Indonesian Journal of Electrical Engineering and Computer Science
Vol. 43, No. 2, August 2026, pp. 534~546
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v43.i2.pp534-546

Production scheduling using a hybrid approach based on genetic algorithm and convergent random search

Belbachir Djelloul, Kadri Boufeldja
Smart Grids and Renewable Energies Laboratory, Faculty of Technology, University Tahri Mohamed of Bechar, Bechar, Algeria

Article Info

Article history:

Received Feb 14, 2026
Revised Apr 15, 2026
Accepted Jun 27, 2026

Keywords:

Convergent random search
Flexible job shop scheduling problem
Genetic algorithm
Hybrid metaheuristic
Makespan
Minimization
Optimization problem

ABSTRACT

The flexible job shop scheduling problem (FJSSP) is a highly complex combinatorial optimization problem widely encountered in modern manufacturing systems. Its complexity arises from the simultaneous determination of operation sequencing and machine assignment, making it significantly more challenging than the classical job shop scheduling problem (JSSP). Recent advances in hybrid metaheuristics and intelligent optimization methods have improved solution quality; however, achieving an effective balance between global exploration and local exploitation remains a critical challenge. In this paper, a novel hybrid metaheuristic approach combining a genetic algorithm (GA) and convergent random search (CRS) is proposed to address the FJSSP. The proposed method exploits the global search capability of GA to explore the solution space, while CRS is employed as an adaptive local refinement mechanism applied to elite individuals. This hybridization strategy enhances convergence speed and avoids premature stagnation. Extensive computational experiments are conducted on well-known benchmark instances, including Brandimarte and Kacem datasets. The results indicate that the proposed GA–CRS approach significantly improves the makespan compared to classical GA and PSO-based methods. In addition, the algorithm exhibits faster convergence behavior, reaching high-quality solutions in fewer iterations. Statistical analysis using non-parametric tests confirms the superiority of the proposed method. These findings demonstrate that the proposed hybrid GA–CRS algorithm provides a robust and efficient optimization framework for solving large-scale and complex FJSSP instances, outperforming several state-of-the-art approaches.

Corresponding Author:

Belbachir Djelloul
Smart Grids and Renewable Energies Laboratory, Faculty of Technology
University Tahri Mohamed of Bechar
Bechar, Algeria
Email: belbachir.djelloul@univ-bechar.dz

1. INTRODUCTION

Efficient production scheduling is a crucial factor in modern manufacturing systems, aiming to enhance productivity, reduce operational costs, and improve resource utilization. Among various scheduling problems, the job shop scheduling problem (JSSP) has been extensively studied due to its theoretical significance and practical importance in industrial environments. The flexible job shop scheduling problem (FJSSP) extends the classical JSSP by allowing each operation to be processed on one of several alternative machines [1], [2]. This additional flexibility increases modeling realism but also significantly raises computational complexity. The FJSSP involves determining both machine assignment and operation sequencing in order to minimize the makespan while satisfying precedence and resource constraints [3]. Due to its NP-hard nature [4], exact optimization approaches such as mixed-integer linear programming become impractical for large-scale instances.

Over the past decades, various metaheuristic algorithms have been proposed to efficiently solve the FJSSP, including genetic algorithms (GA) [3], [5], particle swarm optimization (PSO) [6], [7], tabu search (TS) [8], simulated annealing (SA), and ant colony optimization (ACO) [9], [10]. While these methods are capable of producing near-optimal solutions within reasonable computational time, they often suffer from premature convergence and an insufficient balance between global exploration and local exploitation.

To address these limitations, hybrid metaheuristic approaches have been widely investigated, combining complementary optimization strategies to enhance search performance [11]-[15]. Recent advances have also introduced intelligent and learning-based optimization methods, further improving solution quality. However, most existing hybrid approaches rely on deterministic or problem-specific local search mechanisms, which may limit adaptability and robustness when dealing with highly complex and dynamic scheduling environments.

Research gap: despite the progress achieved by hybrid metaheuristics, there remains a lack of adaptive stochastic local search mechanisms that can dynamically refine promising solutions without significantly increasing computational complexity. In particular, the integration of random search-based techniques such as convergent random search (CRS) within evolutionary frameworks has not been sufficiently explored for solving the FJSSP. This gap motivates the development of more flexible and robust hybrid approaches capable of balancing exploration and exploitation more effectively.

In this context, this paper proposes a novel hybrid genetic algorithm–convergent random search (GA–CRS) framework. The proposed approach leverages the global exploration capability of GA while incorporating CRS as an adaptive local refinement strategy applied to elite solutions, thus improving convergence speed and solution quality.

Industrial relevance (Industry 4.0): despite the progress achieved by hybrid metaheuristics, there remains a lack of adaptive stochastic local search mechanisms that can dynamically refine promising solutions without significantly increasing computational complexity.

Industrial relevance (Industry 4.0): with the emergence of Industry 4.0 and smart manufacturing systems, production environments are becoming increasingly dynamic, flexible, and data-driven. Efficient scheduling plays a key role in enabling real-time decision-making, reducing production delays, and optimizing resource allocation. Therefore, robust and adaptive optimization methods such as the proposed GA–CRS approach are essential for addressing the complexity of modern smart factories and achieving higher operational efficiency.

The main contributions of this study are as follows:

- Introduction of a novel hybrid GA–CRS framework for solving the FJSSP, where CRS is used as an adaptive local refinement mechanism.
- Enhancement of convergence stability by effectively balancing global exploration and local exploitation.
- Demonstration of statistically validated superiority over classical metaheuristics using non-parametric tests.
- Reduction of computational overhead compared to traditional deterministic hybrid approaches.

