

Joint optimization of flexible job shop scheduling and preventive maintenance under high-frequency production switching

Yu Wang, Tangbin Xia*, Yuhui Xu, Yutong Ding, Meimei Zheng, Ershun Pan, Lifeng Xi

State Key Laboratory of Mechanical System and Vibration, School of Mechanical Engineering, Shanghai Jiao Tong University, SJTU-Fraunhofer Center, Shanghai, 200240, China



ARTICLE INFO

Keywords:

Flexible job shop scheduling
Preventive maintenance
High-frequency production switching
Variable operating conditions
Hybrid meta-heuristics algorithm

ABSTRACT

To secure a larger market share in the dynamically evolving personalized market demand, enterprises have to adopt a more flexible production mode. In this context, the implementation of parallel production with multiple product types necessitates frequent switching between different products, imposing elevated requirements on machine fault-free operation and stable production. There is a compelling motivation to investigate the joint optimization of the flexible job shop scheduling problem and preventive maintenance (FJSSP-PM). However, existing research has primarily focused on utilizing predetermined maintenance as inputs for optimizing production scheduling, while overlooking the uncertain maintenance requirements of machines during the production process. Therefore, this paper proposes a two-stage joint optimization model to simultaneously address three subproblems: machine assignment, operation sequencing, and maintenance arrangement. Specifically, the first stage involves constructing a mathematical model for the FJSSP to minimize penalties for tardiness and workload balancing. Subsequently, a comprehensive failure rate model is developed to determine maintenance requirements for machines with production-dependent failure behavior, particularly under high-frequency production switching scenarios. Furthermore, a feedback-updated strategy is proposed to achieve synchronous optimization of production and maintenance activities in an economically and operationally efficient manner. Given the complexity of the problem, a hybrid meta-heuristic algorithm is designed to effectively handle large-scale cases. The effectiveness of the proposed model in terms of cost-savings has been verified compared with existing relevant methods.

1. Introduction

In response to the increasing demand for personalization and customization, industries such as automobile, aerospace, and semiconductor are adopting flexible production systems to enhance competitiveness (Ding and Gu, 2020). The complexity of multi-variety, multiple-type parallel production often results in frequent switching between products, imposing higher requirements on machine fault-free operation and stable production. Therefore, preventive maintenance (PM) should be given significant attention in addressing the flexible job shop scheduling problem (FJSSP) as an effective approach for maintaining machine performance while enhancing overall production efficiency.

Extensive research in recent decades has primarily focused on the simple joint optimization of FJSSP and PM (FJSSP-PM), where a pre-determined maintenance plan is utilized as an input/constraint for

optimizing the production scheme. However, high-frequency production switching inevitably induces alterations in process and production quality, subjecting machines to various operating conditions. The failure time of a machine typically varies under different operating conditions. It is crucial to acknowledge the significant impact of high-frequency production switching on maintenance requirements and production continuity in the joint optimization of FJSSP-PM.

Moreover, the job shop scheduling problem (JSSP) is known to be strongly NP-hard (Garey et al., 1976). The joint optimization of FJSSP-PM, as its extension, is also strongly NP-hard. To discover high-quality solutions, meta-heuristics algorithms progressively explore promising options using well-designed heuristic rules, such as Genetic Algorithm (GA) (Li and Gao, 2016), Discrete Particle Swarm Optimization (DPSO) (Hou et al., 2022), Variable Neighborhood Search (VNS) (Liu, 2019), Simulate Annealing Algorithm (Defersha et al., 2022), Ant Colony Optimization (Rossi, 2014). However, the inherent limitations in

* Corresponding author.

E-mail addresses: 22wangyu@sjtu.edu.cn (Y. Wang), xtbxtb@sjtu.edu.cn (T. Xia).

the fundamental theory and algorithm framework restrict the potential improvement of a single algorithm. Consequently, blending algorithms has become a favored approach for enhancing performance.

Motivated by the aforementioned context, this paper focuses on jointly optimizing production and maintenance in a flexible job shop environment, especially considering the impact of high-frequency production switching on machine reliability and the interdependence between maintenance activities and production scheduling. The primary objective is to reduce downtime losses from machine failures, alleviate dissatisfaction with delayed deliveries, and optimize system-wide machine utilization. Specifically, an initial job scheduling scheme is generated by considering delivery time and workload balancing. Subsequently, we investigate the comprehensive effects of cumulative service time, operating conditions, and maintenance on machine failure. Regarding maintenance, the difference between considering operating conditions or not is that the former mode leads to significantly faster machine deterioration. Importantly, both job scheme and maintenance plan will be dynamically optimized simultaneously. The main contributions are as follows.

- (1) The paper proposes a two-stage joint optimization model for production and maintenance in a flexible job shop environment to effectively address dynamic market demands, bridging the existing gap. This model helps enterprises develop cost-effective plans for production and maintenance simultaneously, ensuring seamless machine operation and uninterrupted production continuity.
- (2) A comprehensive model is developed to analyze the failure rates of a machine under variable operating conditions, focusing on the multi-stage failure process under high-frequency production switching. Maintenance requirements exhibit dynamic uncertainty along with the production process, which can be more accurately determined by considering both cost and availability.
- (3) A feedback-updated strategy is proposed to address the interdependence between stable production and maintenance requirements, specifically machine reassignment and task rescheduling. Meanwhile, a dynamic equilibrium between job scheduling and PM activities is achieved by minimizing tardiness penalty, workload balancing penalty, and maintenance cost.
- (4) The hybrid DPSO-VNS algorithm is designed to efficiently optimize the production and maintenance scheme by integrating global search and local enhancement techniques, aiming for an optimal solution. Multiple initial feasible solutions are simultaneously improved using the DPSO algorithm, generating high-quality initial populations that are further optimized through the VNS algorithm's neighborhood disturbance mechanism.

The paper is structured into six sections. Following this introduction, the second section provides a comprehensive literature review. **Section 3** presents an in-depth analysis of the identified problem and introduces the proposed two-stage joint optimization model. **Section 4** elaborates on the developed hybrid DPSO-VNS algorithm. The effectiveness and performance of the proposed model are demonstrated in **Section 5**. Finally, **Section 6** concludes the paper and discusses further research opportunities.

2. Literature review

The issue of production and maintenance in a flexible job shop environment requires addressing three subproblems: machine assignment, operation sequencing, and maintenance arrangement. This study focuses on accurately predicting maintenance requirements based on enterprise objectives and machine performance to ensure seamless machine operation and uninterrupted production, specifically considering the characteristics of high-frequency switching in the actual production process. Relevant literature will be reviewed in this section to

emphasize the contributions of this research.

The JSSP is a classical combinatorial optimization problem where jobs are processed on machines in specific sequences without pre-emption (Tamssaouet and Dauzère-Pérès, 2023). Numerous studies have been published since Johnson (1954) first studied the JSSP, including detailed analysis by Wein and Chevalier (1992), Zhang et al. (2019), and Xiong et al. (2022).

With the continuous advancements and growing demand for personalized manufacturing, the integration of flexibility into JSSP has given rise to a novel scheduling problem known as FJSSP (Li and Gao, 2016; Yang et al., 2022). FJSSP has garnered considerable attention due to its proximity to modern production systems. Shen et al. (2018) investigated sequence-dependent setup times intending to minimize makespan. Božek and Werner (2018) proposed a two-stage optimization procedure that incorporates lot streaming and variable subplot lot sizing. Park and Ham (2022) considered time-of-use pricing and scheduled downtime to minimize both makespan and total energy cost. Recently, Dauzère-Pérès et al. (2023) have provided comprehensive insights into FJSSP, encompassing criteria, constraints, additional problem characteristics, and solution approaches.

The aforementioned studies assume the constant availability of machines. However, in practical production scenarios, machines inevitably encounter periods of unavailability due to unforeseen failures and maintenance activities. In recent years, researchers have integrated the FJSSP and PM into a comprehensive model. Zribi et al. (2008) incorporated predetermined PM execution times as constraints in a scheduling model, representing machine unavailable periods. Wang and Yu (2010) addressed the FJSSP with availability constraints by considering PM activities for each machine during the planning period. The starting times of these activities were either flexible within a time window or fixed in advance. Moradi et al. (2011) utilized fixed time interval PM activities in FJSSP to minimize both the makespan and system unavailability. Mokhtari and Dadgar (2015) sought the optimal maintenance interval that ensures minimal system availability, aiming to minimize the total number of tardy jobs under varying machine failure rates. Zandieh et al. (2017) determined the maintenance execution time by the machine's degraded state and a preset threshold value, which is used as a constraint to optimize production scheduling. An et al. (2021) considered both maintenance activity and repairman competence under maintenance time window and employee timetable constraints in FJSSP. Later, An et al. (2023) established a multi-objective optimization model to minimize makespan, instability, and cumulative production cost. In this study, the authors assumed that machine failure follows a Gamma process and sought to identify optimal maintenance plans for each machine to minimize maintenance costs.

From the above analysis, it is clear that current research merely integrates FJSSP and PM; fundamentally, they remain distinct entities. These studies assume predetermined maintenance activities as constraints/inputs to optimize production scheduling. However, they overlook uncertain maintenance requirements arising from varying machine deterioration during the production process. Implementing proactive maintenance before unacceptable performance levels can reduce unnecessary activities and costs, aligning more effectively with practical industrial demands.

3. Problem description and formulation

3.1. Problem description

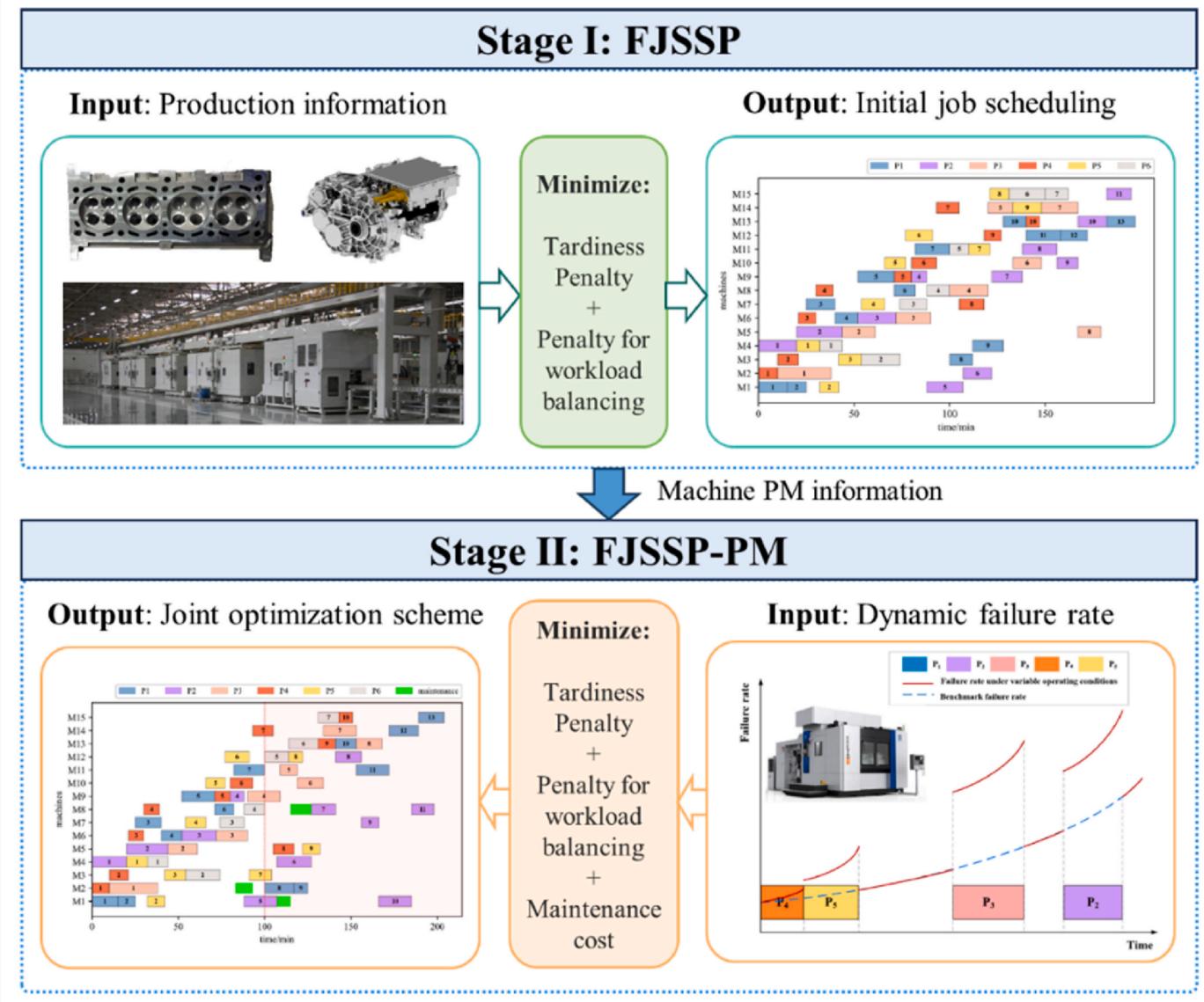
This study proposes a two-stage joint optimization model for production and maintenance in a flexible job shop environment with K machines processing multiple jobs, each with a unique process flow representing specific operating conditions. Machines frequently switch between different jobs due to the simultaneous production of multiple types, enabling them to operate under variable conditions. Each machine experiences independent degradation, with failure rates

increasing over time and inevitable failures occurring. A well-designed maintenance plan is crucial to ensure seamless machine operation and uninterrupted production continuity. Therefore, the proposed model comprehensively considers customer satisfaction, system utilization, and machine reliability to develop cost-effective production and maintenance plans for enterprises.

Specifically, there are I independent jobs $P = \{P_1, P_2, \dots, P_I\}$ and K machines $M = \{M_1, M_2, \dots, M_K\}$. Each job P_i comprises a collection of operations $O_i = \{O_{i1}, O_{i2}, \dots, O_{in_i}\}$ that are subject to precedence constraints. O_{ij} can be assigned to any machine M_k from its candidate machine set M_{ij} . The processing time PT_{ijk} for O_{ij} varies depending on the selected machine M_k . Moreover, the failure rates of a machine are determined by cumulative service time, operating conditions, and maintenance effects. Multiple influencing factors (e.g., maintenance cost, machine availability, production process) need to be considered when making maintenance decisions. The computational complexity of the problem studied surpasses that of FJSSP and increases with the problem scale. Some assumptions are summarized as follows.

- (1) All jobs and machines are ready and available at the beginning of the planning horizon. Jobs are mutually independent and there are no priorities among them.
- (2) Each operation can only be processed by one machine, while each machine can process only one operation at a time.
- (3) After each PM, the machine will be restored to a condition “as good as new”. In the event of a failure, MR is conducted to restore the failed machine to its operational state without altering its failure rate. The period of unavailability for a PM/MR action is associated with each machine and known in advance.

The framework of the two-stage joint optimization model for FJSSP-PM is depicted in Fig. 1. Firstly, an initial job scheduling scheme is generated using the hybrid DPSO-VNS algorithm to minimize penalties related to tardiness and workload balancing. Subsequently, a failure rate model of the machine under variable operating conditions is constructed, and maintenance requirements for each machine are analyzed considering optimization objectives such as maintenance cost rate and machine availability. Finally, taking into account costs associated with tardiness, workload balancing, and maintenance activities, the machine assignment and operation sequencing are revised based on a feedback-



updated strategy to ensure robust integration of maintenance arrangement.

3.2. Mathematical model of FJSSP at stage I

The parameters utilized in this section are initially introduced.

