Lecture Notes in Mechanical Engineering

Chen-Fu Chien
Runliang Dou
Li Luo *Editors*

Proceedings of Industrial Engineering and Management

International Conference on Smart Manufacturing, Industrial and Logistics Engineering and Asian Conference of Management Science and Applications

Springer

Lecture Notes in Mechanical Engineering

Lecture Notes in Mechanical Engineering (LNME) publishes the latest developments in Mechanical Engineering—quickly, informally and with high quality. Original research or contributions reported in proceedings and post-proceedings represents the core of LNME. Volumes published in LNME embrace all aspects, subfields and new challenges of mechanical engineering.

To submit a proposal or request further information, please contact the Springer Editor of your location:

Europe, USA, Africa: Leontina Di Cecco at Leontina.dicecco@springer.com
China: Ella Zhang at ella.zhang@springer.com
India: Priya Vyas at priya.vyas@springer.com
Rest of Asia, Australia, New Zealand: Swati Meherishi at swati.meherishi@springer.com

Topics in the series include:

- Engineering Design
- Machinery and Machine Elements
- Mechanical Structures and Stress Analysis
- Automotive Engineering
- Engine Technology
- Aerospace Technology and Astronautics
- Nanotechnology and Microengineering
- Control, Robotics, Mechatronics
- MEMS
- Theoretical and Applied Mechanics
- Dynamical Systems, Control
- Fluid Mechanics
- Engineering Thermodynamics, Heat and Mass Transfer
- Manufacturing Engineering and Smart Manufacturing
- Precision Engineering, Instrumentation, Measurement
- Materials Engineering
- Tribology and Surface Technology

Indexed by SCOPUS, EI Compendex, and INSPEC.

All books published in the series are evaluated by Web of Science for the Conference Proceedings Citation Index (CPCI).

To submit a proposal for a monograph, please check our Springer Tracts in Mechanical Engineering at https://link.springer.com/bookseries/11693.

Chen-Fu Chien · Runliang Dou · Li Luo
Editors

Proceedings of Industrial Engineering and Management

International Conference on Smart Manufacturing, Industrial and Logistics Engineering and Asian Conference of Management Science and Applications

Springer

Editors
Chen-Fu Chien
Department of Industrial Engineering
and Engineering Management
National Tsing Hua University
Hsinchu, Taiwan

Runliang Dou
Department of Information Management
and Management Science
Tianjin University
Tianjin, China

Li Luo
Department of Industrial Engineering
and Management
Sichuan University
Chengdu, China

ISSN 2195-4356 ISSN 2195-4364 (electronic)
Lecture Notes in Mechanical Engineering
ISBN 978-981-97-0193-3 ISBN 978-981-97-0194-0 (eBook)
https://doi.org/10.1007/978-981-97-0194-0

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

With the promotion of Industry 4.0 and China's Smart Manufacturing 2025, smart manufacturing and industrial logistics engineering have attracted attention from various countries and are key to achieving digital transformation of the manufacturing industry. In 2023, the 3rd International Conference on Smart Manufacturing, Industrial and Logistics Engineering (SMILE2023) and the 7th Asian Conference of Management Science and Applications (ACMSA2023) aims to provide a platform for global experts and scholars to exchange the research developments and latest practice on big data and smart manufacturing, Industrial Internet of Things, industrial automation, intelligent robots, artificial intelligence, virtual reality, and so on. Furthermore, the involved research and applications are not limited to conventional manufacturing domains and can be extended to manufacturing-based services as well as emerging areas such as green supply chains, logistics and so on.

Proceedings of Industrial Engineering and Management—International Conference on Smart Manufacturing, Industrial and Logistics Engineering and Asian Conference of Management Science and Applications caters to the purpose by recording the new research findings and development in the above domains. All the papers included in the proceedings have undergone rigid peer review. What is more excited, you are the experts or scholars with significant achievements in the fields. We believe that the proceedings will serve as the guidebook for the potential development in smart manufacturing and industrial logistics engineering and play great role in promoting smart manufacturing and industrial logistics engineering development.

SMILE2023 is sponsored by Sichuan University, Tianjin University and Taiwan Tsing Hua University and organized by Business School, Sichuan University. We would like to extend our sincerest thanks to Springer for their generous support in the compilation of the proceedings. We also would like to extend sincerest thanks to Business School, Sichuan University for holding such an excellent event, to all the delegates, keynote speakers, and the staff of the organization committee for their great contribution to the success of the conference in various ways.

Thank you very much!

