



HAL
open science

Multitasking Scheduling Problem with Uncertain Credit Risk

Feifeng Zheng, Zhaojie Wang, Yinfeng Xu, Ming Liu

► **To cite this version:**

Feifeng Zheng, Zhaojie Wang, Yinfeng Xu, Ming Liu. Multitasking Scheduling Problem with Uncertain Credit Risk. IFIP International Conference on Advances in Production Management Systems (APMS), Sep 2021, Nantes, France. pp.266-273, 10.1007/978-3-030-85906-0_30 . hal-04022128

HAL Id: hal-04022128

<https://inria.hal.science/hal-04022128>

Submitted on 9 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

Multitasking Scheduling Problem with Uncertain Credit Risk^{*}

Feifeng Zheng¹, Zhaojie Wang¹, Yinfeng Xu¹, and Ming Liu²

¹ Glorious Sun School of Business and Management, Donghua University, 200051, Shanghai, China

² School of Economics and Management, Tongji University, 200092, Shanghai, China

Abstract. Parallel machine scheduling problem in multitasking environment plays an important role in modern manufacturing industry. Multitasking is a special scheduling method, in which each waiting job interrupts the primary job, causing an interruption time and a switching time. The existing literatures discuss the problem of multitasking scheduling, however, few studies consider credit risk into such a realm of scheduling models. In this work, we combine customer credit risk into a multitasking scheduling problem. Besides, due to the existence of credit risk and the constraint of deadline of each accepted job, we also consider the job rejection into this problem. To hedge against the worst-case performance (total profit in the worst-case), we then propose a robust stochastic mathematical model with the objective of minimising the maximum difference between total job rejection cost and total revenue. As commercial solvers cannot directly solve this robust stochastic programming model, a sample average approximation model is proposed to further solve this problem. Numerical experiments are conducted to demonstrate the effectiveness of the proposed sample average approximation approach.

Keywords: Multitasking· Scheduling· Stochastic· Credit risk· Robust

1 Introduction

In the industry 4.0 environment, scheduling should deal with an intelligent manufacturing system supported by new and emerging manufacturing technologies, such as mass customization (Zhang et al. 2019). Hence, customer credit risk (or payment probability) is becoming a main challenge faced by modern manufacturing industry (Liu et al. 2020).

Parallel machine scheduling is a classical subject of study and common problem in practice (Wu and Che 2019). As a special parallel scheduling mode, multitasking allows the machine to perform multiple tasks (Liu et al. 2017). In multitasking settings, each waiting job interrupts the primary job, causing an interruption time and a switching time. In fact, parallel machine scheduling in

^{*} This work was supported by the National Natural Science Foundation of China (Grant No. 71771048, 71832001, 72021002, 72071144 and 71972146) and the Fundamental Research Funds for the Central Universities (Grant No. 2232018H-07).

multitasking environment is a very common phenomenon such as chemical manufacturing, food processing, and oil refining (Gaglioppa et al. 2008). Despite this, few studies investigate the impact of customer credit risk on parallel machine scheduling under multitasking environment.

Motivated by scheduling practices, we investigate a multitasking scheduling problem considering customer credit risk. Due to the existence of credit risk and the limitation of deadline of each accepted job, the option of job rejection is a very practical and realistic feature (Mor et al. 2020). Hence, we make a reasonable assumption in this work that any job can be rejected. For solving this problem, we first establish a robust stochastic mathematical model to hedge against the worst-case performance (i.e., total profit in the worst-case), and then propose a sample average approximation (SAA) approach to solve this problem.

1.1 Literature Review

As parallel machine scheduling in multitasking environment is a very common phenomenon in manufacturing industry, the research on multitasking scheduling problem has attracted wide attention of scholars.