Parameters

i	Index of jobs, $i = 1, 2, \dots, I$
j	Index of operations of jobs $i, j = 1, 2, \dots, n_i$
k	Index of machines, $k = 1, 2, \dots, K$
z	Index of positions on machine $k, z = 1, 2, \dots, z_k$
P	Set of jobs
O_i	Set of operations of job i
M	Set of machines
Z	Set of the number of processing positions for all machines
D_i	Due date of job i
C_i^d	Tardiness penalty per unit time of job i
C^b	Penalty of workload balancing among machines
TTC_i	Tardiness penalty of job i
TBC	Total penalty for workload balancing
Decision variables	
x_{ijkz}	Binary variable: 1, O_{ij} is processed on the z th position of machine k ; 0, otherwise.
t_{ij}^s	Start time of O_{ij}
t_{ij}^e	Completion time of O_{ij}

The optimization objective in stage I includes two cost components: (1) incorporating a penalty for tardy delivery to assess responsiveness to time-varying market demands; and (2) introducing a penalty for workload balancing among machines to prevent excessive utilization of any single machine. Consequently, a mathematical model is formulated as presented below.

$$\min \sum_{i \in P} TTC_i + TBC \quad (1)$$

s. t.

$$TTC_i = \begin{cases} C_i^d (t_{in_i}^e - D_i), & t_{in_i}^e > D_i \\ 0, & t_{in_i}^e \leq D_i \end{cases}; \forall i \in P \quad (2)$$

$$TBC = C^b \sqrt{\sum_{k \in M} \left(ZT_k \frac{\sum_{k \in M} ZT_k}{K} \right)^2} / K \quad (3)$$

$$ZT_k = \sum_{z \in Z} \sum_{i \in P} \sum_{j \in O_i} x_{ijkz} PT_{ijk}; \forall k \in M \quad (4)$$

$$\sum_{k \in M} \sum_{z \in Z} x_{ijkz} = 1; \forall i \in P, j \in O_i \quad (5)$$

$$t_{i(j+1)}^s \geq t_{ij}^e; \forall i \in P, j + 1 \in O_i \quad (6)$$

$$t_{ij}^e = t_{ij}^s + \sum_{k \in M} \sum_{z \in Z} x_{ijkz} PT_{ijk}; \forall i \in P, j \in O_i \quad (7)$$

$$\sum_{i \in P} \sum_{j \in O_i} x_{ijkz} \leq 1; \forall k \in M, z \in Z \quad (8)$$

$$\sum_{i \in P} \sum_{j \in O_i} x_{ijkz} \geq \sum_{i \in P} \sum_{j \in O_i} x_{ijk(z+1)}; \forall k \in M, z + 1 \in Z \quad (9)$$

$$\sum_{i \in P} \sum_{j \in O_i} x_{ijk(z+1)} t_{ij}^s \geq \sum_{i \in P} \sum_{j \in O_i} x_{ijkz} t_{ij}^e; \forall k \in M, z + 1 \in Z \quad (10)$$

$$x_{ijkz} \in \{0, 1\}; \forall i \in P, j \in O_i, k \in M, z \in Z \quad (11)$$

$$t_{ij}^s \geq 0; \forall i \in P, j \in O_i \quad (12)$$

$$t_{ij}^e \geq 0; \forall i \in P, j \in O_i \quad (13)$$

The objective criterion in stage I is represented by Eq. (1) and is subject to the constraints of Eqs. (2)–(13). Constraint (2) assigns a penalty for tardiness to each job. Constraints (3) and (4) quantify the penalty for workload balancing among machines and the total workload of M_k , respectively. Constraint (5) guarantees that each operation can only be processed on one machine. Constraint (6) ensures that the start of an operation cannot be earlier than the completion of a previous operation in the same job. Furthermore, constraint (7) states that once an operation has started, it cannot be interrupted before completion. Constraint (8) indicates that each position on machine M_k can accommodate at most one operation. Constraint (9) forces sequential assignment of positions on each machine to operations. Constraint (10) ensures non-overlapping operations assigned to the same machine. Constraints (11)–(13) define decision variables domains.

To enhance comprehension, we provide a numerical example in Table 1 with five jobs and four machines, where “-” denotes the unavailability of corresponding operations on specific machines. A feasible solution is illustrated in Fig. 2.

3.3. Mathematical model of FJSSP-PM at stage II

This section presents a detailed analysis of maintenance requirements for each machine under high-frequency production switching, along with a feedback-updated joint optimization strategy for production and maintenance. Firstly, the subsequent parameters are introduced.

Parameters

r	Index of maintenance cycle for machine $k, r = 1, 2, \dots, r_k$
$\lambda_{rk}(t X)$	Failure rate function of machine k in the r th maintenance cycle
t_k^p	Time duration of a PM activity for machine k
C_k^p	PM cost for machine k
T_k^f	Time duration of an MR for machine k
C_k^f	MR cost for machine k
u_{kz}	Realized number of failures happens before the z th position on machine k
W	A big constant
T_{rk}^*	Optimal PM interval of machine k in the r th maintenance cycle
T_{rk}^s	Actual PM interval of machine k in the r th maintenance cycle
t_{rk}^{*s}	Optimal PM time point of machine k in the r th maintenance cycle
t_{rk}^{*e}	Optimal PM end time point of machine k in the r th maintenance cycle
t_{rk}^e	Actual PM end time point of machine k in the r th maintenance cycle
B	Set of the number of maintenance cycles for all machines
TMC	Total maintenance cost
Decision variables	
y_{rkz}	Binary variable: 1, a PM activity is performed after the z th position on machine k in the r th maintenance cycle; 0, otherwise.
t_{rk}^s	Actual PM time point of machine k in the r th maintenance cycle

Table 1
Processing times of a numerical example.

Job	operation	Processing time			
		M_1	M_2	M_3	M_4
P_1	O_{11}	–	11	–	14
	O_{12}	13	–	10	12
	O_{13}	–	16	–	13
P_2	O_{21}	–	–	6	8
	O_{22}	–	11	12	–
	O_{23}	10	–	–	13
P_3	O_{31}	9	–	–	7
	O_{32}	11	12	–	–
P_4	O_{41}	8	10	–	–
	O_{42}	6	–	5	7
	O_{43}	–	16	–	14
P_5	O_{51}	9	–	8	–
	O_{52}	–	9	10	10

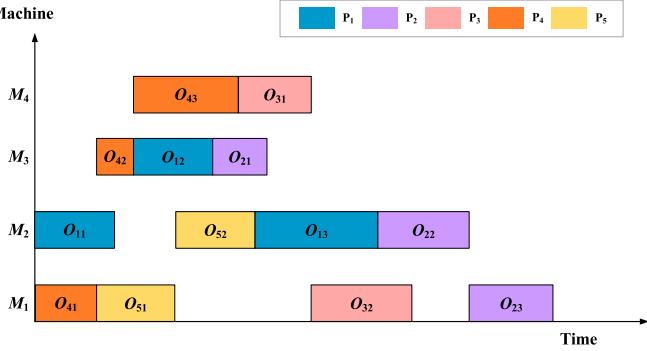


Fig. 2. Gantt chart of a feasible scheme.

3.3.1. Maintenance requirements under high-frequency production switching

The high-frequency production switching induces variations in operating conditions, including applied load, cutting speed, feed speed, and other factors. While the machine reliability remains continuous when operating conditions change, the failure rate experiences sudden jumps (Hu et al., 2017; Wang et al., 2019). Taking machine M_1 in Fig. 2 as an example, Fig. 3 illustrates how changes in operating conditions can cause the failure rate to deviate from the benchmark, increasing the probability of machine failure (Li and Gao, 2016; Kundu et al., 2019). Deteriorating operating conditions also lead to more frequent and costly maintenance activities. The proportional hazard model (PHM) and the accelerated failure-time model are commonly used for modeling failure rates, with PHM being particularly effective due to its versatility as demonstrated by Lu et al. (2012), Zhou et al. (2014), and Brenière et al. (2020).

The machine failure rate under high-frequency production switching is influenced by multiple factors, including operation sequencing, processing time for each operation, job materials, and specific customer requirements. Therefore, a comprehensive model for machine failure rate is developed,

$$\lambda_{rk}(t|\mathbf{X}) = \alpha\lambda_k(t)f(\beta\mathbf{X}) \quad (14)$$

where, $\mathbf{X} = [X_1, X_2, \dots, X_f, \dots, X_F]$ represents the factors affecting machine failure rates. The coefficients $\beta = [\beta_1, \beta_2, \dots, \beta_F]$ corresponding to these factors in \mathbf{X} are estimated using the maximum likelihood estimation method (Hu et al., 2021).

The Weibull distribution is commonly considered a suitable choice for parameterizing the benchmark failure rate.

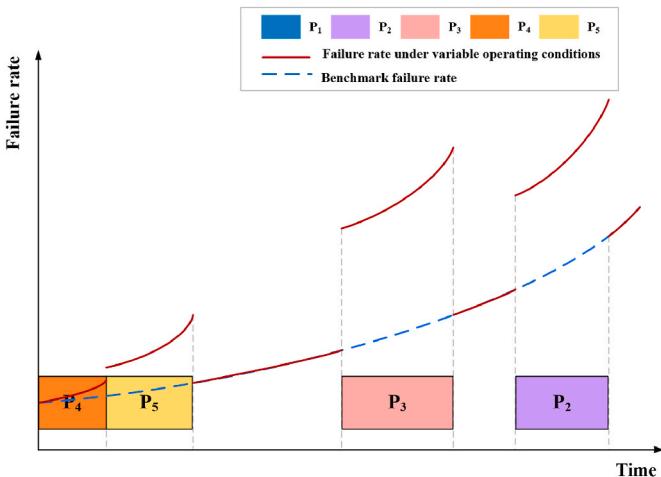


Fig. 3. Evolution of failure rate under variable operating conditions.

$$\lambda_k(t) = \frac{\theta_k}{\eta_k} \left(\frac{t}{\eta_k} \right)^{\theta_k - 1} \quad (15)$$

where, θ_k and η_k represent the shape parameter and the scale parameter of M_k , respectively.

Subsequently, a global maintenance decision goal V_{rk} (Xia et al., 2017) is employed to determine the optimal PM interval for machine M_k during each maintenance cycle by comprehensively considering the maintenance cost rate V_{rk}^C and the availability V_{rk}^A .

$$V_{rk} = \omega_1 \frac{V_{rk}^C}{V_{rk}^{C*}} + \omega_2 \frac{V_{rk}^A}{V_{rk}^{A*}} \quad (16)$$

where, ω_1 and ω_2 ($\omega_1 \geq 0, \omega_2 \geq 0, \omega_1 + \omega_2 = 1$) denote the weight of maintenance cost rate and availability, respectively. ω_1 and ω_2 are set to 0.5 in this study. V_{rk}^C and V_{rk}^A of M_k are defined as follows.

$$V_{rk}^C = \frac{C_k^p + C_k^f \int_0^{T_{rk}} \lambda_{rk}(t|\mathbf{X}) dt}{T_{rk} + T_k^p + T_k^f \int_0^{T_{rk}} \lambda_{rk}(t|\mathbf{X}) dt} \quad (17)$$

$$V_{rk}^A = \frac{T_{rk}}{T_{rk} + T_k^p + T_k^f \int_0^{T_{rk}} \lambda_{rk}(t|\mathbf{X}) dt} \quad (18)$$

The minimum maintenance cost rate V_{rk}^{C*} and maximum availability V_{rk}^{A*} are determined by solving the equations $d V_{rk}^C / d T_{rk} = 0$ and $d V_{rk}^A / d T_{rk} = 0$, respectively. Then, the objective is to minimize V_{rk} to obtain the optimal PM interval T_{rk}^* . Additionally, the optimal time point $t_{(r-1)k}^*$ is determined by T_{rk}^* and the completion time $t_{(r-1)k}^*$ of the r -th maintenance cycle.

3.3.2. Feedback-updated joint optimization strategy

Simultaneous arrangement of production and maintenance activities may deviation from the initial job scheduling scheme generated in Section 3.2 and the optimal PM intervals for machines derived in Section 3.3.1. Regarding maintenance, there exist five potential scenarios where the optimal time range $[t_{rk}^*, t_{rk}^{**}]$ for performing PM activity on machine M_k falls within its operation sequencing (as exemplified by machine M_1 in Fig. 2), as depicted in Fig. 4(a). In Case (i), the PM activity can be directly scheduled during idle periods of production, i.e., $t_{rk}^* = t_{rk}^{**}$. However, cases (ii)–(v) would interrupt operation O_{32} and thus necessitate further analysis employing the advance-postpone balancing method (Xia et al., 2015) to determine the actual time for conducting PM activity.

- (1) The PM activity of M_k is performed before the start time of O_{32} , with t^a as the PM time point, CM_{rk}^a representing the cost-saving effect of MR, and CP_{rk}^a indicating the cost-increasing impact of PM (defined in Eqs. (19)–(21)).

$$t^a = \begin{cases} t_3^s - T_k^p, & t_3^s - t_2^e > T_k^p \\ t_2^e, & \text{otherwise} \end{cases} \quad (19)$$

$$CM_{rk}^a = C_k^f \cdot \left(\int_0^{T_{rk}^*} \lambda_{rk}(t|\mathbf{X}) dt - \int_0^{T_{rk}^*} \lambda_{rk}(t|\mathbf{X}) dt \right) = C_k^f \cdot \int_{t^a}^{T_{rk}^*} \lambda_{rk}(t|\mathbf{X}) dt \quad (20)$$

$$CP_{rk}^a = \frac{t_{rk}^* - t^a}{T_{rk}^* - (t_{rk}^* - t^a)} C_k^p \quad (21)$$

- (2) The PM activity is performed after the completion time of O_{32} , where the PM time point t^d , the cost-increasing impact of PM CM_{rk}^d , and the cost-saving effect of PM CP_{rk}^d are calculated using Eqs. (22)–(24), respectively.

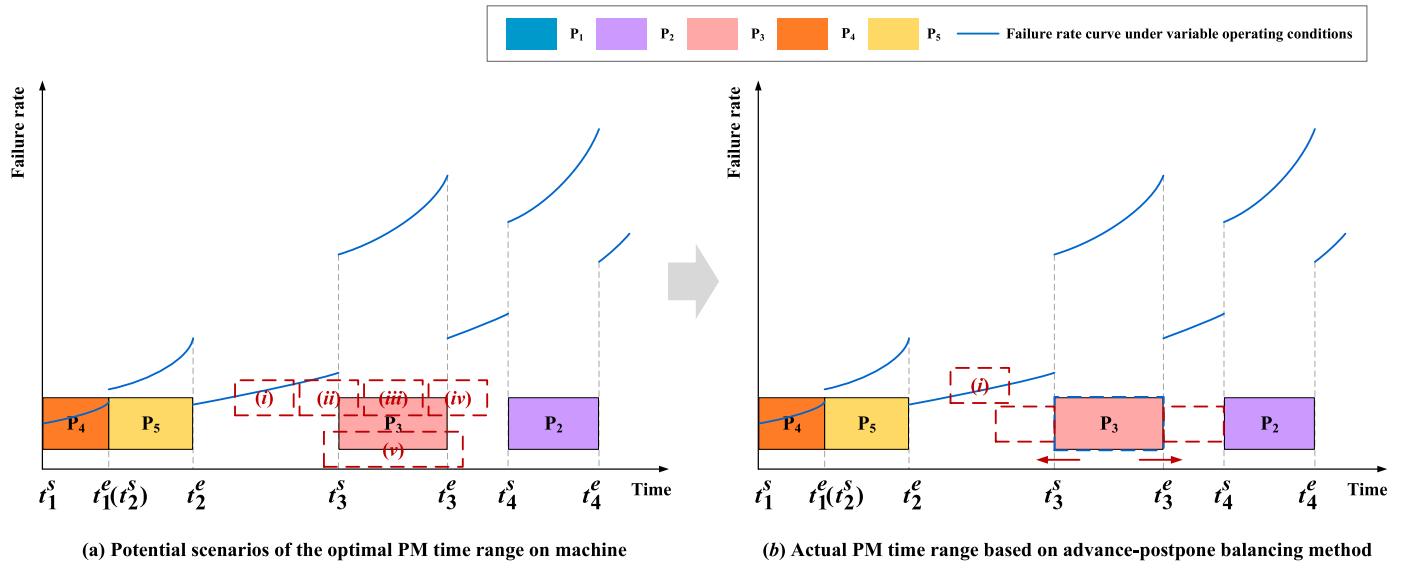


Fig. 4. Schematic diagram of the optimal and actual PM time range.

$$t^d = t^e \quad (22)$$

$$CM_{rk}^d = C_k^f \cdot \left(\int_0^{T_{rk}} \lambda_{rk}(t|\mathbf{X}) dt - \int_0^{T_{rk}} \lambda_{rk}(t|\mathbf{X}) dt \right) = C_k^f \cdot \int_{t_{rk}^*}^{t^d} \lambda_{rk}(t|\mathbf{X}) dt \quad (23)$$

$$CP_{rk}^d = \frac{t^d - t_{rk}^*}{T_{rk}^* + t^d - t_{rk}^*} C_k^p \quad (24)$$

Evaluate the cost savings associated with advanced and postponed implementation to determine the actual time point for PM t'_{rk} for machine M_k :

$$t'_{rk} = \begin{cases} t^d, & (CM_{rk}^a - CP_{rk}^a) > (CP_{rk}^d - CM_{rk}^d) \\ t^d, & \text{otherwise} \end{cases} \quad (25)$$

where the locations of t_{rk}^* can be categorized into two situations: (1) during the operation processing and (2) before the start time of an operation. Fig. 5 illustrates the advancement or postponement of PM activity in both situations.