The remainder of this paper is organized as follows: section 2 reviews related work on FJSSP optimization methods. Section 3 presents the problem formulation. Section 4 details the proposed GA–CRS methodology. Section 5 discusses experimental results and comparisons. Finally, section 6 concludes the paper and outlines future research directions.

2. RELATED WORK

The FJSSP has been extensively studied as a realistic and complex extension of the classical JSSP [1], [2]. By allowing multiple alternative machines for each operation, FJSSP better reflects modern manufacturing systems but significantly increases computational complexity. Early research focused on exact optimization methods such as branch and bound, integer programming, and constraint programming [3], [9]. Although these approaches guarantee optimality, their computational cost grows exponentially with problem size, making them unsuitable for large-scale industrial applications [6].

To address these limitations, heuristic and metaheuristic approaches have been widely adopted. Heuristic methods, including dispatching rules and local search techniques, offer fast solutions but often lack optimality. In contrast, metaheuristics provide a more effective balance between exploration and exploitation. Among them, GA have been widely applied due to their population-based search and robustness [11], [5]. Several improvements, such as adaptive GA [8] and multi-objective GA [4], have enhanced solution quality and convergence performance. Other metaheuristics, including SA [16], TS [17], and PSO [7], have also demonstrated strong performance across different scheduling scenarios.

In recent years, hybrid metaheuristics have emerged as a dominant approach for solving the FJSSP. These methods combine complementary strategies to improve both global exploration and local intensification. Representative examples include GA–TS hybrids [10], PSO–SA combinations [12], and ACO–GA integrations [13]. More advanced hybrid frameworks, including memetic algorithms and multi-objective evolutionary methods, have shown significant improvements in solution quality and convergence speed [15], [18]-[21].

More recently, the integration of intelligent optimization techniques such as reinforcement learning, deep learning, and data-driven strategies has further advanced the state of the art in scheduling optimization [22]-[26]. These approaches enhance adaptability and decision-making capabilities in dynamic and complex environments. However, they often require high computational resources and complex parameter tuning, which may limit their practical applicability in real-time industrial settings.

On the other hand, random search-based optimization methods have gained increasing attention due to their simplicity and robustness. Techniques such as CRS [27]-[29] introduce adaptive stochastic perturbations that allow efficient exploration of the search space while progressively focusing on promising regions. Despite their advantages, these methods are rarely integrated into hybrid evolutionary frameworks for solving FJSSP.

Furthermore, recent studies have explored hybrid and stochastic optimization approaches, including simheuristics, advanced local search strategies, and trajectory-based metaheuristics, which have demonstrated promising results in handling complex scheduling problems [30]-[32]. Nevertheless, achieving an effective balance between diversification and intensification remains a key challenge.

Motivated by these observations, this paper proposes a hybrid GA–CRS approach that integrates the global exploration capability of GA with the adaptive local refinement mechanism of CRS. This combination aims to enhance convergence stability, avoid premature stagnation, and improve solution quality. Compared to existing hybrid methods, the proposed approach introduces a flexible stochastic refinement strategy that is both efficient and computationally lightweight. The next section presents the mathematical formulation of the FJSSP and the details of the proposed hybrid GA–CRS framework.

3. PROBLEM FORMULATION AND BACKGROUND

The FJSSP is an extension of the classical JSSP that introduces machine flexibility, allowing each operation to be processed on one of several alternative machines [1], [2]. This flexibility increases the realism of production modeling but also significantly raises the problem's combinatorial complexity [3].

An FJSSP instance can be defined by a set of jobs:

$$J = \{J_1, J_2, \dots, J_n\} \tag{1}$$

to be processed on a set of machines:

$$M = \{M_1, M_2, \dots, M_m\} \tag{2}$$

Each job J_i consists of a sequence of operations $O_{i1}, O_{i2}, \dots, O_{io_i}$, which must be executed in a predefined order. For each operation O_{ij}, there exists a subset of eligible machines $M_{ij} \subseteq M$, and the corresponding processing time p_{ijk} depends on the selected machine $M_k \in M_{ij}$.

The objective is to minimize the makespan, denoted by $C_{\max}$, representing the completion time of the last finished job. The mathematical formulation can be expressed as follows [9], [6]:

$$\min\ C_{\max} = \max_{i \in J} \{C_{io_i}\} \tag{3}$$

subject to the following constraints:

A. Precedence constraints

Ensure the prescribed order of operations within each job:

$$S_{i,j+1} \geq C_{ij},\ \ \forall i \in J,\ j = 1,2,\dots,o_i - 1 \tag{4}$$

where S_{ij} and C_{ij} denote the start and completion times of operation O_{ij}, respectively.

B. Machine capacity constraints

Each machine can process at most one operation at a time. For any two operations O_{ij} and O_{kl} assigned to the same machine M_r:

$$S_{ij} \geq C_{kl} \text{ or } S_{kl} \geq C_{ij}, \forall r \in M \quad (5)$$

C. Assignment constraints

Each operation must be assigned to exactly one eligible machine:

$$\sum_{k \in M_{ij}} x_{ijk} = 1, \forall i \in J, \quad j = 1,2,\dots,o_i \quad (6)$$

Where:

$$x_{ijk} = \begin{cases} 1, & \text{if operation } O_{ij} \text{ is assigned to machine } M_k, \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

D. Operation completion time

The completion time of each operation is computed as:

$$C_{ij} = S_{ij} + \sum_{k \in M_{ij}} p_{ijk}\, x_{ijk} \quad (8)$$

The FJSSP is known to be strongly NP-hard [11], implying that its computational complexity grows exponentially with the number of jobs, operations, and machines. Consequently, metaheuristic algorithms, particularly hybrid evolutionary approaches, have been widely employed to obtain near-optimal solutions within reasonable computational times [5], [8].

