frontiers | Frontiers in Industrial Engineering

TYPE Review
PUBLISHED 27 January 2025
DOI 10.3389/fieng.2025.1540022

Check for updates

OPEN ACCESS

EDITED BY
Dimitris Zissis,
Athens University of Economics and Business, Greece

REVIEWED BY
Aleksandar Trifunović,
University of Belgrade, Serbia
Ioannis Avgerinos,
Athens University of Economics and Business, Greece

*CORRESPONDENCE
Wenqiang Zhang,
zhangwq@haut.edu.cn
Xinchang Hao,
haoxc@outlook.com

RECEIVED 05 December 2024
ACCEPTED 02 January 2025
PUBLISHED 27 January 2025

CITATION
Zhang W, Bao X, Hao X and Gen M (2025) Metaheuristics for multi-objective scheduling problems in industry 4.0 and 5.0: a state-of-the-arts survey.
Front. Ind. Eng. 3:1540022.
doi: 10.3389/fieng.2025.1540022

Metaheuristics for multi-objective scheduling problems in industry 4.0 and 5.0: a state-of-the-arts survey

Wenqiang Zhang[1]*, Xuan Bao[2], Xinchang Hao[3]* and Mitsuo Gen[4]

[1]School of Artificial Intelligence and Big Data, Henan University of Technology, Zhengzhou, Henan, China, [2]College of Information Science and Engineering, Henan University of Technology, Zhengzhou, Henan, China, [3]School of Art and Design, Changzhou Institute of Technology, Changzhou, Jiangsu, China, [4]Research Institute of Science and Technology, Tokyo University of Science, Tokyo, Japan

The advent of Industry 4.0 and the emerging Industry 5.0 have fundamentally transformed manufacturing systems, introducing unprecedented levels of complexity in production scheduling. This complexity is further amplified by the integration of cyber-physical systems, Internet of Things, Artificial Intelligence, and human-centric approaches, necessitating more sophisticated optimization methods. This paper aims to provide a more comprehensive perspective on the application of metaheuristic algorithms in shop scheduling problems within the context of Industry 4.0 and Industry 5.0. Through a systematic review of recent literature (2015–2024), we analyze and categorize various metaheuristic approaches, including Evolutionary Algorithms (EAs), swarm intelligence, and hybrid methods, that have been applied to address complex scheduling challenges in smart manufacturing environments. We specifically examine how these algorithms handle multiple competing objectives such as makespan minimization, energy efficiency, production costs, and human-machine collaboration, which are crucial in modern industrial settings. Our survey reveals several key findings: 1) hybrid metaheuristics demonstrate superior performance in handling multi-objective optimization compared to standalone algorithms; 2) bio-inspired algorithms show promising results in addressing complex scheduling and multi-objective manufacturing environments; 3) tri-objective and higher-order multi-objective optimization problems warrant further in-depth exploration; and 4) there is an emerging trend towards incorporating human factors and sustainability objectives in scheduling optimization, aligned with Industry 5.0 principles. Additionally, we identify research gaps and propose future research directions, particularly in areas such as real-time scheduling adaptation, human-centric optimization, and sustainability-aware scheduling algorithms. This comprehensive review provides insights for researchers and practitioners in the field of industrial scheduling, offering a structured understanding of current methodologies and future challenges in the evolution from Industry 4.0 to 5.0.

KEYWORDS

metaheuristics, multi-objective optimization, scheduling problems, Industry 4.0, Industry 5.0, smart manufacturing, human-centric manufacturing

1 Introduction

The manufacturing industry is undergoing a profound transformation, characterized by rapidly expanding production scales and intensifying global competition. Traditional manufacturing approaches, heavily dependent on manual labor, are becoming increasingly inadequate in meeting the demands of modern industrial production. In response to these challenges, the emergence of cutting-edge technologies, including the Internet of Things (IoT), Artificial Intelligence (AI), and advanced automation systems, has provided unprecedented opportunities for manufacturing transformation (Mourtzis, 2020a; Mourtzis, 2022). These technological innovations have not only revolutionized production processes through enhanced automation and intelligence but have also optimized resource allocation, reduced operational costs, and significantly improved both efficiency and quality of products.

Shop scheduling represents a fundamental challenge in manufacturing systems and has been extensively studied in the literature. The core objective of scheduling is to optimally allocate limited production resources (e.g., machines, workers, materials) to various tasks while satisfying multiple constraints and objectives. While traditional scheduling approaches, based on human expertise and simplified mathematical models, have served well historically, they prove inadequate in addressing the complexity and dynamism of contemporary manufacturing environments characterized by volatile demand patterns and high uncertainty. Consequently, the integration of advanced information technologies and intelligent algorithms for scheduling optimization has emerged as a critical research focus.

In the evolution of industrial paradigms, Industry 4.0 emphasizes the comprehensive integration of information and communication technologies with cyberspace virtual systems, encompassing IoT, Cyber-Physical Systems, smart factories, and cloud computing, to enhance manufacturing automation and intelligence (Mourtzis, 2020b; Jain et al., 2022). Within the Industry 4.0 framework, shop scheduling faces both new opportunities and challenges (Ferreira et al., 2023). Data-driven scheduling optimization, intelligent decision support, and flexible production models have transformed production processes into more efficient and controllable systems. The vast amount of production data collected through IoT infrastructure enables the development of more accurate scheduling models. Furthermore, the integration of big data analytics and machine learning algorithms facilitates data-driven optimization, while smart factories and autonomous decision-making systems enable real-time production adjustments and anomaly detection, significantly enhancing system adaptability and efficiency.

With the continued advancement of Industry 4.0, it has become increasingly evident that, while technological progress has significantly enhanced productivity, achieving more comprehensive and sustainable industrial development requires greater consideration of human factors, environmental sustainability, and system resilience. Consequently, Industry 5.0 has emerged as an extension of Industry 4.0, adopting a more holistic approach that emphasizes human-centricity, sustainability, and resilience. This shift represents not merely a technological upgrade but a profound reevaluation and transformation of industrial production paradigms, aiming to establish a more harmonious, green, and flexible industrial ecosystem.

In this context, shop scheduling must evolve beyond technical optimization to incorporate human factors and social responsibility (Destouet et al., 2023). The human-centric manufacturing paradigm prioritizes worker wellbeing and satisfaction through smart wearable devices and health monitoring systems. Sustainable production practices focus on minimizing resource consumption and environmental impact through green manufacturing and circular economy principles. Additionally, resilient production systems are designed to withstand disruptions through flexible production networks and robust supply chain architectures. These emerging requirements have significantly increased the complexity of scheduling problems, necessitating the consideration of multiple objectives which often competing.

Metaheuristic algorithms have emerged as powerful tools for addressing these complex multi-objective scheduling challenges. These algorithms demonstrate remarkable effectiveness in handling large-scale problems with multiple competing objectives, offering flexible and efficient solutions that can adapt to varying production scenarios. Their success in solving complex and dynamic shop scheduling problems, coupled with their ability to balance multiple objectives across different production scales, has led to their increased adoption in manufacturing applications. Moreover, the integration of metaheuristics with Industry 4.0 and 5.0 technologies has opened new avenues for scheduling optimization, enabling more sophisticated approaches to improving production efficiency and quality. Compared to previous studies, this paper aims to provide a more comprehensive perspective on the application of metaheuristic algorithms in shop scheduling problems within the context of Industry 4.0 and Industry 5.0, making the following contributions:

(1) This paper focuses on the application of metaheuristic algorithms in multi-objective scheduling problems from 2015 to 2024, systematically evaluating the advantages and disadvantages of single algorithms and hybrid algorithms found in the existing literature.
(2) By conducting an in-depth analysis of the number of objectives across various studies, this paper explores scheduling goals in the context of Industry 5.0 and Industry 4.0, distilling the current scale of research objectives and the trends for future research goals.
(3) This paper provides an overview of the research progress on metaheuristic algorithms combined with Reinforcement Learning (RL) methods in scheduling problems, assesses the advantages of these combined approaches, and suggests potential directions for future research.
(4) Addressing the classification of scheduling problems, this paper summarizes and examines the current trends in scheduling issues from the perspectives of flow-shop and job-shop scheduling problems.

This comprehensive survey examines the application of metaheuristic algorithms in multi-objective shop scheduling within the Industry 4.0 and 5.0 frameworks. The paper is structured as follows: Section 2 provides an overview of multi-objective optimization concepts and methodologies. Section 3 presents a systematic classification of scheduling objectives in modern manufacturing environments. Sections 4, 5 analyze the application of metaheuristics in flow-shop and job-shop scheduling problems, respectively, including their variants. Section 6 explores the integration of metaheuristics with RL and other advanced techniques. Finally, Section 7 summarizes key findings and identifies promising directions for future research.

2 Multi-objective optimization

In contemporary manufacturing processes, multiple objectives often conflict with each other, such as product quality versus production cost, or production efficiency versus energy consumption. The complexity of modern manufacturing systems, particularly in the context of Industry 4.0 and 5.0, necessitates simultaneous optimization of multiple objectives, making multi-objective optimization a crucial research focus.

2.1 The essence of multi-objective optimization

Multi-objective optimization problems are characterized by the presence of multiple, often conflicting objectives where the optimization of one objective typically results in the degradation of others. This inherent trade-off necessitates finding solutions that achieve an acceptable balance among all objectives. The mathematical formulation of a Multi-objective Optimization Problem (MOP) can be expressed as follows:

Definition 1. MOP (Coello, 2007) (Equation 1):

$$\min \quad F(\boldsymbol{x}) = \left(f_1(\boldsymbol{x}), \dots, f_m(\boldsymbol{x})\right)^{\mathrm{T}}, \boldsymbol{x} \in \Omega \tag{1}$$

where $\boldsymbol{x} = (x_1, x_2, \dots, x_n)^{\mathrm{T}}$ represents the decision variable vector in n-dimensional space, Ω denotes the feasible solution space, and m represents the number of objective functions. $F(\boldsymbol{x})$ maps the decision space to the objective space through m objective functions.

Definition 2. Pareto Dominance Relationship:

For two solutions $\boldsymbol{x}$ and $\boldsymbol{y}$ of MOP, $\boldsymbol{x}$ dominates $\boldsymbol{y}$ (denoted as $\boldsymbol{x} \prec \boldsymbol{y}$) if and only if $\forall i \in 1, 2, \dots, m, f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{y})$ and $\exists j \in 1, 2, \dots, m, \text{s.t.} f_j(\boldsymbol{y}) < f_j(\boldsymbol{y})$. Conversely, if $\boldsymbol{y}$ dominates $\boldsymbol{x}$ (denoted as $\boldsymbol{x} \succ \boldsymbol{y}$). In this context, $\boldsymbol{x}$ is termed a non-dominated solution and $\boldsymbol{y}$ a dominated solution relative to $\boldsymbol{x}$. The collection of all non-dominated solutions forms the Non-dominated Set.

Definition 3. Pareto Optimal Solution (POS):

A solution **x** of the MOP is Pareto optimal if $\exists \mathbf{y} \in \Omega$, s.t. $\quad y < x$. Such solutions represent the best possible trade-offs among the multiple objectives.

In the context of Multi-objective Evolutionary Algorithms (MOEA), the goal is to identify a set of Pareto optimal solutions, formally defined as (Equation 2):

$$POS = \{\boldsymbol{x} \in \Omega | \nexists \boldsymbol{y} \in \Omega, \boldsymbol{y} < \boldsymbol{x}\} \tag{2}$$

Definition 4. Pareto-optimal Front (POF) (Equation 3):

$$POF = \{F(\boldsymbol{x}) | \boldsymbol{x} \in POS\} \tag{3}$$

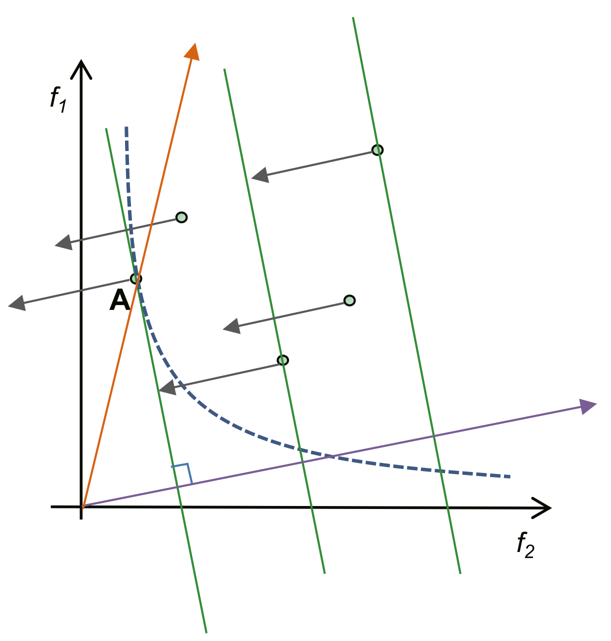

FIGURE 1
Weighted sum aggregation approach contours.

Depending on the evolutionary mechanism, MOEA can be divided into three categories: decomposition-based MOEA, dominance relationship-based MOEA, and indicator-based MOEA.

2.1.1 Decomposition based multi-objective evolutionary algorithm

The core idea of the decomposition-based MOEA algorithm is to decompose a multi-objective problem into a set of single-objective optimization problems, and approximate the solution set of the multi-objective problem by solving the subproblems simultaneously. Among them, the commonly used decomposition methods include the following three: weighted sum approach, Tchebycheff approach and penalty-based boundary intersection approach.

The weighted sum approach was proposed by Hillermeier (2001) and it is a commonly used linear multi-objective aggregation method. The aggregation function of this method can be expressed as (Equation 4):

$$\min g^{ws}(\boldsymbol{x}|\lambda) = \sum_{i=1}^{m} \lambda_i f_i(\boldsymbol{x}) \tag{4}$$

where $\boldsymbol{x} \in \Omega$ is the decision vector, and $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \dots, \lambda_m)^{\mathrm{T}}$ is the weight vector satisfying $\lambda_i \geqslant 0, \quad i = 1, \dots, m$ and $\sum_{i=1}^{m} \lambda_i = 1$.

As shown in Figure 1, take two objective values as an example, when the real Pareto front is convex, purple is the weight vector, the green solid line perpendicular to the purple weight vector is the contour line. Connecting the objective point and the origin in the objective space constitutes a vector, and the purpose of the aggregation function is to multiply this vector with the pair of weight vectors point by point. When the weight vector is unchanged, minimizing/maximizing the length value is optimizing the vector.

Tchebycheff approach is a nonlinear multi-objective aggregation method, proposed by the aggregation function is defined as (Equation 5):

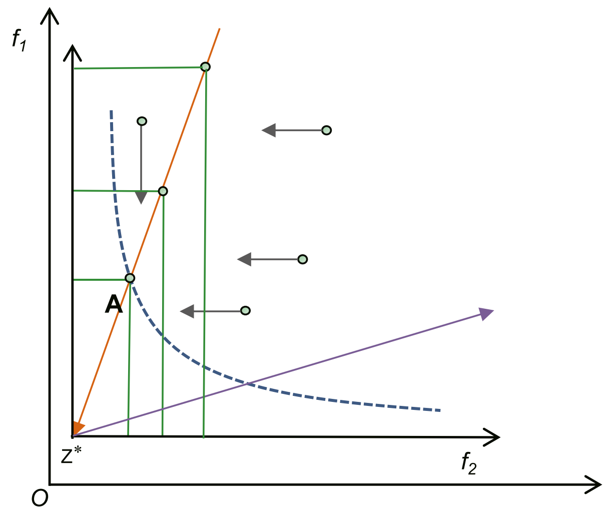

FIGURE 2
Tchebycheff approach contours.

$$\text{minimize}\quad g^{te}(\boldsymbol{x}|\boldsymbol{\lambda},\boldsymbol{z}^{*}) = \max\{\boldsymbol{\lambda}|f_i(\boldsymbol{x}) - z_i^{*}|\}, \boldsymbol{x} \in \Omega \tag{5}$$

The Tchebycheff approach continuously forces individuals to approach a predetermined ideal point when transforming a multi-objective problem into a single-objective optimization by means of an aggregation function, which ultimately results in a Pareto optimal solution under the constraints. As shown in Figure 2, by definition the contour of each function in this method is a straight line parallel to that function, respectively; moreover, the intersection of two objective contours lies in the direction of the weight vector. In order to solve the problem that the solutions obtained by the standard Tchebycheff approach are not homogeneous, Qi et al. (2014) proposed the conversion of the reference vector into $\boldsymbol{\lambda}^{*} = \left(\frac{\frac{1}{\lambda_1}}{\sum_{\mathrm{i}=1}^{\mathrm{m}}\frac{1}{\lambda_c}}, \ldots, \frac{\frac{1}{\lambda_{\mathrm{m}}}}{\sum_{\mathrm{i}=1}^{\mathrm{m}}\frac{1}{\lambda_c}}\right)$, which yields uniformly distributed solutions.

The penalty-based boundary intersection approach was proposed by Zhang and Li (2007) which combines the Tchebycheff method and the penalty mechanism, which allows the solution to deviate from the direction of the weight vector, but a penalty will be imposed on the deviated solution. During the process of multi-objective optimization, the convergence and distribution of the solution can be controlled by adjusting the penalty parameter, so that the solution evolves directly along the direction of the weight vector, which is defined as follows:

$$\begin{aligned} g^{pbi}(\boldsymbol{x}|\boldsymbol{\lambda},\boldsymbol{z}^{*}) &= d_1 + \theta d_2 \\ d_1 &= \frac{\|(F(\boldsymbol{x}) - \boldsymbol{z}^{*})^{\mathrm{T}}\boldsymbol{\lambda}\|}{\|\boldsymbol{\lambda}\|} \\ d_2 &= \|F(\boldsymbol{x}) - \left(\boldsymbol{z}^{*} + d_1\frac{\boldsymbol{\lambda}}{\|\boldsymbol{\lambda}\|}\right)\| \end{aligned} \tag{6}$$

Equation 6 give the calculation of the method. Where θ is a custom parameter and $\theta > 0$, which is used to control the magnitude of the weights d_1 and d_2. In general d_1 and d_2 control the convergence and distributivity of the solution, respectively, and the larger θ is, the method focuses on d_1. In other words, it tends to be distributed. Zhang and Li (2007) based on the idea of decomposition, they combine mathematical planning methods and Evolutionary Algorithms (EAs), convert a multi-objective optimization problem into a set of single-objective optimization problems, and seek the global optimal solution through the co-evolution of multiple sub-problems.

2.1.2 Pareto dominance based multi-objective evolutionary algorithm

Schaffer extended the simple Genetic Algorithm (GA) and proposed the Vector Evaluation Genetic Algorithm (Schaffer, 1985) (VEGA) in 1985, but the method cannot make trade-offs according to the attributes of each sub-objective, and it can only find the extreme points on the optimality boundary. In addition, the non-dominated solutions produced by VEGA are not necessarily globally non-dominated individuals. Fonseca and Fleming made use of the idea of sorting and proposed a kind of Multi-Objective Genetic Algorithm (MOGA) (Fonseca and Fleming, 1993), which determine the classification order of individuals based on the dominance relationship between them. Subsequently, Goldberg et al. (1993) proposed a dominance relationship-based MOEA method, which mainly uses a Pareto-based fitness allocation strategy to find all non-dominated individuals in an evolving population.

Currently, there is a great number of MOEA problems that use dominance relationship methods. The Non-dominated Sorting Genetic Algorithm (NSGA) proposed by Srinivas and Deb (1994) uses Goldberg's idea directly. This algorithm is an improvement of the selection of the next-generation of sub-populations method based on GA, which stratifies each individual according to their dominance and non-dominance relationship before screening, which in turn makes the algorithm get the satisfactory results as possible on multiple objectives. Non-dominated Sorting Genetic Algorithm-II (NSGA-II) proposed by Deb et al. (2002), they uses fast non-dominated sorting on the basis of NSGA, which reduces the complexity of the algorithm and improves the running speed of the algorithm. Subsequently Jain and Deb (2014) proposed Non-dominated Sorting Genetic Algorithm-III (NSGA-III), which uses reference point based non-dominated sorting based on NSGA-II, this approach helps to identify the non-dominated solutions better in high dimensional objective space, NSGA-III is more suitable for the case where the number of objectives to be optimized is large. Zhang et al. (2014) employed a Pareto dominance-based strategy, proposed a hybrid sampling approach that combines VEGA with a fitness function grounded in novel Pareto dominance relationships, to balance the distribution and convergence of solutions. In addition, Gen et al. (2008) provided a comprehensive analysis of the advantages and applications of the GA. Figure 3 illustrates the development of MOEA based on dominance relationships.

In the MOEA based on dominance relationship, the evolutionary population is divided into several layers according to the dominance relationship. The first layer is the set of non-dominated individuals of the evolutionary population, the second layer is the set of non-dominated individuals after removing the first layer of individuals in the evolutionary population, the third layer is the set of non-dominated individuals after removing the first and second layer of individuals in the evolutionary population, and so on. When making the selection, the first layer of the non-dominated set is considered first, and individuals are selected from the first layer according to a certain strategy, and then individuals are considered to be selected sequentially in the other layers until the size

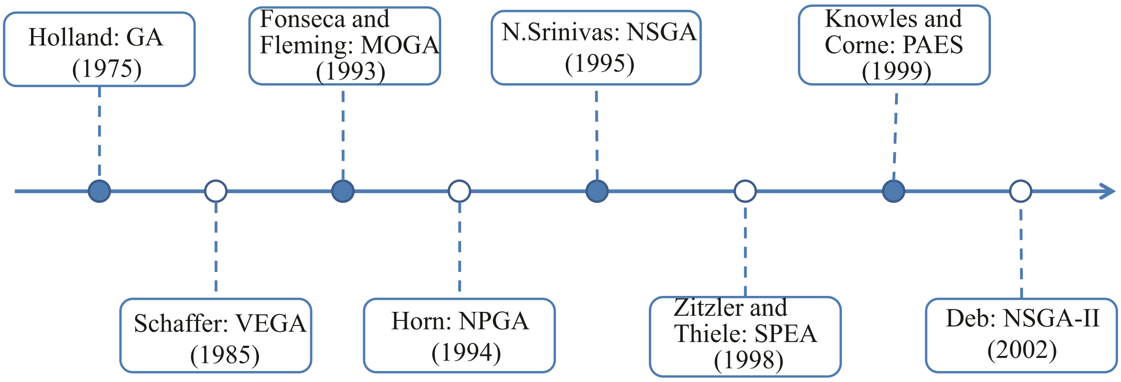

FIGURE 3
Development of MOEA based on dominance relations.