Gaglioppa et al. (2008) consider the planning and scheduling of production in a multitask batch manufacturing process which is typical industries. In their problem, they allow instances in which multiple sequences of tasks may be used to produce end products. For solving this problem, they first establish a mixed-integer linear program, and then propose a new family of efficient inequalities to reduce the solution time. Liu et al. (2019) investigate a multitasking scheduling problem on parallel machines with uncertain processing times. The objective of their problem is to minimize the weighted sum of the earliness and tardiness. They then propose a two-stage stochastic programming formulation based on scenarios to solving this problem. Zhu et al. (2016) study multitasking scheduling problems with a rate-modifying activity. In the problems, the processing of a selected task suffers from interruptions by other tasks that are available but unfinished, and the human operators regularly engage rest breaks during work shifts allowing them to recover or mitigate some of the negative effects of fatigue. Hall et al. (2015) develop optimal algorithms for some fundamental and practical single machine scheduling problems with multitasking. Ji et al. (2018) consider the problem of parallel-machine scheduling with machine-dependent slack (SLK) due-window assignment in the multitasking environment. The objectives of their studied problem is to minimise the total cost that comprises the earliness, tardiness, and due-window-related costs. Xiong et al. (2019) consider the multitasking scheduling problem on unrelated parallel machines to minimize the total weighted completion time. For solving this problem, they propose an exact branch and price algorithm. Li et al. (2020) present four fuzzy optimization models for scheduling multiple heterogeneous complex tasks with distributed resources/services in consideration of multiple criteria with fuzzy uncertainty in service-oriented manufacturing environment. For solving those models, they then develop corresponding heuristic algorithms in their study.

In sum, we know that few studies consider external factors, such as the impact of credit risk on multitasking scheduling problem. Liu et al. (2020) study a stochastic flowshop scheduling problem with the objective of maximizing the profit level. They show that credit risk has an important impact on the scheduling decision of enterprises. Inspired by Liu et al. (2020), we consider credit risk into parallel machine scheduling under multitasking environment. The main contributions of this work include: (1) Credit risk is considered in the multitasking scheduling problem, which makes this research more in line with the practical production; (2) A robust stochastic optimization model is established to study this problem; (3) A SAA method is proposed to further solve this problem.

The remainder of this paper is organized as follows. Section 2 describes the considered problem and establishes a robust mathematical model for the this problem. Section 3 proposes a SAA model for ease of adopting commercial solvers to solve this problem. Finally, Section 4 conducts numerical experiments to evaluate the performance of the proposed model.

2 Problem statement

Suppose that there are $|\mathcal{J}|$ jobs to be processed on $|\mathcal{M}|$ parallel machines, and the release time of all jobs are 0. Due to the constraint of the deadline of each job, we make a reasonable assumption, that is, any job can be rejected.

For describing the details of multitasking scheduling process, we assume that only one machine is available. We use \mathcal{J}_i to represent the job set that will be processed on the machine i . In \mathcal{J}_i , any job scheduled for processing at any time is called a primary job, while all the other unsatisfied (or unfinished) jobs at that time are referred to as waiting jobs (Hall et al. 2015). Beside, in \mathcal{J}_i , any primary job must be interrupted by all the waiting jobs during its processing. Figure 1 describes a multitasking environment in which two accepted jobs are processed on a machine. Since any accepted job have to be completed, each job is selected as a primary job exactly once, and a primary job is always completed before other waiting jobs.

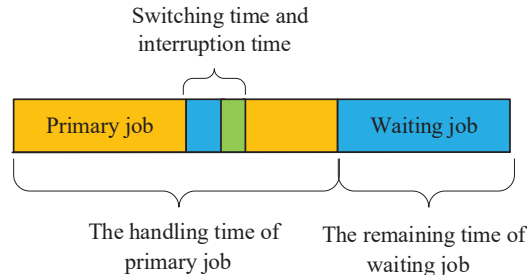


Fig. 1. Illustration of multitasking processing

Due to the uncertainty of credit risk, the total profit usually cannot be evaluated deterministically by decision makers. To address this problem, we investigate a robust multitasking scheduling problem considering uncertain customer credit risk. The objective is to minimise the maximum difference between total penalty cost, i.e., job rejection cost and total revenue.

2.1 Mathematical model

Below we first present basic parameters and decision variables, and then establish a robust stochastic programming model.