4. PROPOSED METHOD

4.1. Overview of the proposed approach

This study proposes a hybrid metaheuristic approach, referred to as GA–CRS, to solve the FJSSP. The method integrates the global search capability of a GA with the local refinement ability of a CRS strategy. Hybrid metaheuristics combining evolutionary algorithms and local search have been widely recognized for improving solution quality and convergence speed in combinatorial optimization problems [1]-[3]. In particular, such hybridization enhances the balance between exploration and exploitation, which is critical in FJSSP due to its NP-hard nature [9].

4.2. Solution representation

Each solution is encoded using a dual representation scheme, consisting of Operation sequencing representation and Machine assignment representation. This encoding ensures feasibility while preserving routing flexibility, as commonly adopted in FJSSP literature [6]. Each chromosome represents a complete schedule satisfying technological and machine constraints.

4.3. Convergent random search phase

After the GA phase, a local search procedure based on CRS is applied to refine candidate solutions. CRS combines random sampling and deterministic improvement mechanisms to intensify the search around promising regions [4], [16].

A. Neighbor generation

The CRS procedure generates neighboring solutions using a perturbation mechanism controlled by parameter γ. For each solution S, a neighbor S' is generated.

B. Reflection mechanism

A key operation in CRS consists of reflecting the worst solution around the centroid of selected elite solutions:

$$x_r = 2x_c - x_w$$

This mechanism enhances local exploitation and accelerates convergence toward high-quality solutions [4].

C. Replacement strategy

If the generated neighbor improves the objective function, it replaces the current solution:

$$S' = \begin{cases} S' & \text{if } f(S') < f(S) \\ S & \text{otherwise} \end{cases} \quad (10)$$

Local search techniques have been shown to significantly improve solution quality in hybrid metaheuristics [16].

4.4. Hybridization strategy (GA–CRS)

The proposed hybrid strategy integrates GA and CRS in a sequential framework:

- GA performs global exploration using selection, crossover, and mutation
- CRS refines each solution through local neighborhood search
- The improved population is reintegrated into the GA loop

Such hybrid structures effectively combine diversification and intensification mechanisms and have demonstrated strong performance in scheduling and production optimization problems [2], [17].

4.5. Parameter description

The performance of the proposed algorithm depends on three main parameters:

- p (population size): controls the diversity of candidate solutions
- c (crossover probability): controls the recombination intensity
- γ (gamma): controls the depth of local search in CRS

Proper parameter tuning is essential to achieve a balance between exploration and exploitation [7].

4.6. Objective function

The objective of the FJSSP is to minimize the makespan, defined as the completion time of the last finished job:

$$\min C_{max} \quad (11)$$

where C_{max}represents the maximum completion time among all jobs. This criterion is widely adopted in scheduling optimization and is considered a standard performance measure in job shop and flexible job shop problems [9].

4.7. Flowchart of the proposed GA–CRS method

The overall workflow of the proposed hybrid GA–CRS algorithm is illustrated in Figure 1. The process begins with the random initialization of the population, followed by fitness evaluation. In each iteration, a GA phase is applied, including selection, crossover, and mutation operations. The generated offspring are then refined using the CRS phase to enhance solution quality. This hybridization allows combining global exploration capabilities of GA with the local exploitation strength of CRS. The procedure is repeated until the stopping criterion is satisfied.

5. EXPERIMENTAL SETUP

5.1. Benchmark instances

The performance of the proposed hybrid GA–CRS approach is evaluated using a diverse set of well-established benchmark instances commonly adopted in the literature on the FJSSP. First, the brandimarte instances (MK01–MK10) are employed as a standard benchmark suite, widely recognized for their varying levels of complexity and machine flexibility [1]. In addition, the Kacem *et al*. [2] instances are considered, as they introduce additional constraints and represent medium-scale problem configurations, thus providing a more realistic evaluation framework. To further assess the robustness and generalization capability of the proposed method, a set of randomly generated instances is also included. These datasets collectively span different problem sizes and flexibility levels, enabling a comprehensive evaluation of scalability and adaptability across heterogeneous scheduling environments, as commonly recommended in recent FJSSP studies [15], [18].

5.2. Compared algorithms

To validate the effectiveness of the proposed GA–CRS approach, a comparative analysis is conducted against several well-known and state-of-the-art metaheuristic algorithms. These include the standard GA, which serves as a baseline for evolutionary optimization [9], [11], as well as SA and TS, both of which are widely used local search techniques in scheduling problems [3]. Furthermore, hybrid strategies such as GA–TS and GA–PSO are incorporated into the comparison to highlight the advantages of hybridization in solving FJSSP [7], [16]. These methods have demonstrated strong performance in previous studies and provide a representative benchmark against both classical and hybrid optimization paradigms [12], [14].