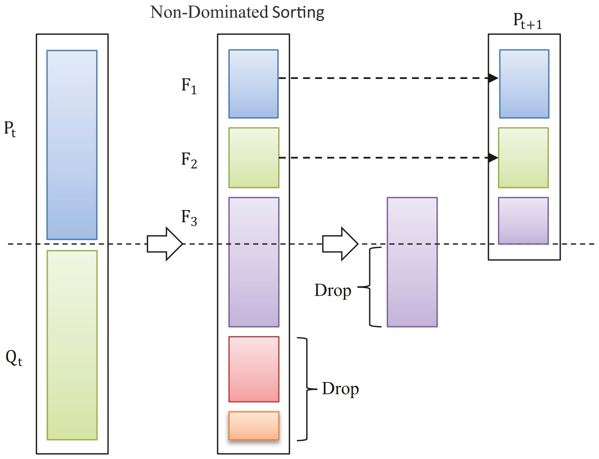

FIGURE 4
Development of MOEA based on dominance relations.

requirement of the new evolutionary population is met. Figure 4 illustrates this process in detail.

2.1.3 Indicator based multi-objective evolutionary algorithm

Indicator based MOEA refers to the use of performance evaluation indictor in optimization algorithms to guide the search selection process of solutions. Based on the decision-making approach, MOEA can be classified into three categories: pre-decision techniques, interactive decision techniques and post-decision techniques.

Pre-decision techniques are those in which the decision maker has specified the preferences and objective weights of the problem before the optimization process begin. These preferences and weights are fixed during the optimization process and the algorithm searches based on these fixed preferences and weights. The advantage of pre-decision techniques is their simplicity and clarity, but the disadvantage is that some potential high-quality solutions may be ignored. The common pre-decision techniques are weighted sum method, ε constraint method and objective planning method.

Interactive decision-making techniques are those in which the decision maker interacts with the algorithm during the optimization process, adjusting preferences and objective weights step by step. This technique allows the decision maker to dynamically adjust the objectives during the optimization process to find a solution that better meets the practical requirements. Common interactive decision-making techniques are: stepwise preference adjustment, interactive multi-objective optimization and dynamic preference adjustment.

The post-decision technique means that at the end of the optimization process, the algorithm generates a set of Pareto optimal solutions from which the decision maker chooses the most satisfactory solution. The advantage of this technique is that the decision maker can make a decision after seeing all possible solutions, but the disadvantage is that it may need to deal with a large set of solutions. Some common post-decision techniques are: non-dominated sorting method, MOGA, Multi-Objective Particle Swarm Optimization (MOPSO) algorithm, Multi-Objective Differential Evolutionary algorithm.

Pre-decision techniques optimize with fixed objective weights and constraints, with the advantage of simplicity and clarity and ease of implementation, but may ignore potential high-quality solutions. Interactive decision-making techniques dynamically adjust preferences during the optimization process by providing real-time feedback and adjusting objective weights, which improves flexibility and adaptability, but may increase time costs. Post-decision techniques generate a set of Pareto optimal solutions after optimization, from which the most satisfactory solutions are selected, ensuring diversity and quality of solutions, but may need to deal with a large set of solutions, increasing the difficulty of decision-making. Depending on the size of the problem, these techniques can be applied in different methods.

3 Classification of scheduling optimization objectives

The selection of optimization objectives in shop scheduling problems fundamentally determines the effectiveness and practical applicability of scheduling solutions. This section presents a systematic classification and analysis of common scheduling objectives, which can be formalized as follows:

3.1 Time-based objectives

Makespan Minimization: The most fundamental objective in shop scheduling, defined as (Equation 7):

$$C_{\max} = \min\left\{\max_{i\in\mathcal{J}} C_i\right\} \tag{7}$$

where C_i represents the completion time of job i.

Total Weighted Completion Time: Considers job priorities through weights (Equation 8):

$$\sum_{i=1}^{n} w_i C_i \rightarrow \min \tag{8}$$

where w_i represents the weight of job i.

3.2 Resource-based objectives

Energy Consumption: Modern scheduling increasingly emphasizes energy efficiency (Equation 9):

$$E_{\text{total}} = \sum_{i=1}^{n}\sum_{j=1}^{m}\left(E_{\text{proc}}^{ij} + E_{\text{idle}}^{ij} + E_{\text{setup}}^{ij}\right) \tag{9}$$

where: E_{proc}^{ij} represents processing energy for job i on machine j, E_{idle}^{ij} represents idle energy consumption, and E_{setup}^{ij} represents setup energy requirements.

Worker-related Objectives: Incorporating human factors (Equation 10):

$$\begin{aligned} W_{\text{balance}} &= \max_{k\in\mathcal{W}}\{|L_k - \bar{L}|\} \rightarrow \min \\ W_{\text{idle}} &= \sum_{k\in\mathcal{W}} T_{\text{idle}}^{k} \rightarrow \min \end{aligned} \tag{10}$$

where L_k represents workload of worker k, and T_{idle}^{k} is idle time.

3.3 Quality-based objectives

Tardiness-related (Equation 11):

$$\sum_{i=1}^{n} w_i \max\{0, C_i - d_i\} \rightarrow \min \tag{11}$$

where d_i represents the due date of job i.

Fuzzy Objectives: For uncertain environments (Equation 12):

$$\tilde{C}_{\max} = (\underline{c}, c, \bar{c}) \tag{12}$$

where $(\underline{c}, c, \bar{c})$ represents triangular fuzzy numbers.

Analysis of the literature reveals several key trends in scheduling optimization objectives:

(1) Objective Prevalence: makespan minimization and energy consumption remain the most frequently studied objectives, reflecting their fundamental importance in production efficiency and sustainability.
(2) Multi-objective Complexity: the majority of studies focus on bi-objective optimization, with fewer addressing three or more objectives simultaneously. This trend reflects the inherent complexity of handling multiple competing objectives.
(3) Emerging Objectives: recent research shows increasing attention to: sustainability metrics (energy, emissions) and human factors (workload, fatigue).
(4) Integration Trends: modern studies increasingly combine traditional time-based objectives with resource and quality considerations, reflecting the complex requirements of contemporary manufacturing environments.

Table 1 summarise the studies related to each objective in detail. This classification demonstrates the evolution of scheduling objectives from simple time-based metrics to comprehensive multi-dimensional optimization problems that better reflect real world manufacturing challenges.

4 Metaheuristic algorithm in flow-shop scheduling problems

Metaheuristic optimization algorithms have emerged as powerful tools for solving complex global optimization problems, particularly in production scheduling. These algorithms excel in generating high-quality solutions within reasonable computational timeframes, making them especially valuable for planning, scheduling, and engineering design applications. Based on their underlying principles, metaheuristics can be broadly categorized into evolutionary mechanisms, physical principles, and swarm intelligence approaches.

To comprehensively analyze the research landscape, we conducted a systematic literature review using Web of Science from 2000 to 2024, focusing on metaheuristic algorithms in multi-objective Flow-shop Scheduling Problem (FSP) from 2015 to 2024.[1]

As illustrated in Figure 5, research activity in FSP reached its peak during 2019–2023, with 2023 recording over 50 publications. The trend analysis reveals a significant increase in academic attention since 2011, with particular intensity after 2020. Figure 6 further demonstrates that, beyond basic FSP, Hybrid Flow-shop

[1] The search query was structured as follows: TS=((metaheuristic* OR "meta-heuristic*" OR "nature-inspired algorithm*" OR "computational intelligence" OR "evolutionary algorithm*" OR "genetic algorithm*" OR "differential evolution" OR "evolution* strategy" OR "evolution* programming" OR "swarm intelligence" OR "particle swarm optimization" OR "ant colony optimization" OR "artificial bee colony" OR "firefly algorithm" OR "bat algorithm" OR "cuckoo search" OR "simulated annealing" OR "tabu search" OR "harmony search" OR "memetic algorithm*" OR "scatter search" OR "variable neighbourhood search" OR "iterated local search") AND ("multi-objective" OR multiobjective OR "multi-criteria" OR multicriteria OR Pareto) AND ("flow shop" OR "flowshop" OR "flow-shop") NOT ("job shop" OR "job-shop" OR "jobshop")) AND PY=(2000–2024) AND DT=(Article).

TABLE 1 Literature objectives and algorithms.

Problem	Reference	Objectives	The number of objectives	Algorithms	Category
FSP	Rahimi-Vahed and Mirghorbani (2006)	weighted average completion time, weighted average tardiness rate	2	SPEA-II	EA
FSP	Sekkal and Belkaid (2023)	makespan, total energy consumption	2	SA	EA
FSP	Vallejos-Cifuentes et al. (2019)	makespan, total energy consumption	2	MOGA	EA
FSP	Anjana et al. (2020)	duration time, average tardiness	2	NSGA-II, hybrid NSGA-II, PSO, hybrid PSO	EA
FSP	Hassanzadeh et al. (2016)	total weighted tardiness, makespan	2	MOPSO, NSGA-II	SI and EA
FSP	Tavakkoli-Moghaddam et al. (2007)	weighted average completion time, weighted average tardiness	2	IA, BFO	SI
FSP	Fekri et al. (2024)	makespan, total idle time of workers	2	GA, SA	EA
FSP	Dong et al. (2024)	fuzzy makespan, the average fuzzy due date agreement index	2	multi-objective non-dominated sorted GSA	PhA
HFSP	Engin and Yılmaz (2021)	agreement index, average agreement index, fuzzy makespan	3	improved GA	EA
HFSP	Wang et al. (2024)	makespan, total energy consumption, system stability	3	multi-objective discrete PSO	SI
HFSP	Schulz et al. (2019)	makespan, total energy cost, peak load	3	multi-stage iterative local search	EA
HFSP	Wang et al. (2023)	fuzzy makespan, total fuzzy energy consumption	2	improved NSGA-II	EA
HFSP	Geng et al. (2020)	worker makespan, total tardiness, workload balance	3	improved MA	SI
PFSP	Rajkumar and Jeen Robert (2019)	makespan, total flow time	2	SA, GA	EA
PFSP	Mishra et al. (2020)	makespan, tardiness cost	2	Jaya	SI
PFSP	Fasihi et al. (2023)	makespan, job tardiness	2	SA, GA	EA
PFSP	Motair (2021)	makespan, maximum tardiness	2	SA	EA
PFSP	Zhang et al. (2021)	makespan, maintenance cost	2	Pareto IG	EA
PFSP	Yüksel et al. (2020)	total delay, total energy consumption	2	multi-objective discrete PSO	SI
PFSP	Öztop et al. (2020)	makespan, total energy consumption	2	multi-objective IG	EA
PFSP	Ding et al. (2016)	energy consumption, carbon emissions	2	multi-objective IG	EA
DPFSP	Huang et al. (2022)	total flow time, total tardiness	2	two-stage evolutionary algorithm	EA
DPFSP	Fathollahi-Fard et al. (2024a)	makespan, energy consumption, the number of lost workdays	3	SA and TS	EA
DPFSP	Schulz et al. (2022)	makespan, carbon emissions	2	IG	EA
DPFSP	Lu et al. (2022)	makespan, total energy consumption	2	Pareto based collaborative multi-objective optimization algorithm	Human based
DPFSP	Chen et al. (2019)	makespan, total energy consumption	2	co-optimization algorithm	Human based
DHFSP	Rifai et al. (2021)	makespan, production cost, tardiness	3	ALNS	EA
DHFSP	Gao et al. (2024)	makespan, worker workload	2	enhanced elite retention strategy MOEA	EA
JSP	Zhang et al. (2024a)	makespan, total workload	2	ABC	PSO
JSP	González et al. (2022)	makespan, energy consumption	2	improved NSGA-II	EA

(Continued on following page)

TABLE 1 (*Continued*) Literature objectives and algorithms.

Problem	Reference	Objectives	The number of objectives	Algorithms	Category
JSP	Afsar et al. (2022)	makespan, non-processing energy consumption	2	hybrid metaheuristics combine MA	SI
JSP	Wei et al. (2021)	total weighted tardiness, precocity, non-processing energy consumption, makespan	4	MOGA	EA
JSP	Li et al. (2020)	makespan, total precocity and tardiness	2	TLA	Human based
JSP	Xie et al. (2023)	tardiness, the number of transferred sub-batches	2	MOJA/D	SI
FJSP	Zhang et al. (2020)	makespan, total tardiness, total workload	3	distribute ACO	SI
FJSP	Shahsavari-Pour and Ghasemishabankareh (2013)	makespan, machine workload, total workload of all machines	3	hybrid GA and SA	EA
FJSP	Liu et al. (2021)	cost, energy consumption	2	VNS and GA	EA
FJSP	Li et al. (2014)	makespan, total workload of machine, workload of crucial machine	3	discrete ABC	SI
FJSP	Caldeira and Gnanavelbabu (2021)	makespan, total workload of machine, workload of crucial machine	3	discrete Jaya	SI
FJSP	Tan et al. (2021)	worker fatigue, makespan	2	improved NSGA-II	EA
Dist.FJSP	Shao et al. (2022)	total weighted tardiness, energy consumption	2	MA	SI
Dist.FJSP	Xu et al. (2021)	makespan, cost, quality, carbon emissions	4	GA and TS	EA
Dist.FJSP	Luo et al. (2020)	makespan, maximum workload, workload of the workers	3	improved MA	SI
Dist.FJSP	Li et al. (2018a)	makespan, total workload, early arrival/late arrival criteria	3	Pareto based hybrid TS	EA
Dyn.JSP	Wang et al. (2019)	discontinuity rate of new jobs, makespan deviation of the initial schedule, order deviation on the machine	3	improved PSO	SI
Dyn.FJSP	Liu et al. (2024)	makespan, workload imbalance	2	adaptive ALNS	EA
Dyn.FJSP	Zhang et al. (2013)	schedule efficiency, schedule stability	2	GA and TS	EA

Note: Abbreviations used in this table can be found in the Glossary section.

Scheduling Problem (HFSP) and Permutation Flow-shop Scheduling Problem (PFSP) have garnered substantial scholarly interest.

4.1 Flow-shop scheduling

FSP represents a fundamental challenge in production scheduling, where the primary goal is to optimize the processing sequence of multiple jobs across a series of sequentially connected machines. As illustrated in Figure 7, research interest in multi-objective FSP has shown significant growth since 2019, reflecting the increasing complexity of modern manufacturing requirements.

4.1.1 Problem model

The problem can be formally defined as follows: Let $J = \{J_1, J_2, \ldots, J_n\}$ be a set of n jobs and $M = \{M_1, M_2, \ldots, M_m\}$ be a set of m machines. Each job J_i must be processed on all machines in the order $M_1 \rightarrow M_2 \rightarrow \ldots \rightarrow M_m$, with processing time p_{ij} on machine M_m. The scheduling problem is subject to the following constraints:

(1) Each machine can process only one job at a time;
(2) Each job can be processed on only one machine at a time;
(3) The processing sequence is identical for all jobs;
(4) No preemption is allowed.

This chapter aims to construct a basic FSP model with the objectives of minimizing makespan and total flow time. Table 2 illustrates the parameter definitions for this problem.

$$\min C_{\max}(\pi) = C^{m}_{\pi(n)} \tag{13}$$

$$\min F(\pi) = \sum_{i=1}^{n} F(\pi(i)) = \sum_{i=1}^{n} C^{m}_{\pi(i)} \tag{14}$$

$$\text{s.t. } C^{1}_{\pi(1)} = p_{\pi(1),1} \tag{15}$$

$$C^{i}_{\pi(1)} = C^{i-1}_{\pi(1)} + p_{\pi(1),i} \tag{16}$$

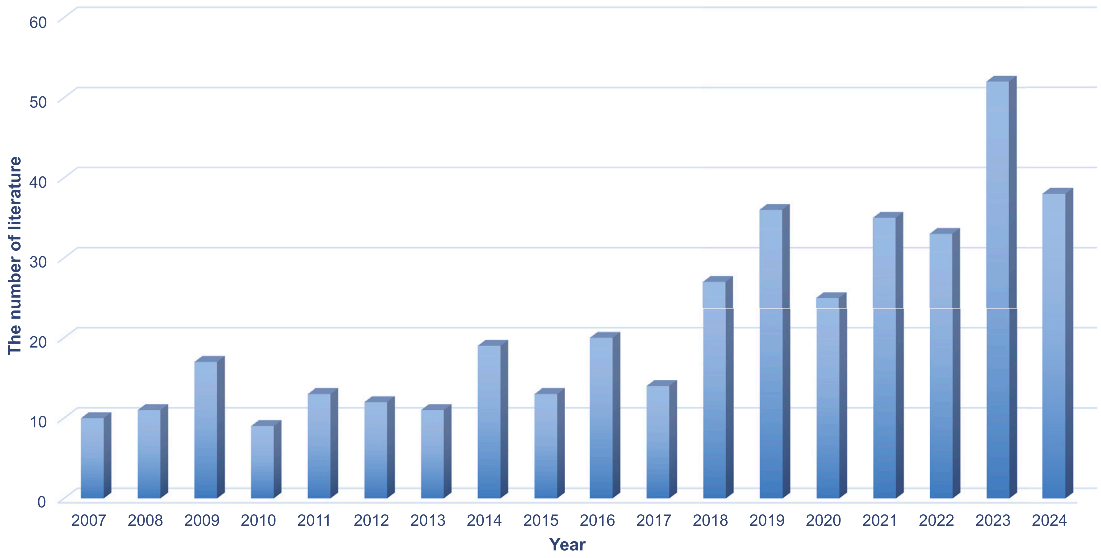

FIGURE 5
Statistics on the number of metaheuristic algorithms solving muti-objective FSP and variant problems.

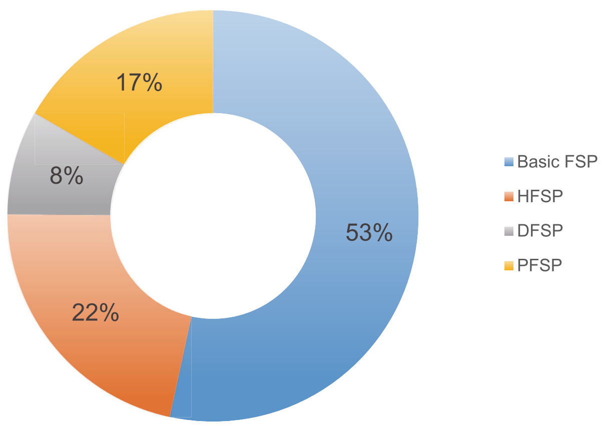

FIGURE 6
Proportion of metaheuristic algorithms solving muti-objective FSP and variant problems posted.

$$C^1_{\pi(j)} = C^1_{\pi(j-1)} + p_{\pi(j),1} \tag{17}$$

$$C^j_{\pi(i)} = \max\left\{C^j_{\pi(i-1)}, C^{j-1}_{\pi(i)}\right\} + p_{\pi(i),j} \tag{18}$$

$$\left(t_{kh} - p_{kh} - t_{ij}\right) x_{ij} x_{kh} y_{ijkh} > 0, \forall (i, j), (k, h) \tag{19}$$

$$y_{ij} = 0, \forall (i, j) \tag{20}$$

$$\sum_{i=1}^{m} x_{ij} = 1, \forall (i, j) \tag{21}$$

$$\sum_{j=1}^{n} x_{ij} = 1, \forall (i, j) \tag{22}$$

$$y_{ijkh} \in \{0, 1\}, \forall (i, j), (k, h) \tag{23}$$

$$x_{ij} \in \{0, 1\}, \forall (i, j) \tag{24}$$

$$t_{ij} \geqslant p_{ji} \geqslant 0, \forall (i, j) \tag{25}$$

Equations 15–18 represents the completion time of each job, while Equation 19 defines the processing sequence on each machine. Equation 20 ensures the feasibility of the operation sequence, and Equation 21 imposes constraints on job uniqueness. Similarly, Equation 22 enforces machine uniqueness, and Equations 23–25 specify non-negativity conditions. The optimization objectives of this model are defined in Equation 13, 14. Among these, Equation 13 represents the minimization of the makespan. As derived from Equations 15–18, the makespan is determined by $C^m_{\pi(n)}$, which corresponds to the completion time of the last job $\pi(n)$ in the processing sequence π on the last machine M_m. Equation 15 defines the minimization of the total flow time, which is the sum of the completion times of all jobs.

4.1.2 Solution algorithm

The evolution of FSP research can be categorized into three main phases: classical optimization phase (pre-2010), modern manufacturing transition phase (2010–2020) and industry 5.0 and sustainability phase (2020-present).

Classical Optimization Phase (Pre-2010): Early research focused on fundamental scheduling objectives. Rahimi-Vahed and Mirghorbani (2006) designed an effective MOPSO considering minimizing the average weighted completion time and weighted average tardiness, which finds the local Pareto frontiers of the problem by employing ideal points to specify the position vectors of the dominant particles in the swarm. Tavakkoli-Moghaddam et al. (2007) in order to minimize the weighted average completion time and weighted average tardiness time, proposed a hybrid multi-objective algorithm based on Immune Algorithm (IA) and Bacterial Foraging Optimization Algorithm.

Modern Manufacturing Transition Phase (2010–2020): This period witnessed the convergence of various modern manufacturing constraints, integrating technological advancements. Rossit et al. (2022) proposed a decomposition-

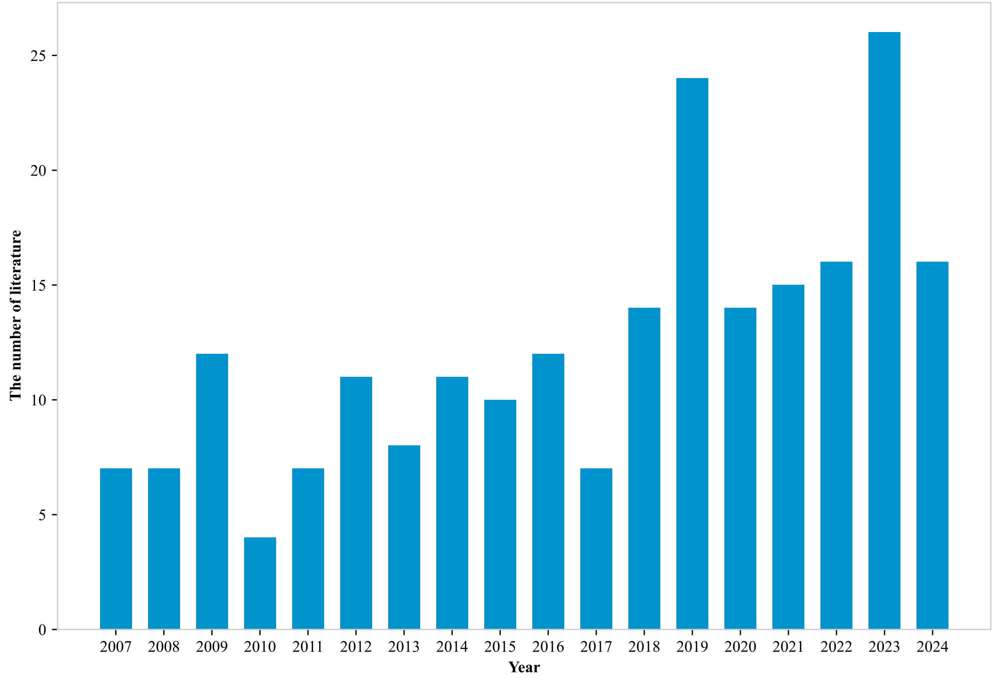

FIGURE 7
The number of literature on metaheuristic algorithms solving muti-objective FSP and variant problems over the years.

