Advances in Industrial Engineering, December 2024, 58(2): 325-340
DOI: 10.22059/aie.2024.378291.1898
RESEARCH PAPER

Flexible Job Scheduling under Consideration of Time and Energy Consumption Using Enhanced Iterative Deferred Acceptance Algorithm

Jalal Delaram[1]*

[1]*Assistant Professor, School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran.*

Received: 19 June 2024, Revised: 30 June 2024, Accepted: 7 July 2024

Abstract

This paper highlights the shift in the industrial sector towards a decentralized structure, focusing the importance of energy efficiency for manufacturers and the need for quick job completion to satisfy customers. The study proposes a matching game approach using the Job Scheduling Problem (JSP) to address both manufacturer and customer concerns. It introduces the Deferred Acceptance (DA) algorithm to create stable and optimal matches between machines and operations, incorporating the W-value concept to represent willingness values between partners. The Enhanced Iterative DA (EIDA) algorithm, enhanced with the W-value, shows improved job completion time, reduced energy consumption, and faster runtime compared to the Genetic Algorithm. Through experiments, our enhanced iterative DA (EIDA) algorithm results in an average 6.40% increase in job completion time and a 16.60% reduction in manufacturers' energy consumption compared to the Genetic Algorithm. Moreover, utilizing the W-value leads to a 19.03% average runtime improvement.

Keywords:
Job Scheduling Problem (JSP),
Matching Game,
Deferred Acceptance (DA) Algorithm,
Stable Resource Allocation.

Introduction

Based on the study by the International Energy Agency (Gong et al., 2018), the energy demand of the world has doubled since 2000, and it has been estimated that it will double again by 2040. Besides, the manufacturing has a 38% share of CO2 emissions and a 33% share of total consumption. Due to increases in energy prices and environmental constraints, industrial firms have to be more aware of controlling their production energy consumption and carbon emissions (Iqbal et al., 2020). Therefore, considering energy-aware indicators is urgent to reduce energy consumption (Armendariz-Lopez et al., 2018).

These days industrial sector is faced with two forces. One force, as mentioned, from the legislative bodies to meet the regulatory rules for energy consumption (Sinha & Chaturvedi, 2018), and the other from the customers to fulfill their orders as soon as possible (Bie et al., 2016). The two forces of meeting energy consumption rules and preparation of customers' orders, usually act in opposite directions (Li et al., 2016). Ignoring one of the forces may entail a loss for the industrial firm, from losing customers to suffering environmental fines (Cai et al., 2016; Raileanu et al., 2017). So, enabling a balance between these two conflicting forces is key

* Corresponding author: (Jalal Delaram)
Email: delaram@ut.ac.ir

to the success of any industrial firm.

Job Scheduling Problem (JSP) has been applied for consideration of time and energy consumption where resource allocation and production scheduling matters (Chaudhry & Khan, 2016; Harjunkoski et al., 2014; Türkyılmaz et al., 2020). Scheduling of operations is one of the most critical problems in the management of industrial systems and processes. JSP refers to the problem of scheduling a set of jobs on a set of machines considering some assumptions and limitations. The difficulty of this decision is how to sequence the operations on the machines in a way that a given performance indicator is optimized (Bruni et al., 2020).

Despite much research and developments in the field of the JSP, to the best of our knowledge, the consideration of time and energy consumption in order to make a balance between the two sides of the problem (customers and manufacturers) is neglected. A vast majority of the previous research has focused on the optimization of time or time-related indicators (makespan, tardiness, and etc.) such as Vital-Soto et al. (2020) that developed a mathematical model for makespan minimization in flexible JSP using a hybridized bacterial foraging optimization algorithm; Baykasoğlu et al. (2020) used a greedy randomized constructive method for job-shop scheduling and optimization of delivery time; Lunardi et al. (2021) used metaheuristics algorithms for the online printing shop scheduling problem in order to minimize the makespan considering unavailability, setup times, and overlapping constraints; and Ahmadian et al. (2021) presented a general review on scheduling problem with time minimization objective over the last four decades.

Also, with the global energy concerns, more attention has been paid to production scheduling considering energy consumption than ever before. Masmoudi et al. (2019) considered energetic aspects of the JSP to minimize production costs in terms of energy respecting a power peak limitation and traditional production constraints. Chou et al. (2020) exploited a robust scheduling algorithm under power consumption constraints for large production systems. Şahman (2021) enabled a discrete spotted hyena optimizer for solving distributed job-shop scheduling problems under consideration of energy consumption minimization. Wu et al. (2019) proposed a multi-objective optimization model for saving energy without delaying makespan using a hybrid pigeon-inspired optimization and simulated annealing algorithm.

There is a common research gap among the previous research; having a one-sided view of the JSP, and optimizing the schedule from the perspective of manufacturers (considering energy consumption models) or customers (considering time-related models). Actually, the JSP delineates a strategic situation between customers (jobs) and manufacturers (machines) in which the customers want to receive their jobs as soon as possible, and manufacturers want to minimize their energy consumption. In this paper, we contribute to the previous literature by modeling a matching game for the allocation of jobs to machines in a way that the utility of each manufacturer and customer is considered. We also exploited the novel properties of the Deferred Acceptance (DA) algorithm in the process of resource allocation (Roth, 2008). The DA is a promising matching algorithm in game theory studies that enables stable and optimal matching in favor of the proposer (Dworczak, 2021; Honda, 2020). We developed an Iterative Deferred Acceptance (IDA) algorithm that enables a stable and optimal matching between the customers and manufacturers at each stage of the resource allocation; meaning that no one can be better off in matching pairs to achieve a more desired state. This process of stable and optimal matching will occur at each stage of the resource allocation in the IDA algorithm. Compared with the existing research, the contributions of this paper can be concluded as follows:

(i) Modeling the JSP as a matching game in which the tasks of the jobs are allocated to the machines of the manufacturers in a stable and optimal manner.
(ii) Development of a heuristic algorithm that outperforms the previous metaheuristic algorithms according to runtime and objective value.
(iii) Elaborating an algorithm that presents a resource allocation for the JSP in which the

considerations of both sides are satisfied; in this work time for jobs and energy consumption for machines.

(iv) Enhancement of the DA algorithm to reach the solution with fewer iterations using the novel concept of the W-value (Willingness value).

The remainder of the paper is organized as follows: the second section presents a review of the current literature in the field of JSP, focusing on the prevailing research in terms of solution methods and objective functions. The third section formally defines the problem and outlines the proposed algorithm based on the DA. In the fourth section, the experiments and their corresponding results are presented. Finally, the fifth section offers concluding remarks and discusses potential future research directions.

Related literature

Over the last decades, the JSP has been one of the most considered problems in academic research and industrial studies (Chaudhry & Khan, 2016). Different methods and algorithms have been developed to solve this class of problems (Türkyılmaz et al., 2020). In the following, an overview of the JSP is presented.

Based on a general view, the JSPs can be classified into two groups classic and flexible problems. In a classic JSP, there is a finite set of n jobs to be processed on a finite set of m machines. Each job comprises a set of operations that must be performed on a different machine and in specified processing times, in a given job-dependent order. Most of the papers in this category considered time (makespan) as the objective function and utilized meta-heuristics for finding an optimal or near-optimal solution, like Bürgy (2017) that developed a Neighborhood Search and Tabu Search (NS-TS) algorithm for makespan minimization, and Tao and Xu-ping (2018) that used Quantum Bacterial Foraging Optimization (QBFO) algorithm for minimizing the tardiness. Wu et al. (2020) addressed the optimization of risks both in performance and stability for the JSP under random machine breakdowns, in which three objectives of makespan, makespan risk, and stability risk are considered simultaneously. They developed a Multi-Objective Predictive Scheduling (MOPS) algorithm to generate a Pareto solution. The MOPS algorithm provides an acceptable Pareto solution set in terms of diversity and convergence.