Hsinchu, Taiwan — Chen-Fu Chien
Tianjin, China — Runliang Dou
Chengdu, China — Li Luo
November 2023

Contents

Optimal Modeling and Decision Analysis for Smart Manufacturing

Quantum Computing Approaches to Optimize Employee Scheduling in Multi-task Call Centers

Cheng Li, Zhaoyang Liu, Yu Song, Haojie Liu, Hanlin Liu, and Xiaodong Liu

Abstract The call center industry urgently needs efficient staff scheduling due to unpredictable customer service demands and challenges in human resource coordination. Appropriate scheduling is vital to enhance operational efficiency, reduce labor costs and improve customer satisfaction. This research is aim to solve the daily multi-task staff scheduling problem by integrating the innovative capabilities of QUBO (Quadratic Unconstrained Binary Optimization) model and Quantum Computing. We formulate the intricate parameters related to the number of employees, task type, and skill proficiency using a traditional mixed-integer programming model. Specifically, the study formulates scheduling constraints and objectives and then we convert this model into a QUBO model this transformation makes the problem to be suitable for Quantum Annealing (QA) and enabling subsequent integration with QA technology. This technology enables rapid and advanced exploration of extensive solution spaces and identify the optimal or near-optimal solutions. This research offers valuable insights and substantial groundwork for future explorations and developments in optimizing call center staff scheduling problem.

Keywords Quantum annealing · Employee scheduling · Call centers · Quadratic unconstrained binary optimization (QUBO)

C. Li (✉)
Comprehensive Research Organization, Fukuoka Institute of Technology, Fukuoka, Japan
e-mail: lanyanseallen@gmail.com

Z. Liu · H. Liu · H. Liu · X. Liu
Graduate School of Engineering, Fukuoka Institute of Technology, Fukuoka, Japan

Y. Song
Department of System Management, Fukuoka Institute of Technology, Fukuoka, Japan

C. Chien et al. (eds.), *Proceedings of Industrial Engineering and Management*, Lecture Notes in Mechanical Engineering, https://doi.org/10.1007/978-981-97-0194-0_1

1 Introduction

The growth of the call center industry has significantly influenced customer service practices across diverse sectors. As the nexus between companies and their clientele, call centers shoulder the responsibility of managing a myriad of inquiries, complaints and support requests. This diverse and unpredictable changing of customer interactions requires robust and adaptive staff scheduling [1]. It is essential to maintain both service quality and operational efficiency.

In the light of these demands, efficient and effective scheduling has become a crucial element. It not only sustains operational excellence but also minimizes labor expenditures, promoting higher levels of customer satisfaction. The inherent complexities of call center staffing scheduling problem can be attributed to factors such as unstable call volumes, a wide range of service domains and the myriad of requisite skill sets. Solving these difficulties is essential for refining workflows and optimizing resource allocation. The ultimate goal is to achieve a harmonious balance between service quality and operational cost.

However, it is difficult to achieve the equilibrium. The multifaceted nature of customer interactions makes scheduling a challenge. The complexities of call center operations further amplify this intricacy. Traditional optimization methods have offered solutions to some extent. They sometimes are inadequate when faced with too many constraints and dynamic variables typical of call center settings. These limitations of traditional methods highlight the importance of exploring new technologies and strategies to generate more adept ways for better call center scheduling.

This research uses Quadratic Unconstrained Binary Optimization (QUBO) and the power of Quantum Computing [2] to attempt solving the daily multi-task shift scheduling challenges. In this study, we aim to assess the efficiency and feasibility of using the QUBO model for daily multi-task shift scheduling in call centers. Our approach effectively converts constraints and objectives to a QUBO model, considering factors such as staff number, task type and skill proficiency. By transforming traditional mixed-integer models into QUBO models, we utilize Quantum Annealing (QA) to solve scheduling challenges. This approach facilitates a comprehensive exploration to find optimal or near-optimal results. Our contribution is the introduction of a novel computational model that underscores the unique potential and opportunities quantum computing presents for combinatorial optimization problems, laying the groundwork for future research in this area.

Section 2 presents a brief overview on shift scheduling and evaluates traditional optimization techniques. Section 3 details the study's methodologies, focusing on Quantum Annealing and QUBO's role in our model. Section 4 discusses the simulation experiment, outlining model formulation, model conversion procedures and experiment results. Finally, Sect. 5 discusses to the consideration, challenges and future implications based on our study.