TABLE 2 Parameter descriptions.

Parameter	Statement
J	Set of jobs, consisting of n jobs $\{J_1, J_2, \ldots, J_n\}$
M	Set of machines, consisting of m machines $M_1, M_2, \ldots, M_m$
O_{ij}	The j-th operation of job J_i
p_{ij}	Processing time of operation O_{ij} on machine M_i
π	The processing sequence of jobs $(\pi(1), \pi(2), \ldots, \pi(n))$
π_a	The a-th job in sequence π
U	All possible operation combinations
$C_{\pi(j)i}$	Completion time of job π_j on machine M_i
t_{J_i}	Completion time of operation O_{ij} on machine M_i
x_{ij}	Binary variable, $x_{ij} = 1$ if operation O_{ij} is processed on machine M_i; otherwise $x_{ij} = 0$
y_{ijkh}	Binary variable, $y_{ijkh} = 1$ if operation O_{ij} is processed exactly before operation O_{kh}; otherwise $y_{ijkh} = 0$

based MOEA to solve the multi-objective FSP with missing operations. The method employs a structured approach to decompose the solution space, ensuring diversity among initial solutions. Hassanzadeh et al. (2016) considered an multi-objective integrated production-distribution FSP with the objective of minimizing the total weighted tardiness and the makespan as the first objective function, and minimizing the sum of total weighted early completion times, total weighted number of late jobs, inventory cost and total delivery cost as the second objective function. Hao et al. (2017) proposed a Multi-objective Estimation of Distribution Algorithm to address stochastic JSP with uncertain processing times. The algorithm effectively balances the trade-off between expected mean completion time and total tardiness. By employing probabilistic model updates and Monte Carlo sampling, it ensures scheduling quality while enhancing computational efficiency. Branda et al. (2021) examined two metaheuristic algorithms combining GA and Harmony Search for solving the bi-objective problem of FSP. Anjana et al. (2020) considered a sequence-dependent setup time FSP with the objective of minimizing duration and average tardiness.

Industry 5.0 and Sustainability Phase (2020-Present): Current research emphasizes human-centric manufacturing and environmental considerations. Key developments include: energy efficiency and worker-centric optimization. Vallejos-Cifuentes et al. (2019) considered achieving energy savings without compromising the productivity of the manufacturing system and proposed an energy-aware FSP with process speed as the main energy related decision variable for a problem that targets production goals with energy efficiency. Boufellouh and Belkaid (2023) investigated the problem of energy saving FSP in the presence of blocking and collision-free transport constraints. For a manufacturing system with a sequential dependent setup times, Automatic Guided Vehicles (AGV), transport speed control and battery management constraints, an enhanced multi-objective Ant Colony Optimization (ACO) was developed for the scheduling problem. With the objective of simultaneously minimizing makespan and total energy consumption, the algorithm proposes a novel high-resolution search strategy, a heuristic AGV scheduling strategy and a critical path based energy saving improvement strategy. Sekkal and Belkaid (2023) considered the learning effect of workers and proposed FSP with learning efficiency. They developed a metaheuristic algorithm based on multi-objective Simulate Anneal (SA) with the minimisation of makespan and energy consumption as two objective functions, in which the search for solutions is enhanced by local search. Focusing on the multi-skilled characteristics of the workers, Fekri et al. (2024) proposed a GA and a SA with the objective of minimizing makespan and the total idle time of workers, and used a special Taguchi method for the parameter adjustment. It is finally concluded that the GA has better performance and efficiency than the SA for the multi-skill resource constrained FSP, but SA is superior to GA in terms of solution time. Dong et al. (2024) solved the multi-objective fuzzy block FSP based on learning and fatigue effects by combining the characteristics of the workers with the background of the FSP scheduling of the prefabricated pods modular cells. They used the objective of minimizing the fuzzy makespan and maximize the average fuzzy due-date agreement index, a multi-objective non-dominated Sorted Gravitational Search Algorithm (GSA) is proposed.

The evolution of FSP research reflects the field's adaptation to emerging manufacturing paradigms, with increasing emphasis on sustainability, worker wellbeing, and system flexibility. Future research opportunities lie in integrating advanced technologies while maintaining computational efficiency and practical applicability.

4.2 Permutation flow-shop scheduling

As shown in Figure 8, research interest in PFSP has exhibited significant volatility between 2007 and 2024, with notable peaks in 2019 and 2022.

4.2.1 Problem model

The PFSP can be described as follows: a set of n jobs is processed on a set of m machines in the same order. Each job i has a series of corresponding operations j on different machines, and the processing times for these operations are predetermined. The rules that must be followed in the workshop are that the processing order of each job on all machines is identical, and each machine must process all jobs in the same sequence. This chapter establishes a mathematical model with the objectives of minimizing makespan and energy consumption. Table 3 presents the detailed parameter list of this model.

$$\min f_1 = C_{\max} \tag{26}$$

$$\min f_2 = \sum_{k=1}^{m} P_k T_k \tag{27}$$

$$\text{s.t. } T_k = C(J_i, k) - \left(C(J_1, k) - p(J_1, k)\right) - \sum_{j=1}^{n} p(J_k, j) \tag{28}$$

$$C(J_1, 1) = p(J_1, 1) \tag{29}$$

$$C(J_i, 1) = C(J_{i-1}, 1) + p(J_i, 1) \tag{30}$$

$$C(J_1, j) = C(J_1, j-1) + p(J_1, j) \tag{31}$$

$$C(J_i, j) = \max\{C(J_{i-1}, j), C(J_i, j-1) + p(J_i, j)\} \tag{32}$$

$$C_{\max} = C(J_n, m) \tag{33}$$

Equations 26–28 define the objective functions of this model. Specifically, Equation 26 represents the first objective, which is the minimization of makespan. Equation 27 presents the second objective. The total energy consumption is calculated as the sum of the products of the idle time and the corresponding idle power for all machines. Equation 28 defines the computation of machine idle times. Equation 29 calculates the completion time of the first job on the first machine. Equation 30 computes the completion time of job i on the first machine. Equation 31 determines the completion time of the first job on machine j. Equation 32 calculates the completion time of job i on machine j. Equation 33 defines the maximum completion time for the job sequence.

4.2.2 Solution algorithm

Tajbakhsh et al. (2014) focused on a three-phase manufacturing system including machining, assembly and batch processing with the objective of minimizing makespan and the sum of early-to-delay costs. Based on the formulation of PFSP as a mixed integer mathematical model, a hybrid algorithm was designed to achieve efficient exploration of the solution space using a metaheuristic approach that combines the advantages of GA and Particle Swarm Optimization (PSO). Rajkumar and Jeen Robert (2019) proposed a hybrid multi-objective optimization algorithm based on GA and SA in order to solve the PFSP problem with the objective of minimizing makespan and the total flow time. The algorithm determines the near-optimal solution by assigning weights to each objective function.

The Jaya algorithm has also been used by scholars to study PFSP due to its advantages of simplicity, efficiency and scalability, Mishra et al. (2020) used the Jaya algorithm with the objective of minimizing makespan and tardiness cost to solve the PFSP problem. The algorithm converts the job preference vector into job ranking vector using maximum order value rule after randomly assigning priority to each job. In order to solve multi-objective, it uses a multi-attribute model based on Apriori method, and after comparing with SA. Zhang et al. (2021) considered preventive and corrective maintenance in manufacturing activities, and with the objective of minimizing makespan and maintenance cost, they designed a restarted iterative Pareto greedy algorithm. During the search process, the algorithm develops a restart mechanism to generate a new initial solution to prevent the algorithm from falling into a local optimum. Fasihi et al. (2023) proposed a two-step procedure by considering the reentrant PFSP with the objective of minimizing makespan and maximizing delay. The first step divides the population into sub-populations and applies GA in each sub-population to obtain the set of approximate Pareto frontier solutions. The second step unifies all the Pareto solution sets into a whole, using a multi-objective hybrid metaheuristic algorithm based on dominance relations.

Considering energy consumption and environmental constraints on manufacturing, Yüksel et al. (2020) proposed a new multi-objective discrete Artificial Bee Colony Algorithm (ABC) with the objective of minimizing both total delay and total energy consumption. By comparing it with metaheuristic algorithms such as traditional MOGA, variations of MOGA with local search, the algorithm is able to adapt to idle free FSP, blocking FSP, and Job-shop Scheduling Problem (JSP), or other higher level integrated manufacturing problems. Öztop et al. (2020) with the objective of minimizing makespan and total energy consumption, developed an enhanced ϵ constraint method to obtain a Pareto

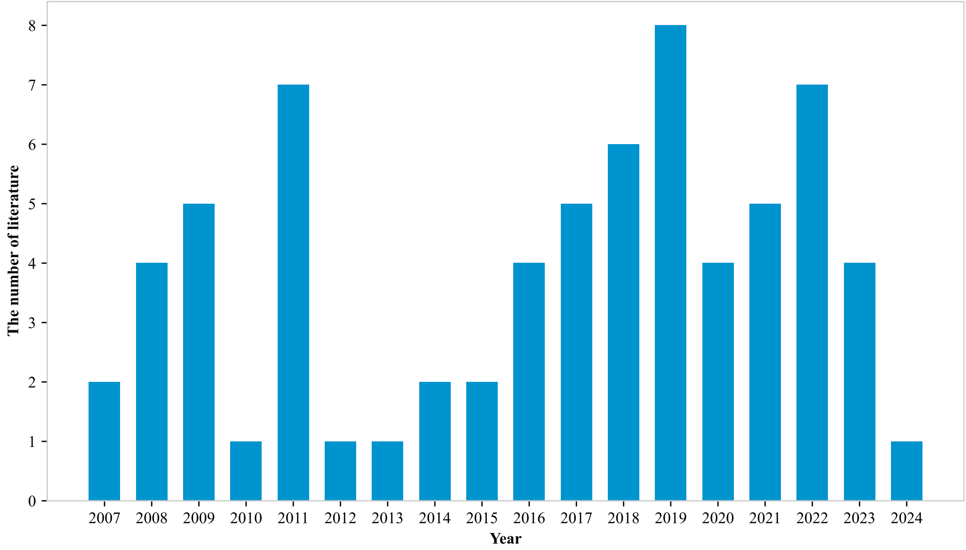

FIGURE 8
The number of literature on metaheuristic algorithms solving muti-objective PFSP over the years.

TABLE 3 Parameter descriptions.

Parameter	Statement
i	Job index, $i = 1, 2, \ldots, n$
j	Machine index, $j = 1, 2, \ldots, m$
n	Total number of jobs
m	Total number of machines
P_k	The fixed idle power of machine k
T_k	The total idle time of machine k after processing all jobs
$p(J_i, j)$	Processing time of job J_i on machine j
$C(J_i, j)$	Completion time of job J_i on machine j

optimal solution. For small scale problems this algorithm obtains an approximation of the Pareto optimal bound using a small ϵ level. In addition, two multi-objective Integrated Gradients (IG) and a multi-objective variable block insertion heuristic algorithm were proposed for initial solution generation.

Ding et al. (2016) concerned about the direct relationship between energy consumption and carbon emissions, with the aim of improving the carbon efficiency of industrial factories and thus reducing the energy cost in the production process. Designed an optimization algorithm for the PFSP with the objective of minimizing the total carbon emissions and the makespan as the objective of the optimization algorithm. The algorithm is based on an extended Nawaz-Enscore-Ham (NEH) insertion procedure, and a multi-objective NEH algorithm and an improved multi-objective IG are designed to solve the problem. Wu and Che (2020) investigated the energy efficient no-waiting PFSP problem, and designed an adaptive multi-objective Variable Neighborhood Search (VNS) with the objective of minimizing makespan and total energy consumption. The algorithm designs two VNS structures to generate neighbours through insertion and swapping operations, and uses an adaptive mechanism to dynamically select the appropriate structure to handle the evolutionary direction of the current solution. Xin et al. (2021) designed an improved discrete Whale Optimization Algorithm (WOA) for PFSP with sequential dependence on setup time, considering that the energy consumption is related to the time of the job transfer process. The algorithm aims to minimize makespan and total energy consumption, and combines Differential Evolution (DE) and enhanced search strategy to improve the performance of the algorithm.

Recent research developments can be categorized into three main streams:

- Manufacturing System Integration: recent studies have expanded PFSP to incorporate real-world manufacturing complexities.
- Advanced Hybrid Approaches: modern PFSP solutions employ sophisticated hybrid algorithms. Such as hybrid GA and SA, Jaya algorithm and hybrid PSO and so on.
- Energy and Environmental Considerations: recent research has increasingly focused on sustainability objectives. For example, carbon emissions, energy consumption and machine availability and so on.

The evolution of PFSP research reflects increasing emphasis on practical manufacturing constraints and environmental considerations. This progression suggests a trend toward more comprehensive and sustainable scheduling solutions that balance traditional performance metrics with modern manufacturing requirements.

4.3 Hybrid flow-shop scheduling

As illustrated in Figure 9, research interest in HFSP has shown significant growth since 2020, reaching unprecedented levels in

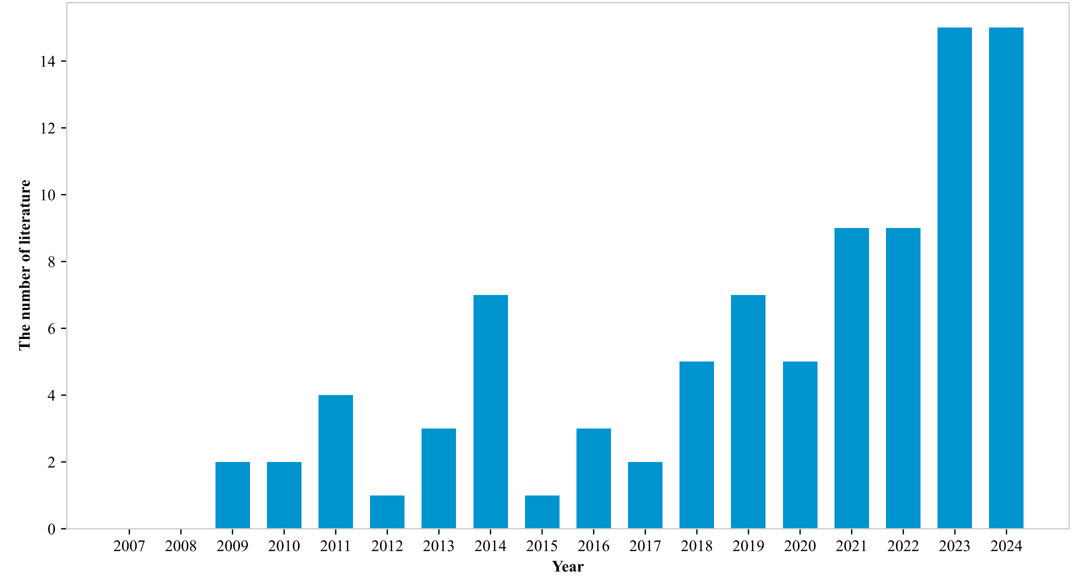

FIGURE 9
The number of literature on metaheuristic algorithms solving muti-objective HFSP and variant problems over the years.

recent years. This surge reflects the increasing complexity and practical relevance of HFSP in modern manufacturing.

4.3.1 Problem model

This chapter takes the HFSP with worker constraints as an example to establish a mathematical model. The problem can be described as follows: there are s $(s > 1)$ consecutive stages, capable of processing b independent jobs that must be handled in the same sequential order. Each stage is equipped with m $(m > 1)$ machines, and each machine can be operated by w $(w > 1)$ workers. All jobs need to be scheduled according to certain optimization objectives. The problem involves addressing three sub-problems:

(1) Sequencing all operations based on operational constraints;
(2) Assigning machines to process the jobs;
(3) Allocating workers to operate the assigned machines.

Table 4 summarizes the parameters and their descriptions for this problem.

$$\min f_1 = \max C_i, \forall i \tag{34}$$

$$\min f_2 = \sum_{i=1}^{n_b} \sum_{j=1}^{n_s} \sum_{m=1}^{n_m} \sum_{w=1}^{n_w} \left(L_{ijmw} \times y_{ijmw} \right) \tag{35}$$

$$\text{s.t. } C_{i(j+1)} - C_{ij} \geq P_{i(j+1)mw} y_{i(j+1)mw}, \forall i, j, m, w \tag{36}$$

$$C_{ij} \geq \left(C_{ng} x_{ngmk} + P_{ijmw} y_{ijmw} \right) - \left(1 - x_{ijm(k+1)} \right) LN, \forall i, h, j, g, m, k, w \tag{37}$$

$$C_{ij} \geq \left(C_{ng} z_{ngwq} + P_{ijmw} y_{ijmw} \right) - \left(1 - z_{ijw(q+1)} \right) LN, \forall i, h, j, g, w, q, m \tag{38}$$

$$\sum_{k=1}^{n_b} \sum_{m=1}^{n_m} x_{ijkm} = 1, \forall i, j \tag{39}$$

$$\sum_{j=1}^{n_s} \sum_{i=1}^{n_b} x_{ijmk} = 1, \forall m, k \tag{40}$$

$$\sum_{j=1}^{n_s} \sum_{i=1}^{n_b} x_{ijm(k+1)} \leq \sum_{j=1}^{n_s} \sum_{i=1}^{n_b} x_{ijmk}, \forall m, k \tag{41}$$

$$\sum_{q=1}^{n_b} \sum_{w=1}^{n_w} z_{ijwq} = 1, \forall i, j \tag{42}$$

$$\sum_{j=1}^{n_s} \sum_{i=1}^{n_b} z_{ijwq} \leq 1, \forall w, q \tag{43}$$

$$\sum_{j=1}^{n_s} \sum_{i=1}^{n_b} z_{ijw(q+1)} \leq \sum_{j=1}^{n_s} \sum_{i=1}^{n_b} z_{ijwq}, \forall w, q \tag{44}$$

$$\sum_{w=1}^{n_w} \sum_{m=1}^{n_m} y_{ijmw} = 1, \forall i, j \tag{45}$$

$$\sum_{j=1}^{n_s} \sum_{i=1}^{n_b} y_{ijmw} \leq n_b, \forall m, w \tag{46}$$

The model aims to minimize the makespan and the total worker cost, as formulated in Equations 34, 35. Equation 36 ensures the precedence constraints for operations. Equation 37 enforces the sequencing constraints for operations on the same machine. Equation 38 ensures the sequencing constraints for operations assigned to the same worker. Equation 39 guarantees that each operation is processed by only one machine at any given time. Equation 40 ensures that each machine processes at most one operation at any given time. Equation 41 prevents gaps in the sequence of operations on the same machine, ensuring that no position is left empty before a position is fully occupied. Equation 42 ensures that each operation is handled by only one worker at any given time. Equation 43 guarantees that each worker handles at most one operation at any given time. Equation 44 prevents gaps in the

TABLE 4 Parameter descriptions.

Parameter	Statement
i, h	Job index, $i, \quad h = 1, 2, \ldots, n_b$
j, g	Stage index, $j, \quad g = 1, 2, \ldots, n_s$
m	Machine index, $m = 1, 2, \ldots, n_m$
w	Worker index, $w = 1, 2, \ldots, n_w$
k	Sequential index of operations processed on the same machine, $k = 1, 2, \ldots, n_b$
q	Sequential index of operations processed by the same worker, $q = 1, 2, \ldots, n_b$
n_b	Number of jobs
n_s	Number of stages
n_m	Number of machines
n_w	Number of workers
WS_j	Available worker set in stage j
MS_j	Available machine set in stage j
nm_j	Number of machines in stage j
nw_j	Number of workers in stage j
O_{ij}	Operation of job i in stage j
P_{ijmw}	Processing time of operation O_{ij} on machine m by worker w
C_{ij}	Completion time of operation O_{ij} in stage j
C_i	Completion time of job i
L_{ijmw}	Cost when operation O_{ij} is processed on machine m by worker w
E_{ijmw}	Green indicator value of operation O_{ij} on machine m by worker w
E_{ijmw}	Energy consumption when operation O_{ij} is processed on machine m by worker w
N_{ijm}	Noise when operation O_{ij} is processed on machine m
R_{ijm}	Recycling rate of tool chips when operation O_{ij} is processed on machine m
S_{ijm}	Safety coefficient when operation O_{ij} is processed on machine m
x_{ijkm}	Decision variable, if O_{ij} is processed at the k-th position on machine m, then 1, otherwise 0
y_{ijmw}	Decision variable, if O_{ij} is processed by worker w on machine m, then 1, otherwise 0
z_{ijwq}	Decision Variable, if O_{ij} is processed by worker w at the q-th position, then 1, otherwise 0
LN	A large enough integer

sequence of operations assigned to the same worker, ensuring that no position is left empty before a position is fully occupied. Equation 45 ensures that each operation is performed by only one worker on one machine at a time. Equation 46 allows each worker to perform multiple operations on the same machine.

4.3.2 Solution algorithm

Behnamian et al. (2009) proposed a multi-objective HFSP with sequence-dependent setup times, focusing on minimizing makespan and the sum of job advance and delay times. Their approach consists of three stages: first, they use a random-keyed GA to approximate the Pareto front; second, they unify non-dominated solutions and enhance the front with a local search; and third, they apply an ϵ-constraint overlay hybrid metaheuristic, combining SA and VNS to refine the solutions further. Fadaei and Zandieh (2013) focused on sequence dependent family setup time for HFSP, considered the two objectives of minimizing makespan and total tardiness, and used three MOGA, subpopulation GA, and nondominated sequential GA to solve the above problems. Engin and Yılmaz, 2021) concerned about the time uncertainty HFSP problem due to human and other factors in the actual production process, proposed a multi-objective processor HFSP method based on fuzzy logic, which formulates the HFSP problem with fuzzy processing time and fuzzy due date. It involves three objectives of maximizing the minimum consistency index, maximizing the average consistency index and minimizing the maximum fuzzy makespan.

