Enhanced Hybrid Gaussian Quantum Particle Swarm Optimization with Adaptive Genetic Algorithm for Flexible Job Shop Scheduling

Mohamed Madassi[1], Imed Eddine Talhi[1], and Sara Sabba[2]

[1] Faculty of New Technologies of Information and Communication, Abdelhamid Mehri University Constantine 2, Constantine 25000, Algeria

[2] Faculty of New Technologies of Information and Communication, Laboratory of Data Science and Artificial Intelligence (LISIA), Abdelhamid Mehri University Constantine 2, Constantine 25000, Algeria

mohamed.madassi@univ-constantine2.dz

Abstract. The Flexible Job Shop Problem (FJSP) is an extension of the classical job shop scheduling problem. It assigns each operation to a machine and sequences the operations on each machine with the aim of minimizing the maximum completion time of all operations. The FJSP is an NP-hard combinatorial optimization problem with critical applications in smart manufacturing, Industry 4.0, and digital twin-based production systems. This paper presents an enhanced version of the Hybrid Gaussian Quantum Particle Swarm Optimization with Adaptive Genetic Algorithm (HGQPSO-AGA) which introduced four key improvements: integration of Gaussian quantum genetic operators into the adaptive genetic algorithm phase, a greedy decoding strategy for intelligent schedule construction, multi-neighborhood local search for solution refinement, and path relinking for intensification. The computational experiment was conducted on 18 benchmark instances from the Kacem, Brandimarte, and Dauzère-Pérès datasets. The results demonstrate that the enhanced algorithm achieves substantial improvements, with an average makespan reduction of 31.7%. Statistical tests confirm the significance of these gains, with the enhanced HGQPSO-AGA achieving a perfect win record across all 90 experimental runs. These findings establish HGQPSO-AGA as a highly effective approach for solving complex scheduling problems.

Keywords: Flexible Job-Shop Scheduling, Quantum Particle Swarm Optimization, Adaptive Genetic Algorithm, Hybrid Metaheuristics, Combinatorial Optimization.

INTRODUCTION

Production scheduling is fundamental to modern manufacturing systems, directly impacting operational efficiency and competitiveness. The Flexible Job-Shop Scheduling Problem (FJSP), first introduced by Brucker and Schlie [1], extends the classical Job-Shop Scheduling Problem (JSP) by allowing each operation to be processed on any machine from a given set of alternatives, thereby offering greater flexibility in produc-

D. Agti et al. (eds.), *Proceedings of the International Conference on Artificial Intelligence Applications in Business Administration in MENA Region (ICAIABA 2026)*, Advances in Economics, Business and Management Research 393,
https://doi.org/10.2991/978-94-6239-711-8_38

tion planning. FJSP is classified as an NP-hard problem [2], meaning that its computational complexity grows exponentially with problem size, rendering exact solution methods impractical for large scale instances.

Building on recent advances, Xu et al. [3] proposed a Hybrid Gaussian Quantum Particle Swarm Optimization with Adaptive Genetic Algorithm (HGQPSO-AGA) that combines GQPSO's global search with AGA's local refinement. This paper proposes an enhanced HGQPSO-AGA with four key innovations:

1. Integration of Gaussian quantum operators into the genetic algorithm phase,
2. Implementation of greedy decoding and multi-neighborhood local search,
3. Addition of path relinking as a dedicated intensification mechanism,
4. Rigorous statistical validation using Friedman and Nemenyi tests.

The remainder of the paper is as follows. Section 2 reviews related work. Section 3 presents methodology. Section 4 reports results. Section 5 concludes.

LITERATURE REVIEW

To address the complexity of FJSP, a wide range of solution approaches has been proposed, broadly categorized into exact and approximate methods. Exact methods [4, 5] guarantee optimal solutions but are often computationally prohibitive beyond small instances. In contrast, approximate methods [6], local search techniques [7], and metaheuristic algorithms [8, 9] seek high quality solutions within reasonable time frames. Among these, hybrid metaheuristics that combine complementary optimization paradigms have shown particular promise.

