Distributed and Flexible Job Shop Scheduling Approaches

Yizheng Shi
Yale University

Abstract

This paper presents a literature review focusing on the evolution of job shop scheduling (JSS) research as it transitions into flexible (FJSSP) and distributed (DJSSP) environments. Genetic Algorithms (GAs) have been refined over time to handle increasing complexity, while Multi-Agent Systems (MAS) provide decentralized, adaptive decision-making suited to large-scale, dynamic, and distributed settings. This synthesis highlights key methodological advancements, the integration of global optimization with localized responsiveness, and future avenues for combining these approaches in modern manufacturing networks.

1 Introduction

As global manufacturing networks evolve, the traditional Job Shop Scheduling Problem (JSSP)—where a finite set of jobs, each composed of ordered operations, must be assigned to specific machines—has become insufficient to capture real-world complexity. Modern production systems often span multiple facilities (distributed) and feature operational flexibility where each operation can be processed by multiple machines (flexible). This evolution has led to the Flexible Job Shop Scheduling Problem (FJSSP) and the Distributed Job Shop Scheduling Problem (DJSSP), both known to be NP-hard and increasingly challenging to solve optimally at scale [1, 5].

Across the literature, researchers have explored a variety of computational strategies. Early methodologies employed classical heuristics and exact methods, which quickly became intractable for larger instances. Subsequently, metaheuristics—particularly Genetic Algorithms (GAs)—gained traction due to their adaptability and capacity to search large solution spaces efficiently [4, 6, 8]. More recently, decentralized and adaptive decision-making frameworks, particularly multi-agent systems (MAS), have emerged as promising tools for handling the dynamic, heterogeneous, and geographically distributed nature of modern job shops [7, 9].

This literature review organizes existing work around three key axes: (1) increasing complexity in the scheduling environment, (2) the evolution of solution techniques, and (3) the integration of global optimization strategies with local, adaptive decision-making. By structuring the discussion in this manner, it elucidates how the field has progressed from single-site, fixed scheduling scenarios toward complex, multi-site networks requiring advanced optimization methods.

2 Increasing Problem Complexity: From Single-Facility to Flexible and Distributed Scenarios

2.1 Classical Job Shop Scheduling (JSSP)

Initial scheduling research focused on the classical JSSP, wherein each operation has a predetermined machine, and jobs follow a strict processing order. Foundational studies using GAs to solve JSSPs illustrated how genetic operators could encode operation sequences and yield near-optimal results within a single facility setting [4]. This early work established a baseline for comparing how much complexity would be added by moving into flexible or distributed environments.

2.2 Flexible Job Shop Scheduling (FJSSP)

With flexibility introduced, each operation can be executed on one of several machines, offering improved robustness against disruptions and the opportunity for better load balancing. Approaches tailored to FJSSP, such as the improved GA proposed in [3], integrated problem-specific encodings that handle flexible machine assignments alongside operation sequencing. This adaptation allowed the genetic search process to exploit the additional degrees of freedom provided by machine flexibility, leading to enhanced resource utilization and shorter makespans [1].

2.3 Distributed Job Shop Scheduling (DJSSP)

Distributed environments raise the complexity further, requiring coordination across multiple facilities—each potentially following its own local constraints and objectives. Studies addressing DJSSP [5, 8] highlight how the scheduling challenge now incorporates inter-factory routing times, communication overhead, and the synchronization of operations across different geographical or organizational units. These considerations make the solution space even larger, pushing classical optimization methods to their limits and necessitating more advanced and adaptive approaches [9].

3 The Evolution of Solution Techniques: From Heuristics to Advanced Metaheuristics and Agent Systems

3.1 Early Heuristics and Metaheuristics

Before the widespread use of GAs, simpler heuristics and dispatching rules were common. While computationally tractable, these methods often struggled to handle complexity or adapt to changes in the environment. Metaheuristics, particularly GAs, emerged as a powerful alternative, as seen in works such as [4, 6]. Their population-based approach and recombination operators provided a means to navigate vast solution spaces and avoid local optima [1].

3.2 Specialized Genetic Algorithms for FJSSP and DJSSP

As complexity increased, researchers refined GAs with specialized operators, hybrid local searches, and advanced encoding schemes. The improved GA presented in [**?**] for the distributed and flexible scenario exemplifies this trend. By accommodating both machine flexibility and multi-site constraints within the chromosome and genetic operators, these GAs maintained feasibility and diversity, accelerating convergence toward high-quality solutions [5]. The adaptability of the GA framework thus proved essential for tackling increasingly complex variants of the scheduling problem [2].