Input parameters:

- \mathcal{M} : set of machines, indexed by i , and $\mathcal{M} = \{1, 2, \dots, |\mathcal{M}|\}$;
- \mathcal{J} : set of jobs, indexed by j , and $\mathcal{J} = \{1, 2, \dots, |\mathcal{J}|\}$;
- \mathcal{W} : set of positions, indexed by w , and $\mathcal{W} = \{1, 2, \dots, |\mathcal{W}|\}$;
- Ω : set of all random events indexed by ω .
- $\xi_j(\omega)$: stochastic payment probability of job j under event $\omega \in \Omega$;
- p_j : the processing time of job j ;
- π_j : revenue for processing job j ;
- e_j : penalty cost for rejecting job j ;
- η : the parameter given in the switching time function $f(|S_j|) = \eta * |S_j|$, where $|S_j|$ denotes the number of waiting jobs processed on the same machine as the job j when processing job j ;
- ν_j : the parameter given in the interruption time function $g_j(p'_j) = \nu_j * p'_j$, where p'_j is the remaining time of job j , $g_j(p'_j)$ is the interruption time;
- d_j : deadline of job j ;
- M : a sufficiently large positive integer.

Decision Variables

- $x_{ij}^w(\omega)$: a binary variable equal to 1 if job j is processed by machine i at position w under event $\omega \in \Omega$, 0 otherwise;
- $\lambda_i(\omega)$: the number of jobs that are processed on machine i under event $\omega \in \Omega$;
- $\tilde{p}_j(\omega)$: the handling time of job j under event $\omega \in \Omega$;
- $y_j(\omega)$: a binary variable equal to 1 if job j is accepted under event $\omega \in \Omega$, 0 otherwise;
- $l_{jj'}(\omega)$: a binary variable equal to 1 if the position assigned to job j ahead of that assigned to job j' on the same machine under event $\omega \in \Omega$, 0 otherwise;
- $h_{ij}^w(\omega)$: the remaining time of job j on machine i at position w under event $\omega \in \Omega$, where $h_{ij}^w(\omega) = (1 - \nu_j)^{w-1} * p_j$;
- $\theta_j(\omega)$: the interruption time of all the waiting jobs for processing job j under event $\omega \in \Omega$;
- $s_j(\omega)$: the start processing time of job j under event $\omega \in \Omega$;
- $c_j(\omega)$: the completion time of job j under event $\omega \in \Omega$.

Robust stochastic programming formulation [P1]:

$$\min \max \left\{ \sum_{j \in \mathcal{J}} (1 - y_j(\omega)) * e_j - \sum_{j \in \mathcal{J}} y_j(\omega) * \xi_j(\omega) * \pi_j \right\} \quad (1)$$

subject to

$$\sum_{i \in \mathcal{M}} \sum_{w \in \mathcal{W}} x_{ij}^w(\omega) = y_j(\omega), \quad j \in \mathcal{J}, \omega \in \Omega \quad (2)$$

$$\sum_{j \in \mathcal{J}} x_{ij}^w(\omega) \leq 1, \quad i \in \mathcal{M}, w \in \mathcal{W}, \omega \in \Omega \quad (3)$$

$$\sum_{j \in \mathcal{J}} x_{ij}^w(\omega) \geq \sum_{j \in \mathcal{J}} x_{ij}^{w'}(\omega), \quad i \in \mathcal{M}, w, w' \in \mathcal{W}, w < w', \omega \in \Omega \quad (4)$$

$$h_{ij}^w(\omega) \geq (1 - \nu_j)^{w-1} * p_j - M * (1 - x_{ij}^w(\omega)), \quad i \in \mathcal{M}, j \in \mathcal{J}, w \in \mathcal{W}, \omega \in \Omega \quad (5)$$

$$\lambda_i(\omega) = \sum_{j \in \mathcal{J}} \sum_{w \in \mathcal{W}} x_{ij}^w(\omega), \quad i \in \mathcal{M}, \omega \in \Omega \quad (6)$$

$$\theta_j(\omega) \geq \sum_{j' \in \mathcal{J}} \sum_{w'=w+1}^{|\mathcal{W}|} \nu_j * (1 - \nu_j)^{w-1} * p_j * x_{ij'}^{w'} - M * (1 - x_{ij}^w(\omega)),$$

