Minimizing Total Weighted Earliness and Tardiness for Parallel Batch-processing Machines with Incompatible Job Families

Pei-Chun WU¹, Ya-Chih TSAI (Hui-Mei Wang)² and Fuh-Der CHOU³,*

¹Department of Airline and Transport Service Management, Vanung University, Tao Yuan, Taiwan, R.O.C.
²Department of Hotel Management, Vanung University, Tao Yuan, Taiwan, R.O.C.
³Department of Industrial Management, Chien Hsin University, Tao Yuan, Taiwan, R.O.C.
E-mail: fdchou@uch.edu.tw
*Corresponding author

Keywords: Parallel batch-processing machines, Incompatible job family, Earliness, Tardiness, Setup time

Abstract. This paper considers the parallel batch-processing machines with incompatible job families, that is, jobs with the same recipe can be processed together as a batch, and the sizes of these jobs in a batch should not exceed the capacity limit of a machine. Additionally, setups are required between batches with different families on a machine. Jobs come from the same customer have identical due dates, and according to contracts, penalty occurs when orders are delivered early or late. Thus, in this paper we attempt to formulate a constrained programming model to find a schedule for minimizing the weighted earliness and tardiness. By examining the model using two instances, the results showed that it requires a lot of computation time to find the optimal solution for the given examples due to its inherent complexity. It leads us to develop some heuristic algorithms to obtain optimal or near-optimal solutions efficiently in the future.

Introduction

This study is motivated by a quantity foodservice company in which contains three main stages: dish process, transportation, and unloading. In dish process, an order contains several dishes; each dish has its own recipe and size; dishes with different recipes cannot cook on the same pot, each pot has its capacity and it can simultaneously cook several dishes with the same recipes as long as the total sizes of dishes on the pot do not exceed its capacity. Besides, the clean operation is required when successive dishes with different recipes have to cook at the same pot. Once an order is completed it could be delivered to the corresponding school by a vehicle, thus all orders have a common due window to ensure all students have their dishes in the due window. For the problem, this paper focuses on the dish process, and according to the above characteristics, the problem could be treated as parallel batch-processing machines problems with incompatible job families where clean time is needed when successive dishes with different recipes cook at the same pot, and the objective is to minimize the total weighted earliness and tardiness based on just-in-time philosophy. Using the well-known three-field notation [1], our problem can be denoted by $P_m|B_k, s_i, incompatible, setup|\sum u_i E_i + w_i T_i$.

In the literature, the BPM problems can be classified into two parts according to whether or not jobs are compatible. In this paper, we consider the case of incompatible job families, which indicates that jobs from the same families can be processed simultaneously as a batch. Azizoglu and Webster [2] considered the problem with the objective of the total weighted completion time, and developed a B&B method that can return optimal solutions for problems with up to 25 jobs. Tangudu and Kurz [3] considered jobs’ release dates and the objective of total weighted tardiness. For this problem, they provided a B&B method which can solve problems up to 32 jobs. Mönch et al. [4] extended the single machine to parallel machines which is motivated by diffusion and oxidation operations in semiconductor wafer fabrication and developed two different decomposition approaches where GA is
used as an assignment mechanism; some studies investigated the same problems with different criteria such as Reichelt and Mönch [5], Koh et al. [6], and Balasubramanian et al. [7]. This paper also considers the parallel batch-processing machines with incompatible job family, however, to our best knowledge, this is the first effort on BPM scheduling for quantity foodservice companies unlike those research cited above. Also, minimizing total weighted earliness and tardiness on a BPM in the presence of sequence-dependent setup time has not been addressed in the literature.

Constrained programming model

For the problem, we address a constrained programming model, in this model, jobs are indexed by $i, j$, batch-processing machines by $k$, job family by $g$ and batches by $v$.

The objective is defined as

$$
\text{Min. } \sum_{i=1}^{n} (u_i \cdot E_i + w_i \cdot T_i)
$$

Subject to

$$
\sum_{k=1}^{m} \sum_{v=1}^{n} X_{ikv} = 1
$$

$$
\sum_{i=1}^{n} q_i \cdot X_{ikv} \leq b_k
$$

$$
X_{ikv} + X_{jkv} \leq 1
$$

$$
X_{ikv} = 1 \rightarrow BY_{kv} = L_i
$$

$$
\sum_{i=1}^{n} X_{ikv} = 0 \rightarrow BY_{kv} = 0
$$

$$
BY_{kv} = L_i \rightarrow BP_{kv} = p_{Li}
$$

$$
\sum_{i=1}^{n} X_{ikv} = 0 \rightarrow BP_{kv} = 0
$$

$$
\sum_{i=1}^{n} X_{ikv} > 0 \rightarrow BZ_{kv} = 1
$$

$$
\sum_{i=1}^{n} X_{ikv} = 0 \rightarrow BZ_{kv} = 0
$$

$$
BZ_{kv} \geq BZ_{kv+1}
$$

$$
(BZ_{kv} = 1) \land (BZ_{k,v+1} = 1) \land (BY_{kv} \neq BY_{k,v+1}) \rightarrow CZ_{k,v+1} = 1
$$

