

Order acceptance and scheduling on unrelated parallel machines

Wencheng Xie*

Department of Automotive Engineering, Hunan Financial & Industrial Vocational-technical College,
Hengyang, China

* Corresponding Author Email: miaokkx@163.com

Abstract. This paper considers an order acceptance and scheduling problem on unrelated parallel machines. The objective is to select and schedule a subset of accepted orders that maximizes the total profit, which is defined as the sum of revenues of all the accepted orders minus their weighted tardiness penalties. We formulate the problem as a mixed-integer linear programming model. To solve this strongly NP-hard problem, we also propose a heuristic algorithm based on the Lagrangian relaxation technique and some greedy search operations.

Keywords: Order acceptance; Scheduling; Unrelated parallel machines.

1. Introduction

The order acceptance and scheduling (OAS) problem consists of simultaneously determine which orders to accept for processing as well as how to schedule these accepted orders. This problem arises in many make-to-order (MTO) production environments, where the limited production capacity and distinct order-delivery requirements necessitate the decision of order acceptance. To the best of our knowledge, there is no research on developing heuristic method for OAS problem in the unrelated parallel machines setting. In this paper, a deterministic OAS problem on unrelated parallel machines is addressed. The objective is to maximize the total profit, which is defined as the difference between the sum of revenues of all the accepted orders and total weighted tardiness penalties. The main contribution of this paper is that we develop a heuristic algorithm based on Lagrangian relaxation to solve the large-sized instances efficiently.

The OAS problem has received extensively research attention over the past two decades [1]. Although the OAS problems have been studied with various objective functions, current researches focus more on two objective functions, that is, (1) maximization the total profit as the sum of revenues minus the total weighted tardiness [2-13], (2) minimization of the makespan of the accepted orders plus the total penalties of all rejected orders [14-16]. As previously stated, this paper studies the OAS problem on unrelated parallel machines with the first objective. With this objective, many researches have been performed on different machine environments, including single machine [2-8], flow shop [9-10] and parallel machines [11-13]. In the unrelated parallel machines context, Wang and Ye [13] developed a formulation-based branch-and-bound algorithm with the idea of “divide and conquer”. For more details with respect to the OAS problem, interested readers can refer to the survey by Slotnick [17].

2. Mathematical description of the problem

At the beginning of planning period, there is a set of potential orders(jobs) $N = \{1, 2, \dots, n\}$ to be processed on unrelated parallel machines $M = \{1, 2, \dots, m\}$. All the orders are available from time zero. A machine can handle only one order at a time and no preemption is permitted. For each order $i \in N$, let d_i , u_i and w_i be its due date, revenue and tardiness penalty, respectively. In addition, let p_{ij} be the processing times, corresponding to the time required to process order $i \in N$ on machine $j \in M$. If an order is accepted, it will be decided which machine should be assigned to process the accepted order. Let C_i denote the completion time of order i . The total penalty for each order i

is defined by $w_i T_i$, where $T_i = \max\{0, C_i - d_i\}$. The objective is to select and schedule a subset of accepted orders $N' \in N$ that maximizes the total profit, which is given by $\sum_{i \in N'} (u_i - w_i T_i)$.

It is easy to know that there exists an optimal solution in which all the accepted orders should have the positive profit. Then, let $Q_i = \lfloor u_i / w_i \rfloor + d_i$, which denotes an upper bound for the completion time of an accepted order $i \in N$. In addition, let $Q = \max\{Q_1, Q_2, \dots, Q_n\}$. Define the binary variables x_{ijt} , which take 1 if order i is accepted and its completion time on machine j is t , and 0 otherwise. The mixed-integer linear programming based on time-indexed variables is expressed as follows.

$$\max Z = \sum_{i=1}^n \sum_{j=1}^m \sum_{t=p_{ij}}^{Q_i} x_{ijt} (u_i - w_i T_i) \tag{1}$$

$$\text{s.t.} \quad \sum_{j=1}^m \sum_{t=p_{ij}}^{Q_i} x_{ijt} \leq 1, \quad \forall i \in N \tag{2}$$

$$\sum_{i=1}^n \sum_{s=\max\{p_{ij}, t\}}^{\min\{t+p_{ij}-1, Q_i\}} x_{ijs} \leq 1, \quad \forall j \in M, 1 \leq t \leq Q \tag{3}$$

$$x_{ijt} \in \{0, 1\}, \quad \forall i \in N, j \in M, 1 \leq t \leq Q \tag{4}$$

Objective function (1) maximizes the total profit. Constraint (2) states that if an order is accepted, then it can be processed at most once. Constraint (3) implies that exactly one order can be processed in each unit time slot on each machine. Finally, Constraints (4) defines the domain of the decision variables.