3.3 Multi-Agent Systems (MAS)

While GAs excel at global optimization under relatively static assumptions, MAS have emerged as a means to handle dynamic and distributed environments. In MAS-based approaches [7, 9], each agent—representing a machine, a shop, or another decision-making entity—acts autonomously and interacts with others to negotiate task assignments and schedules. This decentralized structure inherently supports scalability and responsiveness, allowing the system to quickly adapt to real-time changes such as machine breakdowns, shifting priorities, or rush orders [7, 8].

4 Integrating Global Optimization and Local Adaptation

4.1 Combining GAs and MAS for Enhanced Performance

Although GAs and MAS have been studied largely in parallel, the literature suggests potential synergy. While GAs can generate robust baseline solutions, MAS can provide localized, real-time adjustments that fine-tune schedules in the face of disturbances. For example, GAs could be used to determine initial inter-factory allocations, and agents then refine these allocations as conditions change, making the overall system more resilient and adaptive [7, 9].

4.2 Robustness, Scalability, and Real-Time Adjustments

As FJSSP and DJSSP scenarios grow in scale—both in terms of machine count and geographical dispersion—the importance of robust and scalable methods increases. GAs can maintain good global coverage of the solution space, while MAS distribute decision-making load and handle real-time data inputs. By blending these strengths, the literature points toward frameworks capable of continuous and dynamic rescheduling, ensuring that production lines remain optimized even under uncertainty or rapid environmental changes [3].

5 Conclusion

The trajectory of job shop scheduling research—from classical JSSP to FJSSP and DJSSP—reveals a clear pattern: as the problem domain grows more complex, solution methodologies must evolve accordingly. Genetic algorithms have matured from basic encodings to sophisticated, problem-specific operators that handle flexibility and distribution. Meanwhile, agent-based frameworks provide a decentralized, reactive approach suitable for dynamic, large-scale manufacturing networks. Together, these innovations point toward future solutions that combine global optimization with local adaptation, ultimately yielding more resilient, efficient, and scalable scheduling for modern production environments.

References

[1] H. Chen, J. Ihlow, and C. Lehmann. A genetic algorithm for flexible job-shop scheduling. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, volume 2, pages 1120–1125 vol.2, 1999.

[2] Yaoyao Dai, Lin Zhou, Xutong Shu, Jianyong Chen, Yi Wang, and Yidong Ke. Research on flexible job-shop scheduling problem based on genetic algorithm and

flower pollination algorithm. In *2023 11th International Conference on Intelligent Computing and Wireless Optical Communications (ICWOC)*, pages 61–66, 2023.

[3] Jiang-Ping Huang, Liang Gao, and Xin-Yu Li. A hierarchical multi-action deep reinforcement learning method for dynamic distributed job-shop scheduling problem with job arrivals. *IEEE Transactions on Automation Science and Engineering*, pages 1–13, 2024.

[4] K. Krishna, K. Ganeshan, and D.J. Ram. Distributed simulated annealing algorithms for job shop scheduling. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(7):1102–1109, 1995.

[5] Tung-Kuan Liu, Yeh-Peng Chen, and Jyh-Horng Chou. Solving distributed and flexible job-shop scheduling problems for a real-world fastener manufacturer. *IEEE Access*, 2:1598–1606, 2014.

[6] Bilel Marzouki, Olfa Belkahla Driss, and Khaled Ghédira. Decentralized tabu searches in multi agent system for distributed and flexible job shop scheduling problem. In *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*, pages 1019–1026, 2017.

[7] Shi Yang, Yong Xu, Qi Zheng, Yuhu Zhu, Hongyu Zhang, and Kangrui Zhai. Multi-agent distributed collaborative method for flexible job shop scheduling problem. In *2022 37th Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pages 133–138, 2022.

[8] Zhang Yu-xian, Li Lei, Wang Hong, Zhao Yan-yan, Guo Xu, and Meng Chun-hua. Approach to the distributed job shop scheduling based on multi-agent. In *2008 IEEE International Conference on Automation and Logistics*, pages 2031–2034, 2008.

[9] Henxi Zhang, Minghai Yuan, Liang Zheng, and Fengque Pei. An improved genetic algorithm for solving flexible job shop. In *2021 3rd International Symposium on Robotics Intelligent Manufacturing Technology (ISRIMT)*, pages 269–272, 2021.