Based on the above discussion, **Algorithm 1** is designed to determine the actual PM time point for each machine and assess whether the maintenance activities at this point would interrupt the current job scheme.

where R_{rk} is a 0–1 variable, $R_{rk} = 1$ indicates that the actual PM activity of M_k in the r th maintenance cycle still interrupts the current job scheme.

If $\sum_{k \in M, r \in B} R_{rk} \neq 0$, **Algorithm 2** is executed to reassign available machines for interrupted operations and update operation sequencing on all machines. Specifically, jobs can be categorized into three groups: completed jobs, ongoing jobs, and jobs to be processed. Ongoing jobs continue processing while those in the third category are rescheduled.

The job scheme is updated after rescheduling, and the failure rates for all machines are correspondingly revised. Subsequently, **Algorithm 1** is employed to adjust the maintenance decisions and judge if it still interrupts the current job scheme. If so, **Algorithm 2** is executed to optimize machine assignment and operation sequencing. This iterative process, known as feedback-updated strategy, ensures that there is no

overlapping integration between maintenance and production. It allows for incorporating revised PM activities into the current job scheme to generate a jointly optimized scheme.

The mathematical model in stage II is formulated as follows.

$$\min TC = \sum_{i \in P} TTC_i + TBC + \sum_{k \in M} TMC_k \quad (26)$$

s. t.

$$TMC_k = \sum_{r \in B} \sum_{z \in Z} y_{rkz} [C_k^p + C_k^f \times \int_0^{T_{rk}} \lambda_{rk}(t|\mathbf{X}) dt]; \forall k \in M \quad (27)$$

$$T'_{rk} = T_{rk}^* + (t'_{rk} - t_{rk}^*); \forall k \in M, r \in B \quad (28)$$

$$\sum_{z \in Z} y_{rkz} \leq 1; \forall k \in M, r \in B \quad (29)$$

$$\sum_{z \in Z} y_{rkz} \geq \sum_{z \in Z} y_{(r+1)kz}; \forall k \in M, r+1 \in B \quad (30)$$

$$\sum_{r \in B} y_{rkz} t'_{rk} \geq \sum_{r \in B} y_{rkz} \sum_{i \in P} \sum_{j \in O_i} x_{ijkz} t_{ij}^e - W \left(1 - \sum_{r \in B} y_{rkz} \right); \forall k \in M, z \in Z \quad (31)$$

$$\begin{aligned} \sum_{i \in P} \sum_{j \in O_i} x_{ijk(z+1)} t_{ij}^s &\geq \left(1 - \sum_{r \in B} y_{rkz} \right) \left[\sum_{i \in P} \sum_{j \in O_i} x_{ijkz} t_{ij}^e + u_{k(z+1)} T_k^f \right] \\ &+ \sum_{r \in B} y_{rkz} (t'_{rk} + T_k^p); \forall k \in M, z+1 \in Z \end{aligned} \quad (32)$$

$$y_{rkz} \in \{0, 1\}, \forall k \in M, r \in B, z \in Z \quad (33)$$

$$t'_{rk} > 0, \forall k \in M, r \in B \quad (34)$$

The objective function (26) minimizes the total cost by considering three components: tardiness, workload balancing, and maintenance activities. The cost associated with maintenance is determined by constraints (27) and (28). Constraint (29) stipulates that only 1 p.m. activity can be conducted on each machine during every maintenance cycle. Constraint (30) enforces the sequential arrangement of the maintenance cycle for each machine. Constraints (31) and (32) together prevent overlap between production and maintenance assigned to the same machine. Constraints (33) and (34) define the decision variable domains. Meanwhile, the constraints (2)–(13) at stage I must be strictly

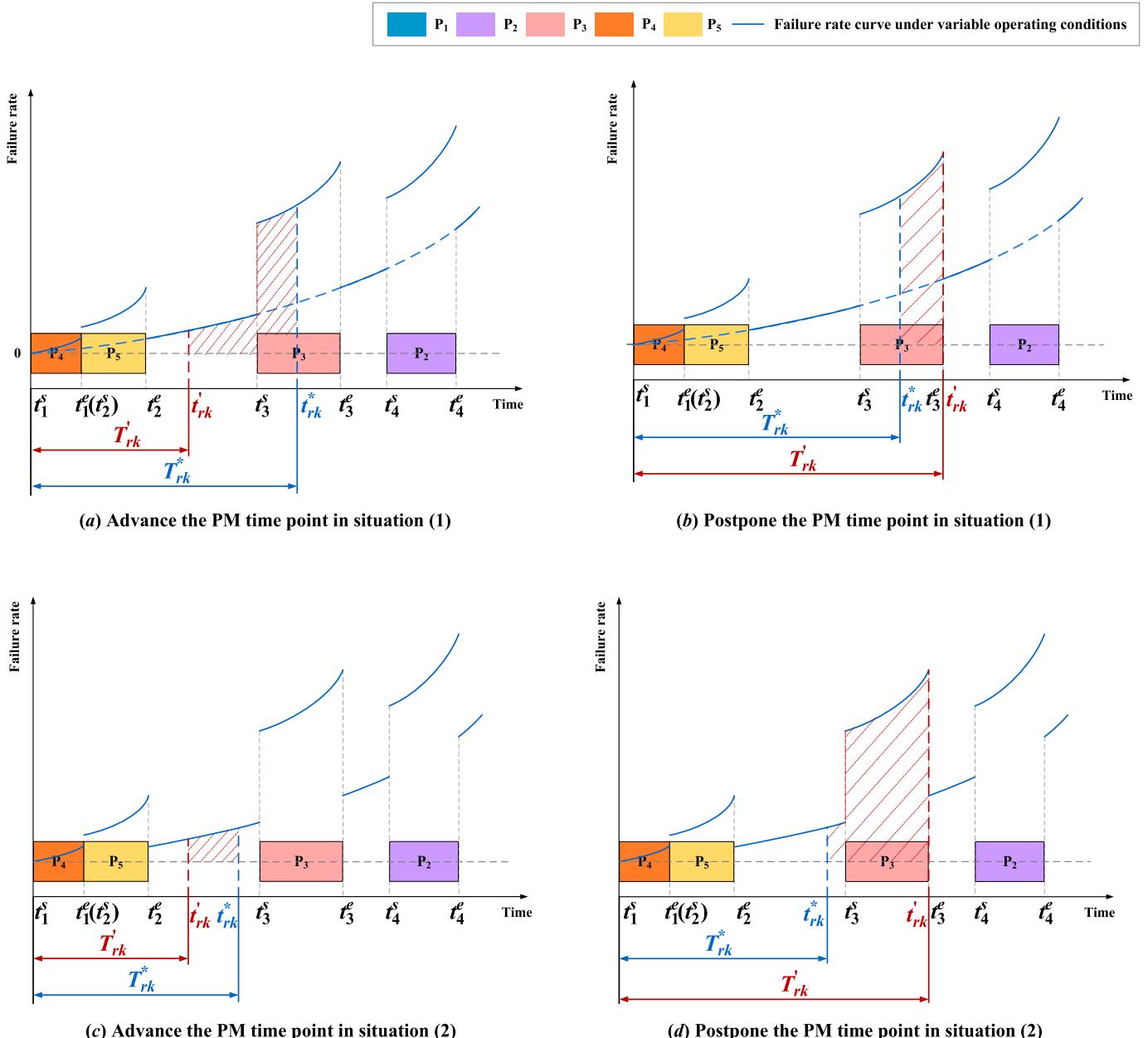


Fig. 5. Schematic diagram of the PM time point advanced or postponed.

followed for both sub-problems involved in the rescheduling process: machine assignment and operation sequencing.

4. Hybrid DPSO-VNS algorithm

4.1. Framework

The framework of the hybrid DPSO-VNS algorithm is illustrated in Fig. 6, with a detailed description provided below.

Step 1 Input the production data, such as the number of jobs, operations, and machines; candidate machines along with their corresponding processing times for each operation.

Step 2 Set parameters for the hybrid DPSO-VNS algorithm, including the swarm size and maximum iteration count, to ensure optimal performance.

Step 3 Initialization of particle swarm is generated by improved hybrid methods.

Step 4 Each particle is decoded and evaluated by a fitness function.

Step 5 Each particle is optimized using the DPSO algorithm, which comprises three parts (individual search, local learning, and global learning). The update is performed by comparing the fitness degrees of newly generated particles.

Step 6 The swarm is updated after each iteration, then the next search and optimization process is executed. If the termination criterion is satisfied, the top 10% best solutions are output, and the DPSO algorithm stops; otherwise, it proceeds to Step 4.

Step 7 Employ the VNS algorithm to further optimize the top 10% particles from the DPSO algorithm, where each particle undergoes updates through S_{ma} and V_{op} operators. The best particle is determined based on meeting the criteria, which serves as the initial job scheduling scheme.

Algorithm 1: Actual PM time point for each machine

Input: a job scheme, maintenance parameters of machines

01. **for** $k=1$ to K
02. **for** $r=1$ to r_k
03. determine failure rate function $\lambda_{rk}(t|\mathbf{X})$ under variable operating conditions
04. calculate T_{rk}^*
05. obtain t_{rk}^* and $[t_{rk}, t_{rk}^*]$
06. **if** t_{rk}^* is far beyond the completion time of the z_k^{th} position on M_k
07. $\sum_{z \in Z} y_{rkz} = 0$
08. **elseif** $[t_{rk}^*, t_{rk}^{e*}]$ interrupts the z^{th} operation on M_k
09. calculate cost savings of advanced and postponed PM activity
10. **if** $(CM_{rk}^a - CP_{rk}^a) > (CP_{rk}^d - CM_{rk}^d)$
11. **if** $\sum_{i \in P} \sum_{j \in O_i} x_{ijkz} t_{ij}^s - \sum_{i \in P} \sum_{j' \in O_i} x_{ij'k(z-1)} t_{ij'}^e \geq T_k^P$
12. $\dot{t}_{rk} = \sum_{i \in P} \sum_{j \in O_i} x_{ijkz} t_{ij}^s - T_k^P$
13. **else**
14. $\dot{t}_{rk} = \sum_{i \in P} \sum_{j \in O_i} x_{ij'k(z-1)} t_{ij'}^e$
15. $y_{rk(z-1)} = 1$
16. **else**
17. $\dot{t}_{rk} = \sum_{i \in P} \sum_{j \in O_i} x_{ijkz} t_{ij}^s$
18. $y_{rkz} = 1$
19. **end if**
20. **if** $[\dot{t}_{rk}, t_{rk}^e]$ interrupts the z^{th} operation or subsequent operations on M_k
21. $R_{rk} = 1$
22. **end if**
23. **elseif** $[t_{rk}^*, t_{rk}^{e*}]$ is during the idle period of production
24. $\dot{t}_{rk} = t_{rk}^*$
25. **for** $z=1$ to z_k
26. **if** $\sum_{i \in P} \sum_{j \in O_i} x_{ijkz} t_{ij}^s > \dot{t}_{rk}$
27. $y_{rk(z-1)} = 1$, and go to step 29
28. **end if**
29. **end for**
30. **end if**
31. **end for**
32. **end for**

Output: $y_{rkz}, \dot{t}_{rk}, R_{rk}$ ($k \in M, r \in B, z \in Z$)

Step 8 Input the maintenance parameters of machines, such as θ_k and η_k of each machine, time durations, and costs for a PM/MR activity.

Step 9 Based on the failure rate functions under variable operating conditions, T_{rk}^* and t_{rk}^* for each machine during every maintenance cycle are determined by minimizing V_{rk} . Perform **Algorithm 1** to revise maintenance decisions and assess whether PM activities of machines interrupt the current job schedule. If $\sum_{k \in M, r \in B} R_{rk} = 0$, arrange the corresponding machine's PM activities in the current job scheme. Subsequently, output the joint optimization scheme and terminate the entire DPSO-VNS algorithm. Otherwise, it proceeds to **Step 10**.

Step 10 **Algorithm 2** is executed for rescheduling, and the result is optimized based on the VNS algorithm. Upon reaching the termination condition, generate a new job scheme and return to **Step 9**.

4.2. DPSO algorithm

The DPSO algorithm starts from initial particles and searches for the optimal solution by moving these particles according to their own flight experience and neighbors. Each particle contains a best solution set p^* during its trajectory, and the whole swarm shares a global best solution

set g^* (Zhou et al., 2022).

$$x_i(t+1) = c_2 \otimes \{c_1 \otimes [w \oslash x_i(t), p^*(t)], g^*(t)\} \quad (35)$$

where $w > 0$ is an inertia weight to maintain the particle flight; \oslash and \otimes represent the mutation and crossover operators, respectively. c_1 and c_2 are personal and social confidence coefficients, respectively.

4.2.1. Encoding and decoding

(1) Encoding

A two-layer coding is adopted to represent a feasible solution, with the layers being operation sequence (OS) and machine sequence (MS). The element in OS is the job index, and the number of occurrences indicates the corresponding job's operation. MS is formed by selecting a processing machine for each operation in the OS.

(2) Decoding

Obtain information from the OS successively and convert it into the corresponding operation O_{ij} . The machine at the same position in the MS is denoted as M_k , and the processing time PT_{ijk} is determined from the input production data. The current completion time of P_i and M_k are

Algorithm 2: Rescheduling

Input: a job scheme, t_{rk}^i ($k \in M, r \in B$)
 01. **while** $\sum_{k \in M} \sum_{r \in B} R_{rk} \neq 0$
 02. record the position finished z of each machine: $\sum_{i \in P} \sum_{j \in O_i} x_{ijkz} t_{ij}^s < \min \{t_{rk}^i | k: R_{rk} = 1\}$
 03. record the operations completed j for each job: $t_{ij}^s < \min \{t_{rk}^i | k: R_{rk} = 1\}$
 04. **for** $k = 1$ to K
 05. **if** $z < z_k$
 06. record $\sum_{i \in P} \sum_{j \in O_i} x_{ijkz} t_{ij}^e$ as ET_k (availability time of M_k)
 07. **end if**
 08. **end for**
 09. **for** $i = 1$ to I
 10. **if** $j < n_i$
 11. record t_{ij}^e as ET_{ij} (completion time of the current operation of P_i)
 12. **for** $j' = j+1$ to n_i
 13. select available machine k for $O_{ij'}$
 14. updating the start time and completion time of $O_{ij'}$, ET_{ij} , and ET_k
 15. **end for**
 16. **end if**
 17. **end for**
 18. **end while**
Output: updated machine assignment, operation sequencing, start time, and completion time of all operations

recorded as ET_{ij} and ET_k , respectively. If $j = 0$ and O_{ij} is in the first position of M_k , $ST_{ij} = 0$; otherwise, $ST_{ij} = \max\{ET_{ij}, ET_k\}$. After assigning a processing machine for O_{ij} , the machine workload and resource available time need to be updated. Repeat this process until the last element in the OS.