3. The proposed heuristic based on Lagrangian relaxation

The OAS problem on unrelated parallel machine is strongly NP-hard, since if order acceptance is determined, the resulting problem becomes a total weighted tardiness problem on unrelated parallel machines, which is strongly NP-hard because it includes the strongly NP-hard problem $1 \parallel \sum w_i T_i$ as a special case when the production environment becomes a single machine [8]. Then, it is justified to designed a heuristic algorithm for solving the problem approximately. With respect to the heuristic algorithm, we firstly adopt the Lagrangian relaxation technique to determine the subset of accepted orders and a feasible schedule. Besides, we also apply some greedy search operations to improve the solution.

3.1 Lagrangian subproblem

By dualizing constraints (3) with Lagrangian multipliers v_{jt} , we obtain the following Lagrangian problem.

$$L(v) = \max \left\{ \sum_{i=1}^n \sum_{j=1}^m \sum_{t=p_{ij}}^{Q_i} x_{ijt} (u_i - w_i \max\{t - d_i, 0\}) + \sum_{j=1}^m \sum_{t=1}^Q v_{jt} \left(1 - \sum_{i=1}^n \sum_{s=\max\{p_{ij}, t\}}^{\min\{t+p_{ij}-1, Q_i\}} x_{ijs} \right) \right\},$$

Subject to (2) and (4).

That is,

$$L(v) = \max \left\{ \sum_{i=1}^n \sum_{j=1}^m \sum_{t=p_{ij}}^{Q_i} x_{ijt} (u_i - w_i \max\{t - d_i, 0\}) - \sum_{s=\max\{t-p_{ij}+1, 1\}}^t v_{js} + \sum_{j=1}^m \sum_{t=1}^Q v_{jt} \right\},$$

Subject to (2) and (4).

Noting with a fixed vector v_{jt} , we can calculate $L(v)$ separately with respect to each order i and the total computational time is $O(nmQ)$. Specifically, for each order i , let

$$L_i(v) = \sum_{j=1}^m \sum_{t=p_{ij}}^{Q_i} x_{ijt} (u_i - w_i \max\{t - d_i, 0\} - \sum_{s=\max\{t-p_{ij}+1, 1\}}^t v_{js}).$$

We may impose the condition that $x_{ijt} = 0$ if $w_i(t - d_i) \geq u_i$, for $i = 1, 2, \dots, n$. Then the objective function is given by

$$L(v) = \max \left(\sum_{i=1}^n L_i(v) + \sum_{j=1}^m \sum_{t=1}^{Q_j} v_{jt} \right).$$

It is clear that for any given non-negative multipliers v , $L(v)$ gives a valid upper bound on optimal value for (1)-(3). The Lagrangian dual problem (LD) aims to minimize such upper bound by solving $\min_{v \geq 0} L(v)$.

3.2 Subgradient method

In order to avoid the heavy computational burden, we apply the subgradient method to determine v so as to find an optimal or near optimal solutions for LD. The subgradient method is explained as follows.

Step 1: Set the number of iterations K , the initial multipliers v^0 , the initial step size ρ , and the factor λ^0 . Let $k = 0$, $L_{best}(v) = 0$, and the lower bound $\bar{Z} = 0$.

Step 2: Calculate x^k and $L^k(v^k)$. If $L^k(v^k) > L_{best}(v)$, let $L_{best}(v) = L^k(v^k)$ and $x_{best} = x^k$.

Step 3: Let $v^{k+1} = \max(0, v^k + \beta^k \gamma^k)$, where γ^k is the subgradient vector at point x^k , $\lambda^k = \rho \lambda^{k-1}$, and $\beta^k = \lambda^k (\bar{Z} - L^k(v^k)) / \|\gamma^k\|^2$.

Step 4: If $k < K$, then let $k = k + 1$, go to Step 3; otherwise, stop.

Note that the factor λ^k must be in the interval $0 < \lambda^k < 2$. By the subgradient method, we obtain an upper bound $L_{best}(v)$ and the vector x_{best} . The latter indicates which orders to accept, which machine assigned to process the accepted orders and when the accepted orders are finished processing in the LD problem.

3.3 Heuristic algorithm

We develop a heuristic algorithm by taking the vector x_{best} as a preliminary result of the decision on order acceptance. As the order acceptance is determined, we can obtain a feasible solution for the accepted orders. Then, we improve the generated solution by applying some greedy search operations. The main steps of the heuristic algorithm are described as follows.

Step 1: Let N' denote the subset of accepted orders through subgradient method and $\bar{N} = N \setminus N'$. Sort the orders of N' in non-decreasing order of their upper bound of completion times Q_i . Let C'_j denote the completion time of the last order processed on machine j .

Step 2: For $i = 1, 2, \dots, |N'|$, perform Step 2-1 or 2-2 repeatedly.