$$i \in \mathcal{M}, j \in \mathcal{J}, w \in \{\mathcal{W}\} \setminus \{|\mathcal{W}|\}, \omega \in \Omega \quad (7)$$

$$\tilde{p}_j(\omega) = h_{ij}^w(\omega) + \eta * (\lambda_i(\omega) - w) + \theta_j(\omega), \quad i \in \mathcal{M}, j \in \mathcal{J}, w \in \mathcal{W}, \omega \in \Omega \quad (8)$$

$$l_{jj'}(\omega) \geq 1 - M * (2 - x_{ij'}^w(\omega) - x_{ij}^w(\omega)),$$

$$i \in \mathcal{M}, j, j' \in \mathcal{J}, w, w' \in \mathcal{W}, w > w', \omega \in \Omega \quad (9)$$

$$s'_j(\omega) \geq c_j(\omega) - M * (1 - l_{jj'}(\omega)), \quad j, j' \in \mathcal{J}, j \neq j', \omega \in \Omega \quad (10)$$

$$c_j(\omega) = s_j(\omega) + \tilde{p}_j, \quad i \in \mathcal{M}, j \in \mathcal{J}, \omega \in \Omega \quad (11)$$

$$c_j(\omega) \leq d_j, \quad j \in \mathcal{J}, \omega \in \Omega \quad (12)$$

$$x_{ij}^w(\omega), y_j(\omega), l_{jj'}(\omega) \in \{0, 1\}, \quad i \in \mathcal{M}, j \in \mathcal{J}, w \in \mathcal{W}, \omega \in \Omega \quad (13)$$

$$s_j(\omega), c_j(\omega), \lambda_i(\omega), \theta_j(\omega), \tilde{p}_j(\omega), h_{ij}^w(\omega) \geq 0, \quad i \in \mathcal{M}, j \in \mathcal{J}, w \in \mathcal{W}, \omega \in \Omega \quad (14)$$

Formula (1) is the objective function, which is used to minimise the maximum difference between total penalty cost and total revenue. Constraint (2) denotes that the accepted job must be processed under event $\omega \in \Omega$. Constraint (3) implies that no more than one job can be processed at a certain position of a machine under event $\omega \in \Omega$. Constraint (4) guarantees that the position of machine $i \in \mathcal{M}$ is continuous when processing the jobs under event $\omega \in \Omega$. Constraint (5) calculates the remaining processing time of job $j \in \mathcal{J}$ on machine $i \in \mathcal{M}$ under event $\omega \in \Omega$, where $(1 - \nu_j)^{w-1}$ represents the $w - 1$ power of ν_j . Constraint (6) calculates the number of jobs assigned to machine $i \in \mathcal{M}$ under event $\omega \in \Omega$. Constraint (7) calculates the interruption time of all the waiting jobs for processing job $j \in \mathcal{J}$ under event $\omega \in \Omega$. Constraint (8) calculates the handling time of job $j \in \mathcal{J}$ on machine $i \in \mathcal{M}$ under event $\omega \in \Omega$, which includes three parts: its remaining processing time, the switching and the interruption time of all the waiting jobs. Constraints (9)-(11) calculate the

start processing time and completion time of job $j \in \mathcal{J}$ under event $\omega \in \Omega$, respectively. Constraint (12) guarantees that the completion time of job $j \in \mathcal{J}$ is less than its deadline under event $\omega \in \Omega$. Constraints (13)-(14) limit the ranges of the variables.

3 SAA approach

Sample average approximation (SAA) has widely used to deal with stochastic optimization problem (e.g., Li and Zhang, 2018). The main idea of this approach is to adopt deterministic optimization techniques with Monte Carlo simulation (Zheng et al. 2018).