A flexible JSP is an extension of the classical JSP problem which allows an operation to be processed by any machine from a given set of alternative machines. A classic JSP requires sequencing of operations on fixed machines, while in flexible JSP the assignment of an operation is not fixed in advance and can be processed on a set of capable machines. In terms of complexity, a flexible JSP stands at a higher level; so hybrid meta-heuristics are more considered in this category. Wang et al. (2012) proposed an effective Bi-population-based Estimation of Distribution (BED) algorithm for the flexible JSPs with the criterion to minimize the maximum completion time. Wang et al. (2013) proposed a hybrid Artificial Bee Colony with Neighborhood Search (ABC-NS) algorithm for flexible JSPs. Sun et al. (2021) formulated the flexible JSP as a new mathematical and network graph model. They proposed a Hybrid Many-objective Evolutionary (HME) algorithm with the TS and the neighborhood structure to improve the local search ability.

In terms of solution approach, there are two major approaches centralized and decentralized view. In the centralized view, the solution approaches focused on finding an optimal solution using heuristic or meta-heuristic algorithms. But in the decentralized approaches, a game theory view is exploited to find a solution in which both the manufacturers' and customers' objectives are optimized in a way that one can better off using a unilateral deviation. Zhou et al. (2009) published one of the first works with a decentralized view of the JSP and using a Genetic Algorithm (GA) developed a mechanism for the minimization of makespan. Of notable other works with a decentralized view is the framework developed by Zhang et al. (2017) for makespan minimization in a real-time JSP in Cloud Manufacturing environments. They also examined different Mixed Integer Linear Programming (MILP) models and their consequences on the solution. Wang et al. (2021) proposed an overall architecture for distributed, flexible and real-time JSP to enhance the decision-making capability of scheduling systems using a decentralized view. Their model included two layers of task assignment on the shop floor and the real-time scheduling of the flexible manufacturing units. They adopted an Evolutionary Game-based Method (EGM) to obtain the optimal allocation.

Most of the related research in the JSP scope is developed using a centralized view. Dai et al. (2013) proposed an energy-efficient model for a flexible JSP. This model was solved using a hybrid Genetic and Simulated Annealing (G-SA) algorithm. Wang et al. (2015) suggested an approach for process planning considering scheduling optimization for sustainable machining including the objectives of energy efficiency and makespan minimization. Mansouri et al. (2016) analyzed the trade-off between energy consumption and makespan and built a heuristic algorithm for a fast trade-off between energy consumption and makespan analysis. Xin et al. (2021) addressed a JSP problem with sequence-dependent setup time considering a novel conveyor speed control energy-saving strategy. The aim was to find an optimal processing sequence for manufacturers to achieve green production. They developed the MILP model to minimize both the completion time and total energy consumption. They designed an improved Discrete Whale Swarm Optimization (DWSO) to solve their model.

To achieve an overview of the current status of JSP research and studies, we classified some of the recent and related papers according to the type of JSP, the view of the solution approach, objective function ingredients, the target side of the optimization problem, and the solution algorithm in Table 1. Based on the type of JSP, Table 1 shows that the flexible JSP has been more popular as it is nearer to real-life applications. Centralized view and application of optimization approaches are more common among the recent research based on Table 1. Existing research about the JSP mainly focuses on meta-heuristics, especially GA. Meta-heuristics are approximate methods and cannot guarantee the optimal solutions even for small-sized problems.

Although the optimization approaches suggest valuable results and solutions in strategic situations with two opposite sides of decision-makers, they are inefficient. Also, Table 1 shows that the applications of time and energy consumption have been considered vastly in recent years. In fact, the time-related function is a performance indicator for customers' satisfaction that want their orders (jobs) as soon as possible. The energy-related function is a performance indicator for manufacturers' satisfaction that want to minimize their energy consumption and related costs. Considering these two objectives simultaneously requires a strategic vision in which the conflicting objectives of both sides have to be met. Therefore, we consider the flexible JSP as a strategic decision-making problem between two sides customers and manufacturers. The customers want their jobs to be fulfilled as soon as possible, and the manufacturers want to minimize their energy consumption. In the subsequent sections, we will formulate this conflicting matching game and develop an algorithm that meets the requirements of this strategic situation.

Problem Statement

There is a set of $|J|$ jobs to be processed on $|M|$ machines. Let J be the set of jobs, $J = \{j_1, j_2, \dots, j_{|J|}\}$, and M the set of machines, $M = \{m_1, m_2, \dots, m_{|M|}\}$. Each job j_i consists of a sequence of n_i operations, $(O_{i,1}, O_{i,2}, \dots, O_{i,n_i})$. The execution of each operation O_{ij} requires one machine out of a set of given machines M_k. The running time of the operation O_{ij} on the machine M_k is t_{ijk}, and the energy consumption of the operation O_{ij} on the machine M_k is e_{ijk}. The JSP determines both the allocation of machines and the sequence of operations on all the machines to realize a certain scheduling objective function. The following assumptions are also considered: all machines and jobs are available at time $t = 0$, each operation can be processed by only one machine at a time, the jobs are independent, no pre-emption of operations is allowed, transportation time of jobs between the machines and time to set up the machine for processing a particular operation are included in the processing time. The standard notation of the considered problem is $FJM\ |t_{ijk}, e_{ijk}, M_k| \sum C_i + E_i$ (Pinedo, 2012).

Table 1. The overview of the recent and related JSP research.

Author(s)	**Class**	**View**	**Objective function**	**Target side**	**Type**	**Algorithm**
Zhou et al. (2009)	Classic	Decentralized	Time	Jobs	Meta.	GA
Wang et al. (2012)	Flexible	Centralized	Time	Jobs	Heuristic	BED
Wang et al. (2013)	Flexible	Centralized	Time	Jobs	Meta.	ABC-NS
Li and Gao (2016)	Flexible	Centralized	Time	Jobs	Meta.	GA-TS
Zhang et al. (2017)	Flexible	Decentralized	Time	Machines	Heuristic	MILP
Bürgy (2017)	Classic	Centralized	Time	Jobs	Meta.	NS-TS
Wu and Sun (2018)	Flexible	Centralized	Energy consumption	Machines	Meta.	NSGA-II
Xie and Chen (2018)	Flexible	Centralized	Time	Jobs	Meta.	GA
Gong et al. (2018)	Flexible	Centralized	Time, Energy consumption	Both	Meta.	GA
Tao and Xu-ping (2018)	Classic	Centralized	Time	Jobs	Meta.	QBFO
Meng et al. (2019)	Flexible	Centralized	Energy consumption	Machines	Heuristic	MILP
Gong et al. (2019)	Flexible	Centralized	Time, Energy consumption	Jobs	Meta.	NSGA-III
Dai et al. (2019)	Flexible	Centralized	Time, Energy consumption	Both	Meta.	GA
Wu et al. (2020)	Classic	Centralized	Time	Machines	Meta.	MOPS
Xin et al. (2021)	Flexible	Centralized	Energy consumption	Machines	Meta.	DWSO
Sun et al. (2021)	Flexible	Centralized	Time	Jobs	Meta.	HME
Wang et al. (2021)	Flexible	Decentralized	Energy consumption	Jobs	Heuristic	EGM
Meng et al. (2023)	Flexible	Centralized	Time	Jobs	Meta.	VNS
Xie et al. (2023)	Flexible	Centralized	Time	Jobs	Meta.	GA-TS
Liu et al. (2024)	Flexible	Centralized	Time	Machines	Meta.	VNS
Berterottière et al. (2024)	Flexible	Centralized	Time	Jobs	Meta.	NS-TS
This work	**Flexible**	**Decentralized**	**Time, Energy consumption**	**Both**	**Heuristic**	**EIDA**

IDA algorithm

We developed our algorithm based on the idea of the DA algorithm in matching games (Bando, 2014b). The DA algorithm was first developed by Gale and Shapley (1962) in the pursuit of finding a stable solution for the allocation of residents to hospitals in a way that no one has an objection about his/her matching partner (Roth, 2003). The DA has remarkable results where it is used, from school choice programs (Abdulkadiroğlu & Sönmez, 2003), airplane landing problems (de Arruda et al., 2015), cloud service sharing (Liu et al., 2017), to public platform design (Delaram et al., 2021).