2 Shift Scheduling in Call Centers

As the call center industry expanded, scheduling diverse tasks became increasingly challenging to execute efficiently. It involved matching employee skills with job needs, predicting call patterns, abiding by labor regulations and ensuring optimal operations [3].

Historically, Integer Programming and Mixed Integer Programming (MIP) were employed to tackle these complexities. While they seemed effective initially, they were less effective when faced with unpredictable call volumes and changing workforce availabilities. To address these challenges, heuristic methods, such as Genetic Algorithms and Simulated Annealing are emerged [4, 5]. These strategies provided solutions that, although approximate, were particularly effective in high-volume data scenarios or when rapid decisions were crucial. Furthermore, the shortcomings of traditional methods became clear as call centers expanded. They found it struggled with scaling to larger datasets and adjust to the constant changes in call center environments. The lack of consistent quality in solutions pointed to the need for more reliable strategies.

In this development dilemma, quantum computers were introduced. Drawing from the principles of quantum mechanics, quantum computers can leverage the unique properties of qubits. They have the potential to explore a multitude of solutions simultaneously. This capability makes them a promising candidate to redefine optimization in shift scheduling [6]. Applying quantum computing concepts to shift scheduling isn't just about technological integration, it's also about addressing the intricate challenges of human resource allocation, task management and efficiency in a way. As call centers continue to develop, quantum computer could be a critical factor to improve efficiency in operations and resource optimization.

3 Methods

Quantum Annealing (QA) is a sophisticated optimization technique underpinned by the principles of quantum mechanics [7]. Its unique capability to explore vast solution spaces addresses complex challenges, such as shift scheduling in call centers. By encoding the optimization problem into a quantum system, each energy state represents a potential solution. The optimal solution is represented by the ground state, which is the state with the lowest energy. Using a superposition of states and leveraging quantum tunneling, the system improves states and avoids common pitfalls like local minima. In the realm of shift scheduling, Quantum Annealing integrates numerous scheduling considerations, ensuring the workforce is allocated efficiently to meet customer requirements.

Quadratic Unconstrained Binary Optimization (QUBO) offers a mathematical framework tailor-made for Quantum Annealing [8]. By modeling combinatorial problems with polynomials of binary variables, the objective is to pinpoint variable assignments that minimize the polynomial's value. For this purpose, we employed the PyQUBO library. This tool facilitates the formulation of QUBO and Ising models and integrates seamlessly with various solvers, including quantum annealers. Shift scheduling complexities, such as employee availability and job proficiency requirements, can be translated into the constraints and objectives of a QUBO model. This translation lays the foundation for an effective implementation using Quantum Annealing.

4 Simulation Experiment

4.1 Model Formulation

To utilize the quantum annealing method effectively, we need to first set up the appropriate MIP model, and then transform it into a QUBO model, finally employ D-wave for problem-solving. This optimization strategy enabled us to depict the scheduling nuances precisely. The call center scheduling issue was approached using MIP firstly. Through defining suitable decision variables and creating an objective function, we captured the complex constraints and objectives of the scheduling endeavor. The resulting MIP model served as our primary mathematical representation, laying the groundwork for subsequent quantum computing applications. The detailed notations, the objective function and MIP model constraints are detailed further in references [9, 10]. Note that p_{ik} and b_{kl} are both integers.

Notations:

I number of staff, i = 1, 2, …, I
K number of tasks, k = 1, 2, …, K
L number of timeslots, l = 1, 2, …, L
p_{ik} staff i's proficiency in task k
b_{kl} amount of demand in timeslot l of task k
z_{kl} shortage of processing capability in timeslot l for every task k

Decision Variables:

$$y_{ikl} = \begin{cases} 1, & \text{when staff } i \text{ performs task } k \text{ in timeslot } l \\ 0, & \text{otherwise} \end{cases}$$

Objective function and constraints:

$$\text{Minimize } C = \sum_{ikl} y_{ikl} \tag{1}$$

s.t.

$$\sum_{i} p_{ik} y_{ikl} + z_{kl} \geq b_{kl}(\forall l, k) \tag{2}$$

$$\sum_{k=1}^{K} y_{ikl} = 1(\forall i, l) \tag{3}$$

$$y_{ikl} = \{0, 1\}(\forall i, k, l) \tag{4}$$

Equation (2) ensures that the total proficiency of all staff i assigned to task k in timeslot l meets the demand for that task in the given timeslot. Using the auxiliary variable z, we ensure the combined capacity of each staff exceeds task demand for all timeslots. This approach helps converting the MIP model to a QUBO format and tackling unexpected task challenges to ensure consistent service quality.