In real production process, energy consumption is one of the concerns of managers, Li et al. (2018b) proposed an energy-aware multi-objective optimization algorithm to solve the HFSP problem, which considers both minimizing makespan and energy consumption. This metaheuristic algorithm two vectors are used to represent the solution using machine assignment priority and scheduling, and four types of decoding methods are used to consider the two objectives simultaneously. Eight types of neighbourhood structures and an adaptive neighbourhood selection method are designed to speed up the search for a solution, effectively balancing the capabilities of global and local search through deep exploitation and exploration strategies. Hosseini (2017) focused on the difference between the manufacturing and assembly phases of a product and proposed a two phase HFSP, where they assumed the first phase to be a HFSP with the same parallel machines and the second phase was treated as assembly. The algorithm uses a two-stage MOGA to solve this problem with the objective of minimizing makespan and the sum of early and late arrivals of jobs. For the uncertain time manufacturing process, Wang et al. (2024) proposed the green HFSP in the case of emergency batch insertion. The problem adopts a multi-objective discrete ABC with the objectives of minimizing makespan, total energy consumption and system stability. The algorithm is able to significantly reduce the cost and substantially improve the accuracy and efficiency by testing and comparing it on small and large scale problems.

Schulz et al. (2019) focused on the importance of energy-aware scheduling and developed a new multi-stage iterative local search algorithm for energy-aware HFSP. The algorithm incorporates three energy saving strategies of reducing energy consumption, reducing energy cost using different energy prices and considering load balancing with the objective of minimizing makespan, total energy cost and peak load. The algorithm incorporates a tabu list, several time and energy dependent list scheduling algorithms, a right shift procedure and a reference point based fitness function to improve the quality of the solution. Chen et al. (2020) used a multi-objective mixed-integer planning model with the objective of minimizing makespan and power consumption. The algorithm uses a GA to obtain the Pareto solution set, and uses a multi-objective energy-efficient scheduling algorithm to compute the

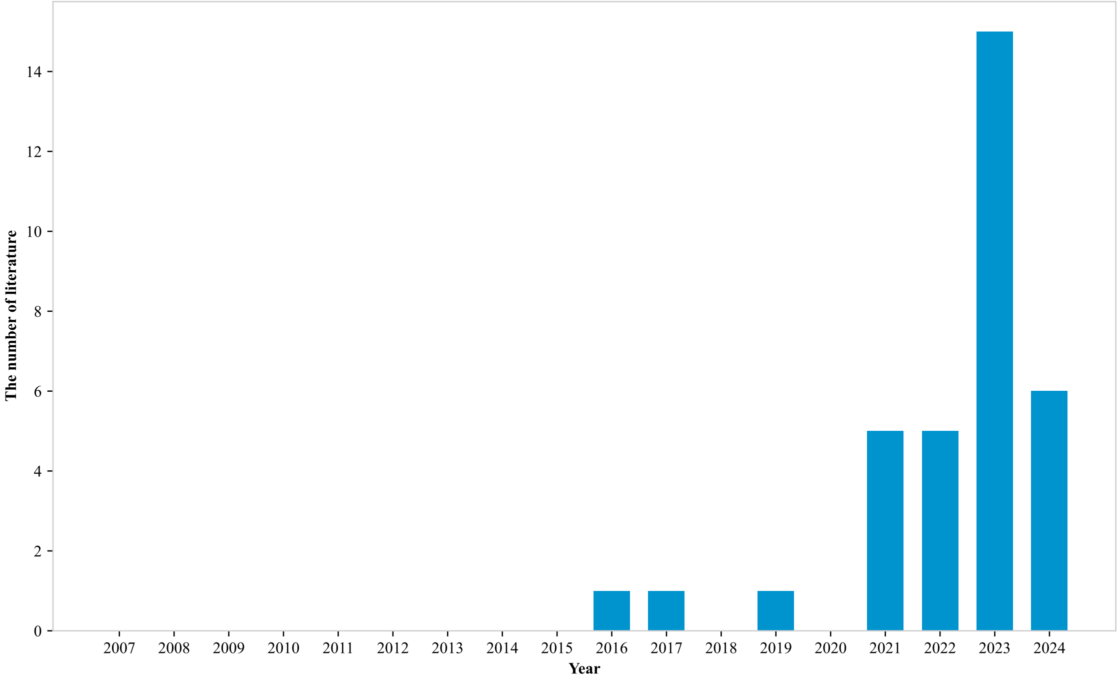

FIGURE 10
The number of literature on metaheuristic algorithms solving muti-objective DFSP and variant problems over the years.

fitness value of each chromosome in GA. Wang et al. (2023) added variable machine speed constraints to the energy-efficient HFSP problem and proposed an energy-efficient fuzzy HFSP considering variable machine speeds, for which an extended NSGA-II was designed. The algorithm aims to minimize both fuzzy makespan and total fuzzy energy consumption, and generates an initial population through inverse learning, and a new strategy based on historical information to produce high-quality solutions.

In addition to the consideration of energy consumption, many scholars have also focused on human-centred manufacturing in the context of Industry 5.0. Geng et al. (2020) investigated multi-objective HFSP with dual resource constraints considering the dual flexibility characteristics of machines and workers. The problem was formulated with the objective of minimizing worker makespan, total tardiness and workload balance, and proposed an improved multi-objective Memetic Algorithm (MA) with parameters set by Taguchi method. This method significantly outperforms other algorithms in terms of convergence, diversity, and dominance of non-dominated solutions. Han et al. (2021) proposed a MOEA with seven heuristic decodings for HFSP considering workers constraints. The algorithm combines machine worker allocation priority rules with the optimization objective of minimizing makespan and total delay. It is demonstrated through numerical experiments that the algorithm enables the solution to achieve a balance between exploration and exploitation, and can reasonably solve the manufacturing problem.

Recent research has focused on several key aspects of HFSP: setup time considerations, energy-aware scheduling, human-centered manufacturing and uncertainty management. The evolution of HFSP research reflects a growing emphasis on practical manufacturing constraints while maintaining computational efficiency. Future research opportunities lie in integrating emerging technologies and addressing increasingly complex real-world scenarios, particularly in the context of smart manufacturing and Industry 5.0 requirements.

4.4 Distributed flow-shop scheduling

Distributed flow-shop scheduling problems encompasses three main variants: Distributed Flow-shop Problem (DFSP), Distributed Permutation Flow-shop Problem (DPFSP), and Distributed Hybrid Flow-shop Problem (DHFSP). In a DFSP system with f factories, each containing a FSP, jobs are distributed across factories through specific allocation mechanisms.

As shown in Figure 10, research interest in distributed scheduling systems has grown significantly in recent years, which may be related to the expansion of manufacturing operations.

4.4.1 Problem model

The DFSP can be described in detail as follows. It involves N jobs and F factories, each containing the same M machines. Each job can only be processed on one machine, and each machine can process only one job at a time. Every job must be assigned to one of the F factories, and all operations must be completed within the same factory, without transferring to another factory during this period. This chapter establishes a mathematical model with the objectives of minimizing makespan and total energy consumption. The detailed parameter list is presented as Table 5.

$$\min C_{\max} = \max\left\{C_{i,\pi_k,j}\right\}, \quad j = 1, 2, \ldots, N, \quad k = 1, 2, \ldots, F \tag{47}$$

$$\min t_{\text{tec}} = \sum_{k=1}^{F} (PEC_i + SEC_i) \tag{48}$$

TABLE 5 Parameter descriptions.

Parameter	Statement
j	Index of job, $j = 1, 2, \ldots, N$
i	Index of machine, $i = 1, 2, \ldots, M$
k	Index of factory, $k = 1, 2, \ldots, F$
s	Index of processing speeds, $s = 1, 2, \ldots, S$
N	The number of jobs
M	The number of machines
F	The number of factories
S	The number of speeds
V_s	The S-th processing speed
n_k	The number of jobs in the k-th factory
π_k	A complete schedule in the k-th factory, $\pi_k = (\pi_{k,1}, \ldots, \pi_{k,n_k})$
$P_{i,k,s}$	The processing power of the job on machine i in factory k at speed S per unit time
$IP_{i,k}$	The idle power of the machine i in factory k per unit time
$t_{i,\pi k_j}$	The standard process time of the j-th job on the i-th machine in the k-th factory
$p_{i,\pi k_j}$	The actual process time of the j-th job on the i-th machine in the k-th factory
SEC_k	The idle time of the job in factory k generate energy consumption
PEC_k	The processing time of the job in factory k generate energy consumption
$C_{i,\pi k_j}$	The completion time of job j on machine m in the k factory
$x_{t,j,k}$	A binary variable that takes value 1 if job j occupies position t in factory k, and 0 otherwise
$y_{j,k}$	A binary variable that takes value 1 if job j is processed in factory k, and 0 otherwise
$z_{j,i,s}$	A binary variable that takes value 1 if job j is processed in machine i at speed s, and 0 otherwise

$$\text{s.t.} \sum_{t=1, t\neq j}^{N} \sum_{k=1}^{F} x_{t,j,k} = 1, \forall j \tag{49}$$

$$\sum_{k=1}^{F} y_{j,k} = 1, \forall j \tag{50}$$

$$\sum_{t=1, t\neq j}^{N} x_{t,j,k} + x_{j,t,k} \leq 2 \cdot y_{j,k}, \forall j, k \tag{51}$$

$$\sum_{j=1, j\neq t}^{N} \sum_{k=1}^{F} x_{t,j,k} = 1, \forall t \in \{1, 2, \ldots, N\} \tag{52}$$

$$x_{t,j,k} + x_{j,t,k} \leq 1, \forall j \in \{1, 2, \ldots, N-1\}, j < t \tag{53}$$

$$c_{0,\pi_{k,j}} = 0, \forall j, k \tag{54}$$

$$c_{t,\pi_{k,0}} = 0, \forall i, k \tag{55}$$

$$c_{i,\pi_{k,j}} \geq c_{i-1,\pi_{k,j}} + \sum_{j=1}^{N} x_{t,j,k} \cdot p_{i,\pi_{k,j}}, \forall i, j, k \tag{56}$$

$$c_{t,\pi_{k,j}} \geq c_{t,\pi_{k,j-1}} + \sum_{j=1}^{N} x_{t,j,k} \cdot p_{t,\pi_{k,j}}, \forall i, j, k \tag{57}$$

$$c_{1,\pi_{k,1}} = \sum_{k=1}^{F} x_{1,1,k} \cdot P\left(1, \pi_{k,1}\right), \forall k \tag{58}$$

$$c_{1,\pi_{k,j+1}} = c_{1,\pi_{k,j}} + \sum_{j=1}^{N-1} x_{t,j,k} \cdot p_{1,\pi_{k,j+1}}, \forall k, j \tag{59}$$

$$c_{t+1,\pi_{k,1}} = c_{t,\pi_{k,1}} + \sum_{l=1}^{N} x_{t,1,k} \cdot p_{t+1,\pi_{k,1}}, \forall k, i \tag{60}$$

$$c_{t,\pi_{k,j}} = \max\left\{c_{t,\pi_{k,j-1}}, c_{t-1,\pi_{k,j}}\right\} + \sum_{j=1}^{N} x_{t,j,k} \cdot p_{t,\pi_{k,j}}, \forall i, j, k \tag{61}$$

$$p_{i,j} = t_{i,j} \big/ v_s, \forall i, j \tag{62}$$

$$PEC_k = \sum_{i=1}^{M} \sum_{j=1}^{N} x_{i,j,k} \cdot p_{i,\pi_{k,j}} \cdot \sum_{s=1}^{S} z_{j,i,s} \cdot PP_{i,k,s}, \forall k \tag{63}$$

$$SEC_k = \left(c_{N,\pi_{k,M}} - \sum_{i=1}^{N} p_{i,j} \cdot \sum_{j=1}^{N} x_{i,l,f} \right) \cdot IP_{f,j}, \forall k, j \tag{64}$$

Equations 47, 48 represent the minimization of the two objectives. Equations 49, 50 ensure that each job is accurately assigned to one factory and exists only once within that factory. Equations 51, 53 indicate that each job can potentially be a predecessor or successor within the assigned factory, while avoiding priority conflicts. Equation 52 explains that all jobs should have the possibility of being assigned to the n positions in each factory. Equations 54, 55 state that jobs are ready at time 0, and machines start working at time 0. Equations 56, 57 specify that a job can only begin after its predecessor in the same factory has completed processing on the machine. Equations 58–61 describe the calculation of completion times, starting from the first job processed on the first machine to the last job processed on the last machine. Equation 62 defines the actual processing time of a job. Equations 63, 64 specify the energy consumption of factory k when machine j operates in processing mode and standby mode, respectively.

4.4.2 Solution algorithm

Rifai et al. (2021) proposed an improved multi-objective Adaptive Large Neighborhood Search (ALNS) for the DPFSP by considering the effect of sequence-dependent setup time for the first time. The algorithm achieves an effective use of balancing and exploration by introducing new destruction and repair heuristics, optimizing the solution acceptance criterion and the non-dominated set update mechanism. Cai et al. (2018) proposed an improved NSGA-II for the DPFSP with transport and qualification constraints aiming to optimize the three objectives of fabrication time, maximum delay, and total cost (including transport and installation cost) simultaneously. The algorithm introduces a new solution representation, a population initialisation strategy, efficient crossover and mutation operators, and local search techniques.

Huang et al. (2022) proposed a two-stage evolutionary algorithm with the objective of minimizing the total flow time and total tardiness time. A two-stage structure is used, in the first stage of this algorithm to increase the population diversity through two constructive heuristics and four crossover and two mutation operators designed. In the second stage the first two populations are integrated to improve the performance of the algorithm by using the normalised objective function and two new crossover operators to extend the solution to the Pareto frontier. Fathollahi-Fard et al. (2024a) redefined the sustainable DPFSP using an online mixed-integer programming model. With the objective of minimizing makespan, energy consumption, and reducing the number of lost workdays, SA and Tabu Search (TS) were designed for the large scale problem. Fathollahi-Fard et al. (2024b) also proposed an ALNS for the sustainable DPFSP. The algorithm considers different modes of operation in which machines can run from manual to automatic, taking into account energy consumption, number of operators required and number of working days lost to training. In addition, the algorithm considers uncertainty problems including machine failures, variable processing times and the random arrival of new jobs.

Since carbon emissions will have a direct impact on the environment, Schulz et al. (2022) designed an IG for the DPFSP under large-scale instances in order to minimize makespan while reducing carbon emissions. Lu et al. (2022) investigated energy efficient scheduling for DPFSP with a finite buffer, aiming to minimize both the manufacturing time and the total energy consumption. They proposed a Pareto based collaborative multi-objective optimization algorithm. The algorithm reduces the total energy consumption through a speed scaling strategy, generates a high quality initial population using a collaborative initialisation strategy, and exploits the properties of the problem to develop a collaborative search operator and a knowledge based local search operator. Chen et al. (2019) investigated the energy efficient DPFSP with idle-free aiming to minimize both makespan and total energy consumption. For this problem, they designed a co-optimization algorithm. The algorithm uses two metaheuristics for population initialisation together to ensure the quality and diversity of the initial solutions. Multiple search operations are designed to collaborate in a competitive manner to enhance the exploration capability. Wang et al. (2022) proposed a method combining a multi-objective mixed integer programming model and a multi-objective WOA to solve the DFSP considering energy consumption. The method has the objective of minimizing the total energy consumption and makespan. For the base WOA, a new initialisation method, update operator and local search strategy are designed for this particular problem. When dealing with large scale problems, this combined approach is able to strike a good balance between computational cost and the quality of the solution.

Zhao et al. (2024) considered the problem of energy efficient DFSP with blocking, and proposed a multi-objective discrete DE with the objective of minimizing makespan and total energy consumption. The algorithm employs a local search strategy based on the knowledge of five operators is introduced to enhance the utilisation of the algorithm, and a non-critical path energy saving strategy is used to reduce energy consumption. In order to cope with environmental problems and to meet the demand of customised production, Yu F. et al. (2024) proposed a mixed integer linear programming model and a multi-objective MA for the study of DHFSP. The method simultaneously aims at minimizing the total delay and total energy consumption. The quality of the initial solution is improved by introducing a speed dependent decoding method, an initialisation strategy based on the problem characteristics, and a search strategy that enhances the local search capability, which guides the population to achieve effective exploration in the solution space. As the worker resources also affect the efficiency of scheduling in the actual production manufacturing process, Gao et al. (2024) studied the DHFSP with assembly to minimize the degree of imbalance between makespan and worker workload. To address the problem, a variation based local search method and an elite search method were proposed, and a MOEA with an enhanced elite retention strategy was developed. Xin et al. (2015) proposed an adaptive binary PSO in the process of solving the multi-skilled worker assignment problem in a complex assembly system in order to achieve a balance between assembly station and process workloads and to minimize the man cost.

Recent research has addressed various practical aspects: sequence-dependent setup time, energy efficiency, sustainability considerations, worker resource management and algorithmic innovations. These developments reflect the field's evolution toward more comprehensive and practical scheduling solutions, incorporating environmental, worker-related, and efficiency considerations. Future research opportunities lie in addressing emerging challenges in distributed manufacturing environments, particularly in areas of real-time adaptation and system resilience.

The evolution of FSP and variant problems research reflects four key trends: Industry 4.0 integration, sustainability considerations, the development of hybrid algorithms and human-centered manufacturing. These developments reflect the field of evolution toward more comprehensive and practical scheduling solutions, incorporating environmental, worker-related, and efficiency considerations. Future research opportunities lie in addressing emerging challenges in smart manufacturing environments, particularly in areas of real-time adaptation and system resilience.

5 Metaheuristic algorithm in job-shop scheduling problems

Job-shop scheduling problems has established itself as a cornerstone of modern manufacturing, particularly valued for its flexibility and customization capabilities. Its applications span diverse industrial sectors, including precision machinery, aerospace manufacturing, and foundry operations (Tan et al., 2021; Wu et al., 2018; Gong et al., 2020). However, this versatility introduces significant complexity in resource utilization and productivity optimization, making it a focal point for academic research.

To comprehensively analyze the research landscape, we conducted a systematic literature review using Web of Science from 2000 to 2024, focusing on metaheuristic algorithms in multi-objective job-shop scheduling problems from 2015 to 2024.[2]

The analysis reveals several significant trends in job-shop scheduling problems research. As illustrated in Figure 11, there has been a marked increase in research activity since 2021, with sustained growth in publications across various JSP variants. Figure 12 demonstrates that Flexible Job-shop Scheduling Problem (FJSP) and Distributed Job-shop Scheduling Problem (Dist.JSP) have emerged as dominant research areas, likely due to their enhanced adaptability to modern production environments characterized by increasing complexity and variability.

5.1 Job-shop scheduling

JSP, classified as NP-hard, finds widespread application in various industrial contexts, from automotive assembly to airport operations. As illustrated in Figure 13, research interest in JSP has shown significant growth since 2016, peaking in 2021, with continued momentum expected in this field.

5.1.1 Problem model

To address the JSP, the following factors are typically considered:

(1) Processing sequence of jobs: each job must be processed on different machines in a specific order;
(2) Machine availability: some machines may be unavailable at specific times, which affects the scheduling of jobs;
(3) Job arrival time: certain jobs may not arrive at the workshop simultaneously, and their arrival times can influence the scheduling plan;
(4) Priority: some jobs may have higher priority than others, requiring preferential scheduling.

This chapter establishes a mathematical model with the objectives of minimizing makespan and total overtime. Table 6 presents the parameters and variables of the model.

$$\min f_1 = \sum_{i=1}^{n} \sum_{j=1}^{n_i} \sum_{q=1}^{t} x_{i,j,q} P_{i,j,q} o_q \tag{65}$$

$$\min f_2 = C_{\max} \tag{66}$$

$$\text{s.t. } C_i - d_i < 0, \forall i \tag{67}$$

$$S_{i,j} + P_{ij} < S_{i,j+1}, \forall i, j \tag{68}$$

$$S_{i,j} > 0, \forall i, j \tag{69}$$

$$S_{i,j} + P_{ij} = C_{i,j}, \forall i, j \tag{70}$$

$$\sum_{q=1}^{t} x_{i,j,q} > 0, \forall i, j \tag{71}$$

$$\sum_{q=1}^{i} P_{i,j,q} = P_{ij}, \forall i, j \tag{72}$$

$$\sum_{k=1}^{m} y_{i,j,k} = 1, \forall i, j \tag{73}$$

$$S_{i,j,k} + P_{i,j,k} < S_{n,l,k} + A\left(1 - z_{i,j,h,l,k}\right), \forall i, j, h, k, l \tag{74}$$

Among them, Equations 65, 66 represent the two objective functions, which are minimizing the total overtime and minimizing the makespan, respectively. Equation 67 defines the no-tardiness constraint. Equations 68, 69 constrain that an operation of a job can only start after its immediately preceding operation has been completed, and its start time must be no earlier than time zero. Equation 70 ensures that an operation cannot be interrupted once it starts. Equations 71, 72 indicate that an operation can be processed within one or more time intervals, with its processing time equal to the total time spent across these intervals. Equations 73, 74 constrain that each operation can be processed on only one machine, and each machine can process only one operation at a time.