Our proposed algorithm is an iterative exploitation of the DA algorithm for machine-operation matching in the process of solving the JSP. At the first step, there are initial operations of the jobs, $O_{i,1}$ for all $i \in J$, which are available, and all machines $M_k \in M$ are free based on the assumption. Before the allocation, a tie-breaking rule is required in order to. For example, if the operation O_{ij} is indifferent between M_k and M_l, order them alphabetically. Then, each operation applies to its most preferred machine. Each machine tentatively accepts its most preferred and available operation and rejects the other. Any operation rejected at step $k-1$ applies to its next highest alternative (if any). Each machine regards both the new operation and the operation accepted at step $k-1$ and tentatively accepts the most preferred and available operation from the joint pool and rejects all others. If an operation remains unmatched in a step it will be passed to the next step until being matched with an available machine. The flowchart of the algorithm is depicted in Fig. 1.

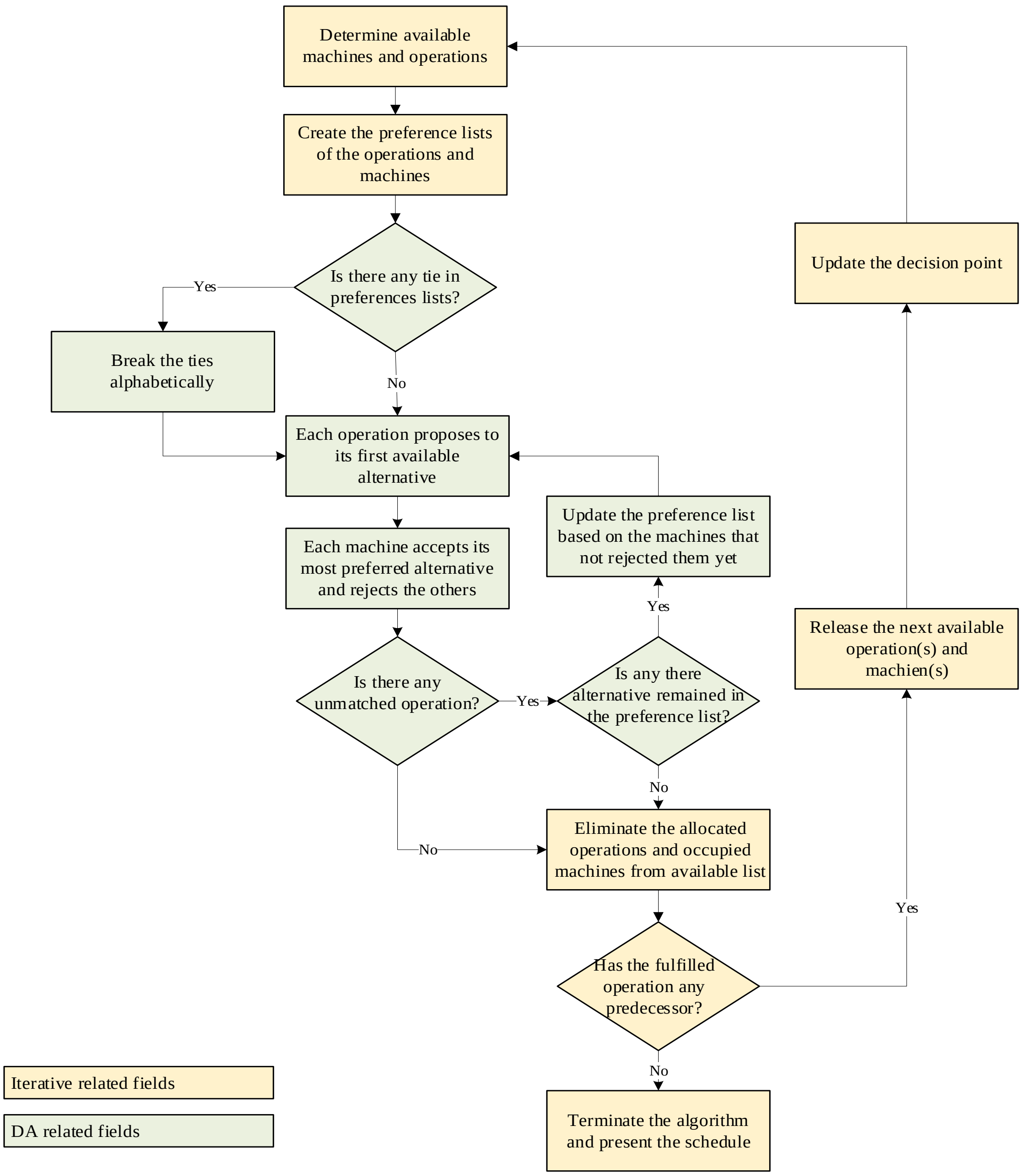


Fig. 1. The flowchart of the IDA algorithm.

Based on the basic assumption that there is at least one machine to do each operation; accordingly, all of the operations will be fulfilled and the algorithm terminates in a finite number of stages and produces a matching. Also, according to the properties of the DA algorithm (Bando, 2014a), the allocation of the resources in each step of the algorithm is stable (Roth, 2008). Also, the solution will be strategy-proof, and optimal for the operations as the operations are considered as the applicants (Abdulkadiroğlu et al., 2009). In terms of computational complexity, the DA algorithm has $O(|M||J|)$ time complexity, in which $|M|$ is the maximum number of machines and $|J|$ is the maximum number of jobs (Maschler et al., 2013), as the algorithm repeats the DA until the allocation of all operations, so at the worst case it will be repeated up to the maximum number of operations. Suppose $|N|$ is the maximum number of operations, so the IDA has the time complexity of $O(|M||J||N|)$. It is noteworthy to say that the complexity of the JSP as a decision problem is categorized in the NP-hard class.

Example

In this section, we used the studies of Wang et al. (2012) and Gong et al. (2018) to develop an example for further clarification. Consider 5 machines and 4 jobs. Table 2 presents the details of the operations processing time (in hours) and energy consumption (in MW) on different machines.

Table 2. Factsheet of the jobs and machines, the parentheses indicate (processing time, energy consumption).

Job	Operation	Machines				
		M1	M2	M3	M4	M5
J1	O11	(5,15)	(3,16)	(4,14)	(∞,∞)	(2,20)
	O12	(∞,∞)	(4,14)	(5,12)	(8,11)	(6,10)
	O13	(2,11)	(∞,∞)	(3,11)	(4,13)	(5,12)
	O14	(5,12)	(4,14)	(∞,∞)	(6,10)	(3,15)
J2	O21	(4,12)	(3,13)	(5,15)	(6,10)	(∞,∞)
	O22	(7,19)	(∞,∞)	(8,18)	(6,20)	(5,17)
	O23	(3,12)	(4,11)	(5,10)	(∞,∞)	(2,14)
	O24	(∞,∞)	(5,13)	(6,12)	(4,16)	(7,11)
J3	O31	(3,10)	(∞,∞)	(5,12)	(4,13)	(2,15)
	O32	(∞,∞)	(5,17)	(6,15)	(∞,∞)	(4,18)
	O33	(4,11)	(5,10)	(3,15)	(7,12)	(6,10)
J4	O41	(6,17)	(5,18)	(∞,∞)	(7,12)	(4,19)
	O42	(9,12)	(∞,∞)	(8,14)	(7,19)	(4,15)
	O43	(3,11)	(6,12)	(3,15)	(5,13)	(∞,∞)
	O44	(5,18)	(∞,∞)	(6,19)	(∞,∞)	(4,17)

At the first iteration, the operations send their request to the machines. The operation O11 requests from M5, O21 requests from M2, O31 requests from M5, and O41 requests from M5, refer to Step 1 in Table 3. Then, the machines have to decide; the machine M2 received one request from O21, and M5 received three requests from O11, O21, and O41. In this step, M2 decides to match with O21, and M5 decides to match with O31 because of its preference list, based on energy consumption in Table 2. Now, the operations O11 and O41 send their requests to their second preferred choices; both O11 and O41 request from M2. So, M2 has three requests from O11, O21, and O41 in Step 2. M2 selects O21 according to its preference list and higher priority of O21 rather than the other. In Step 3, O11 and O41 send request to their three priorities; O11 requests from M3 and O41 requests from M1. As they are free and available the matching of this iteration becomes completed. Table 3 shows the allocation steps of the first iteration of the IDA algorithm.