Equation (3) ensures that each staff member is assigned exactly one task during a specific timeslot l. No task is initiated before the commencement of a staff member's shift, nor does any task continue beyond the conclusion of their assigned shift.

4.2 Model Conversion

Recognizing the computational challenges and scalability concerns of classical solvers in handling large-scale instances of our problem, we get down to converting our MIP formulation into a QUBO model. This model conversion is not only a technical maneuver, but also preparing our model to use the excellent capabilities of quantum annealing. Through a series of introduction of penalties, representation of binary variable and other model conversion tricks, we ensured that the QUBO representation retained the implication of our initial MIP formulation, paving the way for using the power of quantum computing in our staff scheduling problem. Finally, the objective function is expressed as follows.

$$\text{Minimize } Q = \sum_{ikl} y_{ikl} + \left(\sum_{i} p_{ik} y_{ikl} + \sum_{m=0}^{b_{kl}} q_{klm} - b_{kl} \right)^2 + \left(\sum_{k=1}^{K} y_{ikl} - 1 \right)^2 \tag{5}$$

where q_{klm} are binary variables and

$$z = \sum_{m=0}^{b_{kl}} q_{klm} \tag{6}$$

4.3 Experiment Results

The call center scheduling problem was meticulously formulated using a QUBO model, incorporating the constraints and objectives typical of the staff scheduling problem. The experiments were conducted in a quantum computing environment, using the platform provided by D-Wave Systems Inc. This platform facilitates the execution of quantum algorithms, utilizing the unique computational capabilities of quantum hardware. Using this platform and its distinct computational capabilities, we tested the model's accuracy with a small-scale dataset to evaluate the potential of quantum annealing in addressing the call center scheduling problem.

Table 1 delineates the assignment of tasks among five staff members, each identified by a unique Staff ID (ranging from 1 to 5). The tasks, labelled as Task1, Task2 and Task3, are assigned across five distinct timeslots, sequentially numbered from 1 through 5. Here take staff 1 and staff 2 as examples to explain this table.

Staff ID 1: Begins his schedule with Task2 in the first timeslot and then going to consecutive assignments of Task3 from timeslot 2 through 5.

Staff ID 2: The sequence of assignments begins with Task1, followed by two timeslots of Task3, a single Task2 assignment and concluding with Task3 in the final timeslot.

Due to space constraints, we only show part of the experiment results. Although this is only the results of a small-scale data set, they are representative of the larger dataset and the overall outcomes of our extensive experimentation. The experiment results show the model's robustness, efficiency and practical applicability in addressing the intricate scheduling challenges faced by call centers, showcasing its potential in providing real-world solutions.

Table 1 Task assignments for staff across five timeslots

Staff ID	Timeslot				
	1	2	3	4	5
1	Task2	Task3	Task3	Task3	Task3
2	Task1	Task3	Task3	Task2	Task3
3	Task2	Task2	Task1	Task2	Task2
4	Task1	Task2	Task2	Task2	Task2
5	Task1	Task3	Task3	Task3	Task3

5 Conclusion

Quantum annealing offers a unique approach and unprecedented capabilities. It presents a bright future in solving complex combinatorial optimization problems, notably in call center scheduling. It's essential to identify and tackle present obstacles, ranging from quantum resource accessibility, related expenses, system disturbances and error percentages. These are fundamental considerations during current stage of quantum technology. Despite these challenges, we aim to broaden the use of quantum computing for optimization problems and overcome the implementation difficulties as much as possible.

This study emphasizes quantum computing's potential, especially in intricate scheduling challenges faced by call centers. Quantum computing tools like PyQUBO and D-Wave offer innovative solutions for combinatorial optimization problems. It aims to fully utilize quantum computing's potential across various optimization challenges. The implications of this study show the practical value and successful endeavors of quantum computing in real-world scenarios, particularly in improving call center efficiency.