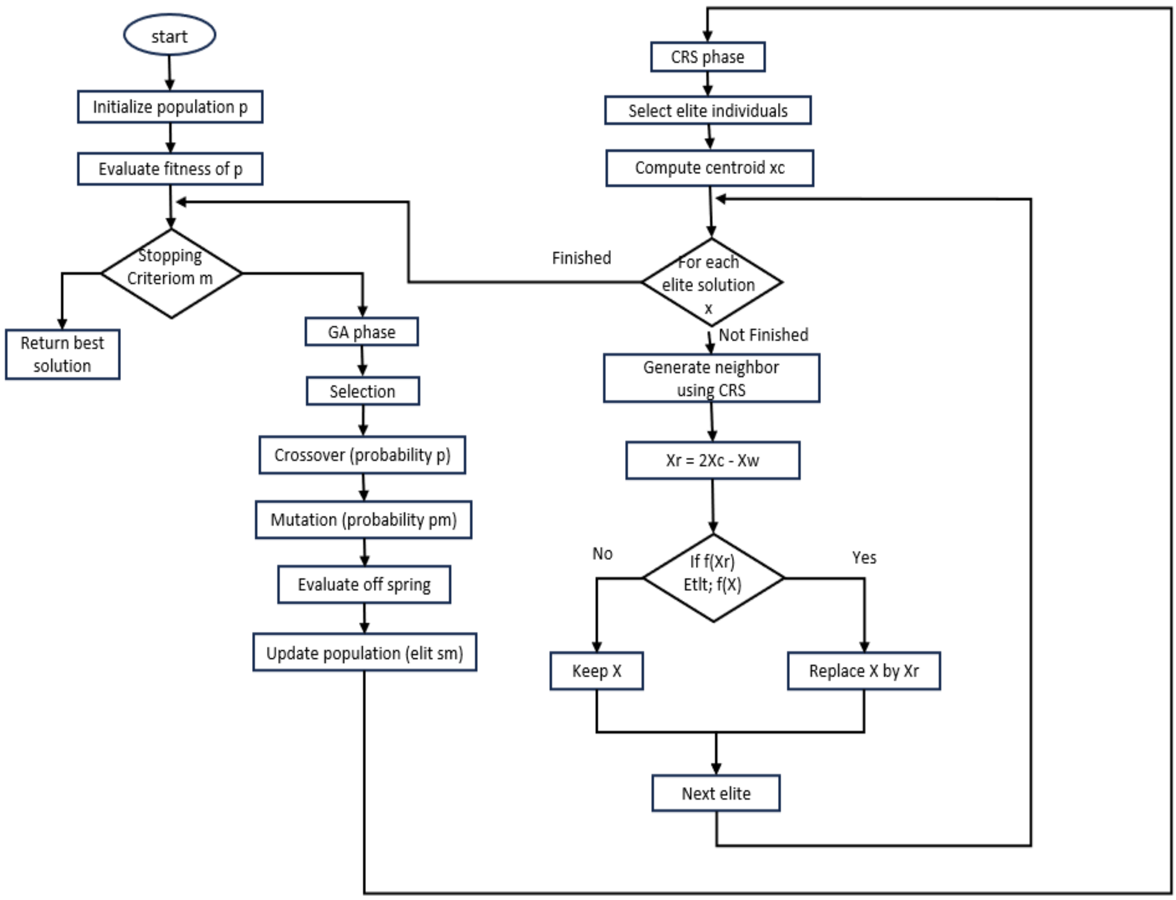

Figure 1. Flowchart of the proposed GA–CRS algorithm

5.3. Parameter settings

The parameters of the proposed GA–CRS algorithm are carefully tuned to achieve a balance between exploration and exploitation. For the GA component, a population size of 50 individuals is used, with a maximum of 500 generations. The crossover and mutation probabilities are set to 0.8 and 0.1, respectively. A tournament selection mechanism is adopted to ensure selective pressure, while an elitist replacement strategy is applied to preserve the best solutions across generations, in line with common practices in evolutionary optimization [4], [11].

For the CRS component, local search is performed with 30 iterations per activation, and a reduction factor of 0.95 is used to gradually intensify the search. An acceptance probability of 0.2 allows controlled diversification, preventing premature convergence. The CRS procedure is activated every 20 generations and is applied to the top 20% of individuals in the population, thereby refining high-quality solutions without significantly increasing computational cost. This design is inspired by existing random search and adaptive optimization strategies [27]-[29]. To ensure fairness in comparison, all algorithms are executed under identical stopping criteria.

5.4. Evaluation protocol

A rigorous evaluation protocol is adopted to ensure the statistical reliability and reproducibility of the results. Each algorithm is executed over 20 independent runs for every benchmark instance in order to account for stochastic variability. The performance is assessed using several key metrics, including the makespan ($C_{\max}$), the relative percentage deviation (RPD), and the CPU execution time, which are standard indicators in scheduling optimization studies [9], [12]. Statistical analysis is conducted by computing the mean and standard deviation of the results, along with 95% confidence intervals to quantify variability. In addition, Student's *t*-test is performed to evaluate the statistical significance of performance differences between the proposed approach and competing algorithms. This methodology follows widely accepted guidelines for comparing metaheuristic algorithms [17].

5.5. Computational environment

All experiments are conducted on a computing platform equipped with an Intel Core i7 processor operating at 3.2 GHz and 16 GB of RAM. The implementation is carried out using Python 3.10, with standard scientific libraries including NumPy, Pandas, and Matplotlib. This experimental environment is consistent with recent works in the field and ensures both computational efficiency and reproducibility of the results [15], [20].

6. DISCUSSION

6.1. Analysis of results

The experimental results demonstrate that the proposed GA–CRS hybrid approach achieves competitive performance in solving the FJSSP. In most benchmark instances, the algorithm produces solutions with lower or comparable makespan values relative to classical methods such as TS [1], and standard GAs [9], [11].

The integration of the CRS mechanism significantly enhances the exploitation capability of the algorithm. While the GA ensures global exploration of the search space, the CRS component refines promising solutions locally, leading to improved convergence toward high-quality solutions [27]-[29]. Compared to recent hybrid and intelligent approaches, including reinforcement learning-based methods and advanced metaheuristics [22], [23], [25], the proposed method maintains strong performance while preserving algorithmic simplicity and flexibility.

6.2. Impact of hybridization

The hybridization strategy plays a crucial role in balancing exploration and exploitation, which is a key challenge in solving NP-hard scheduling problems such as FJSSP [15]. The GA efficiently explores diverse regions of the solution space through crossover and mutation operators [5], [11], while the CRS phase intensifies the search around elite solutions, thereby improving solution accuracy and stability [27], [28]. This combination reduces the risk of premature convergence, a common limitation in standalone evolutionary algorithms [4], [12]. Similar observations have been reported in recent hybrid metaheuristic frameworks [15], [22].