2-1: If order i can at least be completed on time on one machine, then get machine h with $\arg \min_{j \in M} \{C'_j + p_{ij}\}$ and assign order i on machine h . Update C'_h and next i .

2-2: If order i cannot be completed on time on any machine in the current schedule, then get machine h with $\arg \min_{j \in M} \{C'_j + p_{ij}\}$ and assign order i on machine h . Compute the profit of i . If it is non-positive, then discard order i and let $\bar{N} = \bar{N} \cup \{i\}$. Update C'_h and next i .

Step 3: For $j = 1, 2, \dots, m$, determine successively one order θ with $\arg \max_{i \in \bar{N}} \{u_i / p_{ij}\}$ and insert order θ in all positions on the machine j . Record the objective values of the new generated schedules with respect to the positions. If the objective value among the generated schedules is better than the current schedule, update the latter. Next j .

Step 4: Beginning with the first order on each machine, exchange the positions of two adjacent orders to generated a new schedule successively. If the generated schedule has a better objective than the current schedule, update the latter.

4. Conclusions

In this paper we have studied an order acceptance and scheduling problem on unrelated parallel machines, which has few results in available literature. A mixed-integer linear programming based on time-indexed variables is developed. Applying the Lagrangian relaxation technique and some greedy search operations, we also design a constructive heuristic algorithm for solving the OAS problem under study. As for future research, one can try to apply meta-heuristics to tackle the OAS problem under study. Additionally, studying the OAS problem under more general or complex machine environments is also valuable.

References

- [1] Esmaeilbeigi R, Charkhgard P, Charkhgard H. Order acceptance and scheduling problems in two-machine flow shops: New mixed integer programming formulations[J]. *European Journal of Operational Research*, 2016, 251(2): 419-431.
- [2] Slotnick S A, Morton T E. Order acceptance with weighted tardiness[J]. *Computers & Operations Research*, 2007, 34(10): 3029-3042.
- [3] Nobibon F T, Leus R. Exact algorithms for a generalization of the order acceptance and scheduling problem in a single-machine environment[J]. *Computers & Operations Research*, 2011, 38(1): 367-378.
- [4] Oguz C, Salman F S, Yalcin Z B. Order acceptance and scheduling decisions in make-to-order systems[J]. *International Journal of Production Economics*, 2010, 125(1): 200-211.
- [5] Cesaret B, Oguz C, Salman F S. A tabu search algorithm for order acceptance and scheduling[J]. *Computer & Operations Research*, 2012, 39(6): 1197-1205.
- [6] Lin S W, Ying K. Increasing the total net revenue for single machine order acceptance and scheduling problems using an artificial bee colony algorithm[J]. *Journal of the Operational Research Society*, 2013, 64: 293-311.
- [7] Chaurasia S N, Singh A. Hybrid evolutionary approaches for the single machine order acceptance and scheduling problem[J]. *Applied Soft Computing*, 2017, 52: 725-747.
- [8] Silva Y L T V, Subramanian A, Pessoa A A. Exact and heuristic algorithm for order acceptance and scheduling with sequence-dependent setup times[J]. *Computers & Operations Research*, 2018, 90: 142-160.
- [9] Wang X, Xie X, Cheng T C E. A modified artificial bee colony algorithm for order acceptance in two-machine flow shops [J]. *International Journal of Production Economics*, 2013, 141(1): 14-23.
- [10] Wang X, Xie X, Cheng T C E. Order acceptance and scheduling in a two-machine flowshop[J]. *International Journal of Production Economics*, 2013, 141(1): 366-376.
- [11] Wang X, Huang G, Hu X, et al. Order acceptance and scheduling on two identical parallel machines[J]. *Journal of the Operational Research Society*, 2015, 66(10): 1755-1767.
- [12] Wu G H, Cheng C Y, Yang H I, et al. An improved water flow-like algorithm for order acceptance and scheduling with identical parallel machines[J]. *Applied Soft Computing*, 2018, 71: 1072-1084.
- [13] Wang S, Ye B. Exact methods for order acceptance and scheduling on unrelated parallel machines[J]. *Computers & Operations Research*, 2019, 104:159-173.
- [14] Ou J, Zhong X. Bicriteria order acceptance and scheduling with consideration of fill rate[J]. *European Journal of Operational Research*, 2017, 262(3): 904-907.
- [15] Ou J, Zhong X, Wang G. An improved heuristic for parallel machine scheduling with rejection[J]. *European Journal of Operational Research*, 2015, 241(3): 653-661.
- [16] Ou J, Zhong X, Qi X. Scheduling parallel machines with inclusive processing set restrictions and job rejection[J]. *Naval Research Logistics*, 2016, 63: 667-681.
- [17] Slotnick S A, Order acceptance and scheduling: A taxonomy and review[J]. *European Journal of Operational Research*, 2011, 212(1): 1-11.