In this section, we assume that the random vector $\xi_j(j \in \mathcal{J})$ has finite support, i.e. Ω contains a finite number of random events, and $\omega = 1, 2, \dots, |\Omega|$ index its possible realisations. [P1] can be approximately transformed into [P2].

$$[\mathbf{P2}] : \min \theta \tag{15}$$

subject to

$$(2) - (14) \\ \sum_{j \in \mathcal{J}} (1 - y_j(\omega)) * e_j - \sum_{j \in \mathcal{J}} y_j(\omega) * \xi_j(\omega) * \pi_j \leq \theta, \quad \omega \in \Omega \tag{16}$$

Constraint (16) implies that the maximum difference between total job rejection cost and total revenue under event $\omega \in \Omega$ cannot be larger than θ .

4 Numerical experiments

For demonstrating the effectiveness of the proposed SAA approach, we carry out numerical experiments via Matlab R2017a. All numerical experiments are conducted on a PC with AMD Ryzen 7 4800U with Radeon Graphics 1.80 GHz.

4.1 Experiment setup

In this work, we test two probability distribution of $\xi_j(j \in \mathcal{J})$, i.e., uniform and normal distributions. For normal distributions, according to Ng (2015), the variance (σ^2) of ξ_j is set as $\sigma^2 = 0.005 * \mu^2$, where μ is the estimates of mean of ξ_j . In order to compare the results of the two distributions, we set the mean value of the normal distribution to 0.5, which is the same as that of the uniform distribution. For each probability distribution, the instances to be tested and the values or ranges of generated other parameters are shown in Table 1.

Table 1. Parameter settings

<i>Instance</i>	p_j	d_j	e_j	η	ν_j	π_j
$(\mathcal{M} = 2; \mathcal{J} = 5; \Omega = 3)$	[5,10]	[20, 30]	[1, 3]	0.5	[0.1, 0.3]	[5,10]
$(\mathcal{M} = 2; \mathcal{J} = 7; \Omega = 3)$	[5,10]	[30, 50]	[3, 5]	1	[0.3, 0.5]	[8,10]
$(\mathcal{M} = 3; \mathcal{J} = 5; \Omega = 3)$	[5,10]	[20, 30]	[1, 3]	0.5	[0.1, 0.3]	[5,10]
$(\mathcal{M} = 3; \mathcal{J} = 7; \Omega = 3)$	[5,10]	[30, 50]	[3, 5]	1	[0.3, 0.5]	[8,10]
$(\mathcal{M} = 2; \mathcal{J} = 5; \Omega = 5)$	[5,10]	[20, 30]	[1, 3]	0.5	[0.1, 0.3]	[5,10]
$(\mathcal{M} = 2; \mathcal{J} = 7; \Omega = 5)$	[5,10]	[30, 50]	[3, 5]	1	[0.3, 0.5]	[8,10]
$(\mathcal{M} = 3; \mathcal{J} = 5; \Omega = 5)$	[5,10]	[20, 30]	[1, 3]	0.5	[0.1, 0.3]	[5,10]
$(\mathcal{M} = 3; \mathcal{J} = 7; \Omega = 5)$	[5,10]	[30, 50]	[3, 5]	1	[0.3, 0.5]	[8,10]
$(\mathcal{M} = 2; \mathcal{J} = 5; \Omega = 8)$	[5,10]	[20, 30]	[1, 3]	0.5	[0.1, 0.3]	[5,10]
$(\mathcal{M} = 2; \mathcal{J} = 7; \Omega = 8)$	[5,10]	[30, 50]	[3, 5]	1	[0.3, 0.5]	[8,10]
$(\mathcal{M} = 3; \mathcal{J} = 5; \Omega = 8)$	[5,10]	[20, 30]	[1, 3]	0.5	[0.1, 0.3]	[5,10]
$(\mathcal{M} = 3; \mathcal{J} = 7; \Omega = 8)$	[5,10]	[30, 50]	[3, 5]	1	[0.3, 0.5]	[8,10]

4.2 Experimental results

Experimental results of SAA approach under the two distributions are shown in Table 2.

Table 2 summarizes that the average objective value of SAA under uniform distribution is -16.5, which about 29.8% larger than that under normal distribution (-23.5). It can also be observed that the increase of the number of scenarios and instance size may lead to the increase of computation time. Decision makers can first determine the distribution of stochastic payment probability of each job according to historical data, and then use the proposed robust stochastic programming model and SAA approach to make a scheduling plan to hedge against the worst-case performance.