Within the family of metaheuristics, Particle Swarm Optimization (PSO), introduced by Kennedy and Eberhart [10], has been successfully applied to scheduling problems due to its simplicity and effectiveness. Quantum PSO (QPSO), proposed by Sun et al. [11], leverages principles from quantum mechanics to improve population diversity, while Gaussian QPSO (GQPSO) [12] further refines search behavior using Gaussian probability distributions. Singh and Mahapatra [13] demonstrated QPSO's effectiveness on FJSP, and Xu et al. [14] advanced the approach by integrating GQPSO with chaotic maps for improved initialization.

Adaptive Genetic Algorithms (AGA), introduced by Srinivas and Patnaik [15], offer robust evolutionary search through self adjusting crossover and mutation rates. Pezzella et al. [16] demonstrated AGA's effectiveness for FJSP, while quantum-inspired genetic operators [17] further bridged evolutionary and quantum paradigms. Hybrid approaches have shown particular promise: Zhang et al. [18] combined tabu search and

PSO, Xie et al. [19] enhanced hybrid algorithms for faster convergence, and path re-linking an intensification strategy from Glover and Laguna [20]has been successfully integrated to refine elite solutions.

METHODOLOGY

Problem Formulation

FJSP involves scheduling n jobs on m machines, where each job J_i consists of operations $\{O_{i,1}, O_{i,2}, ..., O_{i,n\_i}\}$. Each operation can be processed on a subset of machines with specific processing times. The objective is to minimize the makespan C_{max}, which represents the total completion time of the schedule (i.e., the time at which the last operation across all jobs finishes):

$$min\ C_{ma}{}^{x} = min\ max_{i}\square J\ C_{i,ni} \qquad (1)$$

Subject to the constraints that each operation is assigned to exactly one machine (2), each operation starts after its predecessor completes (3), and the completion time equals start time plus processing time (4).

Original HGQPSO-AGA Algorithm

The original algorithm [3] employs a two-phase hybrid framework that alternates between global exploration via Gaussian Quantum PSO (GQPSO) and local exploitation via an Adaptive Genetic Algorithm (AGA).

Phase 1 – GQPSO. The swarm is evolved using a quantum-inspired position update rule that enhances population diversity:

$$x^{t+1}{}_{i} = S \pm \beta \cdot |m^{best} - x^{t}_{i}| \cdot ln(1/u) \qquad (5)$$

where $S = \theta \cdot p_i + (1 - \theta) \cdot g$ acts as a local attractor combining the particle's personal best (p_i) and the global best (g), m^{best} denotes the mean of all personal best positions, and β is a time-varying contraction–expansion coefficient defined as:

$$\beta = \beta_{m}{}^{In} + (\beta_{ma}{}^{x} - \beta_{m}{}^{In}) \cdot (t_{ma}{}^{x} - t) / t_{ma}{}^{x} \qquad (6)$$

Phase 2 – AGA. The population is split into superior and inferior subpopulations based on fitness. Superior individuals undergo adaptive crossover; inferior ones undergo adaptive mutation, with both probabilities adjusted dynamically during evolution.

Chaotic Encoding. To improve initial population diversity, four chaotic maps are employed for position initialization: Iterative, Sinusoidal, Logistic, and Cubic maps. These generate non-repeating, ergodic sequences that help avoid premature convergence during initialization.

Enhanced HGQPSO-AGA Algorithm

We extend the original two-phase HGQPSO-AGA into a four-phase framework to better balance exploration, adaptation, refinement, and intensification.

Phase 1 – GQPSO with Exponential Decay. We replace the original linear decay of the contraction–expansion coefficient with an exponential schedule to maintain stronger exploration early on and enable faster convergence later:

$$\beta(t) = \beta_m{}^{In} + (\beta_{ma}{}^{x} - \beta_m{}^{In}) \cdot e^{-3t}/T \qquad (7)$$

Phase 2 – Gaussian Quantum AGA. We introduce quantum-inspired genetic operators to enhance diversity and convergence. The adaptive crossover probability and Gaussian quantum crossover/mutation operators are defined as follows:

$$p^d(f) = (p^d{}_{,m}{}^{In} + p^d{}_{,ma}{}^{x})/2 + (p^d{}_{,m}{}^{In} - p^d{}_{,ma}{}^{x})/\pi \cdot \arctan((2f - f_m{}^{In} - f_{av}{}^{k})/(f_{av}{}^{k} - f_m{}^{In} + \varepsilon)) \qquad (8)$$

Phase 3 – Multi-Neighborhood Local Search. A systematic local search refines the top elite solutions every 2 iterations using three complementary neighborhoods: swap, insert, and 2-opt.

