimally designed, or that no better options are in past data used for the BKB extraction. The lowest cost is achieved with with *BKB-BC*. If demand is planned weekly, which is a reasonable assumption given the use case, TEC drops to 75 % of the actual costs. Six instances remain with batches that have to be completed during the next time period.

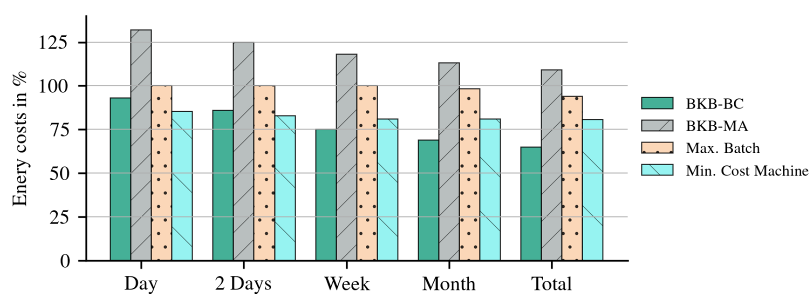

Figure 4: Energy costs for different batching rules and for aggregation of demand over different periods of time. As-is costs equal 100 %.

7 CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

This paper investigates a p-batching problem with incompatible families, unequal job sizes and parallel BPMs with non-identical capacities. In particularly we consider the case of undefined job sizes and machine capacities. The objective of the problem is to batch jobs and assign them to machines in a way that minimizes total energy consumption. Two mathematical models and a dispatching rule are presented. The MILP model for the case of undefined sizes could find optimal solutions for instances covering two days of production in an acceptable time. The dispatching rule can be used for larger problem instances or to generate a lower bound for the number of batches required in the exact approaches.

The approaches were evaluated based on a real use case. Assuming that the demand is processed on a daily basis, potential energy savings of over 20 % were identified. If the demand of several days can be aggregated, the potential increases. In the investigated case study savings primarily result from an increased use of autoclaves, which consume relatively less energy. From a practical point of view, this is not surprising, as those responsible have extensive experience in batching, while energy consumption has only recently been systematically documented. From an academic perspective, this finding is noteworthy, as the extent to which individual, but interdependent decisions (batching, assignment, and sequencing) influence the target values is commonly not investigated. A comprehensive evaluation of the presented approaches, based on parameterizable, random instances and with regard to different problem structures should follow. In the context of the case study, the initial focus of implementation is on improving machine allocation.

There are several other directions for future work. The model can be further refined to align more accurately with the real-world use case. For instance, the autoclave area is closely linked to the upstream production area, which the model does not address. It is also interesting to explore different objective functions, e. g. use time-of-use electricity tariffs or to flatten demand peaks. In both cases, it is necessary to add a time component to the model. Finally, the evaluation has shown that, although the proposed dispatching rule is fast, the results deviate from the optimum by at least $\sim$15 %. When calculating short planning periods, the number of batches increases considerably compared to the actual situation. More complex heuristics, such as population-based algorithms or neighborhood search techniques, as presented in other p-batching studies, could close this gap.

ACKNOWLEDGMENTS

This research was conducted for the KI-noW research project, funded by the Bavarian Ministry of Economic Affairs, Regional Development and Energy. The responsibility for the content lies with the authors.

REFERENCES

Abedi, M., R. Chiong, N. Noman, X. Liao, and D. Li. 2022. "A metaheuristic framework for energy-intensive industries with batch processing machines". *IEEE Transactions on Engineering Management* 71:4502–4516.

Azami, A., K. Demirli, and N. Bhuiyan. 2018. "Scheduling in aerospace composite manufacturing systems: a two-stage hybrid flow shop problem". *The International Journal of Advanced Manufacturing Technology* 95:3259–3274.

Bezanson, J., A. Edelman, S. Karpinski, and V. B. Shah. 2017. "Julia: A fresh approach to numerical computing". *SIAM Review* 59(1):65–98 https://doi.org/10.1137/141000671.