4.2.2. Initialization

The OS is initialized randomly, while the initial MS is generated using four methods in a certain proportion: 30% through global selection method I, 30% through global selection method II, 20% through local selection method, and 20% through random generation. Specifically, global selection methods I and II select machine M_k with the minimum workload for O_{ij} from M_j ; the difference lies in whether the processing time of O_{ij} is included in M_k 's current workload. In the local selection method, the machine with the shortest processing time for O_{ij} is selected.

4.2.3. Search strategy of particle optimization

The fitness function f is employed to evaluate the quality of particles,

$$f = 1 / \left[\sum_{k=1}^K TMC_k + \sum_{i=1}^I TTC_i \right] \quad (36)$$

In addition, the value of w is updated adaptively in each iteration,

$$w(t+1) = w + (w + w')^{2m/(m+1)} \quad (37)$$

$$m = [f^*(t) - f_i(t)] / [f^*(t) + f_i(t)] \quad (38)$$

where w' denotes the final inertia weight. w and w' are set to 0.9 and 0.4, respectively. $f^*(t)$ and $f_i(t)$ represent the fitness degree of $g^*(t)$ and $x_i(t)$, respectively.

Subsequently, the DPSO algorithm optimizes each particle through three parts.

Part I :

$$P_i^I(t) = \begin{cases} w^{\otimes^1} x_i(t), \text{rand}() < w/2 \\ w^{\otimes^2} x_i(t), w/2 \leq \text{rand}() < w \\ x_i(t), \text{otherwise} \end{cases} \quad (39)$$

where, $\text{rand}()$ is a random number between 0 and 1. \otimes^1 and \otimes^2 represent two mutation operations, V_{ma} and V_{op} , respectively.

Part II :

$$P_i^{II}(t) = \begin{cases} c_1 \otimes^1 (x_i(t), p^*(t)), \text{rand}() < c_1/2 \\ c_1 \otimes^2 (x_i(t), p^*(t)), c_1/2 \leq \text{rand}() < c_2 \\ x_i(t), \text{otherwise} \end{cases} \quad (40)$$

where, \otimes^1 and \otimes^2 represent two crossover operations, noted as C_{job} and C_{op} , respectively.

Part III :

$$P_i^{III}(t) = \begin{cases} c_2 \otimes^1 (x_i(t), g^*(t)), \text{rand}() < c_2/2 \\ c_2 \otimes^2 (x_i(t), g^*(t)), c_2/2 \leq \text{rand}() < c_2 \\ x_i(t), \text{otherwise} \end{cases} \quad (41)$$

These mutation and crossover operators, as illustrated in Fig. A1 in the Appendix, are further explained through a numerical example. The job scheme depicted in Fig. 2 can be represented using the two-layer coding method, with OS = [4, 5, 1, 4, 5, 1, 4, 3, 2, 1, 3, 2, 2] and MS = [1, 1, 2, 3, 2, 3, 4, 4, 3, 2, 1, 2, 1]. The V_{ma} and V_{op} operators are employed for the self-learning of particles. Specifically, an auxiliary sequence S = [0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1] is randomly generated to ressign machines to the operation in OS based on Table 1 when encountering position "1" in S. The V_{op} operator is utilized to change the operation sequence on machines while conforming to the process flow of jobs. The resulting feasible solution is presented as a Gantt chart in Fig. 7 (a). In contrast, the C_{job} and C_{op} operators serve as means for interparticle information exchange. The five jobs are randomly divided into two classes: $P^1 = \{1, 2, 3\}$ and $P^2 = \{4, 5\}$. A new particle (3) is generated by assimilating knowledge from particles (1) and (2), resulting in the OS = [4, 5, 1, 4, 5, 1, 4, 3, 2, 1, 3, 2, 2] and the MS = [1, 3, 2, 1, 2, 3, 2, 4, 3, 2, 1, 2, 1]. Furthermore, particle (4) acquires position-based

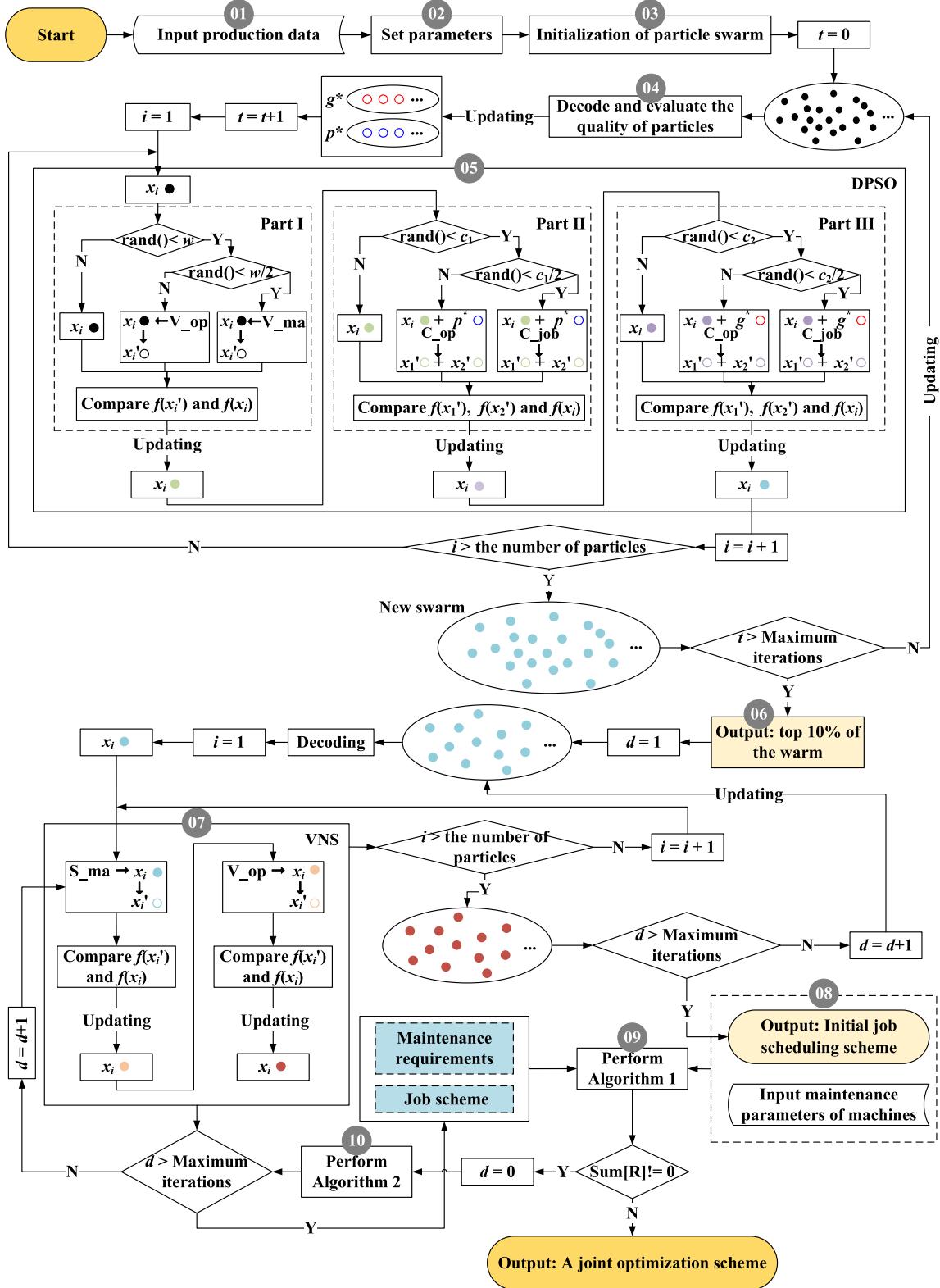


Fig. 6. Framework of the hybrid DPSO-VNS algorithm.

coding information using the C_{op} operator from particles (2) and (3), as illustrated in Fig. 7 (b).

4.3. VNS algorithm

The VNS algorithm is employed to locally search the current solution

by two operations, S_{ma} and V_{op} . Here, S_{ma} represents random changes in the processing machine of an operation. The specific search process is provided in the pseudo-code, shown as **Algorithm 3**.

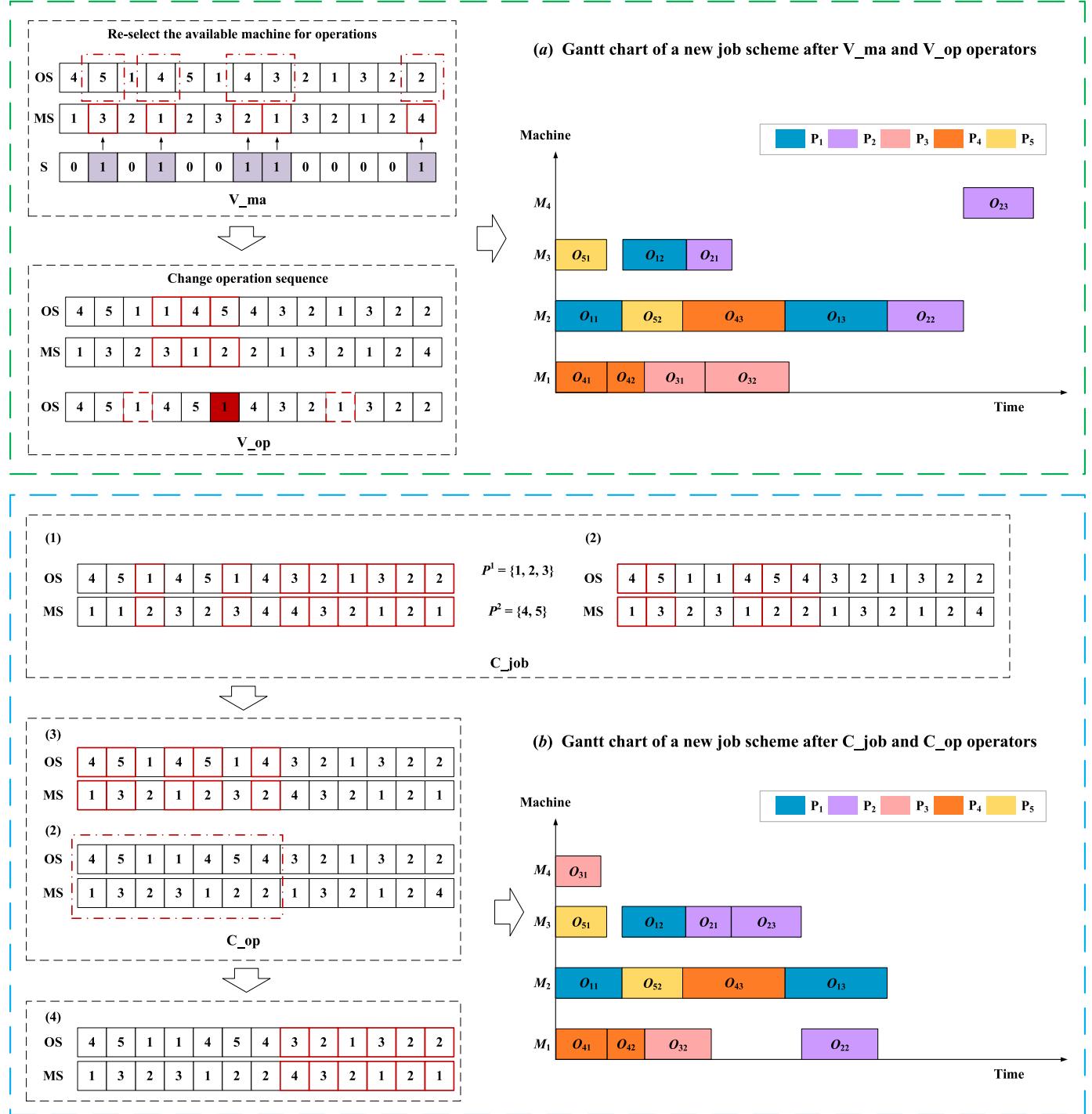


Fig. 7. A numerical example of mutation and crossover operators.

5. Case study

5.1. Experimental setup

In this section, a case of an automotive powertrain manufacturing system is studied to verify the performance of the model. The job shop produces various products, such as cylinder heads, housings, valve bodies, etc., each requiring multiple processes. Drilling machines, grinders, lathes, and other machinery provide scheduling flexibility. Detailed maintenance parameters of machines and production information can be found in Table A1 and Table A2 in the Appendix. The

machining parameters of operations are characterized by the width of cut, feed speed, ambient temperature, and cutting force. Additionally, a penalty of balancing among machine workloads C^b is set to 500. The experiments are implemented using Python version 3.9 on a laptop with AMD Ryzen 7 5800 H at 3.20 GHz with 16.0 GB RAM.

5.2. Results and discussion

Firstly, the initial job scheduling scheme is generated (Fig. 8(a)) with a minimum objective value of 1612.45, ensuring timely completion of all jobs (i.e., $TTC = 0$). The workload of each machine is illustrated in

Algorithm 3: VNS algorithm

```

Input: a feasible solution  $x$  (OS, MS)
1. while the termination criterion is not met do
2.    $m=1$ 
3.   while  $m < m_{\max}$  do
4.      $x' = S_{\text{ma}}(x)$ 
5.     if  $f(x') > f(x)$  then
6.        $x \leftarrow x'$ 
7.     end if
8.      $x' = V_{\text{op}}(x)$ 
9.     if  $f(x') > f(x)$  then
10.       $x \leftarrow x'$ 
11.    end if
12.     $m = m+1$ 
13.  end while
14. end while

```

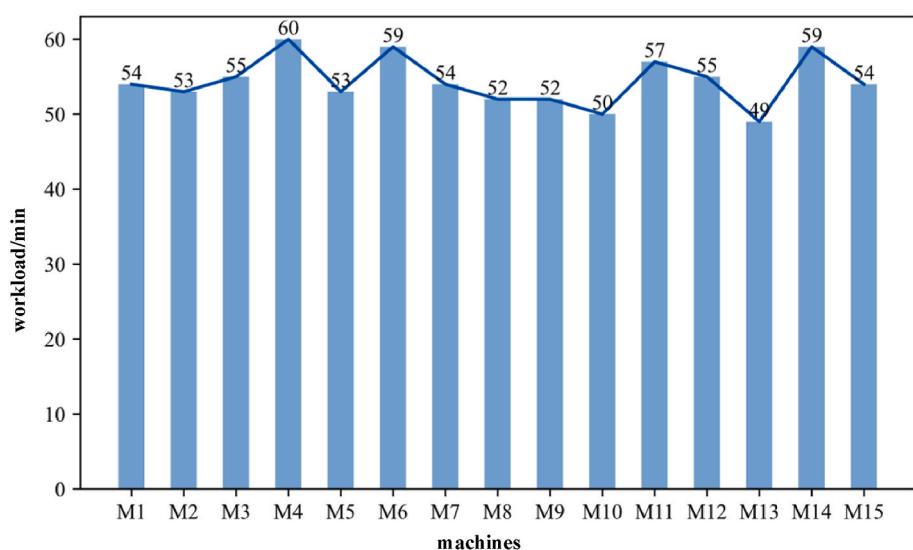
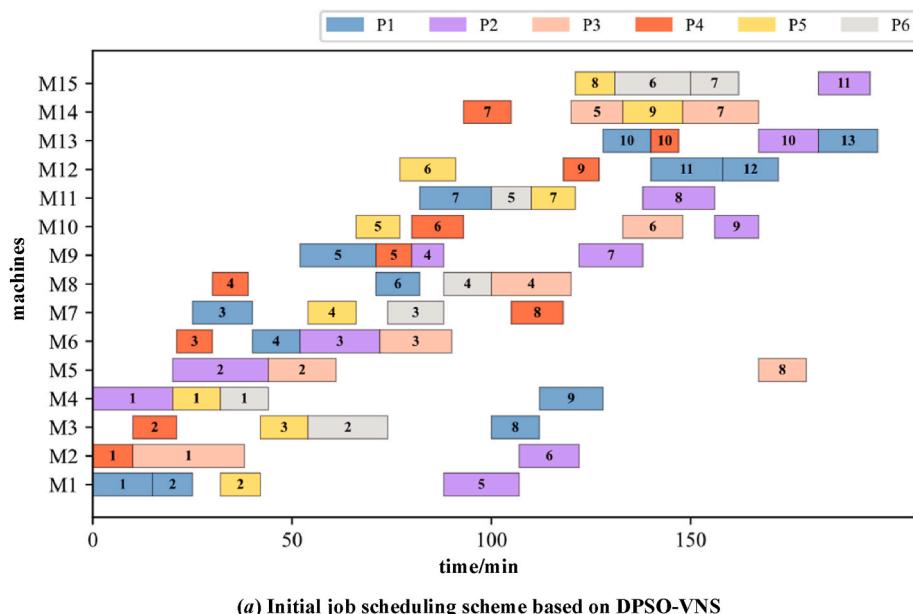


Fig. 8. Initial job scheduling scheme and machine workloads.