$$
(BZ_{kv} = 0) \lor (BZ_{k,v+1} = 0) \lor (BY_{kv} = BY_{k,v+1}) \rightarrow CZ_{k,v+1} = 0
$$

$$
BS_{k1} = 0
$$

$$
BC_{kv} \geq BS_{kv} + s_k \cdot CZ_{kv} + BP_{kv}
$$

$$
BS_{kv} \geq BC_{k,v-1}
$$

$$
X_{ikv} = 1 \rightarrow C_i = BC_{kv}
$$

$$
E_i \geq d_i - C_i
$$

$$
T_i \geq C_i - d_i
$$

The objective (1) is to minimize the sum of weighted earliness and tardiness. Constraint set (2) specifies that job $i$ only can be assigned to a batch $v$ and be processed once by machine $k$. Constraint set (3) give the bounds for each machine $k$ can process the jobs simultaneously as long as the total sizes of jobs in a batch $v$ do not exceed the capacity limit of machine $k$. Constraints (4) forces jobs $i, j$ with different job family cannot be assigned to batch $v$ and processed together by machine $k$. The family of batch $v$ being processed on machine $k$ is decided by constraints (5) and (6). The time required for processing batch $v$ on machine $k$ is specified by constraints (7) and (8). Constraints (9) and (10) aim to find out that batch $v$ is processed by machine $k$ or not. For machine $k$, batch $v$ is processed before the next batch $v+1$ by constraints (11). Constraints (12) and (13) force the setup operation is required for processing successive two batches $v$, $v+1$ by machine $k$ when the family of
batches $v$, $v+1$ is different. Constraints (14) force the start time for the first batch to being processed on machine $k$ to zero. Constraints (15) formulate the completion time of batch $v$ be greater than or equal to the sum of start time, setup time and the processing time of batch $v$ for being assigned to machine $k$. Constraints (16) specify the relationship for two batches $v$, $v+1$ on machine $k$ by their completion time. Constraints (17) define the completion time of job $i$ as the completion time of batch $v$ by machine $k$ if job $i$ is in batch $v$. Constraints (18) and (19) define the earliness and tardiness for each job respectively.

**Results**

With the complexity of the considered problem, we took two small instances to examine the validity of the proposed model. The two instances are as shown in Table 1:

<table>
<thead>
<tr>
<th>Table 1. the information of the two instances.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance 1</td>
</tr>
<tr>
<td>Number of schools (customers)</td>
</tr>
<tr>
<td>Number of courses (dishes)</td>
</tr>
<tr>
<td>Number of pots (machines)</td>
</tr>
<tr>
<td>Number of jobs</td>
</tr>
<tr>
<td>Capacity of pots</td>
</tr>
<tr>
<td>Clean time for each pot</td>
</tr>
<tr>
<td>Processing time for each course on each pot</td>
</tr>
<tr>
<td>Each job from which school</td>
</tr>
<tr>
<td>Course type of each job</td>
</tr>
<tr>
<td>Due date of each job</td>
</tr>
<tr>
<td>Earliness weight of each job</td>
</tr>
<tr>
<td>Tardiness weight of each job</td>
</tr>
</tbody>
</table>

Remarks: 1. (10,12) means that the time needed respectively for the first course cook on the two pots.
2. (1, *, *, *, *, *) indicates that the first job is from the first school
3. (*, *, *, *, 3, *) means that the course type of the fifth job is the third course

Each school can choose at most three course types among those provided by the quantity foodservice company where there are four and five course types respectively in instances 1 and 2.

The final schedule for instance 1 is shown in Fig. 1. In Fig. 1, the first clean operation occur at the beginning of the second batch containing jobs 1 and 4, it is due to the course type of the second batch is 1 which is different from the type 2 of the first batch. Based on the schedule, the value of the objective function is equal to 118.

The optimal solution for instance 2 is equal to 379 and the schedule is shown in Fig. 2. From the two results, it is evident that the proposed model could be used to solve the single and parallel BPM problem with the objective function of minimizing total weighted earliness and tardiness. For computation time, the optimal solutions of instances 1 and 2 are obtained in 1338.25 and 223.62 seconds respectively. As anticipated, the computation time increase dramatically with an increase in the number of machine or dimension of problem. To efficiently obtain optimal or near optimal solutions, it is necessary to develop some heuristic algorithms for the considered problem in the future.
Conclusions

In this paper, we addressed the parallel BPM problem with incompatible job families and non-identical job sizes. This problem is motivated by scheduling of the dish operation in the quantity foodservice industry, and based on the just-in-time philosophy the objective function of the total weighted earliness and tardiness is considered. To the best of our knowledge, this is the first study that attempts applying scheduling theory for the quantity foodservice industry. A constrained programming model was proposed and was examined by two instances. The results showed that the model is effectiveness for solving single BPM and parallel BPM problems. Due to the considered problem is NP hard, it is hard to obtain optimal solutions for large problems. Therefore, another important task is to provide efficient heuristics for the general case of the problem in the future.

![Figure 1. The result of instance 1 obtained by the proposed model.](image)

![Figure 2. The result of instance 2 obtained by the proposed model.](image)

Acknowledgment

This work was supported by the Ministry of Science and Technology, R.O.C., under grant number MOST 105-2221-E-238-003.

References