References

1. Wang J, Sun Y, Wu X, Gao J (2022) The contingency and mediation relationships among internal capabilities, external partnerships, and service variety in manufacturing enterprises. Asian J Manag Sci Appl 7(2):147
2. Nielsen MA, Chuang IL (2010) Quantum computation and quantum information. Cambridge University Press
3. Zhang J, Yamamoto H, Sun J, Kajihara Y (2021) A study of optimal assignment with different workers' capacities for each process in a reset limited-cycle problem with multiple periods. Asian J Manag Sci Appl 6(2):163–188
4. Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220(4598):671–680
5. Ernst AT, Jiang H, Krishnamoorthy M, Sier D (2004) Staff scheduling and rostering: a review of applications, methods, and models. Eur J Oper Res 153(1):3–27
6. Hamada N, Saito K, Kawashima H (2023) Applying quantum annealing for shift scheduling problem for call centers. Int J Netw Comput 13(1):2–17
7. Glover F, Kochenberger G, Du Y (2018) A tutorial on formulating and using QUBO models. arXiv preprint arXiv:1811.11538
8. Kadowaki T, Nishimori H (1998) Quantum annealing in the transverse Ising model. Phys Rev E 58(5):5355
9. Liu X, Liu H, Song Y, Li C (2023) Improvement of task pattern generation in multi-task call centers. In: The 8th international conference on business and industrial research (ICBIR), pp 1269–1273
10. Liu H, Liu X, Li C, Song Y (2023) A two-stage model for multi-task call center shift scheduling. In: the 9th international conference on engineering, applied sciences, and technology (ICEAST), pp 75–78

A Job-Shop Scheduling Method Based on Ant Colony Optimization Considering Simultaneous Processing

Shuqing Cui, Yoshitaka Tanimizu, and Kotomichi Matsuno

Abstract In this paper, Ant Colony Optimization (ACO) is used to determine whether or not to perform simultaneous processing and the combination of jobs that make up a batch in a scheduling problem to achieve the objective of minimizing the make-span. After constructing the scheduling model and decision-making model for simultaneous processing, computer experiments in conjunction with a reactive scheduling system based on genetic algorithms (GA) that considered simultaneous processing and preparatory operations of jobs are conducted to verify the effectiveness of the proposed method.

Keywords Ant colony optimization · Simultaneous processing · Reactive scheduling system

1 Introduction

In processing factories that manufacture a wide variety of products, there are an extremely large number of job types, and production scheduling is becoming increasingly important. In each process, simultaneous processing is performed in which multiple jobs with the same processing specifications are collected and processed together to improve productivity and reduce costs by reducing setup time. Normally, there is a certain degree of freedom in the number of parts that make up a batch, and the size of the batch determined within that range affects not only the productivity of the equipment itself, but also the productivity, lead time, and cost at the factory level. Therefore, it is necessary to decide whether to perform simultaneous processing and the combination of parts that make up a batch.

Potts et al. [1] review the literature on scheduling with batching, giving details of the basic algorithms, and referencing other significant results. Fowler et al. [2] survey the literature on parallel batching and focus primarily on deterministic scheduling. This paper provides a taxonomy of parallel batching problems which make-span, flow

S. Cui (✉) · Y. Tanimizu · K. Matsuno
Waseda University, Tokyo 169-8555, Japan
e-mail: cuishuqing@akane.waseda.jp

C. Chien et al. (eds.), *Proceedings of Industrial Engineering and Management*, Lecture Notes in Mechanical Engineering, https://doi.org/10.1007/978-981-97-0194-0_2

time, and due date-related measures are considered. Zeng et al. [3] create a mixed-integer nonlinear programming model to describe the problem with a machine availability constraint, which is divided into two stages: batch processing and machine processing. This paper develops an auction-based approach in which jobs are categorized into batches during the batch processing stage and resources are allocated to operating machines during the machine processing stage. Zhou et al. [4] consider a single batch processing machine scheduling problem with unequal release times and job sizes. A self-adaptive differential evolution algorithm which mutation operators are adaptively chosen based on their historical performances is developed to address the problem. Jia et al. [5] study the problem of scheduling on parallel batch processing machines with different capacities under a fuzzy environment to minimize the make-span. A fuzzy ant colony optimization (FACO) algorithm is proposed, and based on the unoccupied space of the solution, heuristic information is designed for each candidate list to guide the ants. Gagne et al. [6] compare several heuristics for solving a single machine scheduling problem and the objective is to minimize total tardiness. They describe an ACO algorithm having a new feature using look-ahead information in the transition rule.

We focused on the issues for job shop scheduling problems including simultaneous processing and preparatory work. By adding a simultaneous processing process to the conventional scheduling process, we developed a reactive scheduling system that takes into account preparatory operations and simultaneous processing of multiple jobs. In this paper, the production schedule with minimum make-span will be solved by combining the decision method of simultaneous processing parts using ACO and the reactive scheduling system based on GA that considers preparatory operations.

2 Scheduling Model

2.1 Reactive Scheduling System Including Simultaneous Processing

In this study, the terms expressing production schedule are defined as follows.