Fig. 8(b).

Subsequently, employing Eqs. (14) and (15), the machine's failure rates are calculated separately for both scenarios - considering and not considering operating conditions - in this initial job scheduling scheme, as illustrated in Fig. 9. It is apparent that each machine's failure rate increases over time. The failure rate curves under operating conditions exhibit dynamic and multi-stage characteristics, represented by the red line in Fig. 9. In a flexible job shop environment, the machine failure process exhibits production-dependent behavior influenced by factors such as intrinsic deterioration, operation sequencing, processing time per operation, cumulative service time, and specific process flow of jobs. Changes in operating conditions cause a sudden jump in the machine's failure rates. Machines have higher failure rates under variable operating conditions at any given time compared to neglecting such conditions. In other words, operating conditions accelerate the probability of machine failure. These findings align with real-world scenarios.

The optimal PM intervals and time points for each machine in the two situations mentioned above, based on Eqs. (14)–(18), are presented in Table 2. ψ_{rk}^* and t_{rk}^* represent the optimal PM interval and optimal PM time point of machine M_k are solely determined by its natural stochastic failure process. Due to the acceleration of machine failure rate during variable operating conditions, T_{rk}^* is found to be less than ψ_{rk}^* for any k within each maintenance cycle. Moreover, the value of T_{rk}^* also varies in accordance with the machine's performance and the specific tasks it performs. This implies that machines in actual production processes require more frequent maintenance. Neglecting operating conditions when estimating machine reliability can lead to significant errors.

Observe the distribution of $[t_{rk}^*, t_{rk}^{e*}]$ for each machine in the initial job scheduling scheme (Fig. 10). It was found that maintenance time points for M_2 and M_5 coincided with the idle period of these machines, i.e., t_{12}^*

$= t_{12}^*$, and $t_{15}^* = t_{15}^*$. However, PM activities for M_1, M_3, M_8, M_{10} , and M_{12} caused interference with production. There, Algorithm 1 is employed to determine the actual PM activities of these machines as listed in Table 3. PM activities are advanced for M_8, M_{10} , and M_{12} , while postponed for M_1 and M_3 .

The actual PM time ranges of M_8, M_{10} , and M_{12} still cause disruptions to subsequent operations. Subsequently, operations with a start time greater than or equal to $\min\{t_{18}^*, t_{110}^*, t_{112}^*\} = t_{18}^*$ are rescheduled using Algorithm 2. Following the feedback-update strategy, Algorithms 1, 2, and 3 are iteratively executed until there is no interference between PM activities/jobs. Consequently, a joint optimization scheme with $TC = 8665.865$ is obtained wherein $TTC = 0.0$, $TBC = 4934.813$, and $TMC = 3731.052$; furthermore, the joint scheme and the corresponding machine workloads are shown in Fig. 11.

5.3. Comparative study and analysis

5.3.1. Comparison of strategies

The effectiveness of Strategy V, the feedback-update strategy, has been demonstrated through a comparative analysis involving four commonly employed strategies: Strategy I incorporating periodic maintenance (Vieira et al., 2003), Strategy II employing a right-shifting strategy (Wang et al., 2018), Strategy III utilizing a left-shifting strategy (Li and Pan, 2012), and Strategy IV implementing a partial rescheduling strategy (Lv et al., 2022).

The actual PM time ranges of 15 machines on the five strategies are listed in Table 4. The number of PM activities in strategy I is related to the preset T . Strategies II–IV share a common value determined by the initial job scheduling scheme. Strategy IV ignores the influence of changes in job schemes on machine maintenance requirements. Our proposed strategy considers the dynamic balance between production

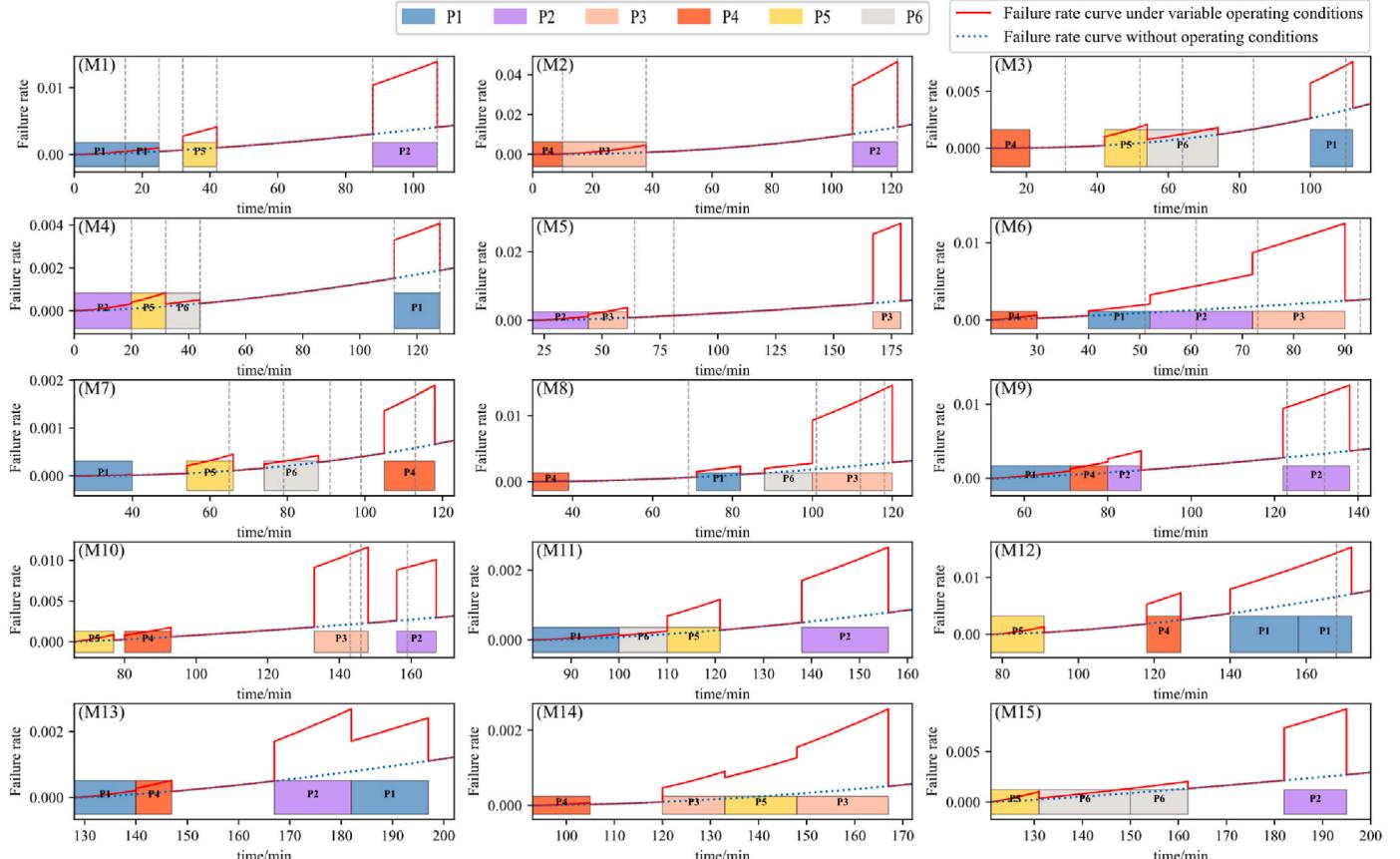
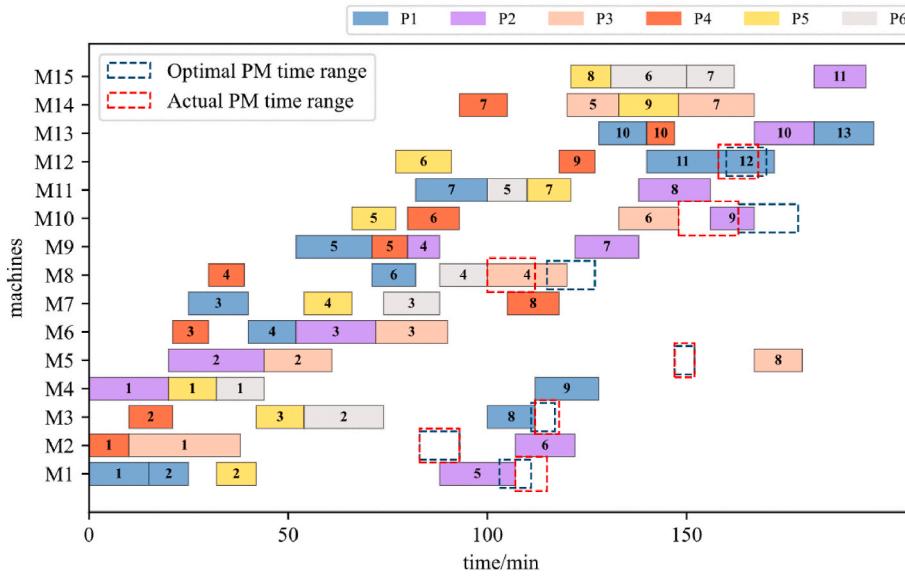
**Fig. 9.** Failure rates of machines under the initial job scheduling scheme.

Table 2

Optimal PM intervals and optimal PM time points of the above two situations.

	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}	M_{11}	M_{12}	M_{13}	M_{14}	M_{15}
T_{1k}^*	103	83	101	177	127	80	193	85	99	97	178	83	163	205	124
t_{1k}^*	103	83	111	177	147	101	218	115	151	163	260	160	291	298	245
Ψ_{1k}^*	136	90	116	189	137	134	198	125	127	148	188	105	174	221	141
ψ_{1k}^*	136	90	126	189	157	155	223	155	179	214	270	182	302	314	262

**Fig. 10.** Comparison of optimal and actual PM time ranges.**Table 3** T_{rk}^* , t_{rk}^* , and t_{rk}^e of machines.

	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}	M_{11}	M_{12}	M_{13}	M_{14}	M_{15}
T_{1k}^*	107	83	102	–	127	–	–	70	–	82	–	81	–	–	–
t_{1k}^*	107	83	112	–	147	–	–	100	–	148	–	158	–	–	–
t_{1k}^e	115	93	118	–	152	–	–	112	–	163	–	168	–	–	–

continuity and machine reliability, aligning more closely with contemporary flexible production demands.

Additionally, Figs. 12 and 13 present the corresponding total costs and the joint optimization schemes, respectively.

It noted that the proposed strategy V demonstrates a minimum total cost $TC = 8665.865$, while strategies I-IV exhibit costs of 15139.993, 15599.231, 25237.297, and 19346.979 respectively. Compared to the previous four strategies, there is a reduction in total cost by 42.762%, 44.447%, 65.662%, and 55.208% respectively. Specifically, the TBC of strategies I-III are the same, and the TMC of strategy I is inversely proportional to the value of T . In contrast to strategy IV, strategy II incorporates additional TMC to ensure $TTC = 0$; conversely, strategy III reduces TMC by sacrificing TTC. Strategy V reduces TMC by assigning operations predominantly to highly reliable machines during rescheduling while ensuring the timely completion of all jobs.

5.3.2. Comparison of algorithms

Moreover, the performance of the DPSO-VNS algorithm has also been compared with GA, DPSO, VNS, and JAYA algorithms. All four compared algorithms are adjusted to suit the specific problem at hand.

From a cost perspective, the initial scheduling scheme in stage I, based on five algorithms (GA, DPSO, VNS, JAYA, and DPSO-VNS), yielded objective values of 2914.659, 2021.904, 2336.562, 2555.107, and 1612.452 respectively. All five algorithms successfully meet the

delivery time requirements of jobs while our proposed algorithm demonstrates superior quality in providing an initial solution. Subsequently, the total costs of these algorithms in stage II under the joint optimization scheme are compared and depicted in Fig. 14. The five algorithms exhibit schemes with $TTC = 0$ and corresponding total goals of 10640.423, 9399.174, 1758.460, 10341.748, and 8665.865, respectively. Among them, the proposed DPSO-VNS algorithm achieves a reduction in total cost by 18.557%, 7.802%, 26.301%, and 16.205% respectively. These findings highlight the crucial role played by high-quality baseline/initial plans in determining the quality of realized plans, which aligns with prior contributions made by Framinan et al. (2019), Ghaleb et al. (2020), and Ghaleb et al. (2021).

