Quantum Whale Optimization Algorithm for Job Shop Scheduling Problem (Postprint)

Authors: Yan Xu, Ye Chunming, Yao Yuanyuan

Date: 2018-05-02T00:00:00+00:00

Abstract

To overcome the deficiencies of the basic Whale Optimization Algorithm (WOA) in solving job shop scheduling problems, including low convergence accuracy and susceptibility to local optima, a Quantum Whale Optimization Algorithm (QWOA) is proposed by leveraging quantum computing and optimization principles, with accompanying computational complexity analysis, global convergence proof, and simulation experiments. Simulation experiments conducted on 11 benchmark instances of job shop scheduling problems reveal that, compared with the basic Whale Optimization Algorithm (WOA), Cuckoo Search algorithm (CS), and Grey Wolf Optimizer (GWO), the QWOA algorithm achieves superior results in terms of minimum values, average values, and optimization success rates. The study demonstrates that the Quantum Whale Optimization Algorithm exhibits higher convergence accuracy and enhanced global search capability when addressing job shop scheduling problems, and is capable of escaping local optima.

Full Text

Preamble

Quantum Whale Optimization Algorithm for Solving Job-Shop Scheduling Problems

Yan Xu, Ye Chunming, Yao Yuanyuan

(Business School, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: To overcome the limitations of the basic Whale Optimization Algorithm (WOA)—namely low convergence precision and susceptibility to local optima—when solving job-shop scheduling problems, this paper proposes a Quantum Whale Optimization Algorithm (QWOA) based on quantum computation and optimization principles. The paper presents computational complexity analysis, global convergence proof, and simulation experiments for QWOA. Through simulations on 11 benchmark job-shop scheduling problem instances, QWOA demonstrates superior performance compared to the basic WOA, Cuckoo Search (CS), and Grey Wolf Optimizer (GWO) in terms of minimum values, average values, and optimization success rates. The research shows that QWOA achieves higher convergence accuracy and better global search capability while effectively escaping local optima when solving job-shop scheduling problems.

Keywords: whale optimization algorithm; quantum computing and optimization; job-shop scheduling problem; convergence proof; hybrid algorithm

---

0 Introduction

The Job-Shop Scheduling Problem (JSP) represents the most classical discrete manufacturing system scheduling challenge. As one of the most difficult combinatorial optimization problems, JSP has been proven to be NP-hard, meaning no algorithm can find its exact optimal solution in polynomial time. Consequently, algorithmic research for this problem has become a hot topic in the field of shop scheduling [?]. This paper employs the emerging Whale Optimization Algorithm to address the job-shop scheduling problem.

The Whale Optimization Algorithm (WOA), proposed by Mirjalili et al. in 2016 [?], draws its bionic inspiration from the hunting behavior of humpback whales. Current research on WOA remains in its early stages, with the algorithm gradually being applied to various domains including resource scheduling [?], location and path planning [?, ?], and industrial design [?, ?]. Algorithmic improvements primarily focus on enhancing convergence accuracy and speed while overcoming weaknesses such as susceptibility to local optima, slow convergence, and low precision. These improvements involve optimizing population initialization methods, maintaining population diversity, balancing local exploitation and global search capabilities, and ensuring feasible solution space traversal. For instance, Jangir et al. [?] proposed an Adaptive Whale Optimization Algorithm (AWOA), Ling et al. [?] introduced the LWOA algorithm, Zhong Minghui et al. [?] developed an Efficient Whale Optimization Algorithm (EWOA) with stochastically adjusted control parameters, and Liu Zhusong et al. [?] proposed the CSCWOA algorithm using sine-cosine chaotic dual strategies. However, WOA has primarily been applied to continuous optimization problems and has not yet been utilized for solving job-shop scheduling problems. Therefore, this paper draws upon quantum computation and optimization concepts to improve WOA and applies it to discrete job-shop scheduling problems.

---

1 Mathematical Description of the Job-Shop Scheduling Problem

The Job-Shop Scheduling Problem (JSP) is generally described as follows: given the processing technology for each job, the machine sequence for each job, and the processing time for each operation, arrange a processing sequence for the jobs such that n jobs to be processed on m machines satisfy certain optimization criteria. This paper uses the performance metric of "minimizing the maximum completion time" (makespan). The problem must satisfy the following assumptions and constraints:

a) Machines do not break down and begin processing jobs from time zero.

b) Each machine can process only one job at a time; each job can be processed on only one machine at a time.

c) Job processing operations strictly adhere to processing times; once started, operations cannot be interrupted, and preemption between operations is not allowed.

d) The process route for each job is fixed, meaning the machine sequence for each job is given, and the processing time for each operation in the process route is fixed. However, no precedence constraints exist between operations of different jobs.