An operation is the smallest unit of work assigned to a machine or worker. Operation $O_{i,j,k}^{l,m}$ is the l-th process of job J_i and the m-th process of machine M_j or worker L_j. k represents the simultaneous processing number, and operations with $k = 1$ perform simultaneous processing, operations with $k = 0$ represent not simultaneous processing. Operation $O_{i,j,k}^{l,m}$ contains information about starting time $st_{i,j,k}^{l,m}$, finishing time $ft_{i,j,k}^{l,m}$, and processing time $pt_{i,j,k}^{l,m}$.

Preparatory Operations. There are two main types of preparatory operations: internal preparatory operation, in which workers attach jigs to the machine and inspect them, and external preparatory operation, in which workers work on parts without stopping the machine. Furthermore, considering the distinction between before and after process operations, four preparatory operations are defined: external setup operation (es), which refers to work that is done before machining without stopping the machine, such as installing jigs, and work can begin as soon as the operator and job are arranged; internal setup operation (is), which refers to work such as product installation and positioning that is performed before processing when the machine is stopped; internal cleanup operation (ic), which refers to work performed after processing, such as removing a product, while the machine is stopped; external cleanup operation (ec), which refers to work that is performed after processing, such as cleaning the product, without stopping the machine, and work can begin as soon as the worker and job are arranged.

Mathematical Model. This paper assumes that simultaneous processing can be performed for processing operations performed by machines. Therefore, the preparation work should be considered as independent work rather than simultaneous processing. However, the internal setup operation, process operation, and internal cleanup operation must be performed consecutively on the same machine.

The mathematical model of the make-span (MS) is as follows.

$$Minimize\,MS \tag{1}$$

$$s.t.(es \cdot is \cdot ic \cdot ec)ft_{i,j,k}^{l,m} = (es \cdot is \cdot ic \cdot ec)st_{i,j,k}^{l,m} + (es \cdot is \cdot ic \cdot ec)pt_{i,j,k}^{l,m} \tag{2}$$

$$bisst_{i,j,k}^{l,m} = \min\{isst_{i,j,k}^{l,m}, isst_{i',j',k'}^{l',m'}, isst_{i'',j'',k''}^{l'',m''}, \ldots\} \tag{3}$$

$$bst_{i,j,k}^{l,m} = bisft_{i,j,k}^{l,m} = bisst_{i,j,k}^{l,m} + ispt_{i,j,k}^{l,m} + ispt_{i',j',k'}^{l',m'} + ispt_{i'',j'',k''}^{l'',m''} + \cdots \tag{4}$$

$$bicst_{i,j,k}^{l,m} = bft_{i,j,k}^{l,m} = bst_{i,j,k}^{l,m} + bpt_{i,j,k}^{l,m} \tag{5}$$

$$bicft_{i,j,k}^{l,m} = bicst_{i,j,k}^{l,m} + ispt_{i,j,k}^{l,m} + ispt_{i',j',k'}^{l',m'} + ispt_{i'',j'',k''}^{l'',m''} + \cdots \tag{6}$$

The starting time, finishing time, and processing time of each operation when they are not subject to simultaneous processing are defined with the relational expressions as Eq. (2). Considering that the internal setup operation, process operation, and internal cleanup operation are consecutive, the starting time of the process operation is the same as the finishing time of the internal setup operation, and the starting time of the internal cleanup operation is the same as the finishing time of the process operation. In addition, the internal setup operation for machining operations that are to be processed simultaneously requires that the external setup operation for that operation is completed and that the same machine is available for work.

When machining operations are to be processed simultaneously, the process operations are the only operations that are processed simultaneously. Considering that the internal setup operation, process operation, and internal cleanup operation are consecutive, the starting time of internal setup operations is the earliest starting time of them, and it is expressed as Eq. (3). Then the $bisft^{l,m}_{i,j,k}$, $bst^{l,m}_{i,j,k}$,$bft^{l,m}_{i,j,k}$ $bicst^{l,m}_{i,j,k}$, and $bicft^{l,m}_{i,j,k}$ are expressed from Eq. (4–6).

2.2 Ant Colony Optimization

ACO is an algorithm proposed by Marco Dorigo [7] in 1992 based on the foraging behavior of ants. In ACO, the artificial ants are set to search the solution space and in each iteration, the ants generate paths and leave pheromone on the paths. The pheromone on a path transmits messages of the path to other ants when they intend to follow it. Ants tend to choose paths with a high concentration of pheromones. Pheromones evaporate over time, whereas if the path is short, the pheromone is reinforced by traveling further. The closer the pheromone is to the shortest path, the faster the pheromone is reinforced than it evaporates, forming a highly concentrated path.