As no operation remained, M4 remains unused until the next decision point. The next decision point occurs when operation O31 ends, refer to point d1 in Fig. 2. At d1, operation O32 becomes available and the machines M4 and M5 are free. But M4 is unable to do O32. So, O32 will be allocated to M5. d2 is the next decision point in which operation O21 ends, and

O22 becomes available. M2 and M4 are free machines at this time. As M2 is unable to do O22, it will be allocated to M4. At the third decision point (d3), O11 ends and O12 becomes available, and M2 and M3 are free machines. Using the preference list of O12, M2 is the preferred choice, so O12 matches with M2. At point d4, operations O33 and O42 become available. Also, the machines M1, M3, and M5 are free. According to the preference lists of the operations and machines, O33 matches with M3, and O42 matches with M5. At the fifth decision point (d5), operations O23 and O13 become available, and the machines M1, M2, M3, and M4 are free for allocation. In this iteration, O13 is allocated to M1 and O23 is allocated to M2. At the sixth decision point (d6), O43 becomes available and machines M3 and M4 are free and able to perform O43. Because of the higher priority of the M3, O43 is allocated to M3 in this iteration. At the point d7, O14 becomes available and machines M1, M4 and M5 are free. M5 is the first choice for O14. So, O14 is allocated to M5. At the point d8, operations O24 and O44 become available, and machines M1, M2, M3, and M4 are free. This is the last iteration in which the remaining operations of the job 2 and 4 have to be allocated. The algorithm allocates O24 to M4 and O44 to M1. So, the schedule finishes at t=18.

Table 3. The first iteration of the algorithm.

Machines	Step 1	Step 2	Step 3	Result
M1			O41	(M1, O41)
M2	O21	O21, ~~O11~~, ~~O41~~	O21	(M2, O21)
M3			O11	(M3, O11)
M4				(M4, -)
M5	~~O11~~, O31, ~~O41~~	O31	O31	(M5, O31)

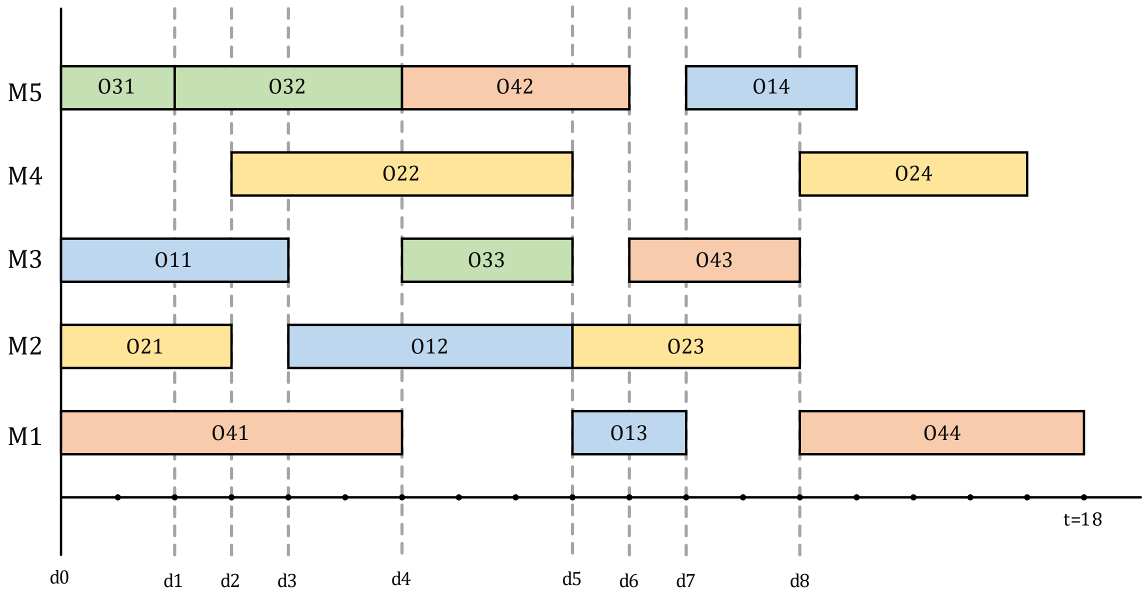


Fig. 2. The schedule for 5 machines and 4 jobs considering the operations as applicants using the IDA algorithm.

Experiments and results

We developed our experiments using the popular data set of Brandimarte (1993) for analyzing our proposed algorithm. The two first columns of Table 5 present the details of the problem instances in the experiments.

The Genetic Algorithm (GA) as a well-known metaheuristic algorithm has been successfully applied to a wide range of problems, especially JSP (Fan et al., 2021; Guo et al., 2021; Liu et al., 2021). We adopted the GA procedure of Pezzella et al. (2008) for comparing the results of the algorithms. The initial parameters of the GA algorithm are presented in Table 4.

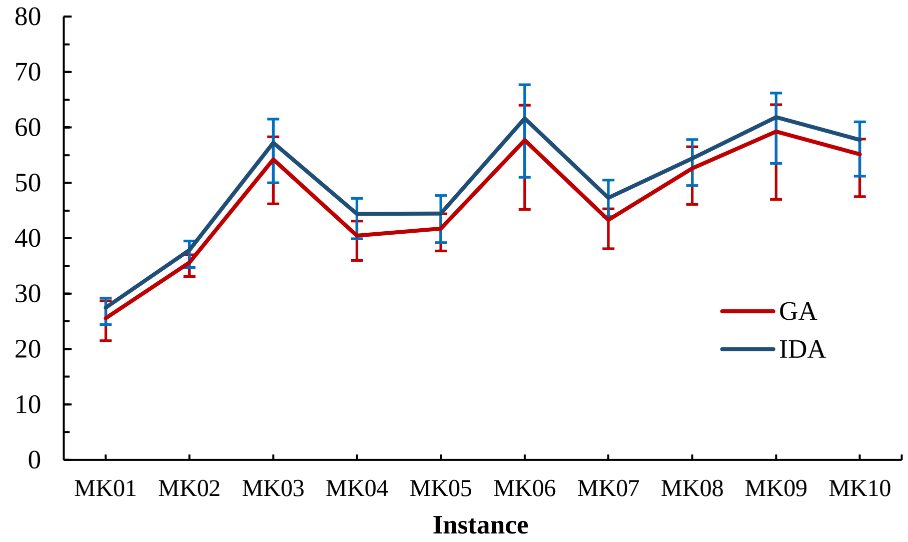
Table 4. The initial parameters for the GA algorithm.

Parameter name	Parameter value
Population	10,000
Crossover type	Uniform
Mutation probability	1%
Mutation type	Exchange value
Selection type	Roulette wheel

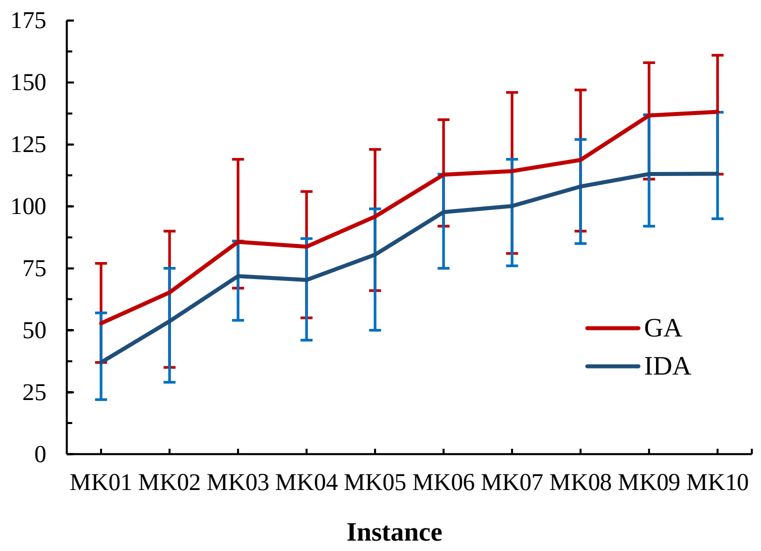
We compare the performance of our algorithm rather than the GA results. Table 5 presents the results of the IDA algorithm in comparison with the GA. According to Table 5, the GA presents slightly better results for demanders as the average of the makespan is lower than the IDA. The GA solves an optimization problem to minimize the makespan of the jobs. So, the results of the GA are not stable and (necessarily) optimal. However, the IDA enables a stable and optimal resource allocation through the phases of the allocation of operations to machines. As Table 5 shows, the IDA has better results in energy consumption. The IDA also reaches the solution in a lower time rather than the GA.