Phase 4 – Path Relinking. Path relinking is applied every 5 iterations between the best and a guiding elite solution to explore high-quality trajectories in the search space.

Algorithmic Comparison

Table 1 summarizes the key differences between the original and enhanced algorithms. The enhanced algorithm strengthens both exploration (via exponential decay and quantum-inspired operators) and exploitation (via multi-neighborhood local search and path relinking), leading to significantly improved solution quality.

Table 1. Comparison of Original and Enhanced HGQPSO-AGA.

Feature	Original	Enhanced
Algorithm Phases	2	4
β Decay Function	Linear	Exponential
Genetic Operators	AGA	Gaussian Quantum AGA
Local Search	None	Multi-neighborhood
Path Relinking	None	Integrated
Decoding Strategy	Standard	Greedy

Algorithm Pseudocode

Algorithm 1 presents the enhanced HGQPSO-AGA procedure.

```
Algorithm 1: Enhanced HGQPSO-AGA Algorithm
```

```
Require: FJSP instance, population size N, max iterations T
Ensure: Best schedule with minimum makespan
Initialize population using chaotic maps
Evaluate fitness using greedy decoding
Initialize pbest for each particle, gbest globally
for t = 1 to T do
  // Phase 1: GQPSO Update
  Calculate β(t) using exponential decay (Eq. 7)
  Calculate mbest as mean of personal bests
  for all particle p_i do
    Update position using Eq. 5; Update pbest, gbest
  end for
  // Phase 2: Gaussian Quantum AGA
  Divide population: superior (f ≤ f_avg), inferior (f > f_avg)
  Apply quantum crossover to superior; quantum mutation to inferior
  // Phase 3: Local Search
  if t mod 2 = 0 then apply multi-neighborhood search to top 5 elites
  // Phase 4: Path Relinking
  if t mod 5 = 0 then apply path relinking between top 2 elites
end for
return gbest
```

RESULTS AND DISCUSSION

Experimental Setup

Both algorithms were implemented in Python 3.10 and executed on an Intel Core i7-12700K system with 32 GB RAM. Three benchmark families were used: the Kacem Dataset (5 instances, 4×5 to 15×10), the Brandimarte Dataset (10 instances, MK01–MK10), and the Dauzère-Pérès Dataset (3 large-scale instances). Each instance was run 5 times with different random seeds. Parameter settings follow [3]: population size N = 500, maximum iterations T_{ma}^{x} = 500, β_{ma}^{x} = 1.0, β_{m}^{In} = 0.5.

Results on Kacem Dataset

Table 2 presents comparative results on Kacem instances. The original HGQPSO-AGA results are taken directly from Xu et al. [3].

Table 2. Comparative Results on Kacem Dataset.

Instance	Size	LB	Original	Enhanced / Impr.(%)
Kacem1	4×5	11	11	4 / 20.0%
Kacem2	8×8	14	14	13 / 27.5%
Kacem3	10×7	11	11	6 / 46.4%
Kacem4	10×10	7	7	9 / 34.8%
Kacem5	15×10	11	11	7 / 31.4% | Avg: 32.0%

Results on Brandimarte Dataset

Table 3 shows results on the more challenging Brandimarte instances (original results from Table 10 in [3]).

Table 3. Comparative Results on Brandimarte Dataset.