Additionally, a perfect scheme should ensure balanced workload distribution among all machines and avoid excessive utilization of any single machine. The workload of each machine is closely associated with the job scheme, which in turn impacts the failure rates and maintenance requirements of the machines. To provide a more comprehensive analysis, we compare and analyze the individual machine workloads and the overall workload balance in the joint optimization scheme using five algorithms. These results are presented in Table 5 and Fig. 15, respectively. Range and standard deviation (SD) have been selected as evaluation metrics to assess the effectiveness of these algorithms in achieving workload balance across machines. The range values of machine workloads based on the five algorithms are observed to be 50, 51, 70, 53,

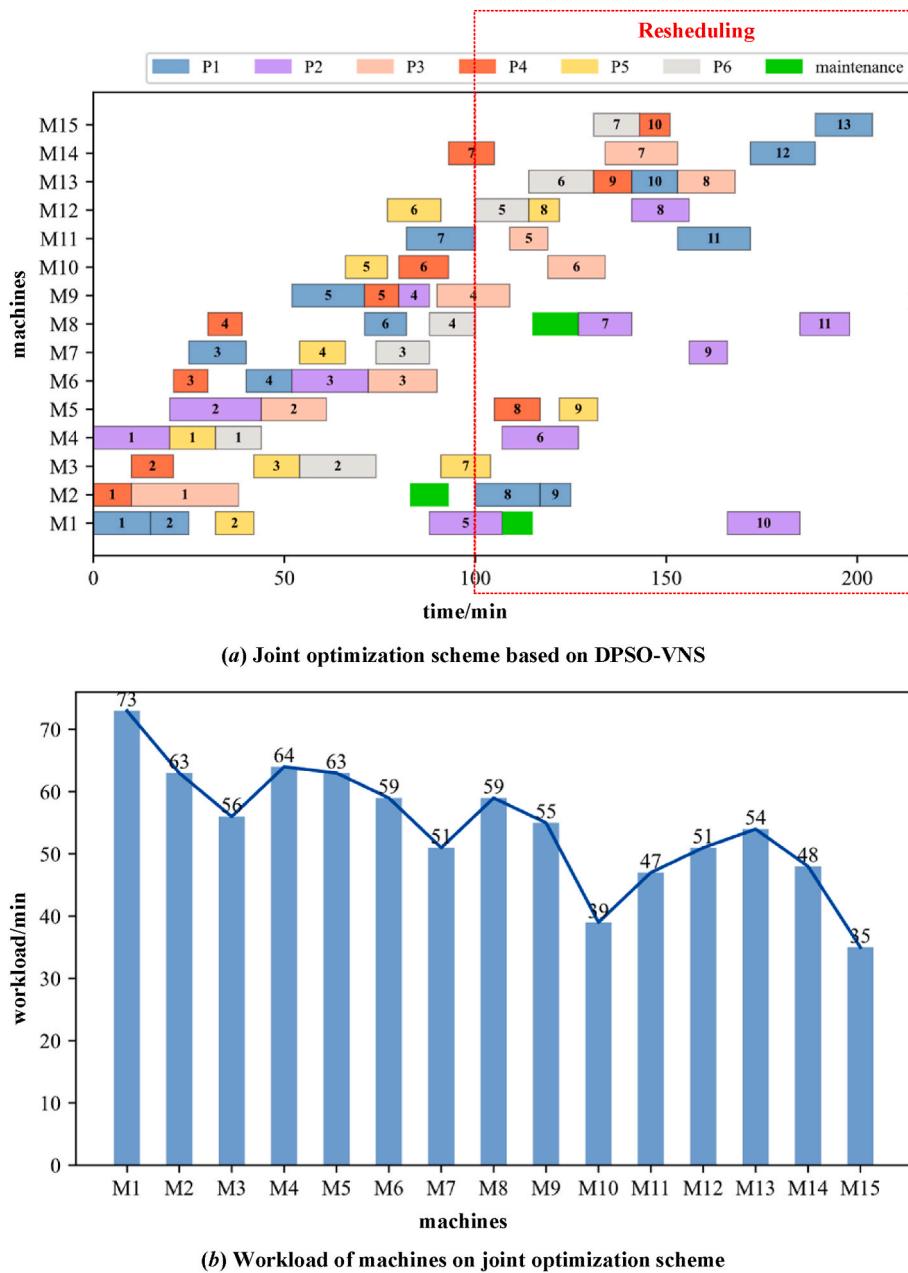


Fig. 11. Joint optimization scheme and machine workloads.

Table 4
Actual PM time ranges of machines on five strategies.

	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}	M_{11}	M_{12}	M_{13}	M_{14}	M_{15}
Strategy I	$t_{1k}^{'}$	90	90	100	90	110	—	—	—	156	—	—	—	—	—
	t_{1k}^{t}	98	100	106	103	115	—	—	—	171	—	—	—	—	—
Strategy II	$t_{1k}^{'}$	107	83	112	—	147	—	—	120	—	167	—	172	—	—
	t_{1k}^{t}	115	93	118	—	152	—	—	132	—	182	—	182	—	—
Strategy III	$t_{1k}^{'}$	80	83	94	—	147	—	—	100	—	148	—	158	—	—
	t_{1k}^{t}	88	93	100	—	152	—	—	112	—	163	—	168	—	—
Strategy IV	$t_{1k}^{'}$	107	83	112	—	147	—	—	100	—	148	—	158	—	—
	t_{1k}^{t}	115	93	118	—	152	—	—	112	—	163	—	168	—	—
Strategy V	$t_{1k}^{'}$	107	83	—	—	—	—	—	115	—	—	—	—	—	—
	t_{1k}^{t}	115	93	—	—	—	—	—	127	—	—	—	—	—	—

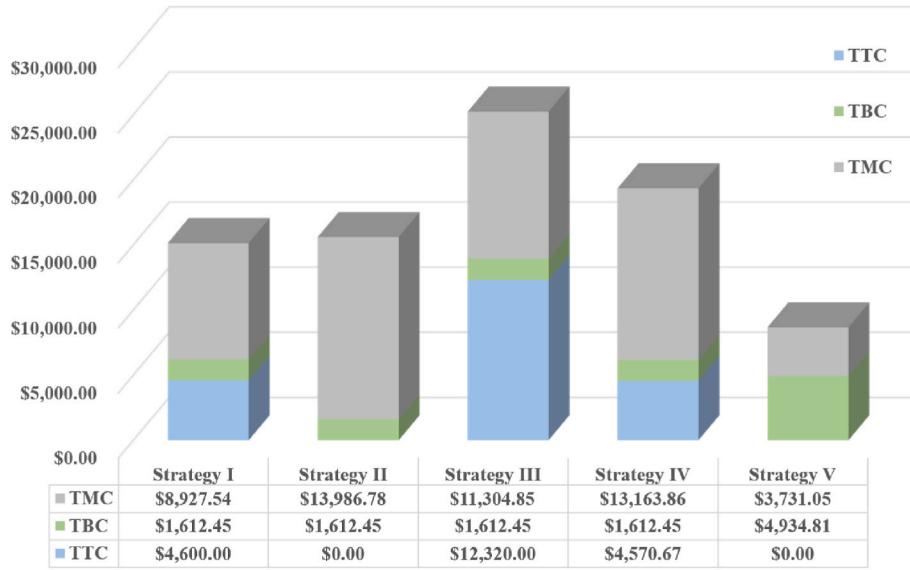


Fig. 12. Cost comparison of joint optimization schemes among five strategies.

and 38 respectively, with corresponding SD values of 14.899, 15.197, 17.537, 14.257, and 9.870. A smaller SD indicates closer proximity of overall machine workloads to the mean value and lower volatility levels in balancing machine workloads across different algorithmic approaches studied here. This conclusion can be further visually observed from Fig. 15, where it is evident that the proposed algorithm achieves a more balanced distribution of workload for machines around the mean value with minimal deviation compared to other algorithms considered in this study. Therefore, it can be concluded that our proposed algorithm demonstrated superior performance in achieving cost savings and effectively balancing machine workloads.

5.4. Larger-sized computational experiment

This subsection aims to provide a larger-sized computational example to further demonstrate the performance of the proposed model in addressing three subproblems: machine assignment, operation sequencing, and maintenance arrangement. Notably, this research emphasizes the previously overlooked aspect of incorporating maintenance decisions into the joint optimization of FJSSP-PM. To illustrate this, we consider processing 10 randomly selected jobs on a set of 15 machines. Each job consists of different numbers of operations: 12, 15, 13, 10, 9, 7, 11, 13, 8 and 10 respectively. By employing the proposed two-stage joint optimization model along with the DPSO-VNS algorithm can obtain the final scheme (as shown in Fig. 16). The maintenance decisions for each machine are presented in Table 6.

This example illustrates the implementation of the proposed model in terms of both random failures and the number of maintenance activities. For instance, there is a failure occurring on machine M_7 before its 7th processing position, the start time of the 7th position $\sum_{i \in P} \sum_{j \in O_i} x_{ij7,7} t_{ij}^e$

should be greater than $\left(\sum_{i \in P} \sum_{j \in O_i} x_{ij7,6} t_{ij}^e + T_7^f \right)$. In other words, the start time of $O_{8,11}$ processed on the 7th position of machine M_7 should be updated as $t_{8,11}^e = \max[(t_{4,6}^e + T_7^f), t_{8,10}^e]$. Furthermore, the results depicted in Fig. 16 and Table 6 indicate that an increase in machine service time necessitates a higher frequency of maintenance activities to improve performance; however, the actual execution times of these activities depend on both machine performance and the specific jobs being processed. Fortunately, based on parameters of production information and machine performance set by enterprise managers, this study can generate cost-effective and efficient production as well as

maintenance schemes.

6. Conclusion

The study proposes a two-stage joint optimization model to achieve seamless machine operation and uninterrupted flexible production in contemporary industrial settings, aiming to provide cost-effective and operable schemes for production and maintenance. Specifically, the parallel production of multiple types necessitates frequent switching between different jobs enabling machines to operate under varying operating conditions. A comprehensive evolution model for machine failure rate considering the influence of cumulative service time, variable operating conditions, and maintenance effect is constructed based on the PHM model. The machine in a flexible job shop environment exhibits production-dependent failure behavior, with results showing that operating conditions will accelerate machine failures and affect both the failure rate and maintenance requirements. In addition, the feedback-update strategy is proposed to eliminate the coupling and constraint relationships between production and maintenance, thereby achieving a dynamic balance between the two domains. This strategy has been compared with four current popular strategies (straight insertion, right shifting, left shifting, partial rescheduling) to minimize the objective with the minimal number of PM activities for machines while ensuring all jobs are delivered on time. Moreover, a hybrid DPSO-VNS algorithm is developed to efficiently obtain an approximate optimal solution. Comparison results with GA, DPSO, VNS, and JAYA algorithms have shown that the DPSO-VNS algorithm consistently provides higher quality solutions for both initial job scheduling schemes and joint optimization schemes. The model and algorithm presented in this paper demonstrate the efficacy of cost-savings in the joint optimization of FJSSP-PM.

6.1. Managerial insights

In this study, a machine in a flexible job shop environment characterized by production-dependent failure behavior is considered. In other words, the machine failure rates depend on the production state (e.g., cutting force, applied load, cutting speed, feed speed, cutting depth, and other factors during processing different jobs), and the natural degradation machine over time. Performing maintenance activities to ensure high machine reliability is a crucial prerequisite for achieving stable and uninterrupted production, which holds immense practical domains. In

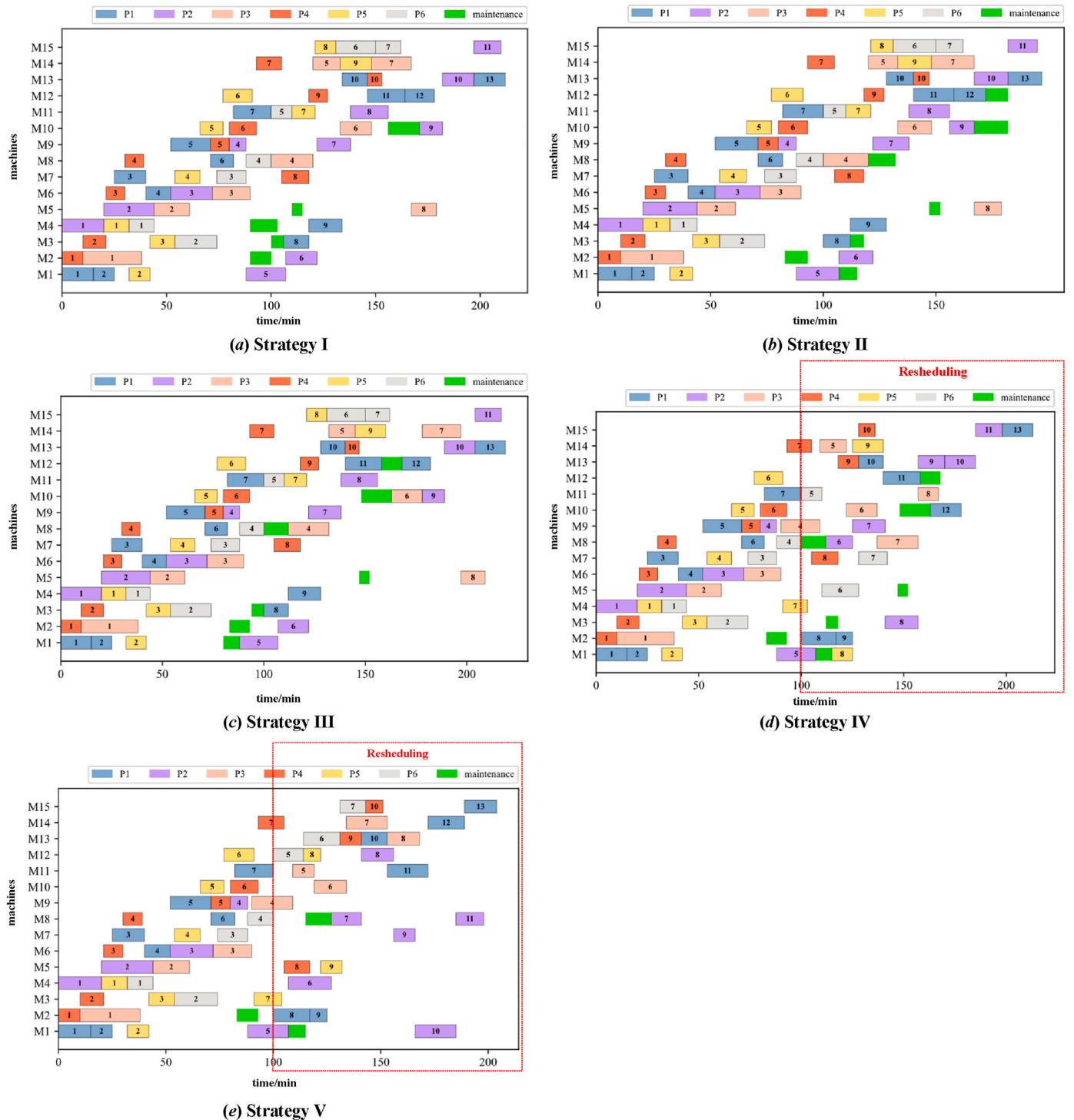


Fig. 13. Joint optimization schemes of five strategies.

conventional studies of FJSSP-PM, the implementation of maintenance revolves around time-based methods. However, excessive maintenance leads to escalated costs, increased machine unavailability duration, and heightened downtime losses. Conversely, insufficient maintenance fails to effectively mitigate the probability of failures. In this context, maintenance engineers should judiciously determine the optimal timing for implementing maintenance interventions based on the machine's operational state within the actual production process from both economic and operational perspectives. We establish a two-stage joint optimization framework and model that allows practitioners to address

machine assignment, operation sequencing, and maintenance arrangement simultaneously while minimizing the overall cost incurred, as depicted in Fig. 1. This peculiar feature provides the possibility to conduct simultaneous optimization of production and maintenance planning, instead of scheduling maintenance in advance to optimize production scheduling. Furthermore, the case study findings provide valuable managerial insights for strategic decision-making regarding implementation, thereby aiding researchers and practitioners in recognizing the significance and necessity of considering high-frequency production switching characteristics within flexible production

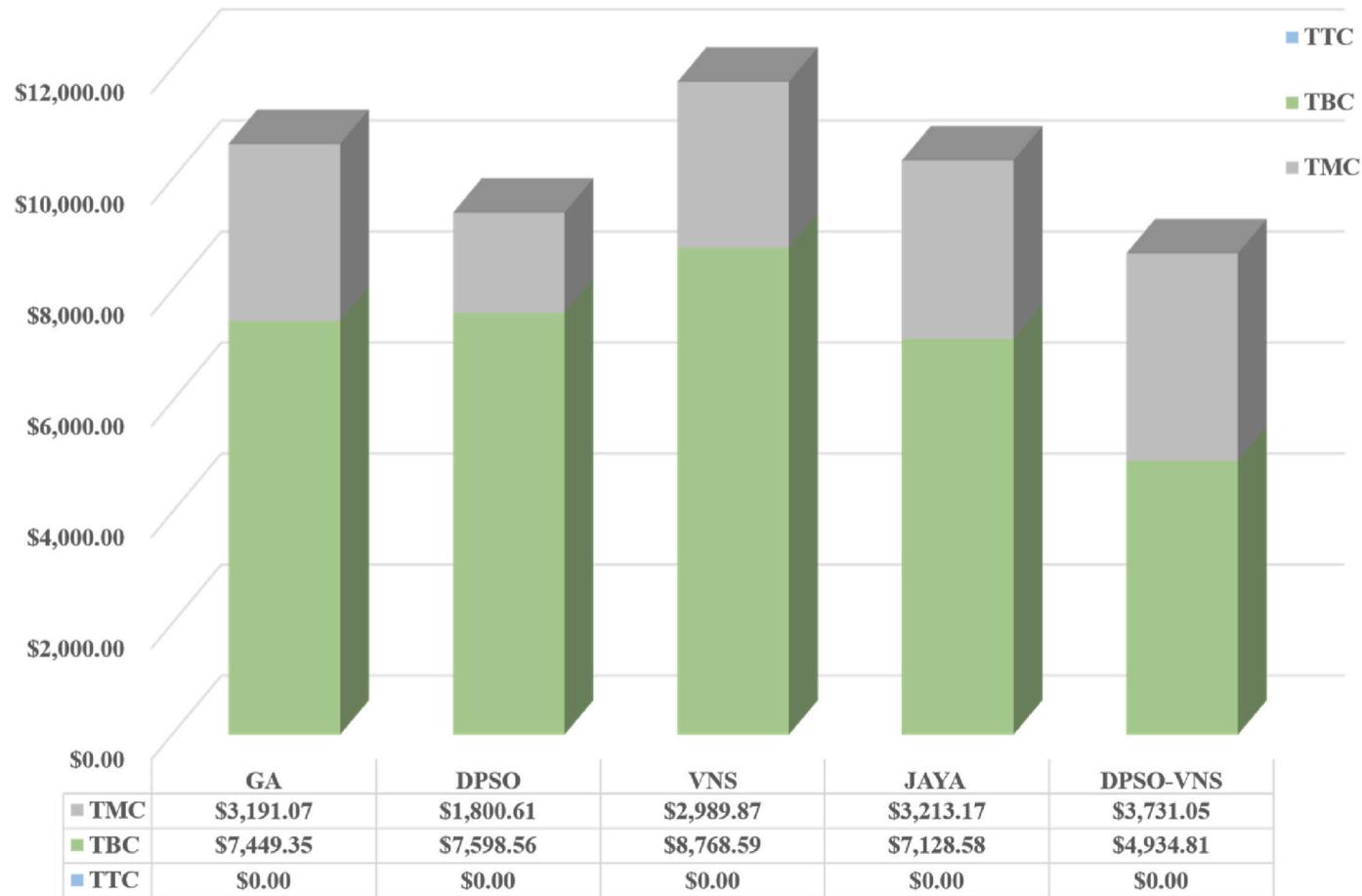


Fig. 14. Cost comparison of joint optimization schemes among five algorithms.