Common mathematical model descriptions for JSP include linear programming (LP) models, integer programming (IP) models, and disjunctive graph models. This paper employs an integer programming model described as follows:

$$\begin{aligned}
\min \quad & C_{\max} && (1)\\
\text{s.t.} \quad & c_{ik} - t_{ik} + M(1 - a_{ihk}) \geq c_{ih}, \quad i = 1, 2, \dots, n; \quad h, k = 1, 2, \dots, m && (2)\\
& c_{jk} - c_{ik} + M(1 - x_{ijk}) \geq t_{jk}, \quad i, j = 1, 2, \dots, n; \quad k = 1, 2, \dots, m && (3)\\
& c_{ik} \geq 0, \quad a_{ihk}, x_{ijk} \in \{0, 1\}, \quad i = 1, 2, \dots, n; \quad h, k = 1, 2, \dots, m && (4)
\end{aligned}$$

where t_{ik} and c_{ik} represent the processing time and completion time of job i on machine k, respectively. In equation (2), $a_{ihk} = 1$ indicates that machine h processes job i before machine k; $x_{ijk} = 1$ indicates that job i is processed before job j on machine k (otherwise, both are 0). M is a large number. Equation (2) represents precedence constraints between operations, while equation (3) represents non-blocking constraints. Equation (1) is the objective function representing the minimization of the makespan performance metric [?].

2 Basic Whale Optimization Algorithm (WOA)

The WOA algorithm' s bionic principle is inspired by the unique feeding behavior of humpback whales (bubble-net feeding method). After discovering prey, humpback whales hunt by creating bubble nets along spiral-shaped paths. The algorithm models three primary behaviors: "bubble-net attacking method," "encircling prey," and "search for prey." The bubble-net attacking behavior can be further divided probabilistically into "spiral updating position" and "shrinking encircling mechanism." The mathematical description of humpback whale position updates is as follows:

2.1.1 Bubble-Net Attacking

Whales hunt in groups. During foraging, individual whales choose between two behaviors based on a random probability value: "spiral updating position" and "shrinking encircling mechanism." The mathematical model for "spiral updating position" can be described as:

$$\vec{X}(t+1) = \vec{D} \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t)$$

where $\vec{D} = |\vec{C} \cdot \vec{X}^*(t) - \vec{X}(t)|$, t represents the current iteration number, vector $\vec{X}(t)$ represents the current position of a humpback whale, vector $\vec{X}(t+1)$ represents the updated position after iteration, vector $\vec{X}^*(t)$ represents the best foraging position in the whale group, parameter b is a constant defining the shape of the logarithmic spiral, and parameter l is a random number in $[-1, 1]$ that together control the spiral position update pattern of whale individuals. When $l = -1$, the whale individual is closest to the foraging target; when $l = 1$, it is farthest.

During the bubble-net attacking process, the probability of a whale individual selecting a position update method is set to p^*. The complete bubble-net attacking method can be described as:

$$\vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - \vec{A} \cdot \vec{D}, & p < p^* \\ \vec{D} \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t), & p \geq p^* \end{cases}$$

2.1.2 Encircling Prey

As described in Section 2.1.1, when $p < p^*$ (with $p^* = 0.5$), whale individuals update their positions in space, moving toward better positions to achieve shrinking encirclement of prey. The mathematical model is described as:

$$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A} \cdot \vec{D}$$

where $\vec{D} = |\vec{C} \cdot \vec{X}^*(t) - \vec{X}(t)|$, $\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a}$, $\vec{C} = 2\vec{r}_2$, $\vec{a}$ linearly decreases from 2 to 0 with iteration number, and $\vec{r}_1$ and $\vec{r}_2$ are random vectors in $(0, 1)$.

Coefficient vectors $\vec{A}$ and $\vec{C}$ control the movement patterns of whale individuals. It should be noted that this position update occurs when $|\vec{A}| < 1$.

In Section 2.1.2, the position update in equation (11) assumes $|\vec{A}| < 1$. When $|\vec{A}| \geq 1$, the following random position update method is adopted, called "search for prey" :

$$\vec{X}(t+1) = \vec{X}_{\text{rand}} - \vec{A} \cdot \vec{D}_{\text{rand}}$$

where $\vec{D}_{\text{rand}} = |\vec{C} \cdot \vec{X}_{\text{rand}} - \vec{X}(t)|$ and $\vec{X}_{\text{rand}}$ represents a randomly selected whale individual from the whale group. Equations (15) and (16) indicate that when $|\vec{A}| \geq 1$, whale individuals randomly move toward other individuals.