There are two keys to using the ACO to solve problems, one is determining the encoding and pheromone matrix of the problem solution, and the other is how to randomly generate the encoding of the solution given the information. Assuming there are n jobs in total, multiple jobs can be processed simultaneously as a batch, and each batch can process up to at most B jobs at the same time. The situation with the largest number of batches is that each batch has only one job, so there will be at most n batches. Therefore, use an n-dimensional integer vector as the encoding of the solution, each component of which is an integer from 1 to n, representing the batch number of the corresponding job.

Coding. It is assumed that the code of a certain solution is $(z_1, z_2, z_3, \ldots, z_i, \ldots, z_n)$, where z_i represents the batch number of the job J_i. According to the setting of the Burn-in type [8] batch sorting problem, the processing time of the φ-th batch is the maximum processing time of all jobs in the batch, and is expressed by Eq. (7).

$$b_\varphi = bpt^{l,m}_{i,j,k,\varphi} = \max\left\{pt^{l,m}_{i,j,k}|z_i = \varphi, i = 1, 2, \ldots, n\right\}, \varphi = 1, 2, \ldots, n \quad (7)$$

The Method for Searching Feasible Solutions. Information plays a very important role in the ACO. Ants choose routes based on the pheromone and heuristic information on each path. If a code is regarded as a path, then choosing the route is to decide whether the φ-th batch contains the job J_i. $\tau_{i\varphi}$ represents the pheromone that the job J_i contains in the φ-th batch. $\eta_{i\varphi}$ represents the heuristic information that tends to put jobs with similar processing times into the same batch, and is expressed by Eq. (8). t_i is the processing time of job J_i, and ω_φ is the average value of the processing time of jobs in the φ-th batch.

$$\eta_{i\varphi} = \frac{1}{1 + \left|\omega_\varphi - t_i\right|} \quad (8)$$

Initially, the number of jobs allowed to be included in each batch is B. If a job is placed into a certain batch, the number of jobs allowed to be included in the batch will be reduced by 1. If this number becomes 0, the batch does not appear in the set of batches that are allowed to be selected, otherwise it is in it.

For ant a at the t-th iteration, let $allowed_a$ represent the set of batches that are allowed to be selected. The ants select the next batch according to pheromone and heuristic information, whose weights are respectively α and β. Then the probability $p_{i\varphi}^{a}(t)$ of putting the job J_i into the φ-th batch is expressed by Eq. (9).

$$p_{i\varphi}^{a}(t) = \begin{cases} \frac{\tau_{i\varphi}(t)^{\alpha}\eta_{i\varphi}(t)^{\beta}}{\sum_{S\in allowed_a}\tau_{iS}(t)^{\alpha}\eta_{iS}(t)^{\beta}}, & \varphi \in allowed_a \\ 0, & \varphi \notin allowed_a \end{cases} \tag{9}$$

After calculating the probability of job J_i, a batch is randomly selected in $allowed_a$ according to the roulette method. Repeating the above operations in sequence to generate a feasible solution.

Selection and evolution of paths. The corresponding solution of each ant will be brought into the reactive scheduling system and obtain its MS^a. Then π_a will be set as the solution according to the ant a and π_* as the solution with the best MS. π_* is updated by π_a if $MS^a < MS^*$.

The pheromone is updated according to the global update rule:

$$\tau_{i\varphi}(t+1) = \rho \cdot \tau_{i\varphi}(t) + \sum_{a=1}^{N} \triangle\ \tau_{i\varphi}^{a}(t) \tag{10}$$

$$\triangle\ \tau_{i\varphi}^{a}(t) = \begin{cases} \frac{Q}{MS^{a}(t)}, & z_i = \varphi \\ 0, & otherwise \end{cases} \tag{11}$$

In Eq. (10), the total number of ants is N, and ρ is the evaporating parameter that controls the loss of pheromone on the paths. In Eq. (11), Q is a constant and its appropriate value needs to be found by experiment.

It will be iterated to a preset number C of iterations. At the end of all iterations, the MS^* and its corresponding solution π_* will become the approximately optimal solution.

3 Numerical Experiments

In this study, a reactive scheduling system combined with ACO was developed using the object-oriented programming language Smalltalk, and carried out experiments on a computer with Intel(R) Core(TM) i5-12400F @2.50 GHz and 16.0 GB RAM.