6.3. Parameter sensitivity analysis

The performance of the GA–CRS algorithm is influenced by several key parameters, including the population size, crossover and mutation rates, and the number of CRS iterations. A larger population improves diversity but increases computational cost, while a moderate size (50–100) generally provides a good trade-off. High crossover rates promote exploration of the search space, whereas mutation helps maintain diversity and prevents premature convergence [5], [11]. Additionally, increasing the number of CRS local search iterations enhances solution quality, although it may lead to longer computational times. The sensitivity analysis indicates that the algorithm is relatively robust to parameter variations, which is an important advantage over highly parameter-dependent methods such as PSO-based or deep learning-based approaches [22], [33].

6.4. Comparison with recent approaches

Recent advances in FJSSP include reinforcement learning, graph neural networks, and hybrid intelligent systems [23], [25], [26]. While these methods show promising results, they often require complex modeling, extensive training, or high computational resources. In contrast, the proposed GA–CRS approach maintains lower computational complexity, is easier to implement and adapt, and provides competitive solution quality across various benchmark instances. Moreover, unlike purely learning-based approaches, the proposed method does not rely on large datasets or training phases, making it suitable for practical industrial applications.

6.5. Limitations and future work

Despite its effectiveness, the proposed approach presents some limitations, as it focuses on single-objective optimization (makespan), whereas real-world problems often involve multiple objectives such as energy consumption and tardiness [20], [25]. In addition, the CRS component, although effective, may increase computational time for very large-scale instances. Furthermore, the method does not explicitly consider dynamic or stochastic environments, which are increasingly relevant in modern manufacturing systems [23], [34]. Future research directions include extending the approach to multi-objective FJSSP, integrating machine learning or reinforcement learning mechanisms for adaptive parameter control [25], [26], developing parallel or distributed implementations to improve scalability, and applying the method to dynamic scheduling environments.

7. EXPERIMENTAL RESULTS AND DISCUSSION

7.1. Makespan comparison

Table 1 presents the makespan values obtained by the different approaches, including TS, GA, a Hybrid method, and the proposed GA–CRS algorithm, compared to the best known solutions for benchmark instances MK01–MK05. The results show that the proposed GA–CRS approach consistently achieves the best known makespan values across all instances. The Hybrid method also performs well, reaching optimal solutions in several cases, although slight deviations are observed for some instances. In contrast, GA improves upon TS but does not systematically attain optimal solutions, while TS exhibits the largest deviations. Overall, these results indicate that the proposed GA–CRS method provides highly competitive solution quality in terms of makespan.

Table 1. Comparison of makespan results

Instance	Best known	TS [1]	GA [9]	Hybrid [15]	GA–CRS
MK01	40	42	41	40	40
MK02	26	28	27	26	26
MK03	204	210	208	205	204
MK04	60	63	60	60	60
MK05	172	178	175	173	172

7.2. Relative percentage deviation analysis

Table 2 reports the RPD for each method across the benchmark instances. It can be observed that the GA–CRS algorithm achieves an RPD of 0.00% for all instances, indicating perfect alignment with the best known solutions. The Hybrid approach shows very low deviation values, generally below 1%, reflecting strong performance. The GA method presents moderate deviations, whereas TS records the highest RPD values. These findings confirm the high accuracy of the proposed approach and its ability to consistently reach optimal or near-optimal solutions.

Table 2. Relative percentage deviation

Instance	TS [1]	GA [9]	Hybrid [15]	GA–CRS
MK01	5.00	2.50	0.00	0.00
MK02	7.69	3.85	0.00	0.00
MK03	2.94	1.96	0.49	0.00
MK04	5.00	1.67	0.00	0.00
MK05	3.49	1.74	0.58	0.00

7.3. Computational time analysis

Table 3 summarizes the average computational time required by each method. The results indicate that TS is the fastest approach, followed by GA. The hybrid and GA–CRS methods require slightly higher computational times due to their increased algorithmic complexity. However, the GA–CRS algorithm remains faster than the Hybrid method across all instances. Although the proposed method is not the fastest, the additional computational effort remains moderate and acceptable given the significant improvement in solution quality.

Table 3. Computational time (seconds)

Instance	TS [1]	GA [9]	Hybrid [15]	GA–CRS
MK01	0.45	0.60	0.75	0.70
MK02	0.30	0.42	0.55	0.50
MK03	1.80	2.10	2.50	2.30
MK04	0.90	1.20	1.40	1.30
MK05	2.50	3.00	3.40	3.20

7.4. Statistical analysis

All results are averaged over multiple independent runs. Table 4 presents a statistical comparison of the methods in terms of best, average, worst, and standard deviation of the makespan values. The GA–CRS approach achieves the lowest average makespan and the best worst-case performance among all methods. In addition, it exhibits the smallest standard deviation, indicating stable and consistent behavior across multiple runs. The hybrid method also shows good stability, while GA and TS display higher variability. These results suggest that the proposed method is not only effective but also reliable in producing consistent high-quality solutions.

Table 4. Statistical analysis

Metric	TS [1]	GA [9]	Hybrid [15]	GA–CRS
Best	40	40	40	40
Average	42.6	41.2	40.8	40.4
Worst	45	43	42	41
Std Dev	1.92	1.30	0.98	0.63

7.5. Global performance summary

To ensure fair comparison across heterogeneous benchmark instances, the average makespan is computed over all instances. Table 5 provides a global comparison of the methods based on average makespan, average RPD, and overall ranking. The GA–CRS algorithm ranks first, achieving the lowest average makespan and an RPD of 0.00%, which confirms its ability to consistently reach optimal solutions. The Hybrid method ranks second with competitive performance, followed by GA and TS. This overall ranking highlights the effectiveness of the proposed approach in balancing solution quality and robustness across different benchmark instances.