2.2.1 Quantum Computation and Quantum Optimization

Quantum computation (QC), first proposed by Feynman in 1982, has become a frontier discipline attracting scholars worldwide. In quantum computing, the basic unit of information storage is called a quantum bit (qubit). Unlike classical bits, qubits can continuously and randomly exist in any superposition of two polarization states. Qubits can be categorized as single, double, or multiple qubits. The superposition state of a single qubit can be expressed as:

$$|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad |\alpha|^2 + |\beta|^2 = 1$$

where $|0\rangle$ and $|1\rangle$ are Dirac notations representing two basic states of microscopic particles in quantum computing, and α and β are probability amplitudes—a pair of complex numbers.

Quantum logic gates are quantum devices that perform logical transformations on quantum states and form the foundation of quantum computing. Commonly used quantum gates include the Hadamard gate, Pauli-X gate, Pauli-Y gate, Pauli-Z gate, phase gate, $\pi/8$ gate, and quantum rotation gate. The quantum rotation gate can be represented as:

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

The quantum state update process can be expressed as:

$$\begin{bmatrix} \alpha' \\ \beta' \end{bmatrix} = R(\theta) \cdot \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

where θ is the rotation angle and α', β' are the updated probability amplitudes [**?**].

2.2.2 Quantum Whale Optimization Algorithm

Due to the basic WOA's limitations of low convergence precision and susceptibility to local optima, this paper employs quantum computation and optimization concepts to improve WOA. By incorporating quantum rotation gate operations into the basic algorithm, the position update mechanism is enhanced to improve population diversity and convergence precision. Let $\vec{X}(t)$ be the position of a whale in the search space, and let $Q\vec{X}(t)$ be the position after quantum rotation gate operation.

From equation (18), we know $|\beta|^2 = 1 - |\alpha|^2$, so the quantum rotation gate update process can be simplified as:

$$Q\vec{X}(t) = 2\cos\theta \cdot \vec{X}(t) - 1 - \sin\theta$$

Compared with equation (20), the quantum rotation gate operation in equation (21) is called single-chain encoded quantum rotation gate operation [**?**]. After a whale individual undergoes quantum rotation gate updating, the quality of its corresponding solution is compared with that before rotation, and the better one is selected. The Quantum Whale Optimization Algorithm is abbreviated as QWOA, with specific steps as follows:

a) Set iteration count, population size, and problem parameters.

b) Generate the initial population using random initialization.

c) Calculate fitness values of whale individuals and identify the current global optimal solution position.

d) Update variables in each dimension of whale individuals based on values of p and $|\vec{A}|$.

e) Perform quantum rotation gate operations on variables in each dimension of whale individuals.

f) Evaluate the update effects from steps d) and e), select the better result, and retain it.

g) Repeat steps c)–f) for the whale population within the maximum iteration count.

h) Output the solution results.

The complete flow of QWOA is described in [Figure 1: see original paper].

---

3 Computational Complexity Analysis and Convergence Proof

3.1 Computational Complexity Analysis

Frequency estimation is used to analyze the time and space complexity of QWOA. Let the maximum iteration count be T, population size be P, and variable dimension be D. From Section 2.2.2, Steps 4–6 constitute the core operations within the innermost loop, with execution frequencies of $T \cdot P \cdot O(D)$. Therefore, the asymptotic time complexity of the algorithm is:

$$O(T \cdot P \cdot D)$$

In the algorithm, variable storage space S is influenced by population size P and variable dimension D, giving it a space complexity of:

$$O(P \cdot D)$$

3.2 Algorithm Convergence Proof [?]

Probability measure theory is employed to prove the global convergence of QWOA. According to global convergence criteria and theorems, an algorithm must satisfy two assumptions to be proven globally convergent:

Assumption 1: $f(D_z) \leq f(\xi)$, and if $S \in \mathbb{R}^n$ is a Borel subset such that $\nu[S] > 0$, then:

$$\prod_{k=0}^{\infty}(1 - \mu_k[S]) = 0$$

where f is the objective function for minimization, D_z is a point in solution space S that minimizes the function or produces an acceptable infimum of function values on S, ξ is a point in S, and μ_k is the probability measure generated by the algorithm on the sample space. This assumption ensures correct algorithm operation so that the solution sequence converges to the function' s infimum on S.

Assumption 2: For any Borel subset A of S:

$$\mu_k[A] \geq \nu[A]$$

where ν is the probability measure generated by distribution D on S. This assumption ensures that, with probability measure greater than 0, the algorithm will not miss any Borel subset A of solution space S after infinite iterations.