Table 5. The comparison of the GA and IDA algorithms.

Ins.	Size (J×O×M)	GA					IDA				
		Makespan		Energy Consumption		Time (s)	Makespan		Energy Consumption		Time (s)
		Mean	Std	Mean	Std		Mean	Std	Mean	Std	
MK01	5×5×5	25.560	2.318	52.800	14.176	4.746	27.460	1.722	37.010	11.900	3.662
MK02	8×5×5	35.613	1.375	65.250	16.060	21.04	37.888	1.368	53.625	14.256	6.534
MK03	8×8×5	54.213	3.871	85.625	14.456	79.708	57.250	3.338	71.875	9.400	9.578
MK04	8×8×8	40.488	2.276	83.750	14.377	91.688	44.388	2.375	70.250	12.627	14.184
MK05	10×8×8	41.730	2.394	95.900	14.501	114.699	44.470	2.557	80.500	15.970	25.725
MK06	10×10×8	57.690	6.921	112.800	12.781	150.195	61.610	6.393	97.700	12.993	36.841
MK07	10×10×10	43.310	2.212	114.200	18.999	275.383	47.270	1.786	100.200	12.998	51.154
MK08	12×10×10	52.600	3.411	118.750	16.912	305.982	54.375	2.938	108.083	12.155	69.227
MK09	12×12×10	59.250	5.461	136.750	16.130	370.829	61.842	4.532	113.083	14.818	85.463
MK10	12×12×12	55.175	3.549	138.167	11.524	474.854	57.750	3.048	113.167	15.253	108.068



(a)



(b)

Fig. 3. The comparison of (a) makespan, and (b) energy consumption between the GA and IDA algorithms.

Although the DA algorithm has a polynomial time complexity to reach the solution, the structure of the algorithm can be improved. The DA can be augmented by considering the level of the willingness between the players on both sides of the problem. The DA does not determine a procedure for the proposition of the players. In other words, there are matching pairs that are mutually the first priority of each other, and the existence of these pairs in the process of matching makes the algorithm longer. To deal with this situation, we define the concept of the Willingness value (W-value) which determines the level of the desirability for a matching between two partners. The W-value has a two-sided nature meaning that it is dependent on the willingness of the provider and the consumer. So, let the $\psi_{ji}(\mu)$ be the willingness function of the machine m_j in matching with the operation o_i under the matching algorithm μ. In formal expression, $\psi_{ji}(\mu)$ is defined as follows:

$$\psi_{ji}(\mu) = \frac{1}{r_{ji}} \tag{1}$$

where r_{ji} is the ranking of the operation o_i in the preference list of the machine m_j. The equation (1) shows that the maximum level of willingness occurs when a player matches with its most preferred alternative (1^{st} alternative in its preference list). Similarly, let $\phi_{ij}(\mu)$ be the willingness function of the operation o_i in matching with the machine m_j under the matching algorithm μ. The formulation of the $\phi_{ij}(\mu)$ is as follows:

$$\phi_{ij}(\mu) = \frac{1}{r_{ij}} \tag{2}$$

where r_{ij} is the ranking of the machine m_j in the preference list of the operation o_i. Using the equations (1) and (2), we introduce the W-value function as follows:

$$W_{ij}(\mu) = \phi_{ij}(\mu).\psi_{ji}(\mu) = \frac{1}{r_{ij}.r_{ji}} \tag{3}$$

The W-value function states the strongness and the level of the desirability for a matching pair between the operation o_i and the machine m_j under the matching algorithm μ. The maximum value of the $W_{ij}(\mu)$ occurs when both sides of the matching pair are the first alternative of each other.

Fig. 4 shows how the concept of the W-value. Our mechanism suggests the calculation of the W-value before utilization of the DA algorithm. If there are machine and operation pairs that are mutually the most preferred alternative for each other, it is not required to enter them into the DA algorithm. The improved mechanism removes the machine-operation pairs with the W-value equal to one and eliminates the allocated operations and the machine's capacity. The mechanism updates the preference lists and the W-values. The mechanism repeats this process to reach no pairs with one W-value. The mechanism decides about the remaining providers and consumers using the DA algorithm. Table 6 presents the results of the application W-value into the IDA mechanism, known as Enhanced IDA (EIDA). According to Table 6, EIDA outperforms the IDA algorithm time performance by 19.03% on average. Fig. 5 also presents the comparison of the runtime analysis for GA, IDA, and EIDA algorithms, which shows the superiority of EIDA algorithm rather than the others.