5.1.2 Solution algorithm

GA have been extensively applied to address the challenges of the classical JSP (Cheng et al., 1996; Cheng et al., 1999). Davis proposed using GA to solve the classical JSP (Davis, 2014). Gonçalves et al. (2005) introduced a hybrid GA to tackle JSP. This method utilizes a chromosome representation based on random keys, where schedules are constructed according to priorities derived from GA. To enhance solution quality, parameterized active schedules are first generated, followed by the application of a local search heuristic for further optimization. Momenikorbekandi and Abbod (2023) proposed a metaheuristic hybrid Parthenogenetic Algorithm (PGA) for traditional JSP with the objective of minimizing makespan and delay time. The algorithm combines with ethnic selection GA and PGA to improve the search efficiency and the quality of the solution by introducing a racial selection mechanism and multiple selection operators. The parthenogenetic algorithm employs exchange, inversion, and insertion operators, while population selection utilizes four different selection operators: random selection, roulette wheel selection, sexual selection, and aging selection. Abedi et al. (2020) concerned with the fact that the speed of the machine has an impact on the productivity, considered JSP that carry out the necessary maintenance activities during the scheduling process. The problem with the objective of minimizing the total weighted delay time and the total energy consumption also determines the appropriate speed of the machine and the location of the maintenance activities. They proposed a multi-cluster, multi-objective MA. The algorithm assigns

[2] The search query was structured as follows: TS=((metaheuristic* OR "meta-heuristic*" OR "nature-inspired algorithm*" OR "computational intelligence" OR "evolutionary algorithm*" OR "genetic algorithm*" OR "differential evolution" OR "evolution* strategy" OR "evolution* programming" OR "swarm intelligence" OR "particle swarm optimization" OR "ant colony optimization" OR "artificial bee colony" OR "firefly algorithm" OR "bat algorithm" OR "cuckoo search" OR "simulated annealing" OR "tabu search" OR "harmony search" OR "memetic algorithm*" OR "scatter search" OR "variable neighbourhood search" OR "iterated local search") AND ("multi-objective" OR multiobjective OR "multi-criteria" OR multicriteria OR Pareto) AND ("*job shop" OR "*job-shop" OR "*jobshop")) AND PY=(2000–2024) AND DT=(Article).

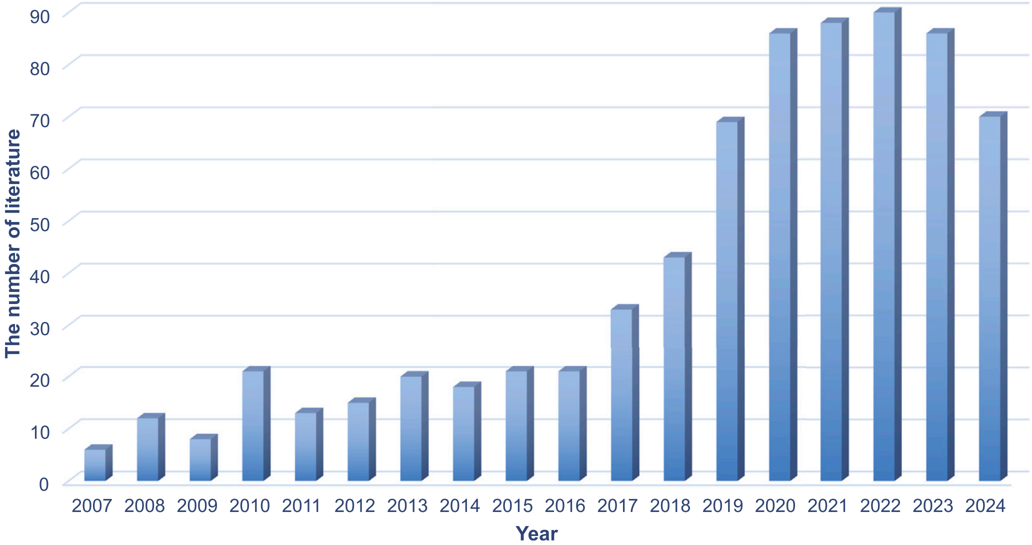

FIGURE 11
Statistics on the number of metaheuristic algorithms solving muti-objective JSP and variants problems.

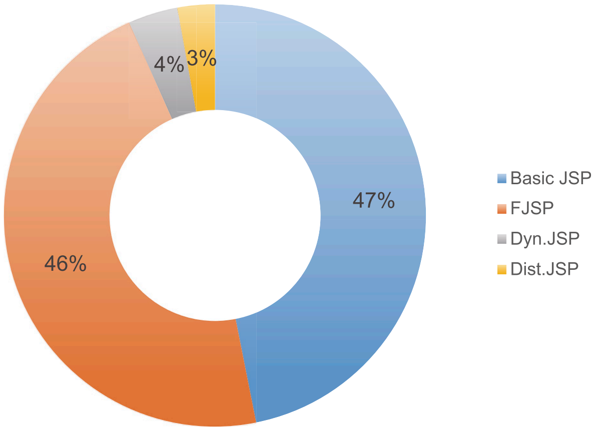

FIGURE 12
Proportion of metaheuristic algorithms solving muti-objective JSP and variant problems posted.

solutions to different sub-populations, and to better cover the solution space, the algorithm combines local search and advanced goal-oriented local search methods. In addition, an efficient non-dominated sorting method is developed for multi-objective optimization. For variable batch JSP, Xie et al. (2023) designed a decomposition-based multi-objective Jaya algorithm. The algorithm aims to achieve minimize tardiness and the least number of transferred sub-batches, based on which it combines forward and backward decoding strategies, problem-specific Jaya updating mechanisms, and local search strategies in order to better balance the exploration and exploitation of solutions. With the development of AI, robots provide new ideas to further improve the productivity of factories. Li et al. (2020) discussed JSP in a robotic cell, where the transport of jobs is handled by robots and the jobs need to be completed within a time window. They used a mixed integer planning model to formulate the problem and proposed a Teaching-based Learning Algorithm (TLA). The algorithm is able to simultaneously determine the operation assignments, the robot assignments to be used for the transport operations, and the robot movement order, aiming to minimize makespan and the overall early arrival and delay rates.

Nowadays, with the growing impact of the Industry 5.0 trend on the manufacturing industry, more and more scholars are exploring the constraints of environmental protection and worker scheduling in JSP as well. González et al. (2022) investigated the problem of minimizing both makespan and energy consumption in classical JSP. The energy model considered allows the machine to be in five states, i.e., off, standby, idle, setup, and processing. They proposed an NSGA-II based evolutionary algorithm combining local and heuristic search. Wei et al. (2021) investigated the problem of simultaneously minimizing makespan, total weighted tardiness and precocity, and non-processing energy consumption in JSP by proposing a algorithm based on unified NSGA. The algorithm incorporates a heuristic that incorporates MinMax and NEH for population initialisation. The algorithm is capable of generating a set of Pareto optimal solutions that enable production managers to select appropriate scheduling schemes based on priorities. González-Rodríguez et al. (2020) solved the fuzzy JSP by referencing fuzzy sets to deal with processing time uncertainty with the objective of minimizing makespan and total non-processing energy. They used a NSGA-II based approach combined with heuristic decoding operators that improves the robustness and applicability of the algorithm in practical applications. Since the processing and handling time of a job is affected by a number of uncertainties in the actual production process, Afsar et al. (2022) proposed a new hybrid metaheuristic approach with the objective of minimizing makespan and minimizing non-processing energy consumption. The method combines the design principles of MA, including an evolutionary component based on explicitness, a forbidden search for makespan and a heuristic search for non-processing energy, as well as a post-

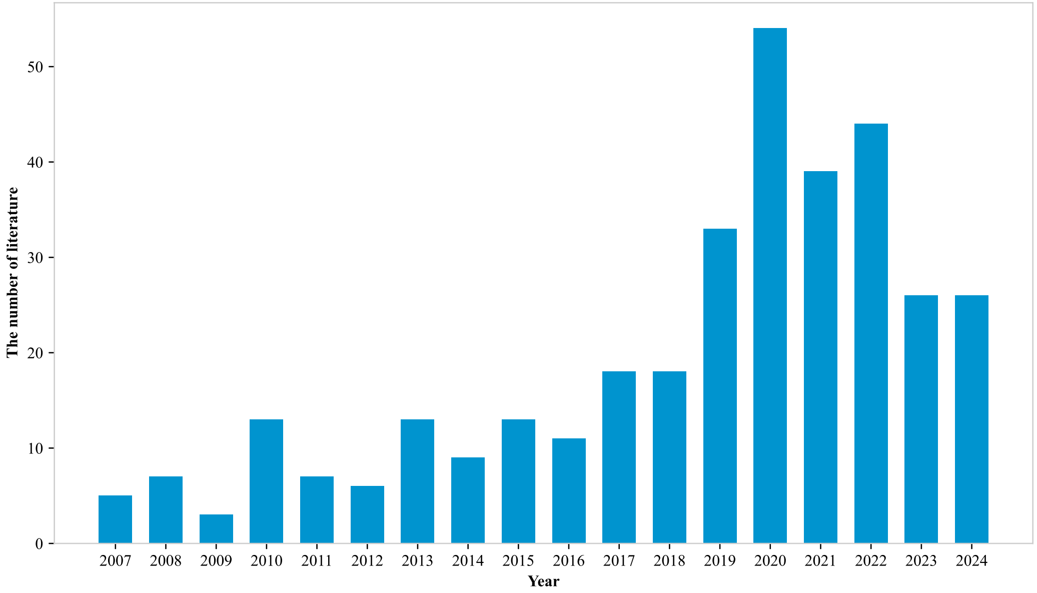

FIGURE 13
The number of metaheuristic algorithms solving muti-objective JSP literature over the years.

processing optimization based on linear programming. The collaborate of multiple algorithms is used to achieve reduced energy consumption and improved environmental sustainability without compromising service levels.

Zhang et al. (2024a) investigated a multi-objective JSP for manual loading and unloading tasks considering the human factor, and proposed a grid-based ABC with the goal of minimizing makespan and total workload. The algorithm combines the artificial bee colony algorithm and the grid technique to reduce the complexity of the problem using a decomposition method with earliest and shortest scheduling rule, and divides the target space into multiple cells by constructing a grid coordinate system to facilitate the localisation and evaluation of individuals. The algorithm also designed to generate observer bees based on a mathematical formulation of priority weights to improve information utilisation and algorithm exploration and development. Efficiency in the manufacturing process is also limited by worker's ability to operate the machine, his work capacity, and the worker's fatigue level, etc. Rodríguez-Espinosa et al. (2024) investigated multi-objective JSP considering overtime work in a make to order manufacturing company, aiming to minimize the total tardiness and overtime costs. They proposed an enhanced NSGA-II. To accelerate convergence, a problem-specific two-stage decoding scheme was designed and an adaptive mechanism was employed to maintain the stability of global convergence. In addition, a local search procedure is introduced to enhance solution exploration.

Recent advances in JSP research have addressed various practical considerations. These developments reflect the evolving nature of JSP research, incorporating increasingly complex real-world constraints while maintaining computational efficiency and practical applicability.

5.2 Flexible job-shop scheduling

FJSP represents a complex manufacturing scenario where each job comprises a sequence of operations with specific processing constraints. The defining characteristic of FJSP is its dual-layer decision making requirement: machine selection and process sequencing. Operations must follow sequential constraints and are non-preemptable, while each operation can be processed on multiple capable machines.

As shown in Figure 14, research interest in FJSP has grown significantly since 2019, indicating its continued relevance in modern manufacturing systems.

5.2.1 Problem model

Addressing FJSP requires not only determining the sequence of jobs but also assigning each operation to an appropriate processing machine, resulting in a complex set of constraints. This section presents a mathematical model formulated with the dual objectives of minimizing the makespan and the total machine delay time. Table 7 provides a detailed definition of the parameters for this problem.

$$\min f_1 : C_{\max} = \max_{1 \le i \le N} c_{iN_i} \tag{75}$$

$$\min f_2 : T_d = \sum_{i=1}^{M} \left[\left(t_{j1} - q_{1j} - 0 \right) + \sum_{v=y}^{M_j} \left(t_{jv} - q_{vj} - t_{j(v-1)} \right) \right] \tag{76}$$

$$\text{s.t.} \quad c_{ik} - c_{i(k-1)} \ge x_{ikj} p_{ijk}, \forall i, j; k = 2, \ldots, N_i \tag{77}$$

$$\sum_{j \in A_k} x_{ikj} = 1, \forall k, i \tag{78}$$

$$c_{hg} - c_{ik} \ge t_{hgj}, \forall (g, h), (i, k), j \tag{79}$$

$$c_{ik} - c_{hg} \ge t_{ikj}, \forall (g, h), (i, k), j \tag{80}$$

$$t_{jv} - t_{j(v-1)} \ge q_{vj}, \forall j, \quad v = 2, \ldots, M_j \tag{81}$$

TABLE 6 Parameter descriptions.

Parameter	Statement
i, h	Job index, $i, \quad h = 1, 2, \ldots, n$
j, l	Operation index, $j, \quad l = 1, 2, \ldots, n$
k	Machine index, $k = 1, 2, \ldots, m$
q	Time interval index, $q = 1, 2, \ldots, t$
n	Number of jobs
n_i	Number of operations for job J_i
m	Number of machines
t	Number of available time intervals
$C_{\max}$	Maximum completion time for all jobs
O_{ij}	Operation j of job J_i
P_{ij}	Processing time of operation O_{ij}
C_{ij}	Completion time of operation O_{ij}
C_i	Completion time of job J_i
d_i	Delivery date of job J_i
$P_{ij,q}$	Time spent on operation O_{ij} in time interval q
F_i	Urgency factor for job J_i
P_{ijk}	Time spent on operation O_{ij} on machine k
S_{ijk}	Start time of operation O_{ij} on machine k
A	A sufficiently large positive number
s_{ij}	Start time of operation O_{ij}
$x_{i,j,q}$	Binary variable, 1 if operation O_{ij} is processed in time interval q, 0 otherwise
$x_{i,j,k}$	Binary variable, 1 if operation O_{ij} on machine k precedes operation $O_{h,l}$, 0 otherwise
$y_{i,j,k}$	Binary variable,1 if operation O_{ij} is processed on machine k, 0 otherwise
o_q	Binary variable,1 if the time interval is an overtime interval, 0 otherwise

$$x_{ikj} \in \{0,1\}, \forall j,k,i \tag{82}$$

$$c_{ik} \geq 0, \forall k,i \tag{83}$$

$$t_{jv} \geq 0, \forall j,v \tag{84}$$

Equation 75, 76 represents the objective function for minimizing the makespan and the total machine delay time. In Equation 76, if a solution assigns only one operation to machine j, then $T_d = t_{j1} - q_{1j} - 0$. If no operation is assigned to machine j, then $T_d = 0$. Equation 77 specifies the precedence of operations within the same job, where operation $o_{i(k-1)}$ must be completed before operation o_{ik} can start. Equation 78 stipulates that when processing an operation, a machine must be selected from the available machine set $A(i,k)$ for operation o_{ik}, and only one machine can be selected. For two operations, o_{ik} and o_{hg}, assigned to the same machine j, if o_{ik} arrives before o_{hg}, the start time of o_{ik} must be greater than or equal to the completion time of o_{hg}, as shown in Equation 79. Similarly, in Equation 80, if o_{hg} arrives before o_{ik}, the start time of o_{ik} must also be greater than or equal to the completion time of o_{hg}. In the same manner, Equations 81 specify that all operations assigned to the same machine j must satisfy the rules outlined in Equations 79, 80. Equation 82 defines the range of values for the machine selection decision variables, and Equations 83, 84 ensure that the completion time of any operation is greater than or equal to zero.

5.2.2 Solution algorithm

Gao et al. (2008) developed a hybrid GA to solve the FJSP with three objectives: minimizing the makespan, minimizing the maximum machine workload, and minimizing the total workload. Pezzella et al. (2008) introduced an integrated approach to tackle the FJSP, which combines various strategies for initial population generation, individual selection for reproduction, and the creation of new offspring. Shahsavari-Pour and Ghasemishabankareh (2013) proposed a hybrid algorithm combining the advantages of GA and SA for solving the multi-objective function optimization problem in FJSP. The algorithm directly deals with three objectives through a Pareto optimal solution approach: minimizing makespan, minimizing machine workload, and minimizing the total workload of all machines. The hybrid algorithm is able to provide higher quality solutions than existing methods in solving multi-objective problems while reducing computation time. Zhang et al. (2020) investigated FJSP with two neighbouring work areas for a problem involving the production of basic parts and subsequent assembly operations. In order to minimize makespan, total tardiness time and total workload, a distributed ACO is proposed to explore the Pareto frontier. In addition, the method was successfully applied to different scenarios of the ball valve production scheduling problem. Liu et al. (2021) provided a new algorithm which combine with VNS and GA for solving FJSP. The algorithm aims to improve productivity, reduce cost and energy consumption, and it proposes several improvement measures for the problems of slow convergence and low accuracy of traditional GA in solving FJSP. These include a mutation operator based on PSO, a hybrid heuristic initialisation strategy, a VNS based on an improved multilevel neighbourhood structure, and a real-number based chromosome representation, encoding, decoding and crossover method. These improvements improve the convergence performance and solution accuracy of the algorithm.

Li et al. (2014) proposed a novel discrete ABC in solving a multi-objective FJSP containing maintenance activities. The algorithm aims to optimize three main performance metrics: minimizing makespan, total machine workload and critical machine workload. The algorithm employs a unique problem solution representation, the application of TS for local search, an efficient initialisation scheme, an adaptive strategy to enhance the exploratory capabilities of the algorithm, and a specially designed decoding method to handle maintenance activities. An external Pareto archive set is employed for recording all the non-dominated solutions found. Kacem et al. (2002) proposed a Pareto-based hybrid approach combining fuzzy logic and EAs to address the FJSP. This method leverages the knowledge representation capability of fuzzy logic and the adaptability of EAs to optimize three objectives: minimizing the makespan, the total machine workload, and the workload of the most heavily

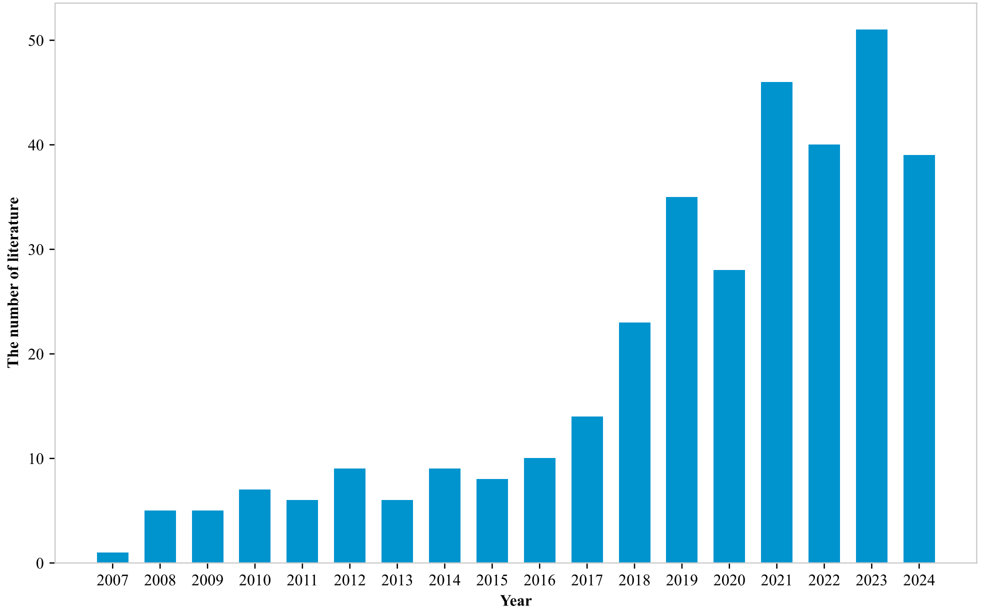

FIGURE 14
The number of literature on metaheuristic algorithms solving muti-objective FJSP over the years.

TABLE 7 Parameters descriptions.

Parameter	Statement
i, h	Job index
j	Machine index
k, g	Operation sequence index of the job
v	Operation sequence index on the machine
N	Total number of jobs
M	Total number of machines
N_i	Total number of operations for job i
M_j	Total number of operations processed on machine j
$A(i,k)$	The set of available machines for the k-th operation of job i
o_{ik}	The k-th operation of job i
p_{ijk}	The processing time of the k-th operation of job i on machine j
w_{jv}	The v-th operation processed on machine j
q_{vj}	The processing time of the v-th operation processed on machine j
x_{ikj}	If machine j is selected to process operation O_{ik}, it is 1; otherwise, it is 0
c_{ik}	If operation O_{ik} is completed, it is 1; otherwise, it is 0
t_{jv}	If operation w_{jv} is completed, it is 1; otherwise, it is 0

utilized machine. Basiri et al. (2020) designed a hybrid intelligent algorithm for FJSP dealing with fuzzy time. The algorithm is optimized based on the Pareto solution set, considering constraints such as the number of un-predetermined parallel machines, sequence-dependent setup times, re-entrant workflows and fuzzy variables, and combines a simple weighted sum method and a technology preference method.

Industry 5.0 further extends the beautiful vision of Industry 4.0 for energy saving by focusing on sustainable development and environmental responsibility while emphasising on human-centred production methods. Luan et al. (2023) developed an enhanced NSGA-II for multi-objective energy-saving FJSP with the objective of minimizing makespan, total delay time and total energy consumption. This algorithm improves the accuracy of the current solution by performing different neighbourhood search procedures in the sparse solution space by performing different neighbourhood search procedures to increase the diversity of the population of children in the local search, thus improving the accuracy of the current solution. In addition, a weighted approach is used to select the desired compromise solution from the set of Pareto solutions in order to achieve a balance between multiple objectives. Burmeister et al. (2023) focused on FJSP with dynamic energy costs and proposed a multi-objective modelling algorithm based on NSGA-II. The algorithm aims to minimize makespan and energy cost, and bridges the gap in the existing literature that is limited to coarse-grained time periods by taking into account frequently changing real-time energy tariffs. He et al. (2022) focusing on FJSP with dual resource constraints of machines and workers, constructed a model considering both machine and worker constraints, with the aim of minimizing makespan and total delay. They developed an improved African Vultures Optimization Algorithm. A neighbourhood search operation was designed to further minimize makespan and total delay. Zhang et al. (2023b) focused on the post processing shop for parallel disassembly/reassembly workstations and FJSP, which needs to decide on the assignment and sequence of disassembly/reassembly jobs, the as well as the sequence of operations and workstation assignments for

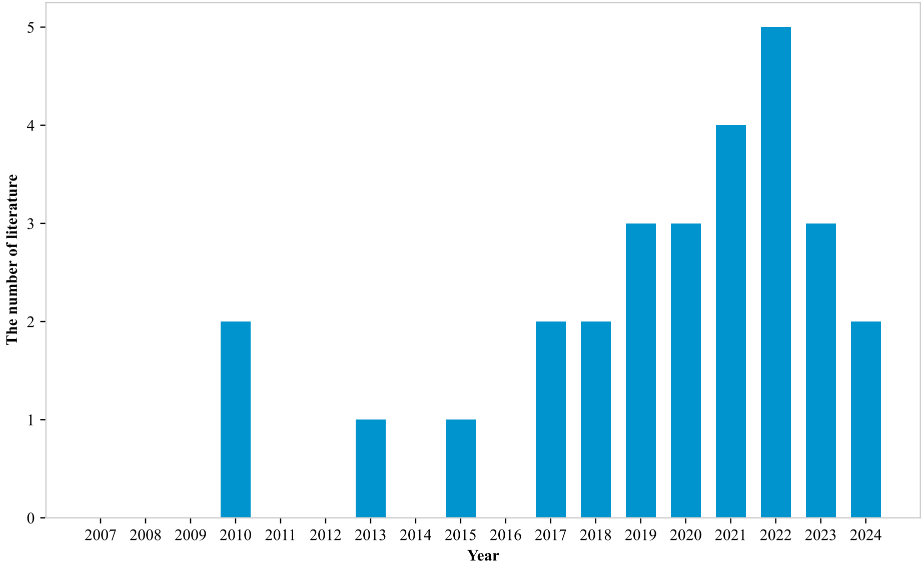

FIGURE 15
The number of literature on metaheuristic algorithms solving muti-objective Dyn.JSP and variant problems over the years.

reprocessing operations. An enhanced Grey Wolf Optimizer is proposed to address the multi-objective FJSP in remanufacturing systems, aiming to minimize both makespan and energy consumption. The algorithm improves accuracy, computational efficiency, solution stability, and convergence performance by incorporating multiple local neighborhood search strategies, stochastic interference methods, and weighted distance updating mechanisms.