Table 5

Machine workloads based on five algorithms.

	GA	DPSO	VNS	JAYA	DPSO -VNS	GA	DPSO	VNS	JAYA	DPSO -VNS
M_1	72	62	55	52	73	M_{11}	57	29	36	47
M_2	50	72	54	55	63	M_{12}	57	38	56	29
M_3	53	80	57	75	56	M_{13}	34	38	40	61
M_4	72	72	89	57	64	M_{14}	61	54	46	65
M_5	69	48	46	68	63	M_{15}	22	54	19	22
M_6	50	49	49	47	59	Mean	52.6	53.3	53.9	53.5
M_7	67	68	70	70	51	Maximum	72	80	89	75
M_8	45	53	83	53	59	Minimum	22	29	19	22
M_9	47	50	60	51	55	Range	50	51	70	53
M_{10}	33	33	48	50	39	SD	14.899	15.197	17.537	14.257
										9.870

environments to achieve seamless machine operation and uninterrupted production continuity. Specifically, our method allows practitioners to set operating conditions (such as factors affecting machine failure rates X , corresponding coefficients β), maintenance parameters (e.g., parameters in Weibull distribution, maintenance cost, time duration, etc.), and other information according to their actual production conditions. Moreover, the design of a hybrid meta-heuristic algorithm helps practitioners to obtain economical joint optimization schemes for production and maintenance within an acceptable timeframe, thereby guiding the production process.

6.2. Limitations and future work

While this study makes significant contributions to the joint optimization of FJSSP-PM and has important implications for practice, it also possesses certain limitations. Firstly, we investigate the impact of

variable operating conditions resulting from high-frequency production switching on machine failure rate and maintenance requirements, assuming that each machine experiences independent degradation. However, in real industrial sites, machine failure can be significantly influenced by intricate failure dynamics, such as escalated degradation caused by defects generated during previous operations of jobs and the degradation of auxiliary resources (e.g., tool wear) (Bouslah et al., 2018). This mechanism engenders interdependence in the failure process among machines, thereby warranting further investigation of this interdependency within a flexible job shop environment. Specifically, emphasis should be placed on investigating the impact of both cumulative deterioration in job quality and degradation of key components on machine reliability. Secondly, the PHM model is employed to model the machine failure rate under variable operating conditions (e.g., width of cut, feed speed, ambient temperature, and cutting force). However, a limitation of this model is its disregard of historical operating conditions

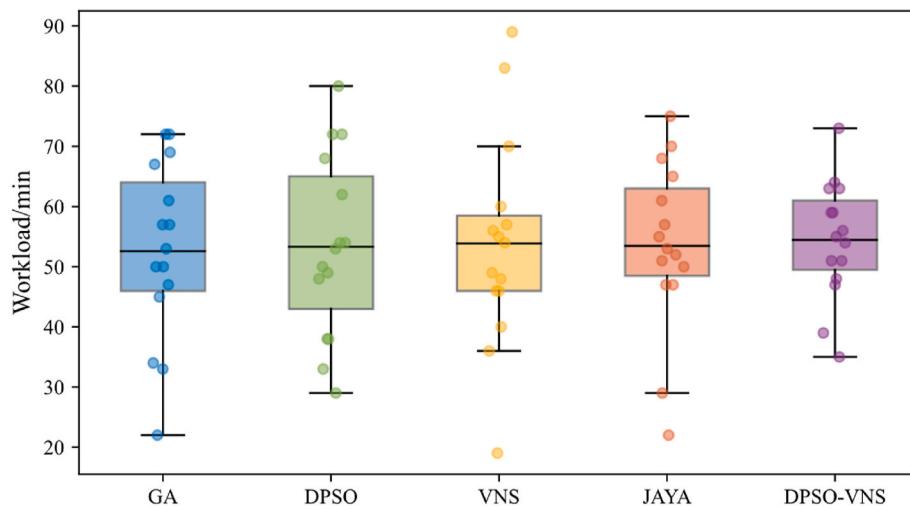


Fig. 15. Distribution of machine workloads based on five algorithms.

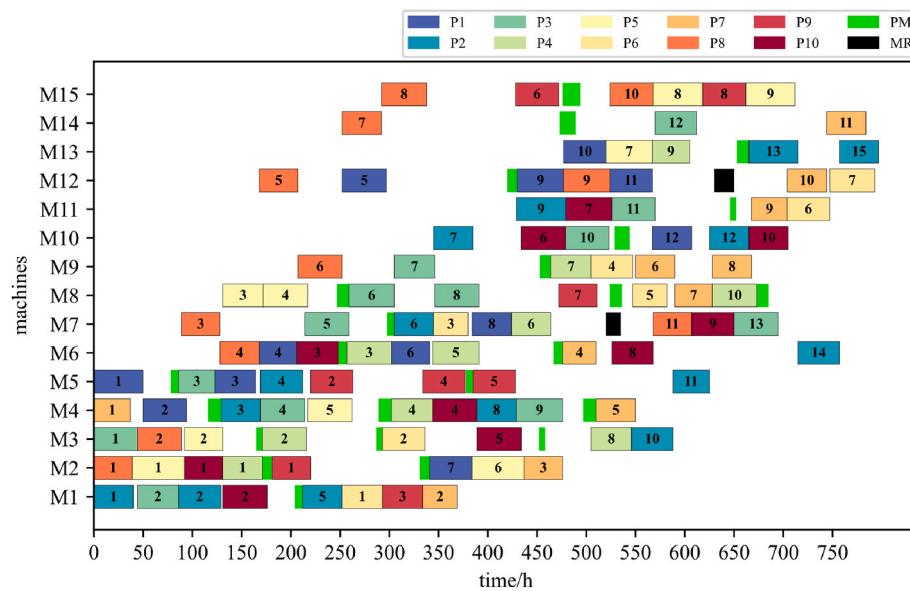


Fig. 16. Joint optimization scheme for a larger-sized computational example.

Table 6
Actual PM time points of machines.

	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}	M_{11}	M_{12}	M_{13}	M_{14}	M_{15}
$y_{1kz} = 1$	$z = 4$	$z = 4$	$z = 3$	$z = 2$	$z = 1$	$z = 3$	$z = 2$	$z = 2$	$z = 2$	$z = 3$	$z = 3$	$z = 2$	$z = 3$	$z = 2$	$z = 2$
t'_{1k}	204	171	165	116	78	248	298	247	453	529	646	420	653	473	476
$y_{2kz} = 1$	–	$z = 5$	$z = 4$	$z = 5$	$z = 6$	$z = 6$	–	$z = 5$	–	–	–	–	–	–	–
t'_{2k}	–	331	287	289	378	467	–	524	–	–	–	–	–	–	–
$y_{3kz} = 1$	–	–	$z = 6$	$z = 9$	–	–	–	$z = 8$	–	–	–	–	–	–	–
t'_{3k}	–	–	452	497	–	–	–	673	–	–	–	–	–	–	–

on the failure rate when these covariates are time-dependent. Future research endeavors could explore the potential application of the accelerated failure-time model (Hu et al., 2017; Zhu and Zhou, 2023) to address the pertinent issue highlighted in this paper. Thirdly, this study investigates the dynamic characteristics of production-dependent stochastic deterioration of machines in flexible production environments and achieves the joint optimization of production and maintenance activities economically and operationally. However, certain other design variables such as new job arrivals, flexible due dates, and variable processing times also render optimal solutions to the original "ideal"

problem impractical or less effective in real-world scenarios. Thus, considering the implementation of robust scheduling techniques (Xiong et al., 2013) or flexible constraints (Li et al., 2022) may provide more feasible solutions for time-varying market demands. Furthermore, an intriguing endeavor would be to incorporate the framework and algorithm proposed by Leoni et al. (2023) for capturing more realistic information about the occurrence of failures into our proposed model, thereby enhancing the efficient response of production scheduling and maintenance decisions in dynamic production environments.

CRediT authorship contribution statement

Yu Wang: Conceptualization, Methodology, Writing – original draft.
Tangbin Xia: Resources, Supervision, Writing – original draft.
Yuhui Xu: Data curation, Formal analysis.
Yutong Ding: Investigation, Software.
Meimei Zheng: Validation, Visualization.
Ershun Pan: Writing – review & editing.
Lifeng Xi: Project administration, Supervision.

Data availability

Data will be made available on request.

Appendix

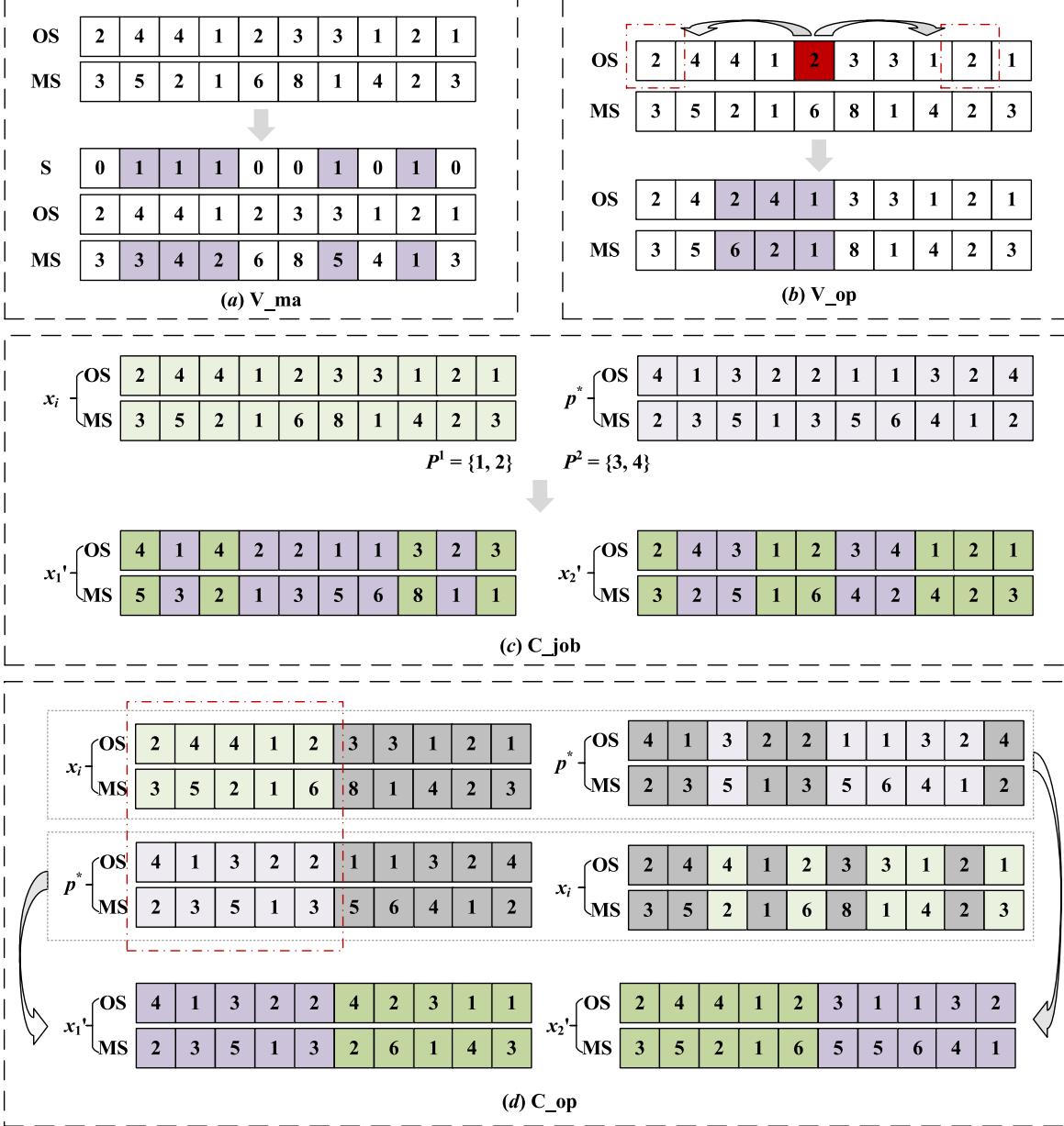


Fig. A1. Mutation and crossover operators

Table A1
Maintenance parameters of machines

	θ_k	η_k	T_k^p	C_k^p	T_k^f	C_k^f		θ_k	η_k	T_k^p	C_k^p	T_k^f	C_k^f
M_1	2.5	215	8 min	\$480	20 min	\$1010	M_9	2.4	200	11 min	\$540	28 min	\$1140
M_2	3.3	150	10 min	\$550	24 min	\$1300	M_{10}	2.2	250	15 min	\$500	28 min	\$1320
M_3	3.3	200	6 min	\$460	16 min	\$1200	M_{11}	2.6	320	6 min	\$510	21 min	\$1260
M_4	2.6	320	13 min	\$470	18 min	\$1150	M_{12}	2.6	160	10 min	\$560	25 min	\$1050
M_5	2.5	240	5 min	\$450	18 min	\$1220	M_{13}	2.4	290	12 min	\$480	26 min	\$1170
M_6	2.2	220	9 min	\$500	22 min	\$1240	M_{14}	2.7	360	16 min	\$490	29 min	\$1080
M_7	3.2	320	7 min	\$570	28 min	\$1190	M_{15}	2.2	220	18 min	\$510	30 min	\$1130
M_8	2.8	210	12 min	\$520	22 min	\$1230							