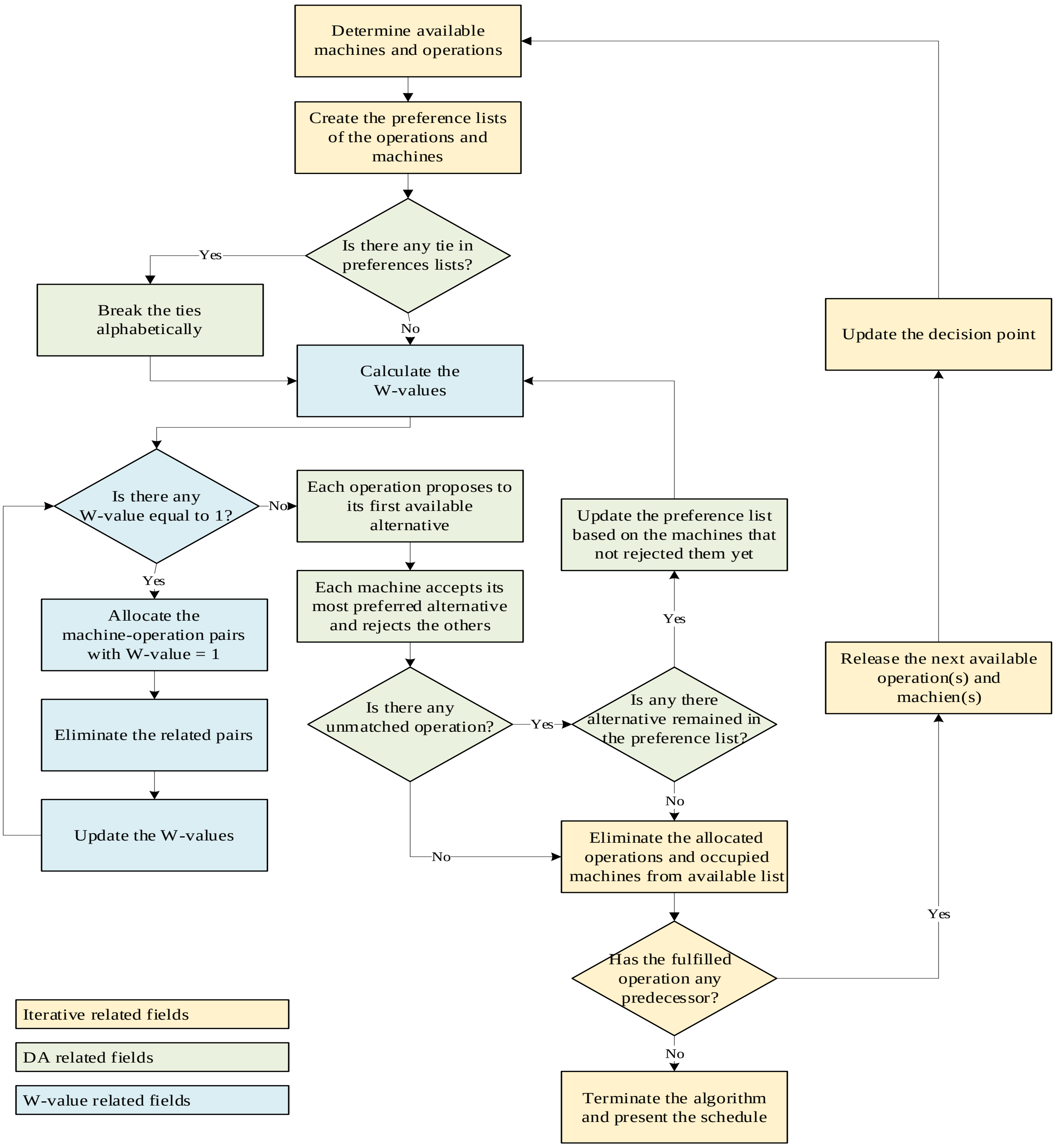


Fig. 4. The flowchart of the EIDA algorithm.

Table 6. The impact of the W-value on the IDA algorithm.

Instance	IDA Time	EIDA Time	Improvement %
MK01	3.662	2.928	20.03%
MK02	6.534	5.135	21.42%
MK03	9.578	7.749	19.09%
MK04	14.184	10.633	25.03%
MK05	25.725	20.923	18.67%
MK06	36.841	29.658	19.50%
MK07	51.154	41.517	18.84%
MK08	69.227	57.736	16.60%
MK09	85.463	72.261	15.45%
MK10	108.068	91.118	15.68%
Average			**19.03%**

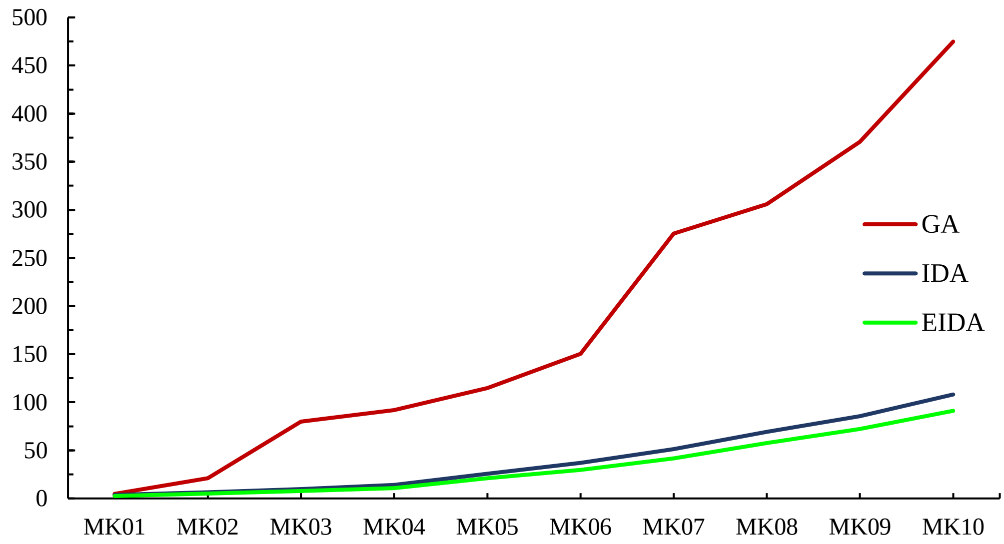


Fig. 5. The runtime analysis of the GA, IDA, and EIDA algorithms.

Conclusion

The industrial sector has evolved from a traditional and centralized structure into a modern and decentralized structure in recent decades. In the modern industrial world, not only the expectations of the customers should be considered but also the manufacturers' concerns have to be regarded. Although the customers expect to receive their orders as soon as possible, on the other side the manufacturers are concerned about energy consumption and environmental issues.

Dealing with the JSP while considering the objectives of one side of the problem may create difficulties for the other side. Keeping a balance between the two sides of the customers' expectations and manufacturers' requirements is an important managerial decision. In this paper, we utilized the DA algorithm for finding the stable and optimal resource allocation through an interval. This interval is the makespan of the jobs and includes the resource allocation phases of the JSP. In each resource allocation phase, the DA is exploited to enable a stable and optimal resource allocation.

The DA algorithm has outstanding characteristics in matching problems such as enabling a stable solution in which no one can complain about his partner (Nash Equilibrium of a matching game). To the best of our knowledge, this is the first application of the DA to manipulate an algorithm for solving the JSP. We manipulated the DA algorithm from a static form into a dynamic form. The original DA algorithm determines a stable and optimal matching between two sides of a problem for a specific time window. However, we manipulated the DA algorithm into an iterative form for assigning the jobs' operations to the manufacturers' machines.

Based on the experiments, we achieved the following results:

- Exploiting the EIDA algorithm the completion time of the jobs increases by 6.40% on average rather than the GA solution.
- The energy consumption of the manufacturers decreases by 16.60% on average in the EIDA algorithm in comparison with the GA.
- The performance of the DA using the W-value concept is enhanced. The W-value helps to reduce the required time to reach the solution. According to our experiments, exploiting the W-value reduces the runtime of the algorithm by 19.03% on average.

The contribution of the paper may be helpful for managers according the following insights that extracted from the results:

- Optimizing Energy Efficiency and Customer Satisfaction: By emphasizing the importance of energy efficiency for manufacturers and the need for quick job completion to meet customer expectations, this study highlights the critical balance between operational efficiency and customer satisfaction in the industrial sector. Implementing the Enhanced Iterative DA (EIDA) algorithm with the W-value concept allows for a comprehensive consideration of time and energy consumption, ultimately leading to improved performance metrics.
- Strategic Resource Allocation: The application of the Deferred Acceptance (DA) algorithm presents a strategic approach to resource allocation in job scheduling, ensuring stable and optimal matches between machines and operations. By incorporating willingness values through the W-value concept, the study provides a practical framework for enhancing decision-making processes in flexible job scheduling scenarios.
- Performance Enhancements over Traditional Methods: A key insight from the experiments conducted in this study is the significant performance improvements achieved by the Enhanced Iterative DA (EIDA) algorithm compared to the Genetic Algorithm. With an average 6.40% increase in job completion time and a 16.60% reduction in manufacturers' energy consumption, the EIDA algorithm proves to be a viable and efficient solution for addressing the challenges of job scheduling in the industrial sector.

A lot has been learned and much more remains to be learned. The authors encourage the following topics as part of future work: comparing the other metaheuristic and heuristic algorithms, considering priority rules in the allocation process, developing other kinds of policies in accordance with the situation, and exploiting multi-objective utility functions for the customers and manufacturers rather than time and energy consumption, respectively.