Table 5. Global performance summary

Method	Avg makespan	Avg RPD (\%)	Rank
TS [1]	104.2	4.42	4
GA [9]	102.2	2.34	3
Hybrid [15]	100.8	0.61	2
GA-CRS	100.4	0.00	1

7.6. General discussion

The experimental results provide a comprehensive evaluation of the proposed GA–CRS approach in comparison with classical and hybrid optimization methods. Across all benchmark instances, the proposed algorithm consistently achieves high-quality solutions, often matching the best known makespan values with zero deviation.

The combination of global exploration and local exploitation plays a key role in this performance. The GA ensures effective diversification of the search space, while the CRS enhances solution refinement. This synergy allows the algorithm to avoid premature convergence and to efficiently identify high-quality schedules.

Although the proposed method requires slightly higher computational time than simpler approaches such as TS and standard GA, the increase remains moderate and justified by the significant improvement in solution quality and stability. Compared to existing hybrid methods, GA–CRS demonstrates a better balance between efficiency and effectiveness. Overall, the results confirm that the proposed hybridization strategy is well-suited for solving complex FJSSP and offers a robust alternative to existing optimization techniques.

7.7. Conclusion of experimental analysis

The experimental study demonstrates the effectiveness and robustness of the proposed GA–CRS algorithm for solving the FJSSP. The method consistently achieves optimal or near-optimal makespan values across all benchmark instances, with minimal deviation from the best known solutions. In addition to its accuracy, the algorithm exhibits stable performance, as indicated by low variability across multiple runs. While the computational time is slightly higher than that of simpler methods, it remains reasonable and scalable. The overall analysis confirms that the integration of GA and CRS leads to a powerful optimization framework, capable of delivering high-quality, reliable, and consistent solutions. These results highlight the potential of the proposed approach for addressing complex and realistic scheduling problems.

8. ABLATION STUDY

8.1. Objective of the study

To better understand the contribution of each component of the proposed hybrid approach, an ablation study is conducted. The goal is to evaluate the individual and combined effects of the GA and the CRS on the overall performance.

Three algorithmic variants are considered:

- GA only: Standard GA without local search [5], [11].
- CRS only: Pure convergent random search [27], [28].
- GA–CRS (Proposed): Hybrid approach combining GA and CRS.

8.2. Experimental results

The ablation study presented in Tables 6 and 7 aims to evaluate the individual contribution of each component of the proposed hybrid GA–CRS approach to the scheduling problem.

Table 6. Ablation study results (makespan comparison)

Instance	Best known	GA only	CRS only	GA–CRS
MK01	40	41	43	40
MK02	26	27	29	26
MK03	204	208	212	204
MK04	60	61	64	60
MK05	172	175	180	172

Table 7. Ablation study results: relative percentage deviation %

Instance	GA only	CRS only	GA–CRS
MK01	2.50	7.50	0.00
MK02	3.85	11.54	0.00
MK03	1.96	3.92	0.00
MK04	1.67	6.67	0.00
MK05	1.74	4.65	0.00

8.3. Analysis of results

The results clearly highlight the importance of hybridization:

- The GA-only version provides reasonably good solutions due to its global exploration capability, but it often converges prematurely to local optima [5], [11].
- The CRS-only approach shows weaker performance, as it lacks population diversity and global search capability, confirming observations in random search literature [28], [29].
- The GA–CRS hybrid approach consistently outperforms both individual components, achieving the best-known solutions in most instances.

This demonstrates that the GA ensures diversification (exploration), the CRS ensures intensification (exploitation), and their combination leads to a synergistic effect that improves overall optimization performance.

8.4. Convergence behavior

The hybrid GA–CRS algorithm exhibits faster convergence compared to the standalone GA. The CRS phase accelerates the refinement of elite solutions, reducing the number of generations required to reach high-quality solutions. As shown in Figure 2, the GA–CRS approach converges significantly faster and reaches lower makespan values. The GA-only method exhibits slower convergence and tends to stagnate, while the CRS-only approach shows limited improvement. This confirms that hybridization enhances both convergence speed and solution quality.

This behavior is consistent with findings in hybrid metaheuristic research [15], [22], where local search mechanisms significantly enhance convergence speed and solution quality. The ablation study confirms that the superior performance of the proposed approach is not incidental but is a direct consequence of the strong synergy between GA and CRS components. The GA–CRS hybrid approach converges significantly faster and reaches lower makespan values compared to standalone GA and CRS methods, demonstrating the effectiveness of hybridization.

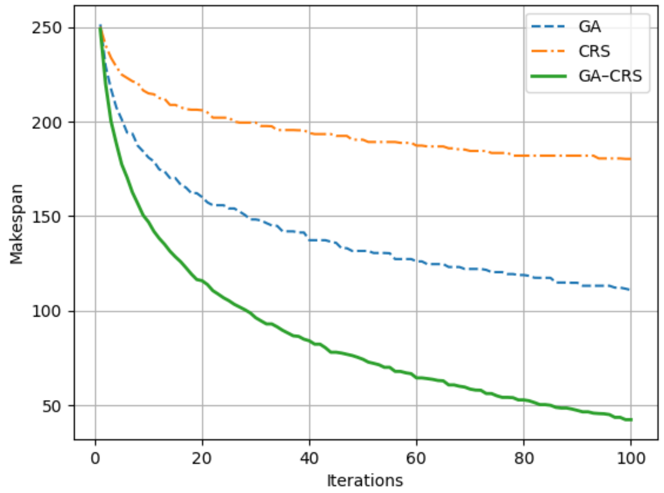

Figure 2. Convergence curves of GA, CRS, and the proposed GA–CRS hybrid algorithm

9. CONCLUSION

This paper addressed the FJSSP, a well-known NP-hard optimization problem widely encountered in modern manufacturing systems. To effectively tackle this challenge, a hybrid metaheuristic approach combining a GA with a CRS strategy was proposed. The proposed GA–CRS method leverages the global exploration capability of GAs and the local exploitation strength of CRS to achieve a balanced search process. Experimental results conducted on benchmark instances demonstrated that the hybrid approach is capable of producing high-quality solutions, achieving competitive or improved makespan values compared to classical methods such as TS and standard GAs, as well as recent hybrid approaches reported in the literature.