Table A2
Production information

Job	Operation	(Machine, process time/min)	Due date	Tardiness penalty	β
P_1	O_{11}	(1, 15); (2, 17); (3, 19); (4, 22)	240 min	\$480	[1.095, 0.479, 0.134, 0.754]
	O_{12}	(1, 10); (2, 8); (4, 9)			
	O_{13}	(5, 12); (6, 17); (7, 15)			
	O_{14}	(5, 8); (6, 12); (7, 14)			
	O_{15}	(8, 23); (9, 19)			
	O_{16}	(8, 11); (9, 15)			
	O_{17}	(8, 22); (9, 16); (11, 18)			
	O_{18}	(2, 13); (3, 12); (5, 17)			
	O_{19}	(4, 16); (7, 12)			
	O_{110}	(10, 9); (13, 12)			
	O_{111}	(11, 19); (12, 18); (15, 21)			
	O_{112}	(10, 15); (12, 14); (14, 17)			
	O_{113}	(13, 15); (14, 12); (15, 15)			
P_2	O_{21}	(1, 18); (3, 16); (4, 20)	200 min	\$460	[1.124, 0.658, 0.278, 1.006]
	O_{22}	(2, 26); (5, 24)			
	O_{23}	(3, 16); (4, 13); (6, 20)			
	O_{24}	(5, 9); (7, 10); (9, 8)			
	O_{25}	(1, 19); (6, 22)			
	O_{26}	(2, 15); (8, 13)			
	O_{27}	(8, 14); (9, 16)			
	O_{28}	(11, 18); (12, 15); (15, 22)			
	O_{29}	(10, 11); (13, 13)			
	O_{210}	(13, 15); (14, 12)			
	O_{211}	(14, 14); (15, 13)			
P_3	O_{31}	(1, 25); (2, 28); (4, 22)	200 min	\$500	[0.875, 1.232, 0.433, 0.986]
	O_{32}	(3, 14); (5, 17)			
	O_{33}	(6, 18); (7, 15); (8, 16)			
	O_{34}	(8, 20); (9, 19)			
	O_{35}	(11, 10); (13, 15); (14, 13)			
	O_{36}	(10, 15); (12, 18)			
	O_{37}	(14, 19); (15, 17)			
	O_{38}	(3, 13); (5, 12)			
P_4	O_{41}	(1, 12); (2, 10); (4, 13)	160 min	\$430	[0.878, 0.956, 0.354, 1.432]
	O_{42}	(2, 14); (3, 11)			
	O_{43}	(5, 8); (6, 9)			
	O_{44}	(8, 9); (9, 7)			
	O_{45}	(8, 10); (9, 9)			
	O_{46}	(7, 15); (10, 13); (11, 11)			
	O_{47}	(12, 11); (14, 12)			
	O_{48}	(5, 12); (7, 13)			
	O_{49}	(12, 9); (13, 10)			
	O_{410}	(13, 7); (15, 8)			
P_5	O_{51}	(2, 14); (3, 13); (4, 12)	150 min	\$450	[1.128, 0.452, 0.126, 0.364]
	O_{52}	(1, 10); (5, 12)			
	O_{53}	(3, 12); (5, 10)			
	O_{54}	(6, 9); (7, 12)			
	O_{55}	(8, 13); (10, 11)			
	O_{56}	(9, 15); (12, 14)			
	O_{57}	(7, 14); (8, 13); (11, 11)			
	O_{58}	(12, 8); (15, 10)			
	O_{59}	(13, 18); (14, 15)			
P_6	O_{61}	(1, 16); (2, 14); (4, 12)	180 min	\$420	[0.768, 0.982, 0.453, 1.057]
	O_{62}	(3, 20); (5, 18)			
	O_{63}	(4, 15); (6, 17); (7, 14)			
	O_{64}	(8, 12); (9, 13)			
	O_{65}	(10, 13); (11, 10); (12, 14)			
	O_{66}	(13, 17); (15, 19)			
	O_{67}	(14, 15); (15, 12)			

References

- An, Y., Chen, X., Gao, K., Zhang, L., Li, Y., Zhao, Z., 2023. A hybrid multi-objective evolutionary algorithm for solving an adaptive flexible job-shop rescheduling problem with real-time order acceptance and condition-based preventive maintenance. *Expert Syst. Appl.* 212 <https://doi.org/10.1016/j.eswa.2022.118711>.
- An, Y., Chen, X., Li, Y., Zhang, J., Jiang, J., 2021. Flexible job-shop scheduling and heterogeneous repairman assignment with maintenance time window and employee timetabling constraints. *Expert Syst. Appl.* 186 <https://doi.org/10.1016/j.eswa.2021.115693>.
- Bouslah, B., Gharbi, A., Pellerin, R., 2018. Joint production, quality and maintenance control of a two-machine line subject to operation-dependent and quality-dependent failures. *Int. J. Prod. Econ.* 195 <https://doi.org/10.1016/j.ijpe.2017.10.016>.
- Božek, A., Werner, F., 2018. Flexible job shop scheduling with lot streaming and subplot size optimisation. *Int. J. Prod. Res.* 56 <https://doi.org/10.1080/00207543.2017.1346322>.
- Brenière, L., Doyen, L., Bérenguer, C., 2020. Virtual age models with time-dependent covariates: a framework for simulation, parametric inference and quality of estimation. *Reliab. Eng. Syst. Saf.* 203 <https://doi.org/10.1016/j.ress.2020.107054>.
- Dauzère-Pérès, S., Ding, J., Shen, L., Tamssouet, K., 2023. The flexible job shop scheduling problem: a review. *Eur. J. Oper. Res.* <https://doi.org/10.1016/j.ejor.2023.05.017>.
- Defersha, F.M., Obimuyiwa, D., Yimer, A.D., 2022. Mathematical model and simulated annealing algorithm for setup operator constrained flexible job shop scheduling problem. *Comput. Ind. Eng.* 171 <https://doi.org/10.1016/j.cie.2022.108487>.
- Ding, H., Gu, X., 2020. Improved particle swarm optimization algorithm based novel encoding and decoding schemes for flexible job shop scheduling problem. *Comput. Oper. Res.* 121 <https://doi.org/10.1016/j.cor.2020.104951>.
- Framinan, J.M., Fernandez-Viegas, V., Perez-Gonzalez, P., 2019. Using real-time information to reschedule jobs in a flowshop with variable processing times. *Comput. Ind. Eng.* 129, 113–125. <https://doi.org/10.1016/j.cie.2019.01.036>.
- Garey, M.R., Johnson, D.S., Sethi, R., 1976. Complexity of flowshop and jobshop scheduling. *Math. Oper. Res.* 1 <https://doi.org/10.1287/moor.1.2.117>.
- Ghaleb, M., Taghipour, S., Zolfaghariania, H., 2021. Real-time integrated production-scheduling and maintenance-planning in a flexible job shop with machine deterioration and condition-based maintenance. *J. Manuf. Syst.* 61, 423–449. <https://doi.org/10.1016/j.jmssy.2021.09.018>.
- Ghaleb, M., Zolfaghariania, H., Taghipour, S., 2020. Real-time production scheduling in the Industry-4.0 context: addressing uncertainties in job arrivals and machine breakdowns. *Comput. Oper. Res.* 123 <https://doi.org/10.1016/j.cor.2020.105031>.
- Hou, Y., Hao, G., Zhang, Y., Gu, F., Xu, W., 2022. A multi-objective discrete particle swarm optimization method for particle routing in distributed particle filters. *Knowl. Base Syst.* 240 <https://doi.org/10.1016/j.knosys.2021.108068>.
- Hu, J., Jiang, Z., Liao, H., 2017. Preventive maintenance of a single machine system working under piecewise constant operating condition. *Reliab. Eng. Syst. Saf.* 168 <https://doi.org/10.1016/j.ress.2017.05.014>.
- Hu, W., Yang, Z., Chen, C., Wu, Y., Xie, Q., 2021. A Weibull-based recurrent regression model for repairable systems considering double effects of operation and maintenance: a case study of machine tools. *Reliab. Eng. Syst. Saf.* 213 <https://doi.org/10.1016/j.ress.2021.107669>.
- Johnson, S.M., 1954. Optimal two- and three-stage production schedules with setup times included. *Nav. Res. Logist. Q.* 1 <https://doi.org/10.1002/nav.3800010110>.
- Kundu, P., Darpe, A.K., Kulkarni, M.S., 2019. Weibull accelerated failure time regression model for remaining useful life prediction of bearing working under multiple operating conditions. *Mech. Syst. Signal Process.* 134 <https://doi.org/10.1016/j.ymssp.2019.106302>.
- Leoni, L., De Carlo, F., Tucci, M., 2023. Developing a framework for generating production-dependent failure rate through discrete-event simulation. *Int. J. Prod. Econ.* <https://doi.org/10.1016/j.ijpe.2023.109034>.
- Li, J.Q., Pan, Q.K., 2012. Chemical-reaction optimization for flexible job-shop scheduling problems with maintenance activity. *Applied Soft Computing Journal* 12, 2896–2912. <https://doi.org/10.1016/j.asoc.2012.04.012>.
- Li, R., Gong, W., Wang, L., Lu, C., Jiang, S., 2022. Two-stage knowledge-driven evolutionary algorithm for distributed green flexible job shop scheduling with type-2 fuzzy processing time. *Swarm Evol. Comput.* 74 <https://doi.org/10.1016/j.swevo.2022.101139>.
- Li, X., Gao, L., 2016. An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem. *Int. J. Prod. Econ.* 174, 93–110. <https://doi.org/10.1016/j.ijpe.2016.01.016>.
- Liu, L., 2019. Outsourcing and rescheduling for a two-machine flow shop with the disruption of new arriving jobs: a hybrid variable neighborhood search algorithm. *Comput. Ind. Eng.* 130, 198–221. <https://doi.org/10.1016/j.cie.2019.02.015>.
- Lu, X., Chen, M., Liu, M., Zhou, D., 2012. Optimal imperfect periodic preventive maintenance for systems in time-varying environments. *IEEE Trans. Reliab.* 61 <https://doi.org/10.1109/TR.2012.2182817>.
- Lv, Y., Li, C., Tang, Y., Kou, Y., 2022. Toward energy-efficient rescheduling decision mechanisms for flexible job shop with dynamic events and alternative process plans. *IEEE Trans. Autom. Sci. Eng.* 19, 3259–3275. <https://doi.org/10.1109/TASE.2021.3115821>.
- Mokhtari, H., Dadgar, M., 2015. Scheduling optimization of a stochastic flexible job-shop system with time-varying machine failure rate. *Comput. Oper. Res.* 61, 31–45. <https://doi.org/10.1016/j.cor.2015.02.014>.
- Moradi, E., Fatemi Ghomi, S.M.T., Zandieh, M., 2011. Bi-objective optimization research on integrated fixed time interval preventive maintenance and production for scheduling flexible job-shop problem. *Expert Syst. Appl.* 38 <https://doi.org/10.1016/j.eswa.2010.12.043>.
- Park, M.J., Ham, A., 2022. Energy-aware flexible job shop scheduling under time-of-use pricing. *Int. J. Prod. Econ.* 248 <https://doi.org/10.1016/j.ijpe.2022.108507>.
- Rossi, A., 2014. Flexible job shop scheduling with sequence-dependent setup and transportation times by ant colony with reinforced pheromone relationships. *Int. J. Prod. Econ.* 153, 253–267. <https://doi.org/10.1016/j.ijpe.2014.03.006>.
- Shen, L., Dauzère-Pérès, S., Neufeld, J.S., 2018. Solving the flexible job shop scheduling problem with sequence-dependent setup times. *Eur. J. Oper. Res.* 265 <https://doi.org/10.1016/j.ejor.2017.08.021>.
- Tamssouet, K., Dauzère-Pérès, S., 2023. A general efficient neighborhood structure framework for the job-shop and flexible job-shop scheduling problems. *Eur. J. Oper. Res.* 311 <https://doi.org/10.1016/j.ejor.2023.05.018>.
- Vieira, G.E., Herrmann, J.W., Lin, E., 2003. Rescheduling manufacturing systems: a framework of strategies, policies, and methods. *J. Sched.* 6, 39–62.
- Wang, D., Yin, Y., Cheng, T.C.E., 2018. Parallel-machine rescheduling with job unavailability and rejection. *Omega* 81, 246–260. <https://doi.org/10.1016/j.omega.2018.04.008>.
- Wang, L., Lu, Z., Han, X., 2019. Joint optimisation of production, maintenance and quality for batch production system subject to varying operational conditions. *Int. J. Prod. Res.* 57, 7552–7566. <https://doi.org/10.1080/00207543.2019.1581956>.
- Wang, S., Yu, J., 2010. An effective heuristic for flexible job-shop scheduling problem with maintenance activities. *Comput. Ind. Eng.* 59 <https://doi.org/10.1016/j.cie.2010.05.016>.
- Wein, L.M., Chevalier, P.B., 1992. Broader view of the job shop scheduling problem. *Manag. Sci.* 38 <https://doi.org/10.1287/mnsc.38.7.1018>.
- Xia, T., Jin, X., Xi, L., Ni, J., 2015. Production-driven opportunistic maintenance for batch production based on MAM-APB scheduling. *Eur. J. Oper. Res.* 240, 781–790. <https://doi.org/10.1016/j.ejor.2014.08.004>.
- Xia, T., Xi, L., Pan, E., Ni, J., 2017. Reconfiguration-oriented opportunistic maintenance policy for reconfigurable manufacturing systems. *Reliab. Eng. Syst. Saf.* 166, 87–98. <https://doi.org/10.1016/j.ress.2016.09.001>.
- Xiong, H., Shi, S., Ren, D., Hu, J., 2022. A survey of job shop scheduling problem: the types and models. *Comput. Oper. Res.* <https://doi.org/10.1016/j.cor.2022.105731>.
- Xiong, J., Xing, L.N., Chen, Y.W., 2013. Robust scheduling for multi-objective flexible job-shop problems with random machine breakdowns. *Int. J. Prod. Econ.* 141, 112–126. <https://doi.org/10.1016/j.ijpe.2012.04.015>.
- Yang, D., Wu, M., Li, D., Xu, Y., Zhou, X., Yang, Z., 2022. Dynamic opposite learning enhanced dragonfly algorithm for solving large-scale flexible job shop scheduling problem. *Knowl. Base Syst.* 238 <https://doi.org/10.1016/j.knosys.2021.107815>.
- Zandieh, M., Khatami, A.R., Rahmati, S.H.A., 2017. Flexible job shop scheduling under condition-based maintenance: improved version of imperialist competitive algorithm. *Applied Soft Computing Journal* 58. <https://doi.org/10.1016/j.asoc.2017.04.060>.
- Zhang, J., Ding, G., Zou, Y., Qin, S., Fu, J., 2019. Review of job shop scheduling research and its new perspectives under Industry 4.0. *J. Intell. Manuf.* 30 <https://doi.org/10.1007/s10845-017-1350-2>.
- Zhou, Q., Son, J., Zhou, S., Mao, X., Salman, M., 2014. Remaining useful life prediction of individual units subject to hard failure. *IIE Trans.* 46, 1017–1030. <https://doi.org/10.1080/0740817X.2013.876126>.
- Zhou, W., Chen, F., Ji, X., Li, H., Zhou, J., 2022. A Pareto-based discrete particle swarm optimization for parallel casting workshop scheduling problem with fuzzy processing time. In: *Knowledge-Based Systems*. Elsevier B.V. <https://doi.org/10.1016/j.knosys.2022.109872>.
- Zhu, M., Zhou, X., 2023. Hierarchical-clustering-based joint optimization of spare part provision and maintenance scheduling for serial-parallel multi-station manufacturing systems. *Int. J. Prod. Econ.* 264 <https://doi.org/10.1016/j.ijpe.2023.108971>.
- Zribi, N., El Kamel, A., Borne, P., 2008. Minimizing the makespan for the MPM job-shop with availability constraints. *Int. J. Prod. Econ.* 112 <https://doi.org/10.1016/j.ijpe.2007.01.014>.