Cheng, J. 2017. *Multi-criteria batch scheduling under time-of-use tariffs*. Ph.D. thesis, Université Paris-Saclay; Northwestern Polytechnical University (Chine).

Cheng, J., J. Cheng, and F. Chu. 2022. "Bi-objective optimization for uniform parallel batch machine scheduling under Time-of-Use tariffs". In *2022 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, 1–6. IEEE.

Feng, L., G. Chen, S. Zhou, and X. Zhou. 2022. "Scheduling on unrelated parallel batch processing machines considering energy cost". In *2022 International Conference on Intelligent Systems and Computational Intelligence (ICISCI)*, 93–97. IEEE.

Fowler, J. W., and L. Mönch. 2022. "A survey of scheduling with parallel batch (p-batch) processing". *European journal of operational research* 298(1):1–24.

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.

Gao, K., Y. Huang, A. Sadollah, and L. Wang. 2020. "A review of energy-efficient scheduling in intelligent production systems". *Complex & Intelligent Systems* 6:237–249.

Garey, M. R., and D. S. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: W. H. Freeman & Co.

Gendreau, M., G. Laporte, and F. Semet. 2004. "Heuristics and lower bounds for the bin packing problem with conflicts". *Computers & Operations Research* 31(3):347–358.

Graham, R. L., E. L. Lawler, J. K. Lenstra, and A. R. Kan. 1979. "Optimization and approximation in deterministic sequencing and scheduling: a survey". In *Annals of discrete mathematics*, Volume 5, 287–326. Elsevier.

Hadj Salem, K., E. Silva, and J. F. Oliveira. 2023. "Cutting and packing problems under uncertainty: literature review and classification framework". *International Transactions in Operational Research* 30(6):3329–3360.

Horn, M., E. Demirović, and N. Yorke-Smith. 2023. "Parallel batch processing for the coating problem". In *Proceedings of the International Conference on Automated Planning and Scheduling*, Volume 33, 171–179.

Huangfu, Q., and J. J. Hall. 2018. "Parallelizing the dual revised simplex method". *Mathematical Programming Computation* 10(1):119–142.

Huynh, N.-T., and C.-F. Chien. 2018. “A hybrid multi-subpopulation genetic algorithm for textile batch dyeing scheduling and an empirical study”. *Computers & Industrial Engineering* 125:615–627.

Jia, Z.-h., Y. Wang, C. Wu, Y. Yang, X.-y. Zhang, and H.-p. Chen. 2019. “Multi-objective energy-aware batch scheduling using ant colony optimization algorithm”. *Computers & Industrial Engineering* 131:41–56.

Jia, Z.-h., Y.-l. 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.

Johnson, D. S. 1973. *Near-optimal bin packing algorithms*. Ph. D. thesis, Massachusetts Institute of Technology.

Johnson, D. S., A. Demers, J. D. Ullman, M. R. Garey, and R. L. Graham. 1974. “Worst-case performance bounds for simple one-dimensional packing algorithms”. *SIAM Journal on computing* 3(4):299–325.

Lubin, M., O. Dowson, J. Dias Garcia, J. Huchette, B. Legat, and J. P. Vielma. 2023. “JuMP 1.0: Recent improvements to a modeling language for mathematical optimization”. *Mathematical Programming Computation* https://doi.org/10.1007/s12532-023-00239-3.

Mathirajan, M., and A. I. Sivakumar. 2006. “A literature review, classification and simple meta-analysis on scheduling of batch processors in semiconductor”. *The International Journal of Advanced Manufacturing Technology* 29:990–1001.

Munien, C., and A. E. Ezugwu. 2021. “Metaheuristic algorithms for one-dimensional bin-packing problems: A survey of recent advances and applications”. *Journal of Intelligent Systems* 30(1):636–663.

Muritiba, A. E. F., M. Iori, E. Malaguti, and P. Toth. 2010. “Algorithms for the bin packing problem with conflicts”. *Informs Journal on computing* 22(3):401–415.