Tan et al. (2021) proposed an optimization model considering worker fatigue for FJSP with dual resource constraints, aiming to simultaneously reduce worker fatigue and increase productivity through joint scheduling of machines and workers. The model aims at minimizing maximum worker fatigue and makespan. This research not only provides practical scheduling tools for decision makers, but also brings fatigue awareness to machine and worker constrained flexible manufacturing systems, facilitating both worker health and productivity. Vital-Soto et al. (2023) developed an elite NSGA-II with innovative operators for a dual resource constrained FJSP considering machine and worker allocation and sorting. The algorithm is able to provide a set of Pareto optimal solutions that help the decision maker to evaluate the trade-offs between different objectives. The algorithm demonstrates its advantages in defining daily schedules, adjusting schedules, priority management, on-time delivery and balancing worker workloads.

With the advent of Industry 5.0, the field of FJSP is increasingly oriented toward energy efficiency and sustainable development. Researchers are placing greater emphasis on integrating factors such as energy consumption, worker fatigue, and environmental responsibility into the scheduling process, aiming to achieve sustainability objectives alongside enhanced production efficiency. Current studies employ various hybrid algorithms designed to address the limitations of traditional methods, thereby improving convergence speed and solution accuracy. This trend indicates that future research will place greater importance on the synergy between algorithmic innovation and practical applications, underscoring the necessity of developing efficient and environmentally friendly scheduling solutions in complex dynamic environments.

5.3 Dynamic job-shop scheduling

Manufacturing environments inherently contain uncertainties, making dynamic scheduling a crucial research direction that closely aligns with real-world production scenarios. Dynamic events in manufacturing can be categorized into four main types: job-related events, operations-related events, machine-related events and other operational events.

Within this domain, the Dynamic Flexible Job-shop Scheduling Problem (Dyn.FJSP) emerged as an extension of Dynamic Job-shop Scheduling Problem (Dyn.JSP), incorporating machine selection flexibility.

As illustrated in Figure 15, research interest in dynamic scheduling problems shows an upward trend, reflecting their increasing practical significance. The ability to respond rapidly to environmental changes while generating optimal scheduling solutions is paramount in this context.

5.3.1 Problem model

The Dyn.FJSP with new job insertion considered in this chapter can be defined in detail as follows. There are n sequentially arriving jobs $J = \{J_1, J_2, \ldots, J_n\}$ to be processed on m machines $M = \{M_1, M_2, \ldots, M_m\}$. Each job J_i consists of n_i operations, where O_{ij} represents the j-th operation of job J_i. Each operation O_{ij} can be processed on any machine M_k selected from its compatible machine set $M_{ij} \subseteq M$. The processing time of operation O_{ij} on machine M_k is denoted as t_{ijk}. The arrival time and due date of job J_i are represented as A_i and D_i, respectively. The actual completion time of operation O_{ij} is denoted as C_{ij}. The urgency level of job J_i is indicated by P_{ri}, where a higher urgency level results in greater penalties for delays.

TABLE 8 Parameter descriptions.

Parameter	Statement
n	Total number of jobs
m	Total number of machines
J_i	Job J_i
n_i	Total number of operations belonging to job J_i
M_k	Machine M_k
O_{ij}	Operation j of job J_i
M_{ij}	Set of available machines for operation O_{ij}
t_{ijk}	Processing time of operation O_{ij} on machine M_k
A_i	Arrival time of job J_i
D_i	Due date of job J_i
P_{ri}	Urgency degree of job J_i
C_{ij}	The completion time of operation O_{ij}
X_{ijk}	If operation O_{ij} is assigned on machine M_k, it is 1; otherwise it is 0
Y_{ijhg}	If operation O_{ij} is a predecessor of O_{hg}, it is 1; If operation O_{ij} is a successor of O_{hg}, it is −1

This chapter establishes a mathematical model with the objectives of simultaneously minimizing the total weighted tardiness and maximizing the average machine utilization. Table 8 presents the detailed parameters of this model.

$$\min TWT = \sum_{i=1}^{n} \max\left(C_{i,n_i} - D_i, 0\right) \cdot Pr_i \tag{85}$$

$$\min \bar{U}_{\text{ave}} = \left(\frac{1}{m}\sum_{k=1}^{m} \frac{\sum_{i=1}^{n}\sum_{j=1}^{n_i} t_{i,j,k}X_{i,j,k}}{\sum_{k=1}^{m}\max_i C_{i,n_i}\cdot X_{i,n_i,k}}\right) \tag{86}$$

$$\text{s.t. } C_{i,0} = 0, \quad C_{i,j} > 0, \forall i, j \tag{87}$$

$$\sum_{k\in M_{i,j}} X_{i,j,k} = 1, \forall i, j \tag{88}$$

$$\left(C_{i,1} - t_{i,1,k} - A_i\right)X_{i,1,k} \geq 0, \forall i, k \tag{89}$$

$$\left(C_{i,j} - t_{i,j,k} - C_{i,j-1}\right)X_{i,j,k} \geq 0, \forall i, j, k \tag{90}$$

$$\begin{aligned}&\left(C_{h,g} - t_{h,g,k} - C_{i,j}\right)X_{i,j,k}X_{h,g,k}\left(Y_{i,j,h,g} + 1\right)\\ &\quad + \left(C_{i,j} - t_{i,j,k} - C_{h,g}\right)X_{i,j,k}\left(1 - Y_{i,j,h,g}\right) \geq 0, \forall i, j, h, g, k\end{aligned} \tag{91}$$

Equation 85 represents the total weighted tardiness of all jobs, where urgency is used as the weight factor (i.e., the penalty factor) for delays. Equation 86 represents the reciprocal of the average machine utilization. Equation 87 indicates that the completion time of each operation must be non-negative. Equation 88 ensures that each operation can only be processed on one available machine. Equation 89 ensures that a job can only be processed after its arrival time. Equation 90 guarantees precedence constraints. Equation 91 ensures capacity constraints are satisfied.

5.3.2 Solution algorithm

When solving scheduling problems under the influence of dynamic events, it is particularly important to design algorithms that can respond quickly to changes in the environment and generate reasonable scheduling schemes in immediately. Chryssolouris and Subramaniam (2001) designed a GA for the Dyn.JSP with the objectives of average job tardiness and average job cost. In scheduling considering dynamic events, most scholars use rescheduling method, i.e., the newly inserted jobs are merged with the existing jobs to reschedule, this scheduling method will make the new scheduling scheme completely different from the old scheduling scheme. In addition, this rescheduling method may lead to stability degradation. Rangsaritratsamee et al. (2004) in order to improve the stability degradation caused by rescheduling and other problems, they proposed a multi-objective rescheduling method based on GA and local search algorithm. The method adopts stability as an additional performance measure based on the objective of minimizing makespan and delay, aiming to balance efficiency and stability. Random job arrivals are the most common class of dynamic events in the dynamic scheduling. Wang et al. (2019) proposed a rescheduling method that combines performance and stability considerations to address this type of Dyn.JSP. The method aims to minimize the discontinuity rate of new jobs, makespan deviation of the initial schedule and order deviation on the machine. Four matching strategies are modified to determine the rescheduling range. An improved PSO is proposed, which consists of a decoding scheme that considers machine unavailability constraints, a population initialisation method based on a new transformation mechanism, and a particle movement method that introduces positional variations and random inertia weights.

Compared to Dyn.JSP, Dyn.FJSP is more challenging in that it needs to consider not only the impact of dynamic events in the manufacturing process, but also the flexibility that jobs can be assigned to multiple machines for processing. Fattahi and Fallahi (2010) proposed a multi-objective mathematical model and a metaheuristic algorithm based on GA to address Dyn.FJSP, aiming at balancing the efficiency and stability of the schedule. Zhang et al. (2013) proposed a rescheduling approach based on hybrid GA and TS to solve Dyn.FJSP. They introduced a simulator to deal with the complexity of real-time events and the difficulty of representing them accurately in a mathematical model. Liu et al. (2024) proposed a multi-objective ALNS for the Dyn.FJSP of self-driving transportation in smart factories. The algorithm adopts a proactive responsive approach to cope with the dynamic interference of new job insertion. The algorithm formulates a two stage multi-objective mixed-integer planning model: the first stage aims to minimize makespan and workload imbalance; the second stage introduces instability minimisation to deal with the effects of disturbances.

From the above research, it can be concluded that in the field of dynamic scheduling, the design of algorithms that can quickly respond to dynamic events is particularly important, as it directly impacts the efficiency and stability of manufacturing environments. Researchers are focusing on employing rescheduling strategies to balance efficiency and stability, particularly in the context of multi-objective optimization, where the goal is to simultaneously enhance scheduling efficiency and robustness. The trend towards the adoption of hybrid algorithms and innovative strategies underscores the critical role of algorithmic innovation in improving scheduling performance.

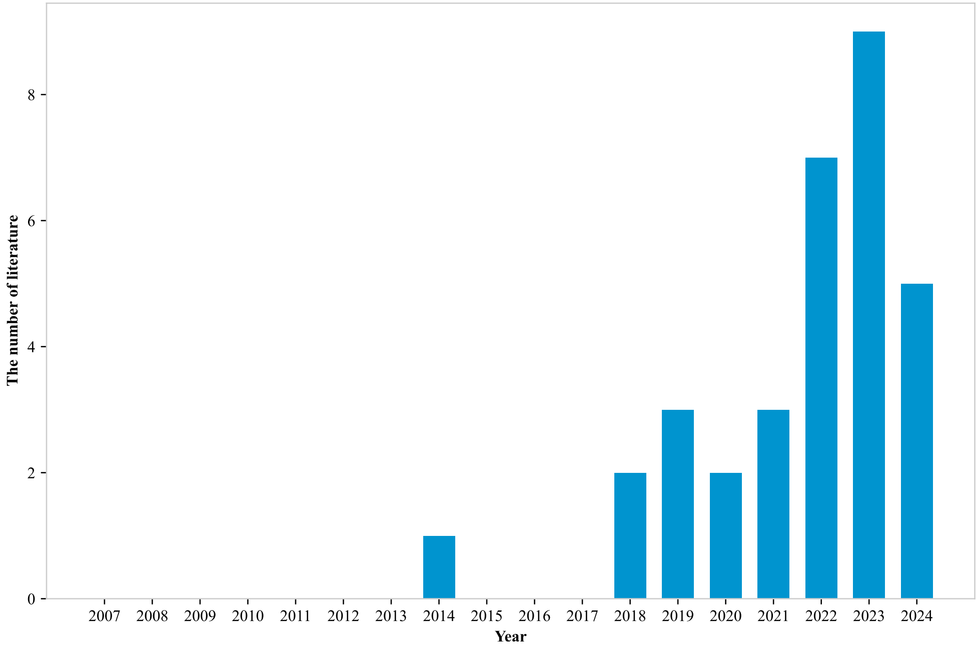

FIGURE 16
The number of literature on metaheuristic algorithms solving muti-objective Dist.JSP and variant problems over the years.

5.4 Distributed job-shop scheduling

Modern manufacturing systems have evolved into complex networks of multiple interconnected job-shop scheduling problems, each characterized by distinct equipment, process capabilities, and resources. This evolution has exposed the limitations of traditional centralized scheduling approaches, highlighting the growing importance of distributed scheduling solutions. The field encompasses two main variants: Dist.JSP and its flexible counterpart, Distributed Flexible Job-shop Scheduling Problem (Dist.FJSP), distinguished by machine optionality.

As illustrated in Figure 16, research interest in Dist.JSP has shown a consistent upward trend, driven by industry demands for enhanced productivity and resource optimization. This trajectory suggests continued growth in this research domain.

5.4.1 Problem model

In production and manufacturing systems, flexible scheduling strategies play a critical role in adapting to dynamic market demands and enhancing production flexibility. As a representative problem of multi-factory scheduling, Dist.FJSP is characterized by allowing each sub-factory to operate as an independent flexible job shop, responsible for processing specific parts of the overall production tasks. The resolution of Dist.FJSP involves addressing three key subproblems:

(1) The allocation of parts to processing factories;
(2) The matching of operations to processing machines;
(3) The optimization of operation scheduling within individual factories.

This chapter uses Dist.FJSP as an example to formulate a mathematical model aimed at minimizing makespan and total energy consumption. Table 9 presents the detailed parameters of this model.

$$\min f_1 = \max\{T_i \mid i = 1, 2, \ldots, n\} \tag{92}$$

$$\begin{aligned} \min\ f_2 &= E^{AH} + E^{AF} + E^{AM} \\ &= \sum_{i=1}^{n} \sum_{j=1}^{p_i} \sum_{a=1}^{f} \sum_{\beta=1}^{\gamma_a} T_{ij\alpha\beta} E_{ij\alpha\beta} Y_{ij\alpha\beta} + \sum_{i=1}^{n} \sum_{j=1}^{p} T^F E^U Y_{ij}^F \\ &\quad + \sum_{i=1}^{n} \sum_{j=1}^{p} T^M E^U Y_{ij}^M \end{aligned} \tag{93}$$

$$\text{s.t. } T_{ij} - T_{i(j-1)} \geq T_{ij\alpha\beta} Y_{ij\alpha\beta}, \forall i, j, \alpha, \beta \tag{94}$$

$$\begin{aligned} &\left[\left(T_{ks} - T_{ij} - T_{ks\alpha\beta}\right) Y_{ks\alpha\beta} Y_{ij\alpha\beta} \geq 0\right] \cup \\ &\left[\left(T_{ij} - T_{ks} - T_{ij\alpha\beta}\right) Y_{ij\alpha\beta} Y_{ks\alpha\beta} \geq 0\right], \forall i, j, k, s, \alpha, \beta \end{aligned} \tag{95}$$

$$\sum_{\alpha=1}^{f} \sum_{\beta=1}^{Y_a} Y_{ij\alpha\beta} = 1, \forall i, j \tag{96}$$

$$0 \leq \left(Y_{ij}^F + Y_{ij}^M\right) \leq 1, \forall i, j \tag{97}$$

Equations 92, 93 define the optimization objectives of the problem, which are minimizing the makespan and total energy consumption, respectively. Equation 94 ensures that each operation satisfies the precedence constraints. Equation 95 guarantees that each machine processes at most one operation at a time. Equation 96 ensures that each operation is processed on one machine in a single factory only once. Equation 97 restricts each operation to undergo only one transportation, either within or between factories.

5.4.2 Solution algorithm

Li et al. (2018a) proposed a Pareto based hybrid TS for multi-objective Dist.FJSP that simultaneously minimizes four objectives: makespan, maximum workload, total workload, and early arrival/late arrival criteria. The algorithm employs multiple methods to

TABLE 9 Parameter definitions.

Parameter	Statement
n	Total number of jobs
p_i	Total number of operations for job i
f	Total number of factories
Y_a	Total number of machines in factory α
i, k	Job indices, $i, \quad k = 1, 2, \ldots, n$
j, s	Operation indices, $j, \quad s = 1, 2, \ldots, p_i$
α	Factory index, $\alpha = 1, 2, \ldots, f$
β	Machine index, $\beta = 1, 2, \ldots, Y_a$
m	Total number of machines across all factories
L_a	Workload of factory α
$u_{ij}\,(u_{ks})$	The j-th (or s-th) operation of job i (or k)
$T_{ij\alpha\beta}\,(T_{ks\alpha\beta})$	Processing time of operation u_{ij} on machine β in factory α
T^M	Transfer time of an operation between different machines within the same factory
T^F	Transfer time of an operation between different factories
$T_i\,(T_k)$	Completion time of job i (or k)
$T_{ij}\,(T_{ks})$	Completion time of operation u_{ij} (or u_{ks})
$E_{ij\alpha\beta}$	Energy consumption rate for operation u_{ij} on machine β in factory α
E^{AU}	Energy consumption rate for transportation
$Y_{ij\alpha\beta}\,(Y_{ks\alpha\beta})$	Binary variable, equals 1 if operation u_{ij} is assigned to machine β in factory α, otherwise 0
$Y_{ij}^M\,(Y_{ks}^M)$	Binary variable, equals 1 if operation u_{ij} is transferred between different machines within the same factory, otherwise 0
$Y_{ij}^F\,(Y_{ks}^F)$	Binary variable, equals 1 if operation u_{ij} is transferred between different factories, otherwise 0
E^{AH}	Total energy consumption for job processing
E^{AM}	Total energy consumption for transferring jobs between different machines within the same factory
E^{AF}	Total energy consumption for transferring jobs between different factories

initialise the solution set, which take into account both problem characteristics and objective characteristics. In addition, five neighbourhood structures are used to balance the exploration and exploitation of solutions and a well-designed inverse method is proposed to optimize the early arrival/late arrival criteria. Luo et al. (2020) focus on the limitation that the job operations in the traditional Dist.FJSP can not be transferred between different factories, and propose a Dist.FJSP with transfers that allow operations to be to be processed between different factories. They proposed an efficient MA for this problem that aims to minimize makespan, maximum workload and total energy consumption. The algorithm devises several crossover and mutation operators as well as three efficient neighbourhood structures are developed to extend the search space and accelerate the convergence of the solution. Jia et al. (2024) focused on the problem of integrating optimization of production and distribution in a furniture supply chain, and proposed an integrated Dist.FJSP aiming to minimize makespan and total delay. They introduced a multi-objective Brain Storming Optimization algorithm based on the construction of a mixed integer mathematical planning model.

The widespread adoption of distributed factories in large and complex manufacturing industries will inevitably lead to a series of problems such as energy consumption and environmental impacts. Xu et al. (2021) proposed a multi-objective low carbon scheduling model, which takes the minimization of makespan, total cost, carbon emission and maximization of quality as the optimization objectives. A hybrid GA with three-layer coding and TS was developed, which combines the global search capability of GA and the local search capability of TS to effectively improve the solution performance. The method not only considers the three traditional subproblems of operation sequencing, job assignment and operation to machine assignment, but also introduces the characteristics of job outsourcing, which makes it more in line with the operation of real manufacturing enterprises. Shao et al. (2022) proposed a multi-objective MA based on multiple neighbourhoods aiming to optimize the total weighted tardiness and energy consumption for an energy-efficient Dist.FJSP with variable machine speeds. The algorithm designs two genetic global search operators to enhance the solution search capability and integrates several multi-neighbourhood strategies including intra and inter factories sequence adjustment, energy saving strategies and speed adjustment to enhance the exploration capability. To address the limitation that worker arrangement is not considered in Dist.FJSP, Luo et al. (2022) investigated Dist.FJSP with worker arrangement. To solve the problem, they proposed an improved MA based on the structure of NSGA-II, aiming to minimize the makespan of the machine, the maximum workload and the workload of the workers at the same time.

With the widespread adoption of distributed factories, research has increasingly focused on issues such as low-carbon scheduling and worker arrangement to enhance the energy efficiency and environmental sustainability of scheduling processes. This trend indicates a shift towards more complex real-world applications, emphasizing the integration of algorithmic innovation with practical operations to address the challenges faced by modern manufacturing industries.

The evolution of research on job-shop scheduling problems and its variants highlights four key trends: integration with Industry 5.0, the implementation of flexible manufacturing, the emergence of distributed manufacturing, and the advancement of hybrid algorithms. These trends underscore the ongoing evolution and adaptation of modern manufacturing systems in response to increasing complexity and diversity.

6 Combining metaheuristic algorithm and reinforcement learning for shop scheduling optimization

Despite the widespread application of metaheuristics in shop scheduling optimization, these algorithms face two significant limitations. First, their computational intensity and inability to retain optimization parameters necessitate repeated initialization, resulting in extended response times. Second, their limited generalization capability often requires parameter adjustments for different problem scales, hindering algorithmic transferability. Recent advances in AI, particularly RL, offer promising solutions to these challenges (Cheng et al., 2020).

6.1 Integration approaches in flow shop scheduling

Q-learning (Clifton and Laber, 2020) has emerged as a particularly effective model-free RL algorithm, offering several advantages including environment independent learning, structural simplicity, and efficient exploration exploitation balance through ϵ-greedy strategy. The Q-learning update rule can be expressed as (Equation 98):

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)\right] \quad (98)$$

Recent research has demonstrated successful integration of Q-learning with various metaheuristic algorithms. Zhang and Cai (2023) developed a dual population GA incorporating Q-learning to minimize makespan and delayed jobs. Their approach features multiple genetic operators and a sophisticated population evaluation method, where one population employs optimal search strategies while the other follows Q-learning guidance. Tao et al. (2023) proposed a NSGA-II with Q-learning for DPFSP. Firstly, an IG is used to generate the initial solution. Then, the NSGA-II is designed to optimize the bi-objective problem, and the algorithm parameters are dynamically adjusted by Q-learning. In addition, two kind of crucial factory based local search strategies, including insertion and exchange operations, are introduced to improve the quality of the solution. Zhang et al. (2024b) focused on the DHFSP in a heterogeneous shop and designed a multi-objective MA combining PSO and local search based on Q-learning to minimize makespan and total energy consumption. The algorithm employs a multi-group PSO to improve the fast convergence performance of the multidirectional solutions on the Pareto front, and two local search strategies are designed to further enhance the quality and diversity of the solutions. Q-learning is used to guide the VNS for a better balance of exploration and exploitation.