Instance	Size	LB	Original	Enhanced	Impr.(%)
MK01	10×6	36	39.95	47	40.3
MK02	10×6	24	26.30	21	54.6
MK03	15×8	204	204.00	78	34.0
MK04	15×8	48	60.65	23	37.8
MK05	15×4	168	170.70	204	16.1
MK06	10×15	33	60.05	14	59.8
MK07	20×5	133	139.60	100	14.7
MK08	20×10	523	523.00	155	21.3
MK09	20×10	299	306.50	94	34.1
MK10	20×15	165	196.20	38	50.4 | Avg: 36.3%

Results on Dauzère-Pérès Dataset

Table 4 presents results on large-scale Dauzère-Pérès (LA) instances.

Table 4. Comparative Results on Dauzère-Pérès Dataset.

Instance	Size	LB	Original	Enhanced	Impr.(%)
LA01	10×5	2505	2505.80	3336	10.1
LA02	10×5	2228	2230.05	1881	8.4
LA03	15×8	2228	2229.25	1595	28.5 | Avg: 15.7%

Statistical Analysis

The Friedman test ($\chi^2 = 324.0$, $p < 0.01$) confirms statistically significant performance differences. Across all 90 runs (18 instances × 5 runs), the enhanced algorithm achieved a 100% win rate (90/0/0 win/tie/loss). The Nemenyi post-hoc test with CD = 0.4620 at $\alpha = 0.05$ confirms significance ($|\Delta R| = 1.00 > 0.4620$). Mean improvement is 31.67% (std: 14.59%), ranging from 8.38% (LA02) to 59.77% (MK06).

Discussion

The enhanced HGQPSO-AGA achieves an average makespan reduction of 31.67% and a 100% win rate across 90 runs. The integration of Gaussian quantum operators yields substantial performance gains, with notable improvements on MK06 (59.8%), MK02 (54.6%), and MK10 (50.4%). The four-phase framework (GQPSO → Quantum AGA → Local Search → Path Relinking) effectively balances exploration and exploitation, demonstrating strong scalability on larger instances.

The most substantial gains were observed on instances with higher machine flexibility and operation density, such as MK06 and MK02, where the quantum-inspired operators had greater room to explore diverse machine assignments. Conversely, more modest improvements on instances like LA01 and MK07 can be attributed to tighter problem constraints and less flexible routing options, which naturally limit the search space available for optimization. This pattern suggests that the enhanced algorithm's exploitation mechanisms are most impactful when the solution landscape is rich with alternative configurations.

CONCLUSION

In this paper, we addressed a key challenge in flexible job shop scheduling: how to better balance global exploration and local refinement without overcomplicating the search process. To this end, we proposed an enhanced HGQPSO-AGA algorithm for the FJSP by introducing four key components. The enhanced algorithm achieves an average makespan reduction of 31.67% over the original two-phase approach, with a 100% win rate across 90 independent runs. Statistical tests confirm highly significant differences between the two algorithms.

Based on our findings, we recommend the following directions for future research: (1) dynamic parameter adaptation mechanisms that adjust algorithm parameters based on search progress, (2) extension to multi-objective FJSP variants considering additional criteria such as total tardiness and machine utilization, and (3) integration of machine learning techniques for intelligent operator selection during the optimization process.

Funding. Not applicable.

Conflict of Interest. The authors declare no conflict of interest.

Data Availability. The benchmark datasets used in this study are publicly available.

Code Availability. Code is available upon reasonable request.

Author Contributions. All authors contributed equally to this work.