Qian, S.-y., Z.-h. Jia, and K. Li. 2020. “A multi-objective evolutionary algorithm based on adaptive clustering for energy-aware batch scheduling problem”. *Future Generation Computer Systems* 113:441–453.

Rocholl, J., L. Mönch, and J. Fowler. 2020. “Bi-criteria parallel batch machine scheduling to minimize total weighted tardiness and electricity cost”. *Journal of Business Economics* 90(9):1345–1381.

Rocholl, J., L. Mönch, and J. W. Fowler. 2018. “Electricity power cost-aware scheduling of jobs on parallel batch processing machines”. In *2018 Winter Simulation Conference (WSC)*, 3420–3431. IEEE https://doi.org/10.1109/WSC.2018.8632342.

Rocholl, J., F. Yang, and L. Mönch. 2022. “Scheduling jobs with uncertain ready times on a single batch processing machine”. In *2022 Winter Simulation Conference (WSC)*, 3349–3359. IEEE https://doi.org/10.1109/WSC57314.2022.10015363.

Santos, L. F. M., R. S. Iwayama, L. B. Cavalcanti, L. M. Turi, F. E. de Souza Morais, and G. t. Mormilho. 2019. “A variable neighborhood search algorithm for the bin packing problem with compatible categories”. *Expert Systems with Applications* 124:209–225.

Schorn, D. S., and L. Mönch. 2023. “Learning priority indices for energy-aware scheduling of jobs on batch processing machines”. *IEEE Transactions on Semiconductor Manufacturing* 37(1):3–15.

Shahmoradi-Moghaddam, H., K. Akbari, S. J. Sadjadi, and M. Heydari. 2016. “A scenario-based robust optimization approach for batch processing scheduling”. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 230(12):2286–2295.

Shahnaghi, K., H. Shahmoradi-Moghadam, A. Noroozi, and H. Mokhtari. 2016. “A robust modelling and optimisation framework for a batch processing flow shop production system in the presence of uncertainties”. *International journal of computer integrated manufacturing* 29(1):92–106.

Tang, L., Y. Meng, Z.-L. Chen, and J. Liu. 2016. “Coil batching to improve productivity and energy utilization in steel production”. *Manufacturing & Service Operations Management* 18(2):262–279.

Tian, Z., and L. Zheng. 2024. “Scheduling unrelated parallel batch processing machines under time-of-use electricity prices”. *IEEE Transactions on Automation Science and Engineering*.

Wang, T., X. Shao, and X. Yan. 2022. “A branch-and-price algorithm for robust parallel batch scheduling problem with uncertain size”. *Industrial Management & Data Systems* 122(10):2351–2370.

Wu, W., T. Hayashi, K. Haruyasu, and L. Tang. 2023. “Exact algorithms based on a constrained shortest path model for robust serial-batch and parallel-batch scheduling problems”. *European Journal of Operational Research* 307(1):82–102.

Zhou, S., X. Li, N. Du, Y. Pang, and H. Chen. 2018. “A multi-objective differential evolution algorithm for parallel batch processing machine scheduling considering electricity consumption cost”. *Computers & Operations Research* 96:55–68.

AUTHOR BIOGRAPHIES

LISA C. GÜNTHER is a member of research staff at the Department of Factory and Process Design at the Fraunhofer Institute for Manufacturing Engineering and Automation IPA in Stuttgart, Germany. She received a MS degree in Logistics Engineering from the University of Duisburg-Essen, Germany and a MS degree in Practical Computer Science from the University of Hagen, Germany. Before joining Fraunhofer IPA Lisa Günther was a research associate at the Institute of Industrial Manufacturing and Management at the University of Stuttgart. Her research interests include modeling of manufacturing systems, deterministic scheduling and data-driven optimization of production processes. Her email address is lisa.charlotte.guenther@ipa.fraunhofer.de.