The experimental scale of this experiment is 10 jobs processed on 3 machines and 2 workers. The second machine and its corresponding process can be processed

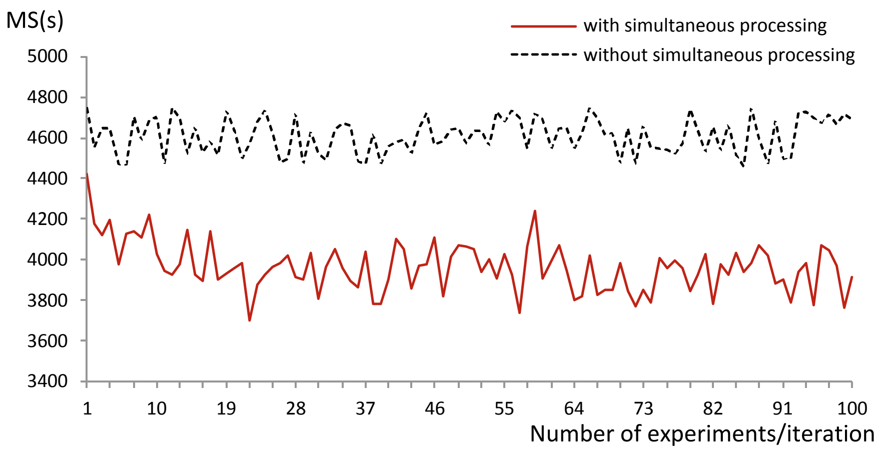

Fig. 1 MS value comparison between with/without simultaneous processing

simultaneously, and the processing time of the batch conforms to the Burn-in type. The parameters of ACO were set as follows: $C = 100, N = 40, \alpha = 10, \beta = 10, \rho = 0.8, Q = 1$. Then the same production data is used to run the reactive scheduling system 100 times without simultaneous processing, and the results are used as control data.

Judging from the results, the MS obtained using simultaneous processing is completely better than that without simultaneous processing. The optimal MS with simultaneous processing is 3700S, but without simultaneous processing, it is 4465S, and the optimization rate is 17.13%. Moreover, the combination of simultaneously processed parts selected by the ACO makes the MS obtained by the reactive scheduling system show a convergence trend (see Fig. 1).

4 Conclusion

In this paper, ACO is used to determine whether to perform simultaneous processing and the combination of jobs that form a batch in a scheduling problem to achieve the purpose of minimizing the make-span. Then the comparative computer experiments in conjunction are conducted with a reactive scheduling system. The experiment demonstrates the proposal theory's efficacy.

There are still many shortcomings in our research, including that the experimental scale is not large enough, the type of the batch processing time is simple, etc. Therefore, in the future, it is needed to expand the scale of experiments to better test the validity of the theory. Batch processing time types for Sum type [9] and Constant type will also be added in addition to Burn-in type, and the corresponding heuristic information will be changed to be able to cope with more production situations.

References

1. Potts CN, Kovalyov MY (2000) Scheduling with batching: a review. Eur J Oper Res 120(2):228–249
2. Fowler JW, Monch L (2022) A survey of scheduling with parallel batch (p-batch) processing. Eur J Oper Res 298(1):1–24
3. Zeng C, Liu Z, Tang J, Fan Z, Yan C (2023) Auction-based approach to the job-shop problem with parallel batch processing and a machine availability constraint. Eng Optim 55(1):71–88
4. Zhou S, Xing L, Zheng X, Du N, Wang L, Zhang Q (2021) A self-adaptive differential evolution algorithm for scheduling a single batch-processing machine with arbitrary job sizes and release times. IEEE Trans Cybern 51(3):1430–1442
5. Jia Z, Yan J, Leung JYT, Li K, Chen H (2019) Ant colony optimization algorithm for scheduling jobs with fuzzy processing time on parallel batch machines with different capacities. Appl Soft Comput 75(1):548–561
6. Gange C, Price WL, Gravel M (2002) Comparing an ACO algorithm with other heuristics for the single machine scheduling problem with sequence-dependent setup times. J Oper Res Soc 53(8):895–906
7. Dorigo M (1992) Optimization, learning and natural algorithms. Ph.D. Thesis, Politecnico di Milano, Italy
8. Brucker P (1998) Scheduling a batching machine. J Sched 1(1):31–54
9. Albers S, Brucker P (1993) The complexity of one-machine batching problems. Discret Appl Math 47(2):87–107