In the context of energy-efficient scheduling, Yu H. et al. (2024) conducted comprehensive research on multi-objective DPFSP, embedding Q-learning into multiple algorithms including ABC, GA, PSO, IG, and Jaya. Their work introduced five critical neighborhood structures, with the Q-learning enhanced Jaya algorithm demonstrating superior performance. Shao et al. (2024) addressed the challenges of uncertain processing times in DHFSP through a novel energy-efficient approach. Their algorithm combines: machine position-based scheduling rules, decomposition-based construction heuristics, problem-specific search operators and a two-phase framework incorporating metatraining and adaptive search. Zhang et al. (2023a) proposed a Q-learning based MOPSO for DFSP aiming to minimize makespan and total energy consumption. The algorithm enhances the convergence speed of PSO by dividing the particles into three subgroups, enabling it to reach three different regions of the Pareto front faster. Q-learning is used to guide VNS, balancing the exploration and exploitation functions. MOPSO uses exchange sequences to update the job sequence vector, and crossover and mutation to update the plant allocation vector to improve the rationality and efficiency of the algorithm.

Research indicates that Q-learning can be integrated with various metaheuristic algorithms, such as GA and PSO, to enhance their performance in addressing complex scheduling tasks. Particularly in the domains of energy-efficient scheduling and multi-objective optimization, the application of Q-learning demonstrates a favorable balance between exploration and exploitation, facilitating faster convergence to high-quality solutions.

6.2 Integration approaches in job shop scheduling

The integration of metaheuristics and RL has shown particular promise in job shop scheduling problems scenarios (Zhang et al., 2024c). Li et al. (2023) developed a sophisticated approach for energy-efficient HFSP, considering time-sharing tariffs and disconnection strategies. Their Q-learning enhanced NSGA-II algorithm demonstrates effective balance between exploration and exploitation.

He et al. (2021) proposed an effective multi-objective Jaya algorithm. The algorithm introduces a strategy based on gray entropy parallel analysis for evaluating and selecting solutions, and an opposition learning based strategy to enhance the algorithm's search capability and convergence speed. The strategy of objection learning is integrated into the search operation and external archiving of Jaya to improve the diversity of solutions and avoid local optimum. This algorithm solves multi-objective JSP with the objective of minimizing makespan, total flow time and average tardiness. Abedi et al. (2020) studied Dyn.FJSP considering random job arrivals and machine failures with event-driven policy rescheduling. To solve the problem, they chose VNS as the rescheduling method with the objective of minimizing makespan and total delay, and dynamically updated the VNS parameters at each rescheduling point by a trained Artificial Neural Network. Li et al. (2022) proposed a RL based Multi-objective Evolutionary Algorithm Based on Decomposition for multi-objective FJSP with fuzzy processing time, aiming to minimize makespan and total machine workload. The algorithm generates high quality initial populations using three initial strategies, and employs a Q-learning based parameter adaptation strategy to guide the populations in selecting optimal parameters to increase diversity. In addition, VNS combined with RL guides the solution to choose the correct local search method.

Chen et al. (2023) proposed a predictive reactive dynamic/static rescheduling model for FJSP with ambiguous processing times, dynamic interrupts and variable processing speeds, aiming to optimize the three objectives of maximum completion time, total energy consumption and average protocol index. To address this problem, a multi-objective IA incorporating Q-learning was proposed to optimize the initial solution using an active decoding heuristic based on the interval insertion mechanism, to improve the exploration and exploitation capabilities through clone selected IA and Q-learning, respectively. Kamali et al. (2023) developed an algorithm to optimize the maximum completion time and the average protocol index for a multi-objective Dyn.FJSP, developed a bio-immune based multi-agent system that aims to simultaneously minimize the four objectives of makespan, total weighted tardiness, maximum machine workload and progress stability. In this algorithm, the immune intelligences are responsible for environment recognition, generating non-dominated schedules and selecting optimal schedules with self adaptive and flexible coordination capabilities. Tang et al. (2024) proposed an improved NSGA-III combined with RL for Dist.FJSP with transport resource constraints. The algorithm is designed with a heuristic rule based initialisation method and an IG decoding method, and the key parameters of NSGA-III are adaptively tuned using double Q-learning with an improved ϵ-greedy strategy.

In the field of job-shop scheduling problems, the integration of metaheuristic algorithms with RL has demonstrated both effectiveness and significance. By introducing novel evaluation strategies and dynamically adjusting parameters, these approaches enhance the search capabilities and convergence speeds of the algorithms. Furthermore, research addressing dynamic environments and uncertainty issues employs event-driven strategies and multi-agent systems to effectively facilitate dynamic rescheduling.

7 Conclusion and future perspectives

The evolution of manufacturing towards intelligent production has highlighted the limitations of traditional scheduling methods in addressing complex production environments. Metaheuristic algorithms have emerged as effective solution tools, demonstrating significant advantages in handling complex, multi-objective scheduling problems through their nature-inspired optimization approaches. Based on recent literature analysis, these methods have become a research hotspot in multi-objective shop scheduling, showing promising results in various applications.

7.1 Current status analysis

Under the background of Industry 4.0, most studies have concentrated on distributed and dynamic scheduling problems, with some research paying special attention to scheduling with transportation time constraints. In the era of Industry 5.0, a growing number of studies have started to consider the environmental impacts of manufacturing processes, alongside factors such as worker fatigue rates and skill levels. While the existing literature addresses various aspects, research on scheduling problems closer to real-world production scenarios, particularly in the domain of heterogeneous distributed scheduling involving workers, remains inadequate and requires further in-depth exploration.

As for research objectives, the optimization goals of multi-objective scheduling problems currently focus primarily on minimizing makespan, total energy consumption, and optimizing tardiness or earliness rates. Almost all studies emphasize makespan as a critical metric, however, the majority are limited to bi-objective optimization. Studies addressing tri-objective or multi-objective optimization remain relatively rare, indicating significant potential for advancing comprehensive optimization methods.

In addressing multi-objective scheduling problems, metaheuristic algorithms, with NSGA-II as a prominent example, dominate the field. Many studies adopt hybrid strategies that integrate multiple techniques, enhancing the exploration capabilities of algorithms while improving their exploitation of existing solutions. These approaches have achieved remarkable optimization results in multi-objective scheduling problems.

It is worth noting that most studies rely on job-based benchmark test cases for extended research on standard datasets, with only a small fraction utilizing real-world production cases. While benchmark test cases contribute to theoretical advancements, they often approach tasks at the job level, whereas actual production processes typically handle tasks at the order level. This mismatch underscores a gap between current testing methodologies and real-world production practices.

Additionally, in solving dynamic scheduling problems, metaheuristic algorithms often require frequent rescheduling, which incurs additional time costs. In the context of Industry 5.0, which emphasizes enhancing system resilience, RL has emerged as an essential complementary method. It effectively reduces the impact of frequent rescheduling while improving the overall adaptability and flexibility of scheduling systems.

7.2 Future research directions

Looking forward, several critical areas demand attention for advancing the field. First, research problems should expand beyond current paradigms to better align with actual production environments. The acceleration of Industry 5.0 necessitates increased focus on human-centric factors, including worker wellbeing and multi-skilled workforce deployment. This evolution requires a more nuanced approach to problem formulation and solution design.

The scope of optimization objectives needs broadening to reflect the multifaceted nature of modern manufacturing. While current bi-objective approaches have proven valuable, future research should embrace more comprehensive optimization objectives.

Dynamic scheduling represents another crucial frontier. Manufacturing environments face various uncertainties, from order modifications to equipment failures and workforce variations. Future research must develop robust approaches for handling these dynamic elements while maintaining system resilience. This includes designing adaptive strategies that can respond to real-time changes while ensuring production stability.

Methodologically, the integration of metaheuristic algorithms with RLpresents promising opportunities. While metaheuristics excel at finding approximate optimal solutions, they often struggle with large-scale problems and generalization. RL offers complementary strengths, particularly in strategy generalization and rapid decision-making. The integration of these two approaches holds promise for enabling researchers to develop more robust and practical scheduling solutions.

7.3 Implementation considerations

Success in advancing these research directions requires careful attention to implementation strategies. Researchers and practitioners should focus on:

- Adopting test cases that better reflect real-world scenarios: strengthening collaboration between academia and industry is essential for developing generalized test cases using real order data from manufacturing enterprises. This approach ensures that test scenarios are closer to actual production environments. By introducing diverse real-world order data and simulating realistic production processes and dynamic changes, such as order modifications and equipment failures, it is possible to provide more reliable and comprehensive performance evaluations. This ensures that algorithms not only demonstrate theoretical advantages but also excel in practical applications.
- Effective integration of RL and metaheuristic algorithms: to meet the demands of resilient manufacturing in Industry 5.0, it is recommended to effectively integrate RL with metaheuristic algorithms. Leveraging offline training and online application capabilities of RL, combined with the powerful search abilities of metaheuristic algorithms, can enable the development of more intelligent, flexible, and adaptable scheduling algorithms. These advanced approaches would facilitate efficient and stable optimization decisions in dynamically changing production environments.
- In-depth exploration of multi-objective research: for modern manufacturing, focusing on only two objectives often fails to comprehensively address the complexities of real-world requirements. Therefore, it is recommended to further explore optimization research involving three or more objectives. Expanding to multi-objective optimization is not only a necessary academic advancement but also a critical step in addressing the challenges of Industry 5.0.

Through these focused efforts, the field can move closer to achieving practical, efficient, and comprehensive scheduling solutions that meet the demands of modern manufacturing environments.

Author contributions

WZ: Conceptualization, Formal Analysis, Funding acquisition, Methodology, Project administration, Supervision, Writing–review and editing. XB: Conceptualization, Data curation, Investigation, Methodology, Writing–original draft, Writing–review and editing. XH: Conceptualization, Funding acquisition, Methodology, Supervision, Writing–review and editing. MG: Conceptualization, Formal Analysis, Funding acquisition, Methodology, Supervision, Validation, Writing–review and editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. Science and Technology Research Project of Henan Province (232102211049), Open Fund of Key Laboratory of Grain Information Processing and Control, Ministry of Education (KFJJ2023005), Zhengzhou Science and Technology Collaborative Innovation Project (21ZZXTCX19), and Open Fund of Institute for Complexity Science, Henan University of Technology (CSKFJJ-2024-29), and Scientific Research (C) of Japan Society of Promotion of Science (JSPS) (19K12148).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Abedi, M., Chiong, R., Noman, N., and Zhang, R. (2020). A multi-population, multi-objective memetic algorithm for energy-efficient job-shop scheduling with deteriorating machines. *Expert Syst. Appl.* 157, 113348. doi:10.1016/j.eswa.2020.113348

Afsar, S., Palacios, J. J., Puente, J., Vela, C. R., and Gonzalez-Rodriguez, I. (2022). Multi-objective enhanced memetic algorithm for green job shop scheduling with uncertain times. *Swarm Evol. Comput.* 68, 101016. doi:10.1016/j.swevo.2021.101016

Anjana, V., Sridharan, R., and Ram Kumar, P. (2020). Metaheuristics for solving a multi-objective flow shop scheduling problem with sequence-dependent setup times. *J. Sched.* 23, 49–69. doi:10.1007/s10951-019-00610-0

Basiri, M.-A., Alinezhad, E., Tavakkoli-Moghaddam, R., and Shahsavari-Poure, N. (2020). A hybrid intelligent algorithm for a fuzzy multi-objective job shop scheduling problem with reentrant workflows and parallel machines. *J. Intelligent & Fuzzy Syst.* 39, 7769–7785. doi:10.3233/jifs-201120

Behnamian, J., Fatemi Ghomi, S., and Zandieh, M. (2009). A multi-phase covering pareto-optimal front method to multi-objective scheduling in a realistic hybrid flowshop using a hybrid metaheuristic. *Expert Syst. Appl.* 36, 11057–11069. doi:10.1016/j.eswa.2009.02.080

Boufellouh, R., and Belkaid, F. (2023). Multi-objective optimization for energy-efficient flow shop scheduling problem with blocking and collision-free transportation constraints. *Appl. Soft Comput.* 148, 110884. doi:10.1016/j.asoc.2023.110884

Branda, A., Castellano, D., Guizzi, G., and Popolo, V. (2021). Metaheuristics for the flow shop scheduling problem with maintenance activities integrated. *Comput. & Industrial Eng.* 151, 106989. doi:10.1016/j.cie.2020.106989

Burmeister, S. C., Guericke, D., and Schryen, G. (2023). A memetic nsga-ii for the multi-objective flexible job shop scheduling problem with real-time energy tariffs. *Flexible Serv. Manuf. J.* 36, 1530–1570. doi:10.1007/s10696-023-09517-7

Cai, S., Yang, K., and Liu, K. (2018). Multi-objective optimization of the distributed permutation flow shop scheduling problem with transportation and eligibility constraints. *J. Operations Res. Soc. China* 6, 391–416. doi:10.1007/s40305-017-0165-3

Caldeira, R. H., and Gnanavelbabu, A. (2021). A pareto based discrete jaya algorithm for multi-objective flexible job shop scheduling problem. *Expert Syst. Appl.* 170, 114567. doi:10.1016/j.eswa.2021.114567

Chen, J.-f., Wang, L., and Peng, Z.-p. (2019). A collaborative optimization algorithm for energy-efficient multi-objective distributed no-idle flow-shop scheduling. *Swarm Evol. Comput.* 50, 100557. doi:10.1016/j.swevo.2019.100557

Chen, T.-L., Cheng, C.-Y., and Chou, Y.-H. (2020). Multi-objective genetic algorithm for energy-efficient hybrid flow shop scheduling with lot streaming. *Ann. Operations Res.* 290, 813–836. doi:10.1007/s10479-018-2969-x

Chen, X.-l., Li, J.-q., and Xu, Y. (2023). Q-learning based multi-objective immune algorithm for fuzzy flexible job shop scheduling problem considering dynamic disruptions. *Swarm Evol. Comput.* 83, 101414. doi:10.1016/j.swevo.2023.101414

Cheng, J., Yang, B., Gen, M., Jang, Y. J., and Liang, C.-J. (2020). Machine learning based evolutionary algorithms and optimization for transportation and logistics. *Comput. & Industrial Eng.* 143, 106372. doi:10.1016/j.cie.2020.106372

Cheng, R., Gen, M., and Tsujimura, Y. (1996). A tutorial survey of job-shop scheduling problems using genetic algorithms—i. representation. *Comput. & Industrial Eng.* 30, 983–997. doi:10.1016/0360-8352(96)00047-2

Cheng, R., Gen, M., and Tsujimura, Y. (1999). A tutorial survey of job-shop scheduling problems using genetic algorithms, part ii: hybrid genetic search strategies. *Comput. & Industrial Eng.* 36, 343–364. doi:10.1016/s0360-8352(99)00136-9

Chryssolouris, G., and Subramaniam, V. (2001). Dynamic scheduling of manufacturing job shops using genetic algorithms. *J. Intelligent Manuf.* 12, 281–293. doi:10.1023/a:1011253011638

Clifton, J., and Laber, E. (2020). Q-learning: theory and applications. *Annu. Rev. Statistics Its Appl.* 7, 279–301. doi:10.1146/annurev-statistics-031219-041220

Coello, C. A. C. (2007). *Evolutionary algorithms for solving multi-objective problems*. Springer.

Davis, L. (2014). “Job shop scheduling with genetic algorithms,” in *Proceedings of the first international conference on genetic algorithms and their applications* (New York: Psychology Press), 136–140.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: nsga-ii. *IEEE Trans. Evol. Comput.* 6, 182–197. doi:10.1109/4235.996017

Destouet, C., Tlahig, H., Bettayeb, B., and Mazari, B. (2023). Flexible job shop scheduling problem under industry 5.0: a survey on human reintegration, environmental consideration and resilience improvement. *J. Manuf. Syst.* 67, 155–173. doi:10.1016/j.jmsy.2023.01.004

Ding, J.-Y., Song, S., and Wu, C. (2016). Carbon-efficient scheduling of flow shops by multi-objective optimization. *Eur. J. Operational Res.* 248, 758–771. doi:10.1016/j.ejor.2015.05.019

Dong, R., Li, J., Song, D., Yang, B., and Zhou, L. (2024). A multi-objective non-dominated sorting gravitational search algorithm for assembly flow-shop scheduling of marine prefabricated cabins. *Mathematics* 12, 2288–7390. doi:10.3390/math12142288

Engin, O., and Yılmaz, M. K. (2021). A fuzzy logic based methodology for multi-objective hybrid flow shop scheduling with multi-processor tasks problems and solving with an efficient genetic algorithm. *J. Intelligent & Fuzzy Syst.* 42, 451–463. doi:10.3233/jifs-219203

Fadaei, M., and Zandieh, M. (2013). Scheduling a bi-objective hybrid flow shop with sequence-dependent family setup times using metaheuristics. *Arabian J. Sci. Eng.* 38, 2233–2244. doi:10.1007/s13369-013-0611-4

Fasihi, M., Tavakkoli-Moghaddam, R., and Jolai, F. (2023). A bi-objective re-entrant permutation flow shop scheduling problem: minimizing the makespan and maximum tardiness. *Operational Res.* 23, 29. doi:10.1007/s12351-023-00770-0

Fathollahi-Fard, A. M., Woodward, L., and Akhrif, O. (2024a). A distributed permutation flow-shop considering sustainability criteria and real-time scheduling. *J. Industrial Inf. Integration* 39, 100598. doi:10.1016/j.jii.2024.100598

Fathollahi-Fard, A. M., Woodward, L., and Akhrif, O. (2024b). A scenario-based robust optimization model for the sustainable distributed permutation flow-shop scheduling problem. *Ann. Operations Res.*, 1–42. doi:10.1007/s10479-024-05940-7

Fattahi, P., and Fallahi, A. (2010). Dynamic scheduling in flexible job shop systems by considering simultaneously efficiency and stability. *CIRP J. Manuf. Sci. Technol.* 2, 114–123. doi:10.1016/j.cirpj.2009.10.001

Fekri, M., Heydari, M., and Mahdavi Mazdeh, M. (2024). Bi-objective optimization of flexible flow shop scheduling problem with multi-skilled human resources. *Eng. Appl. Artif. Intell.* 133, 108094. doi:10.1016/j.engappai.2024.108094

Ferreira, J. J., Lopes, J. M., Gomes, S., and Rammal, H. G. (2023). Industry 4.0 implementation: environmental and social sustainability in manufacturing multinational enterprises. *J. Clean. Prod.* 404, 136841. doi:10.1016/j.jclepro.2023.136841

Fonseca, C. M., and Fleming, P. J. (1993). Genetic algorithms for multiobjective optimization: formulation discussion and generalization. *Icga (Citeseer)* 93, 416–423.

Gao, J., Sun, L., and Gen, M. (2008). A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems. *Comput. & Operations Res.* 35, 2892–2907. doi:10.1016/j.cor.2007.01.001

Gao, Q., Hu, H., Liu, J., Zhuang, C., Ding, H., and Li, H. (2024). Distributed assembly shop scheduling problem for complex products considering multiskilled worker assignment and transportation time. *Int. J. Prod. Res.*, 1–23. doi:10.1080/00207543.2024.2383785

Gen, M., Cheng, R., and Lin, L. (2008). *Network models and optimization: multiobjective genetic algorithm approach.* Springer Science & Business Media.

Geng, K., Ye, C., and Liu, L. (2020). Research on multi-objective hybrid flow shop scheduling problem with dual resource constraints using improved memetic algorithm. *IEEE Access* 8, 104527–104542. doi:10.1109/access.2020.2999680

Goldberg, D. E., Deb, K., Kargupta, H., and Harik, G. (1993). “Rapid, accurate optimization of difficult problems using messy genetic algorithms,” in *Proceedings of the fifth international conference on genetic algorithms (urbana, USA)*, 59–64.

Gonçalves, J. F., de Magalhães Mendes, J. J., and Resende, M. G. (2005). A hybrid genetic algorithm for the job shop scheduling problem. *Eur. J. Operational Res.* 167, 77–95. doi:10.1016/j.ejor.2004.03.012

Gong, G., Chiong, R., Deng, Q., and Gong, X. (2020). A hybrid artificial bee colony algorithm for flexible job shop scheduling with worker flexibility. *Int. J. Prod. Res.* 58, 4406–4420. doi:10.1080/00207543.2019.1653504

González, M. A., Rasconi, R., and Oddi, A. (2022). Metaheuristics for multiobjective optimization in energy-efficient job shops. *Eng. Appl. Artif. Intell.* 115, 105263. doi:10.1016/j.engappai.2022.105263

González-Rodríguez, I., Puente, J., Palacios, J. J., and Vela, C. R. (2020). Multi-objective evolutionary algorithm for solving energy-aware fuzzy job shop problems. *Soft Comput.* 24, 16291–16302. doi:10.1007/s00500-020-04940-6

Han, W., Deng, Q., Gong, G., Zhang, L., and Luo, Q. (2021). Multi-objective evolutionary algorithms with heuristic decoding for hybrid flow shop scheduling problem with worker constraint. *Expert Syst. Appl.* 168, 114282. doi:10.1016/j.eswa.2020.114282

Hao, X., Gen, M., Lin, L., and Suer, G. A. (2017). Effective multiobjective eda for bi-criteria stochastic job-shop scheduling problem. *J. Intelligent Manuf.* 28, 833–845. doi:10.1007/s10845-014-1026-0

Hassanzadeh, A., Rasti-Barzoki, M., and Khosroshahi, H. (2016). Two new meta-heuristics for a bi-objective supply chain scheduling problem in flow-shop environment. *Appl. Soft Comput.* 49, 335–351. doi:10.1016/j.asoc.2016.08.019

He, L., Li, W., Chiong, R., Abedi, M., Cao, Y., and Zhang, Y. (2021). Optimising the job-shop scheduling problem using a multi-objective jaya algorithm. *Appl. Soft Comput.* 111, 107654. doi:10.1016/j.asoc.2021.107654

He, Z., Tang, B., and Luan, F. (2022). An improved african vulture optimization algorithm for dual-resource constrained multi-objective flexible job shop scheduling problems. *Sensors* 23, 90. doi:10.3390/s23010090

Hillermeier, C. (2001). Generalized homotopy approach to multiobjective optimization. *J. Optim. Theory Appl.* 110, 557–583. doi:10.1023/a:1017536311488

Hosseini, S. M. H. (2017). A multi-objective genetic algorithm (moga) for hybrid flow shop scheduling problem with assembly operation. *J. Industrial Syst. Eng.* 10, 132–154.