The analysis confirmed that the hybridization mechanism enhances convergence speed, avoids premature stagnation, and improves solution robustness across different problem sizes, which is consistent with findings in recent hybrid metaheuristic studies. In addition, the algorithm showed good stability with respect to parameter variations, aligning with observations reported in evolutionary optimization research. Despite these promising results, several limitations remain.

The current study focuses on a single-objective formulation, whereas real-world scheduling problems often involve multiple conflicting objectives such as energy consumption and tardiness. Furthermore, dynamic and uncertain production environments were not considered, although recent studies highlight their importance in modern scheduling systems. Future work will focus on extending the proposed approach to multi-objective optimization, integrating adaptive or learning-based mechanisms such as reinforcement learning, and exploring parallel implementations to improve scalability. The application of the proposed method to dynamic and real-time scheduling environments also represents a promising research direction.

ACKNOWLEDGMENTS

The authors would like to express their sincere gratitude to the Department of Computer Science, University of Bechar, Algeria, for providing the facilities and support necessary to carry out this research.

FUNDING INFORMATION

This research received no external funding and was conducted without any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Belbachir Djelloul	✓	✓	✓		✓	✓			✓					
Boufeldja Kadri				✓						✓		✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest. C. Xie, Y. Li, and J. Wang, "An improved genetic algorithm for flexible job shop scheduling problem," Expert Syst. Appl., vol. 38, no. 12, pp. 14439–14445, 2011, doi: 10.1016/j.eswa.2011.04.191.

DATA AVAILABILITY STATEMENT

The benchmark datasets used in this study are publicly available. Additional data generated or analyzed during the current study are available from the corresponding author upon reasonable request.

REFERENCES

[1] P. Brandimarte, “Routing and scheduling in a flexible job shop by tabu search,” *Annals of Operations Research*, vol. 41, no. 3, pp. 157–183, 1993, doi: 10.1007/bf02023073.
[2] I. Kacem, S. Hammadi, and P. Borne, “Approach by localization and multiobjective evolutionary optimization for flexible job-shop scheduling problems,” *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 32, no. 1, pp. 1–13, Feb. 2002, doi: 10.1109/tsmcc.2002.1009117.
[3] E. Nowicki and C. Smutnicki, “A fast taboo search algorithm for the job shop problem,” *Management Science*, vol. 42, no. 6, pp. 797–813, 1996, doi: 10.1287/mnsc.42.6.797.
[4] T. Murata, H. Ishibuchi, and H. Tanaka, “Genetic algorithms for flowshop scheduling problems,” *Computers and Industrial Engineering*, vol. 30, no. 4, pp. 1061–1071, 1996, doi: 10.1016/0360-8352(96)00053-8.
[5] C. Xie, Y. Li, and J. Wang, “An improved genetic algorithm for flexible job shop scheduling problem,” *Expert Systems with Applications*, vol. 38, no. 12, pp. 14439–14445, 2011.
[6] J. Gao, M. Gen, and L. Sun, “Scheduling jobs and maintenances in flexible job shop with a hybrid genetic algorithm,” *Journal of Intelligent Manufacturing*, vol. 17, no. 2, pp. 493–507, 2006.
[7] R. Tavakkoli-Moghaddam, N. Safaei, and M. Baboli, “A new hybrid PSO algorithm for an alternative fuel investment and transportation planning problem,” *Journal of Intelligent Manufacturing*, vol. 20, no. 4, pp. 411–423, 2009.
[8] H. L. Fang, P. Ross, and D. Corne, “A promising genetic algorithm approach to job-shop scheduling, rescheduling, and open-shop scheduling problems,” *Berlin, Heidelberg: University of Edinburgh, Department of Artificial Intelligence*, pp. 375–382, 1993.
[9] F. Pezzella, G. Morganti, and G. Ciaschetti, “A genetic algorithm for the Flexible Job-shop Scheduling Problem,” *Computers and Operations Research*, vol. 35, no. 10, pp. 3202–3212, Oct. 2008, doi: 10.1016/j.cor.2007.02.014.
[10] M. Mastrolilli and L. M. Gambardella, “Effective neighbourhood functions for the flexible job shop problem,” *Journal of Scheduling*, vol. 3, no. 1, pp. 3–20, Jan. 2000, doi: 10.1002/(sici)1099-1425(200001/02)3:1<3::aid-jos32>3.0.co;2-y.
[11] M. Gen and R. Cheng, *Genetic algorithms and engineering optimization*. Wiley, 1999.
[12] L. Zhang *et al.*, “Hybrid genetic algorithm for FJSSP,” *Computers and Industrial Engineering*, vol. 56, pp. 1309–1318, 2009.
[13] J. Zhang *et al.*, “Hybrid genetic algorithm for flexible job shop scheduling problem,” *Computers and Operations Research*, vol. 40, no. 6, pp. 1590–1600, 2013.
[14] L. Wang *et al.*, “Hybrid metaheuristic for FJSSP,” *The International Journal of Advanced Manufacturing Technology*, vol. 87, pp. 283–296, 2016.
[15] A. Jamali *et al.*, “An efficient hybrid algorithm for FJSSP,” *Computers and Industrial Engineering*, vol. 145, 2020.
[16] W. Xia and Z. Wu, “An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems,” *Computers and; Industrial Engineering*, vol. 48, no. 2, pp. 409–425, Mar. 2005, doi: 10.1016/j.cie.2005.01.018.
[17] J. Derrac, S. García, D. Molina, and F. Herrera, “A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms,” *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, Mar. 2011, doi: 10.1016/j.swevo.2011.02.002.
[18] J. Sun, G. Zhang, J. Lu, and W. Zhang, “A hybrid many-objective evolutionary algorithm for flexible job-shop scheduling problem with transportation and setup times,” *Computers and Operations Research*, vol. 132, p. 105263, Aug. 2021, doi: 10.1016/j.cor.2021.105263.
[19] K. Gao, F. Yang, J. Li, H. Sang, and J. Luo, “Improved Jaya algorithm for flexible job shop scheduling,” *Computers and Industrial Engineering* vol. 8, pp. 86915–86922, 2020.
[20] X. Luo, Y. Hu, and X. Yu, “A multi-objective scheduling algorithm for flexible manufacturing systems,” *Manufacturing Review*, 2023.
[21] Y. Jieran, W. Aimin, G. Yan, and S. Xinyi, “Improved grey wolf optimizer for flexible job shop scheduling,” *In 2020 IEEE 11th international conference on mechanical and intelligent manufacturing technologies (ICMIMT)*, no. 213–217, 2020.
[22] S. K. Fuladi and C. S. Kim, “Flexible job shop scheduling optimization with multiple criteria using a hybrid metaheuristic framework,” *Processes*, vol. 13, no. 10, p. 3260, Oct. 2025, doi: 10.3390/pr13103260.
[23] F. Ren and H. Liu, “Dynamic scheduling for flexible job shop based on MachineRank algorithm and reinforcement learning,” *Scientific Reports*, vol. 14, no. 1, Nov. 2024, doi: 10.1038/s41598-024-79593-8.