In the future research, we will explore several rule-based heuristics, and then compare them with existing algorithms through numerical experiments. Besides, multitasking environment with unrelated machines can also be considered to further study the considered problem.

References

1. Gaglioppa, F., Miller, L. A., Benjaafar, S.: Multitask and multistage production planning and scheduling for process industries. *Operations Research* **56**(4),1010–1025 (2008)
2. Hall, N. G., Leung, J., Li ,C.: The effects of multitasking on operations scheduling. *Production and Operations Management* **24**(8), 1248–1265 (2015)
3. Ji, M., Zhang, W., Liao, L., Cheng, T. C. E., Tan, Y.: Multitasking parallel-machine scheduling with machine-dependent slack due-window assignment. *International Journal of Production Research* **57**(6), 1667–1684 (2018)
4. Li, F., Liao, T.W., Cai, W.T., Zhang, L.: Multitask scheduling in consideration of fuzzy uncertainty of multiple vriteria in service-oriented manufacturing. *IEEE Transactions on Fuzzy Systems*. **126**(11), 2759–2771 (2020).

Table 2. Experimental results of SAA approach with uniform and normal distributions

Instance	Uniform distribution		Normal distribution	
	<i>obj</i>	time (s)	<i>obj</i>	time(s)
1	-14.8	5	-16.6	4
2	-16.4	5	-31.6	5
3	-14.8	8	-16.6	8
4	-16.4	9	-31.6	9
5	-12.0	6	-15.9	6
6	-22.7	14	-30.7	14
7	-12.0	5	-15.9	5
8	-22.7	12	-30.7	12
9	-8.0	9	-15.4	9
10	-25.1	21	-30.7	21
11	-8.0	7	-15.4	7
12	-25.1	18	-30.7	18
Average	-16.5	10	-23.5	10

5. Li, X., Zhang, K.: A sample average approximation approach for supply chain network design with facility disruptions. *Computers & Industrial Engineering* **126**, 243–251 (2018)
6. Liu, M., Liu, X., Chu, F., Chu, C.B.: Profit-oriented distributionally robust chance constrained flowshop scheduling considering credit risk. *International Journal of Production Research* **58**(8), 2527–2549 (2020)
7. Liu, M., Wang, S., Zheng, F., Chu, C.: Algorithms for the joint multitasking scheduling and common due date assignment problem. *International Journal of Production Research* **55**(20), 6052–6066 (2017)
8. Liu, M., Liu, R., Liu, X.: Two-stage stochastic programming for parallel machine multitasking to minimize the weighted sum of tardiness and earliness. 16th International Conference on Service Systems and Service Management (ICSSSM) (2019)
9. Mor, B., Mosheiov, G., Shapira, D.: Flowshop scheduling with learning effect and job rejection. *Journal of Scheduling* **23**(6), 631–641 (2020)
10. Ng, M. W.: Container vessel fleet deployment for liner shipping with stochastic dependencies in shipping demand. *Transportation Research Part B* **74**, 79–87 (2015)
11. Wu, X., Che, A.: A memetic differential evolution algorithm for energy-efficient parallel machine scheduling. *Omega* **82**, 155–165 (2019)
12. Xiong, X., Zhou, P., Yin, Y., Cheng, T. C. E., Li, D.: An exact branch and price algorithm for multitasking scheduling on unrelated parallel machines. *Naval Research Logistics* **66**(6), 502–516 (2019)
13. Zhang, J., Ding, G., Zou, Y., Qin, S., Fu, J.: Review of job shop scheduling research and its new perspectives under Industry 4.0. *Journal of Intelligent Manufacturing* **30**(4), 1809–1830 (2019)
14. Zheng, F., Man, X., Chu, F., Liu M., Chu, C.: A two-stage stochastic programming for single yard crane scheduling with uncertain release times of retrieval tasks. *Journal of Production Research* **57**(13), 4132–4147 (2018)
15. Zhu, Z., Zheng, F., Chu, C.: Multitasking scheduling problems with a rate-modifying activity. *International Journal of Production Research* **55**(1), 296–312 (2016)