Huang, Y.-Y., Pan, Q.-K., Gao, L., Miao, Z.-H., and Peng, C. (2022). A two-phase evolutionary algorithm for multi-objective distributed assembly permutation flowshop scheduling problem. *Swarm Evol. Comput.* 74, 101128. doi:10.1016/j.swevo.2022.101128

Jain, A. K., Kumar, S., and Tayal, S. (2022). Life cycle engineering in the era of industry 4.0 [Dataset].

Jain, H., and Deb, K. (2014). An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: handling constraints and extending to an adaptive approach. *IEEE Trans. Evol. Comput.* 18, 602–622. doi:10.1109/tevc.2013.2281534

Jia, Y., Zhou, Y., and Fu, Y. (2024). A multi-objective brain storm optimization for integrated distributed flexible job shop and distribution problems. *Heliyon* 10, e36318. doi:10.1016/j.heliyon.2024.e36318

Kacem, I., Hammadi, S., and Borne, P. (2002). Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic. *Math. Comput. Simul.* 60, 245–276. doi:10.1016/s0378-4754(02)00019-8

Kamali, S. R., Banirostam, T., Motameni, H., and Teshnehlab, M. (2023). An immune-based multi-agent system for flexible job shop scheduling problem in dynamic and multi-objective environments. *Eng. Appl. Artif. Intell.* 123, 106317. doi:10.1016/j.engappai.2023.106317

Li, J.-Q., Duan, P., Cao, J., Lin, X.-P., and Han, Y.-Y. (2018a). A hybrid pareto-based tabu search for the distributed flexible job shop scheduling problem with e/t criteria. *IEEE Access* 6, 58883–58897. doi:10.1109/access.2018.2873401

Li, J.-Q., Pan, Q.-K., and Tasgetiren, M. F. (2014). A discrete artificial bee colony algorithm for the multi-objective flexible job-shop scheduling problem with maintenance activities. *Appl. Math. Model.* 38, 1111–1132. doi:10.1016/j.apm.2013.07.038

Li, J.-Q., Sang, H.-y., Han, Y.-y., Wang, C.-g., and Gao, K.-z. (2018b). Efficient multi-objective optimization algorithm for hybrid flow shop scheduling problems with setup energy consumptions. *J. Clean. Prod.* 181, 584–598. doi:10.1016/j.jclepro.2018.02.004

Li, P., Xue, Q., Zhang, Z., Chen, J., and Zhou, D. (2023). Multi-objective energy-efficient hybrid flow shop scheduling using q-learning and gvns driven nsga-ii. *Comput. & Operations Res.* 159, 106360. doi:10.1016/j.cor.2023.106360

Li, R., Gong, W., and Lu, C. (2022). A reinforcement learning based rmoea/d for bi-objective fuzzy flexible job shop scheduling. *Expert Syst. Appl.* 203, 117380. doi:10.1016/j.eswa.2022.117380

Li, X., Yang, X., Zhao, Y., Teng, Y., and Dong, Y. (2020). Metaheuristic for solving multi-objective job shop scheduling problem in a robotic cell. *IEEE Access* 8, 147015–147028. doi:10.1109/access.2020.3015796

Liu, J., Sun, B., Li, G., and Chen, Y. (2024). Multi-objective adaptive large neighbourhood search algorithm for dynamic flexible job shop schedule problem with transportation resource. *Eng. Appl. Artif. Intell.* 132, 107917. doi:10.1016/j.engappai.2024.107917

Liu, Z., Wang, J., Zhang, C., Chu, H., Ding, G., and Zhang, L. (2021). A hybrid genetic-particle swarm algorithm based on multilevel neighbourhood structure for flexible job shop scheduling problem. *Comput. & Operations Res.* 135, 105431. doi:10.1016/j.cor.2021.105431

Lu, C., Huang, Y., Meng, L., Gao, L., Zhang, B., and Zhou, J. (2022). A pareto-based collaborative multi-objective optimization algorithm for energy-efficient scheduling of distributed permutation flow-shop with limited buffers. *Robotics Computer-Integrated Manuf.* 74, 102277. doi:10.1016/j.rcim.2021.102277

Luan, F., Zhao, H., Liu, S. Q., He, Y., and Tang, B. (2023). Enhanced nsga-ii for multi-objective energy-saving flexible job shop scheduling. *Sustain. Comput. Inf. Syst.* 39, 100901. doi:10.1016/j.suscom.2023.100901

Luo, Q., Deng, Q., Gong, G., Guo, X., and Liu, X. (2022). A distributed flexible job shop scheduling problem considering worker arrangement using an improved memetic algorithm. *Expert Syst. Appl.* 207, 117984. doi:10.1016/j.eswa.2022.117984

Luo, Q., Deng, Q., Gong, G., Zhang, L., Han, W., and Li, K. (2020). An efficient memetic algorithm for distributed flexible job shop scheduling problem with transfers. *Expert Syst. Appl.* 160, 113721. doi:10.1016/j.eswa.2020.113721

Mishra, A. K., Shrivastava, D., Bundela, B., and Sircar, S. (2020). "An efficient jaya algorithm for multi-objective permutation flow shop scheduling problem," in *Advanced engineering optimization through intelligent techniques: select proceedings of AEOTIT 2018* (Springer), 113–125.

Momenikorbekandi, A., and Abbod, M. F. (2023). A novel metaheuristic hybrid parthenogenetic algorithm for job shop scheduling problems: applying an optimization model. *IEEE Access* 11, 56027–56045. doi:10.1109/access.2023.3278372

Motair, H. M. (2021). Exact and hybrid metaheuristic algorithms to solve bi-objective permutation flow shop scheduling problem. *J. Phys. Conf. Ser.* 1818, 012042. IOP Publishing. doi:10.1088/1742-6596/1818/1/012042

Mourtzis, D. (2020a). "Adaptive scheduling in the era of cloud manufacturing," *Scheduling in industry 4.0 and cloud manufacturing*, 61–85.

Mourtzis, D. (2020b). Simulation in the design and operation of manufacturing systems: state of the art and new trends. *Int. J. Prod. Res.* 58, 1927–1949. doi:10.1080/00207543.2019.1636321

Mourtzis, D. (2022). Advances in adaptive scheduling in industry 4.0. *Front. Manuf. Technol.* 2, 937889. doi:10.3389/fmtec.2022.937889

Öztop, H., Tasgetiren, M. F., Eliiyi, D. T., Pan, Q.-K., and Kandiller, L. (2020). An energy-efficient permutation flowshop scheduling problem. *Expert Syst. Appl.* 150, 113279. doi:10.1016/j.eswa.2020.113279

Pezzella, F., Morganti, G., and Ciaschetti, G. (2008). A genetic algorithm for the flexible job-shop scheduling problem. *Comput. & Operations Res.* 35, 3202–3212. doi:10.1016/j.cor.2007.02.014

Qi, Y., Ma, X., Liu, F., Jiao, L., Sun, J., and Wu, J. (2014). Moea/d with adaptive weight adjustment. *Evol. Comput.* 22, 231–264. doi:10.1162/evco_a_00109

Rahimi-Vahed, A., and Mirghorbani, S. (2006). A multi-objective particle swarm for a flow shop scheduling problem. *J. Comb. Optim.* 13, 79–102. doi:10.1007/s10878-006-9015-7

Rajkumar, R., and Jeen Robert, R. (2019). A hybrid algorithm for multi-objective optimization of minimizing makespan and total flow time in permutation flow shop scheduling problems. *Inf. Technol. Control* 48, 47–57. doi:10.5755/j01.itc.48.1.20909

Rangsaritratsamee, R., Ferrell, W. G., Jr, and Kurz, M. B. (2004). Dynamic rescheduling that simultaneously considers efficiency and stability. *Comput. & Industrial Eng.* 46, 1–15. doi:10.1016/j.cie.2003.09.007

Rifai, A. P., Mara, S. T. W., and Sudiarso, A. (2021). Multi-objective distributed reentrant permutation flow shop scheduling with sequence-dependent setup time. *Expert Syst. Appl.* 183, 115339. doi:10.1016/j.eswa.2021.115339

Rodríguez-Espinosa, C. A., González-Neira, E. M., and Zambrano-Rey, G. M. (2024). A simheuristic approach using the nsga-ii to solve a bi-objective stochastic flexible job shop problem. *J. Simul.* 18, 646–670. doi:10.1080/17477778.2023.2231877

Rossit, D. G., Nesmachnow, S., and Rossit, D. A. (2022). A multiobjective evolutionary algorithm based on decomposition for a flow shop scheduling problem in the context of industry 4.0. *Int. J. Math. Eng. Manag. Sci.* 7, 433–454. doi:10.33889/ijmems.2022.7.4.029

Schaffer, J. D. (1985). Some experiments in machine learning using vector evaluated genetic algorithms. Nashville, TN (USA): Vanderbilt Univ. Tech. rep.

Schulz, S., Neufeld, J. S., and Buscher, U. (2019). A multi-objective iterated local search algorithm for comprehensive energy-aware hybrid flow shop scheduling. *J. Clean. Prod.* 224, 421–434. doi:10.1016/j.jclepro.2019.03.155

Schulz, S., Schönheit, M., and Neufeld, J. S. (2022). Multi-objective carbon-efficient scheduling in distributed permutation flow shops under consideration of transportation efforts. *J. Clean. Prod.* 365, 132551. doi:10.1016/j.jclepro.2022.132551

Sekkal, D. N., and Belkaid, F. (2023). A multi-objective optimization algorithm for flow shop group scheduling problem with sequence dependent setup time and worker learning. *Expert Syst. Appl.* 233, 120878. doi:10.1016/j.eswa.2023.120878

Shahsavari-Pour, N., and Ghasemishabankareh, B. (2013). A novel hybrid meta-heuristic algorithm for solving multi objective flexible job shop scheduling. *J. Manuf. Syst.* 32, 771–780. doi:10.1016/j.jmsy.2013.04.015

Shao, W., Shao, Z., and Pi, D. (2022). A multi-neighborhood-based multi-objective memetic algorithm for the energy-efficient distributed flexible flow shop scheduling problem. *Neural Comput. Appl.* 34, 22303–22330. doi:10.1007/s00521-022-07714-3

Shao, Z., Shao, W., Chen, J., and Pi, D. (2024). Mql-mm: a meta-q-learning-based multi-objective metaheuristic for energy-efficient distributed fuzzy hybrid blocking flow-shop scheduling problem. *IEEE Trans. Evol. Comput.*, 1. doi:10.1109/tevc.2024.3399314

Srinivas, N., and Deb, K. (1994). Muiltiobjective optimization using nondominated sorting in genetic algorithms. *Evol. Comput.* 2, 221–248. doi:10.1162/evco.1994.2.3.221

Tajbakhsh, Z., Fattahi, P., and Behnamian, J. (2014). Multi-objective assembly permutation flow shop scheduling problem: a mathematical model and a meta-heuristic algorithm. *J. Operational Res. Soc.* 65, 1580–1592. doi:10.1057/jors.2013.105

Tan, W., Yuan, X., Wang, J., and Zhang, X. (2021). A fatigue-conscious dual resource constrained flexible job shop scheduling problem by enhanced nsga-ii: an application from casting workshop. *Comput. & Industrial Eng.* 160, 107557. doi:10.1016/j.cie.2021.107557

Tang, H., Xiao, Y., Zhang, W., Lei, D., Wang, J., and Xu, T. (2024). A dql-nsga-iii algorithm for solving the flexible job shop dynamic scheduling problem. *Expert Syst. Appl.* 237, 121723. doi:10.1016/j.eswa.2023.121723

Tao, X.-R., Pan, Q.-K., Sang, H.-Y., Gao, L., Yang, A.-L., and Rong, M. (2023). Nondominated sorting genetic algorithm-ii with q-learning for the distributed permutation flowshop rescheduling problem. *Knowledge-Based Syst.* 278, 110880. doi:10.1016/j.knosys.2023.110880

Tavakkoli-Moghaddam, R., Rahimi-Vahed, A., and Mirzaei, A. H. (2007). A hybrid multi-objective immune algorithm for a flow shop scheduling problem with bi-objectives: weighted mean completion time and weighted mean tardiness. *Inf. Sci.* 177, 5072–5090. doi:10.1016/j.ins.2007.06.001

Vallejos-Cifuentes, P., Ramirez-Gomez, C., Escudero-Atehortua, A., and Rodriguez Velasquez, E. (2019). Energy-aware production scheduling in flow shop and job shop environments using a multi-objective genetic algorithm. *Eng. Manag. J.* 31, 82–97. doi:10.1080/10429247.2018.1544798

Vital-Soto, A., Baki, M. F., and Azab, A. (2023). A multi-objective mathematical model and evolutionary algorithm for the dual-resource flexible job-shop scheduling problem with sequencing flexibility. *Flexible Serv. Manuf. J.* 35, 626–668. doi:10.1007/s10696-022-09446-x

Wang, G., Li, X., Gao, L., and Li, P. (2022). An effective multi-objective whale swarm algorithm for energy-efficient scheduling of distributed welding flow shop. *Ann. Operations Res.* 310, 223–255. doi:10.1007/s10479-021-03952-1

Wang, W., Zhang, B., Jiang, X., Jia, B., Sang, H., and Meng, L. (2024). Decomposition-based multi-objective approach for a green hybrid flowshop rescheduling problem with consistent sublots. *Int. J. Prod. Res.* 62, 7904–7932. doi:10.1080/00207543.2024.2333943

Wang, Y.-J., Wang, G.-G., Tian, F.-M., Gong, D.-W., and Pedrycz, W. (2023). Solving energy-efficient fuzzy hybrid flow-shop scheduling problem at a variable machine speed using an extended nsga-ii. *Eng. Appl. Artif. Intell.* 121, 105977. doi:10.1016/j.engappai.2023.105977

Wang, Z., Zhang, J., and Yang, S. (2019). An improved particle swarm optimization algorithm for dynamic job shop scheduling problems with random job arrivals. *Swarm Evol. Comput.* 51, 100594. doi:10.1016/j.swevo.2019.100594

Wei, H., Li, S., Quan, H., Liu, D., Rao, S., Li, C., et al. (2021). Unified multi-objective genetic algorithm for energy efficient job shop scheduling. *IEEE Access* 9, 54542–54557. doi:10.1109/access.2021.3070981

Wu, R., Li, Y., Guo, S., and Xu, W. (2018). Solving the dual-resource constrained flexible job shop scheduling problem with learning effect by a hybrid genetic algorithm. *Adv. Mech. Eng.* 10, 1687814018804096. doi:10.1177/1687814018804096

Wu, X., and Che, A. (2020). Energy-efficient no-wait permutation flow shop scheduling by adaptive multi-objective variable neighborhood search. *Omega* 94, 102117. doi:10.1016/j.omega.2019.102117

Xie, F., Li, L., Li, L., Huang, Y., and He, Z. (2023). A decomposition-based multi-objective jaya algorithm for lot-streaming job shop scheduling with variable sublots and intermingling setting. *Expert Syst. Appl.* 228, 120402. doi:10.1016/j.eswa.2023.120402

Xin, B., Li, Y., Yu, J., and Zhang, J. (2015). An adaptive bpso algorithm for multi-skilled workers assignment problem in aircraft assembly lines. *Assem. Autom.* 35, 317–328. doi:10.1108/aa-06-2015-051

Xin, X., Jiang, Q., Li, S., Gong, S., and Chen, K. (2021). Energy-efficient scheduling for a permutation flow shop with variable transportation time using an improved discrete whale swarm optimization. *J. Clean. Prod.* 293, 126121. doi:10.1016/j.jclepro.2021.126121

Xu, W., Hu, Y., Luo, W., Wang, L., and Wu, R. (2021). A multi-objective scheduling method for distributed and flexible job shop based on hybrid genetic algorithm and tabu search considering operation outsourcing and carbon emission. *Comput. & Industrial Eng.* 157, 107318. doi:10.1016/j.cie.2021.107318

Yu, F., Lu, C., Yin, L., and Zhou, J. (2024). Modeling and optimization algorithm for energy-efficient distributed assembly hybrid flowshop scheduling problem considering worker resources. *J. Industrial Inf. Integration* 40, 100620. doi:10.1016/j.jii.2024.100620

Yu, H., Gao, K., Li, Z., and Suganthan, P. N. (2024). Energy-efficient multi-objective distributed assembly permutation flowshop scheduling by q-learning based meta-heuristics. *Appl. Soft Comput.* 166, 112247. doi:10.1016/j.asoc.2024.112247

Yüksel, D., Taşgetiren, M. F., Kandiller, L., and Gao, L. (2020). An energy-efficient bi-objective no-wait permutation flowshop scheduling problem to minimize total tardiness and total energy consumption. *Comput. & Industrial Eng.* 145, 106431. doi:10.1016/j.cie.2020.106431

Zhang, B., Che, A., and Wang, Y. (2024a). Grid-based artificial bee colony algorithm for multi-objective job shop scheduling with manual loading and unloading tasks. *Expert Syst. Appl.* 245, 123011. doi:10.1016/j.eswa.2023.123011

Zhang, J., and Cai, J. (2023). A dual-population genetic algorithm with q-learning for multi-objective distributed hybrid flow shop scheduling problem. *Symmetry* 15, 836. doi:10.3390/sym15040836

Zhang, L., Gao, L., and Li, X. (2013). A hybrid genetic algorithm and tabu search for a multi-objective dynamic job shop scheduling problem. *Int. J. Prod. Res.* 51, 3516–3531. doi:10.1080/00207543.2012.751509

Zhang, Q., and Li, H. (2007). Moea/d: a multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.* 11, 712–731. doi:10.1109/tevc.2007.892759

Zhang, S., Li, X., Zhang, B., and Wang, S. (2020). Multi-objective optimisation in flexible assembly job shop scheduling using a distributed ant colony system. *Eur. J. Operational Res.* 283, 441–460. doi:10.1016/j.ejor.2019.11.016

Zhang, W., Gen, M., and Jo, J. (2014). Hybrid sampling strategy-based multiobjective evolutionary algorithm for process planning and scheduling problem. *J. Intelligent Manuf.* 25, 881–897. doi:10.1007/s10845-013-0814-2

Zhang, W., Geng, H., Li, C., Gen, M., Zhang, G., and Deng, M. (2023a). Q-learning-based multi-objective particle swarm optimization with local search within factories for energy-efficient distributed flow-shop scheduling problem. *J. Intelligent Manuf.* 36, 185–208. doi:10.1007/s10845-023-02227-9

Zhang, W., Li, C., Gen, M., Yang, W., and Zhang, G. (2024b). A multiobjective memetic algorithm with particle swarm optimization and q-learning-based local search for energy-efficient distributed heterogeneous hybrid flow-shop scheduling problem. *Expert Syst. Appl.* 237, 121570. doi:10.1016/j.eswa.2023.121570

Zhang, W., Xiao, G., Gen, M., Geng, H., Wang, X., Deng, M., et al. (2024c). Enhancing multi-objective evolutionary algorithms with machine learning for scheduling problems: recent advances and survey. *Front. Industrial Eng.* 2, 1337174. doi:10.3389/fieng.2024.1337174

Zhang, W., Zheng, Y., and Ahmad, R. (2023b). An energy-efficient multi-objective scheduling for flexible job-shop-type remanufacturing system. *J. Manuf. Syst.* 66, 211–232. doi:10.1016/j.jmsy.2022.12.008

Zhang, Z., Tang, Q., and Chica, M. (2021). Maintenance costs and makespan minimization for assembly permutation flow shop scheduling by considering preventive and corrective maintenance. *J. Manuf. Syst.* 59, 549–564. doi:10.1016/j.jmsy.2021.03.020

Zhao, F., Zhang, H., Wang, L., Xu, T., Zhu, N., and Jonrinaldi, J. (2024). A multi-objective discrete differential evolution algorithm for energy-efficient distributed blocking flow shop scheduling problem. *Int. J. Prod. Res.* 62, 4226–4244. doi:10.1080/00207543.2023.2254858

Glossary

ABC	Artificial Bee Colony Algorithm
ACO	Ant Colony Optimization
AGV	Automatic Guided Vehicles
AI	Artificial Intelligence
ALNS	Adaptive Large Neighborhood Search
BFO	Bacterial Foraging Optimization Algorithm
DE	Differential Evolution
DFSP	Distributed Flow-shop Problem
DHFSP	Distributed Hybrid Flow-shop Problem
Dist.FJSP	Distributed Flexible Job-shop Scheduling Problem
Dist.JSP	Distributed Job-shop Scheduling Problem
DPFSP	Distributed Permutation Flow-shop Problem
Dyn.FJSP	Dynamic Flexible Job-shop Scheduling Problem
Dyn.JSP	Dynamic Job-shop Scheduling Problem
EA	Evolutionary Algorithm
EAs	Evolutionary Algorithms
FJSP	Flexible Job-shop Scheduling Problem
FSP	Flow-shop Scheduling Problem
GA	Genetic Algorithm
GSA	Gravitational Search Algorithm
HFSP	Hybrid Flow-shop Scheduling Problem
IA	Immune Algorithm
IG	Integrated Gradients
IoT	Internet of Things
JSP	Job-shop Scheduling Problem
MA	Memetic Algorithm
MOEA	Multi-objective Evolutionary Algorithms
MOGA	Multi-Objective Genetic Algorithm
MOJA/D	Multi-Objective Jaya Algorithm Based on Decomposition
MOP	Multi-objective Optimization Problem
MOPSO	Multi-Objective Particle Swarm Optimization
NEH	Nawaz-Enscore-Ham
NPGA	Niched Pareto Genetic Algorithm
NSGA	Non-dominated Sorting Genetic Algorithm
NSGA-II	Non-dominated Sorting Genetic Algorithm II
NSGA-III	Non-dominated Sorting Genetic Algorithm III
PAES	Pareto Archived Evolution Strategy
PFSP	Permutation Flow-shop Scheduling Problem
PGA	Parthenogenetic Algorithm
PhA	Physics-based Algorithms
POF	Pareto-optimal Front
POS	Pareto-optimal Set
PSO	Particle Swarm Optimization
RL	Reinforcement Learning
SA	Simulated Annealing
SI	Swarm Intelligence
SPEA	Strength Pareto Evolutionary Algorithm
TLA	Teaching-Learning-Based Optimization
TS	Tabu Search
VEGA	Vector Evaluated Genetic Algorithm
VNS	Variable Neighborhood Search
WOA	Whale Optimization Algorithm