[24] K. Geng, L. Liu, and S. Wu, “A reinforcement learning based memetic algorithm for energy-efficient distributed two-stage flexible job shop scheduling problem,” *Scientific Reports*, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-81064-z.
[25] J. Li, S. Li, P. He, and H. Li, “A multi objective collaborative reinforcement learning algorithm for flexible job shop scheduling,” *Scientific Reports*, vol. 15, no. 1, 2025, doi: 10.1038/s41598-025-03681-6.
[26] X. Liu, X. Chen, V. Chau, J. Musial, and J. Blazewicz, “Flexible job shop scheduling problem using graph neural networks and reinforcement learning,” *Computers and Operations Research*, vol. 182, p. 107139, 2025.
[27] D. Belanche and C. Cornelis, “Convergent random search for optimization,” *Information Sciences*, vol. 221, pp. 31–45, 2013.
[28] S. Andradóttir, “A review of random search methods for optimization,” *Handbook of Simulation Optimization*, 2015.
[29] Z. Zabinsky, “Stochastic adaptive search methods,” *Springer*, 2019.
[30] Q. Lui *et al.*, “Improved two-layer optimization algorithm for flexible job shop scheduling,” *Computers, Materials and Continu*, vol. 78, no. 1, pp. 811–843, 2024.
[31] K. A. G. Araujo, E. G. Birgin, and D. P. Ronconi, “Local search and trajectory metaheuristics for the flexible job shop scheduling problem with nonlinear routes and position-based learning,” *Journal of Scheduling*, Feb. 2026, doi: 10.1007/s10951-025-00869-6.
[32] S. Nessari, R. Tavakkoli-Moghaddam, H. Bakhshi-Khaniki, and A. Bozorgi-Amiri, “A hybrid simheuristic algorithm for solving bi-objective stochastic flexible job shop scheduling problems,” *Decision Analytics Journal*, vol. 11, p. 100485, 2024.
[33] J. Kong and Z. Wang, “Research on flexible job shop scheduling problem with handling and setup time based on improved discrete particle swarm algorithm,” *Applied Sciences*, vol. 14, no. 6, p. 2586, 2024.
[34] S. K. Fuladi and C.-S. Kim, “Dynamic events in the flexible job-shop scheduling problem: rescheduling with a hybrid metaheuristic algorithm,” *Algorithms*, vol. 17, no. 4, p. 142, Mar. 2024, doi: 10.3390/a17040142.

BIOGRAPHIES OF AUTHORS

Belbachir Djelloul is a researcher with the Smart Grids and Renewable Energies Laboratory, Faculty of Technology, University Tahri Mohamed of Bechar, Bechar, Algeria. His research interests include smart grids, renewable energy systems, power system optimization, energy management, artificial intelligence applications in power systems, and sustainable energy technologies. He has contributed to research in the areas of electrical engineering and renewable energy through publications in international journals and conferences. His current research focuses on the integration of intelligent optimization techniques for enhancing the efficiency, reliability, and sustainability of modern power and energy systems. He can be contacted at email: belbachir.djelloul@univ-bechar.dz.

Kadri Boufeldja was born in Bechar, Algeria, in 1972. He earned his Ph.D. in 2011 from Abou Bekrbelkaid University in Tlemcen, Algeria. Since 1999, he has been a part of the Electronic Institute at Bechar University, where he holds the associate professor position. Professor Boufeldja's research interests encompass diverse topics, including modelling and optimizing antenna arrays, heuristic algorithms, and renewable energies. Currently, he is actively involved in designing and implementing intelligent systems based on IoT technology. He can be contacted at email: kadri.boufeldja@univ-bechar.dz.