References

Abdulkadiroğlu, A., Pathak, P. A., & Roth, A. E. (2009). Strategy-Proofness versus Efficiency in Matching with Indifferences: Redesigning the NYC High School Match. *American Economic Review*, *99*(5), 1954-1978. https://doi.org/https://www.doi.org/10.1257/aer.99.5.1954

Abdulkadiroğlu, A., & Sönmez, T. (2003). School Choice: A Mechanism Design Approach. *American Economic Review*, *93*(3), 729-747. https://doi.org/http://doi.org/10.1257/000282803322157061

Ahmadian, M. M., Khatami, M., Salehipour, A., & Cheng, T. C. E. (2021). Four decades of research on the open-shop scheduling problem to minimize the makespan. *European Journal of Operational Research*. https://doi.org/https://doi.org/10.1016/j.ejor.2021.03.026

Armendariz-Lopez, J. F., Arena-Granados, A. P., Gonzalez-Trevizo, M. E., Luna-Leon, A., & Bojorquez-Morales, G. (2018). Energy payback time and Greenhouse Gas emissions: Studying the international energy agency guidelines architecture. *Journal of Cleaner Production*, *196*, 1566-1575. https://doi.org/https://doi.org/10.1016/j.jclepro.2018.06.134

Bando, K. (2014a). A modified deferred acceptance algorithm for many-to-one matching markets with externalities among firms. *Journal of Mathematical Economics*, *52*, 173-181. https://doi.org/https://doi.org/10.1016/j.jmateco.2014.01.001

Bando, K. (2014b). On the existence of a strictly strong Nash equilibrium under the student-optimal deferred acceptance algorithm. *Games and Economic Behavior*, *87*, 269-287. https://doi.org/https://doi.org/10.1016/j.geb.2014.05.009

Baykasoğlu, A., Madenoğlu, F. S., & Hamzadayı, A. (2020). Greedy randomized adaptive search for dynamic flexible job-shop scheduling. *Journal of Manufacturing Systems*, *56*, 425-451. https://doi.org/https://doi.org/10.1016/j.jmsy.2020.06.005

Berterottière, L., Dauzère-Pérès, S., & Yugma, C. (2024). Flexible job-shop scheduling with transportation resources. *European Journal of Operational Research*, *312*(3), 890-909. https://doi.org/https://doi.org/10.1016/j.ejor.2023.07.036

Bie, Z., Xie, H., Hu, G., & Li, G. (2016). Optimal scheduling of power systems considering demand response. *Journal of Modern Power Systems and Clean Energy*, *4*(2), 180-187. https://doi.org/https://doi.org/10.1007/s40565-015-0136-9

Brandimarte, P. (1993). Routing and scheduling in a flexible job shop by tabu search. *Annals of Operations Research*, *41*(3), 157-183. https://doi.org/10.1007/BF02023073

Bruni, M. E., Khodaparasti, S., & Demeulemeester, E. (2020). The distributionally robust machine scheduling problem with job selection and sequence-dependent setup times. *Computers & Operations Research*, *123*, 105017. https://doi.org/https://doi.org/10.1016/j.cor.2020.105017

Bürgy, R. (2017). A neighborhood for complex job shop scheduling problems with regular objectives. *Journal of Scheduling*, *20*(4), 391-422. https://doi.org/https://doi.org/10.1007/s10951-017-0532-2

Cai, W., Liu, F., Zhou, X., & Xie, J. (2016). Fine energy consumption allowance of workpieces in the mechanical manufacturing industry. *Energy*, *114*, 623-633. https://doi.org/https://doi.org/10.1016/j.energy.2016.08.028

Chaudhry, I. A., & Khan, A. A. (2016). A research survey: review of flexible job shop scheduling techniques. *International Transactions in Operational Research*, *23*(3), 551-591. https://doi.org/https://doi.org/10.1111/itor.12199

Chou, Y.-L., Yang, J.-M., & Wu, C.-H. (2020). An energy-aware scheduling algorithm under maximum power consumption constraints. *Journal of Manufacturing Systems*, *57*, 182-197. https://doi.org/https://doi.org/10.1016/j.jmsy.2020.09.004

Dai, M., Tang, D., Giret, A., & Salido, M. A. (2019). Multi-objective optimization for energy-efficient flexible job shop scheduling problem with transportation constraints. *Robotics and Computer-Integrated Manufacturing*, *59*, 143-157. https://doi.org/https://doi.org/10.1016/j.rcim.2019.04.006

Dai, M., Tang, D., Giret, A., Salido, M. A., & Li, W. D. (2013). Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm. *Robotics and Computer-Integrated Manufacturing*, *29*(5), 418-429. https://doi.org/https://doi.org/10.1016/j.rcim.2013.04.001

de Arruda, A. C., Weigang, L., & Milea, V. (2015). A new Airport Collaborative Decision Making algorithm based on Deferred Acceptance in a two-sided market. *Expert Systems with Applications*, *42*(7), 3539-3550. https://doi.org/https://doi.org/10.1016/j.eswa.2014.11.060

Delaram, J., Fatahi Valilai, O., Houshamand, M., & Ashtiani, F. (2021). A matching mechanism for public cloud manufacturing platforms using intuitionistic Fuzzy VIKOR and deferred acceptance algorithm. *International Journal of Management Science and Engineering Management*, 1-16. https://doi.org/https://doi.org/10.1080/17509653.2021.1892549

Dworczak, P. (2021). Deferred Acceptance with Compensation Chains. *Operations Research*, *69*(2), 456-468. https://doi.org/https://doi.org/10.1287/opre.2020.2042

Fan, J., Shen, W., Gao, L., Zhang, C., & Zhang, Z. (2021). A hybrid Jaya algorithm for solving flexible job shop scheduling problem considering multiple critical paths. *Journal of Manufacturing Systems*, *60*, 298-311. https://doi.org/https://doi.org/10.1016/j.jmsy.2021.05.018

Gale, D., & Shapley, L. S. (1962). College Admissions and the Stability of Marriage. *The American Mathematical Monthly*, *69*(1), 9-15. https://doi.org/10.1080/00029890.1962.11989827

Gong, G., Deng, Q., Gong, X., Liu, W., & Ren, Q. (2018). A new double flexible job-shop scheduling problem integrating processing time, green production, and human factor indicators. *Journal of Cleaner Production*, *174*, 560-576. https://doi.org/https://doi.org/10.1016/j.jclepro.2017.10.188

Gong, X., De Pessemier, T., Martens, L., & Joseph, W. (2019). Energy- and labor-aware flexible job shop scheduling under dynamic electricity pricing: A many-objective optimization investigation. *Journal of Cleaner Production*, *209*, 1078-1094. https://doi.org/https://doi.org/10.1016/j.jclepro.2018.10.289

Guo, W., Lei, Q., Song, Y., & Lyu, X. (2021). A learning interactive genetic algorithm based on edge selection encoding for assembly job shop scheduling problem. *Computers & Industrial Engineering*, *159*, 107455. https://doi.org/https://doi.org/10.1016/j.cie.2021.107455

Harjunkoski, I., Maravelias, C. T., Bongers, P., Castro, P. M., Engell, S., Grossmann, I. E., Hooker, J., Méndez, C., Sand, G., & Wassick, J. (2014). Scope for industrial applications of production scheduling models and solution methods. *Computers & Chemical Engineering*, *62*, 161-193. https://doi.org/https://doi.org/10.1016/j.compchemeng.2013.12.001