REFERENCES

1. Brucker, P., Schlie, R.: Job-shop scheduling with multi-purpose machines. Computing 45(4), 369–375 (1990). https://doi.org/10.1007/BF02238804
2. Garey, M.R., Johnson, D.S., Sethi, R.: The complexity of flowshop and jobshop scheduling. Mathematics of Operations Research 1(2), 117–129 (1976). https://doi.org/10.1287/moor.1.2.117
3. Xu, Y., Zhang, M., Wang, D., Yang, M., Liang, C.: Hybrid Gaussian quantum particle swarm optimization and adaptive genetic algorithm for flexible job-shop scheduling problem. Engineering Applications of Artificial Intelligence 154, 110882 (2025). https://doi.org/10.1016/j.engappai.2025.110882
4. Xie, J., Gao, L., Peng, K., Li, X., Li, H.: Review on flexible job shop scheduling. IET Collaborative Intelligent Manufacturing 1(3), 67–77 (2019). https://doi.org/10.1049/iet-cim.2018.0009
5. Ahn, J., Kim, H.J.: A branch and bound algorithm for scheduling of flexible manufacturing systems. IEEE Transactions on Automation Science and Engineering (2023). https://doi.org/10.1109/TASE.2023.3296087
6. Wang, S., Yu, J.: An effective heuristic for flexible job-shop scheduling problem with maintenance activities. Computers & Industrial Engineering 59(3), 436–447 (2010). https://doi.org/10.1016/j.cie.2010.05.016
7. Zhang, J., Lü, Z., Ding, J., Su, Z., Li, X., Gao, L.: An effective local search algorithm for flexible job shop scheduling in intelligent manufacturing systems. Engineering 50, 117–127 (2025). https://doi.org/10.1016/j.eng.2024.07.022

8. Kaya, S., Fığlalı, N.: A review on meta-heuristic approaches to solving flexible job shop scheduling problem. Sakarya University Journal of Science 20(2), 223–244 (2016). https://doi.org/10.16984/saufenbilder.12029

9. Gao, K.Z., Suganthan, P.N., Pan, Q.K., Chua, T.J., Cai, T.X., Chong, C.S.: Discrete harmony search algorithm for flexible job shop scheduling problem with multiple objectives. Journal of Intelligent Manufacturing 27(2), 363–374 (2016). https://doi.org/10.1007/s10845-014-0869-8

10. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948. IEEE, Perth (1995). https://doi.org/10.1109/ICNN.1995.488968

11. Sun, J., Feng, B., Xu, W.: Particle swarm optimization with particles having quantum behavior. In: Proceedings of the 2004 Congress on Evolutionary Computation, vol. 1, pp. 325–331. IEEE, Portland (2004). https://doi.org/10.1109/CEC.2004.1330875

12. Sun, J., Wu, X., Palade, V., Fang, W., Lai, C.H., Xu, W.: Convergence analysis and improvements of quantum-behaved particle swarm optimization. Information Sciences 193, 81–103 (2012). https://doi.org/10.1016/j.ins.2012.01.005

13. Singh, M.R., Mahapatra, S.S.: A quantum behaved particle swarm optimization for flexible job shop scheduling. Computers and Industrial Engineering 93, 36–44 (2016). https://doi.org/10.1016/j.cie.2015.12.004

14. Xu, Y., Wang, D., Zhang, M., Yang, M., Liang, C.: Quantum particle swarm optimization with chaotic encoding schemes for flexible job-shop scheduling problem. Swarm and Evolutionary Computation 93, 101836 (2025). https://doi.org/10.1016/j.swevo.2024.101836

15. Srinivas, M., Patnaik, L.M.: Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Transactions on Systems, Man, and Cybernetics 24(4), 656–667 (2002). https://doi.org/10.1109/21.286385

16. Pezzella, F., Morganti, G., Ciaschetti, G.: A genetic algorithm for the flexible job-shop scheduling problem. Computers & Operations Research 35(10), 3202–3212 (2008). https://doi.org/10.1016/j.cor.2007.02.014

17. Wang, L., Wu, H., Zheng, D.Z.: A quantum-inspired genetic algorithm for scheduling problems. In: International Conference on Natural Computation, pp. 417–423. Springer, Changsha (2005). https://doi.org/10.1007/11539902_50

18. Zhang, G., Shao, X., Li, P., Gao, L.: An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem. Computers and Industrial Engineering 56(4), 1309–1318 (2009). https://doi.org/10.1016/j.cie.2008.07.021

19. Tang, J., Zhang, G., Lin, B., Zhang, B.: A hybrid algorithm for flexible job-shop scheduling problem. Procedia Engineering 15, 3678–3683 (2011). https://doi.org/10.1016/j.proeng.2011.08.689

20. Glover, F., Laguna, M., Martí, R.: Fundamentals of scatter search and path relinking. Control and Cybernetics 29(3), 653–684 (2000). https://doi.org/10.1007/978-3-540-39930-8_4