Honda, E. (2020). A modified deferred acceptance algorithm for conditionally lexicographic-substitutable preferences. *Journal of Mathematical Economics*, 102458. https://doi.org/https://doi.org/10.1016/j.jmateco.2020.102458
Iqbal, M. W., Kang, Y., & Jeon, H. W. (2020). Zero waste strategy for green supply chain management with minimization of energy consumption. *Journal of Cleaner Production*, *245*, 118827. https://doi.org/https://doi.org/10.1016/j.jclepro.2019.118827
Li, X., & Gao, L. (2016). An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem. *International Journal of Production Economics*, *174*, 93-110. https://doi.org/https://doi.org/10.1016/j.ijpe.2016.01.016
Li, Z., Tang, Q., & Zhang, L. (2016). Minimizing energy consumption and cycle time in two-sided robotic assembly line systems using restarted simulated annealing algorithm. *Journal of Cleaner Production*, *135*, 508-522. https://doi.org/https://doi.org/10.1016/j.jclepro.2016.06.131
Liu, M., Lv, J., Du, S., Deng, Y., Shen, X., & Zhou, Y. (2024). Multi-resource constrained flexible job shop scheduling problem with fixture-pallet combinatorial optimisation. *Computers & Industrial Engineering*, *188*, 109903. https://doi.org/https://doi.org/10.1016/j.cie.2024.109903
Liu, Y., Zhang, L., Tao, F., & Wang, L. (2017). Resource service sharing in cloud manufacturing based on the Gale–Shapley algorithm: advantages and challenge. *International Journal of Computer Integrated Manufacturing*, *30*(4-5), 420-432. https://doi.org/10.1080/0951192X.2015.1067916
Liu, Z., Wang, J., Zhang, C., Chu, H., Ding, G., & Zhang, L. (2021). A hybrid genetic-particle swarm algorithm based on multilevel neighbourhood structure for flexible job shop scheduling problem. *Computers & Operations Research*, 105431. https://doi.org/https://doi.org/10.1016/j.cor.2021.105431
Lunardi, W. T., Birgin, E. G., Ronconi, D. P., & Voos, H. (2021). Metaheuristics for the online printing shop scheduling problem. *European Journal of Operational Research*, *293*(2), 419-441. https://doi.org/https://doi.org/10.1016/j.ejor.2020.12.021
Mansouri, S. A., Aktas, E., & Besikci, U. (2016). Green scheduling of a two-machine flowshop: Trade-off between makespan and energy consumption. *European Journal of Operational Research*, *248*(3), 772-788. https://doi.org/https://doi.org/10.1016/j.ejor.2015.08.064
Maschler, M., Solan, E., & Zamir, S. (2013). *Game Theory*. Cambridge University Press. https://doi.org/https://doi.org/10.1017/CBO9780511794216
Masmoudi, O., Delorme, X., & Gianessi, P. (2019). Job-shop scheduling problem with energy consideration. *International Journal of Production Economics*, *216*, 12-22. https://doi.org/https://doi.org/10.1016/j.ijpe.2019.03.021
Meng, L., Zhang, C., Shao, X., & Ren, Y. (2019). MILP models for energy-aware flexible job shop scheduling problem. *Journal of Cleaner Production*, *210*, 710-723. https://doi.org/https://doi.org/10.1016/j.jclepro.2018.11.021
Meng, L., Zhang, C., Zhang, B., Gao, K., Ren, Y., & Sang, H. (2023). MILP modeling and optimization of multi-objective flexible job shop scheduling problem with controllable processing times. *Swarm and Evolutionary Computation*, *82*, 101374. https://doi.org/https://doi.org/10.1016/j.swevo.2023.101374
Pezzella, F., Morganti, G., & Ciaschetti, G. (2008). A genetic algorithm for the Flexible Job-shop Scheduling Problem. *Computers & Operations Research*, *35*(10), 3202-3212. https://doi.org/https://doi.org/10.1016/j.cor.2007.02.014
Pinedo, M. L. (2012). *Scheduling* (Vol. 29). Springer.
Raileanu, S., Anton, F., Iatan, A., Borangiu, T., Anton, S., & Morariu, O. (2017). Resource scheduling based on energy consumption for sustainable manufacturing. *Journal of Intelligent Manufacturing*, *28*(7), 1519-1530. https://doi.org/10.1007/s10845-015-1142-5
Roth, A. E. (2003). The Origins, History, and Design of the Resident Match. *JAMA*, *289*(7), 909-912. https://doi.org/https://www.doi.org/10.1001/jama.289.7.909
Roth, A. E. (2008). Deferred acceptance algorithms: history, theory, practice, and open questions. *International Journal of Game Theory*, *36*(3), 537-569. https://doi.org/http://doi.org/10.1007/s00182-008-0117-6
Şahman, M. A. (2021). A discrete spotted hyena optimizer for solving distributed job shop scheduling problems. *Applied Soft Computing*, *106*, 107349. https://doi.org/https://doi.org/10.1016/j.asoc.2021.107349
Sinha, R. K., & Chaturvedi, N. D. (2018). A graphical dual objective approach for minimizing energy consumption and carbon emission in production planning. *Journal of Cleaner Production*, *171*, 312-321. https://doi.org/https://doi.org/10.1016/j.jclepro.2017.09.272
Sun, J., Zhang, G., Lu, J., & Zhang, W. (2021). A hybrid many-objective evolutionary algorithm for flexible job-shop scheduling problem with transportation and setup times. *Computers & Operations Research*, 105263. https://doi.org/https://doi.org/10.1016/j.cor.2021.105263
Tao, N., & Xu-ping, W. (2018). Study on disruption management strategy of job-shop scheduling problem based on prospect theory. *Journal of Cleaner Production*, *194*, 174-178. https://doi.org/https://doi.org/10.1016/j.jclepro.2018.05.139

Türkyılmaz, A., Şenvar, Ö., Ünal, İ., & Bulkan, S. (2020). A research survey: heuristic approaches for solving multi objective flexible job shop problems. *Journal of Intelligent Manufacturing*, *31*(8), 1949-1983. https://doi.org/https://doi.org/10.1007/s10845-020-01547-4

Vital-Soto, A., Azab, A., & Baki, M. F. (2020). Mathematical modeling and a hybridized bacterial foraging optimization algorithm for the flexible job-shop scheduling problem with sequencing flexibility. *Journal of Manufacturing Systems*, *54*, 74-93. https://doi.org/https://doi.org/10.1016/j.jmsy.2019.11.010

Wang, J., Liu, Y., Ren, S., Wang, C., & Wang, W. (2021). Evolutionary game based real-time scheduling for energy-efficient distributed and flexible job shop. *Journal of Cleaner Production*, *293*, 126093. https://doi.org/https://doi.org/10.1016/j.jclepro.2021.126093

Wang, L., Wang, S., Xu, Y., Zhou, G., & Liu, M. (2012). A bi-population based estimation of distribution algorithm for the flexible job-shop scheduling problem. *Computers & Industrial Engineering*, *62*(4), 917-926. https://doi.org/https://doi.org/10.1016/j.cie.2011.12.014

Wang, L., Zhou, G., Xu, Y., & Liu, M. (2013). A hybrid artificial bee colony algorithm for the fuzzy flexible job-shop scheduling problem. *International Journal of Production Research*, *51*(12), 3593-3608. https://doi.org/https://doi.org/10.1080/00207543.2012.754549

Wang, S., Lu, X., Li, X. X., & Li, W. D. (2015). A systematic approach of process planning and scheduling optimization for sustainable machining. *Journal of Cleaner Production*, *87*, 914-929. https://doi.org/https://doi.org/10.1016/j.jclepro.2014.10.008

Wu, X., Shen, X., & Li, C. (2019). The flexible job-shop scheduling problem considering deterioration effect and energy consumption simultaneously. *Computers & Industrial Engineering*, *135*, 1004-1024. https://doi.org/https://doi.org/10.1016/j.cie.2019.06.048

Wu, X., & Sun, Y. (2018). A green scheduling algorithm for flexible job shop with energy-saving measures. *Journal of Cleaner Production*, *172*, 3249-3264. https://doi.org/https://doi.org/10.1016/j.jclepro.2017.10.342

Wu, Z., Sun, S., & Yu, S. (2020). Optimizing makespan and stability risks in job shop scheduling. *Computers & Operations Research*, *122*, 104963. https://doi.org/https://doi.org/10.1016/j.cor.2020.104963

Xie, J., Li, X., Gao, L., & Gui, L. (2023). A hybrid genetic tabu search algorithm for distributed flexible job shop scheduling problems. *Journal of Manufacturing Systems*, *71*, 82-94. https://doi.org/https://doi.org/10.1016/j.jmsy.2023.09.002

Xie, N., & Chen, N. (2018). Flexible job shop scheduling problem with interval grey processing time. *Applied Soft Computing*, *70*, 513-524. https://doi.org/https://doi.org/10.1016/j.asoc.2018.06.004

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

Zhang, Y., Wang, J., Liu, S., & Qian, C. (2017). Game Theory Based Real-Time Shop Floor Scheduling Strategy and Method for Cloud Manufacturing. *International Journal of Intelligent Systems*, *32*(4), 437-463. https://doi.org/https://doi.org/10.1002/int.21868

Zhou, G., Jiang, P., & Huang, G. Q. (2009). A game-theory approach for job scheduling in networked manufacturing. *The International Journal of Advanced Manufacturing Technology*, *41*(9), 972-985. https://doi.org/https://doi.org/10.1007/s00170-008-1539-9