The global convergence of QWOA is proven according to these two assumptions.

Lemma 1: QWOA satisfies Assumption 1.

Proof: According to Section 2.2.2, function f in QWOA can be defined as:

$$f(\vec{X}_{i,t+1}) = \begin{cases} g(\vec{X}_{i,t}), & f(g(\vec{X}_{i,t})) \geq f(\vec{X}_{i,t}) \\ \vec{X}_{i,t}, & f(g(\vec{X}_{i,t})) < f(\vec{X}_{i,t}) \end{cases}$$

where $g(\vec{X}_{i,t})$ represents the quantum rotation operation function applied to the current global optimal position, and $\vec{X}_{i,t}$ is the position of whale individual i after t updates. By algorithm definition, the fitness value corresponding to f is monotonically non-increasing and gradually converges to the infimum of the solution space. Therefore, Lemma 1 holds.

Lemma 2: QWOA satisfies Assumption 2.

Proof: The quantum rotation gate operation in QWOA can be expressed as:

$$g(\vec{X}_{i,j,t}) = \cos\theta \cdot \vec{X}_{i,j,t} - \sin\theta$$

where $\vec{X}_{i,j,t}$ represents the j-th dimensional variable of whale individual i at iteration t. Let $\theta = \arccos(\sigma) - \phi$, where $\sigma = \vec{X}_{i,j,t}$ and $\phi \in [0, 2\pi]$ is a random angle. Then $\theta \in [\arccos(\vec{X}_{i,j,t}) - 2\pi, \arccos(\vec{X}_{i,j,t})]$. Since $\vec{X}_{i,j,t} \in [0, 1]$, the range of θ is $[-2\pi, \pi/2]$.

Define $\mu_{i,t}$ as the uniform distribution corresponding to the j-th dimensional variable of whale individual i. From the above, for any Borel subset A of S, when $\theta \in [-2\pi, 3\pi/2]$, the probability of $g(\vec{X}_{i,j,t}) \in A$ is greater than 0. The probability measure generated by $\mu_{i,t}$ on S is:

$$\mu_{i,t}[A] = \int_A \frac{1}{3\pi/2 - (-2\pi)}\, d\theta = \frac{2}{7\pi} \int_A d\theta$$

From equation (26), we obtain:

$$\mu_k[A] = \prod_{i=1}^{P} \prod_{j=1}^{D} \mu_{i,t}[A] = \left(\frac{2}{7\pi}\right)^{P \cdot D} \int_A \cdots \int_A d\theta_1 \cdots d\theta_{P \cdot D}$$

Since $P \cdot D$ is finite and $\theta \in [-2\pi, 3\pi/2]$, we have $\mu_k[A] > 0$. Therefore, Lemma 2 holds.

Thus, based on Lemma 1 and Lemma 2, QWOA satisfies both Assumption 1 and Assumption 2, proving it is a globally convergent algorithm.

4 Simulation Experiments and Analysis

To evaluate and verify the performance of WOA in solving job-shop scheduling problems, this paper selects 11 benchmark problems from the standard JSP test library for solution and comparison with other algorithms from the literature [?]. The computational environment is Windows 10 Professional 32-bit operating system, Intel(R) Core(TM) i7-2600 CPU @ 3.40GHz processor, and 4GB RAM. Algorithm implementation uses Matlab 2015b.

4.1 Comparison of WOA with GWO and CS

The Grey Wolf Optimizer (GWO) [?], proposed by Mirjalili et al. in 2014, and the Cuckoo Search (CS) algorithm [?], proposed by Yang et al. in 2009, are both emerging intelligent algorithms. To ensure accurate comparison with GWO and CS on JSP problems, WOA parameters are set consistent with [?]: population size of 30, maximum iterations of 500, and 30 independent runs. The results are shown in .

The results demonstrate that under identical population size, iteration count, and algorithm runs, WOA successfully converges to currently known optimal solutions for FT06, LA01, LA06, LA11, and other problems (partial Gantt charts shown in [Figure 2: see original paper]). However, WOA' s optimization success rate is inferior to GWO and CS. In terms of minimum values, WOA outperforms CS but underperforms GWO. For average values, WOA is generally worse than GWO and only slightly better than CS in some cases. These results indicate that while WOA can be successfully applied to job-shop scheduling problems, it shows no clear advantage over GWO and CS, leaving significant room for improvement. Therefore, this paper employs the improved QWOA algorithm to solve these problems.

4.2 Comparison of QWOA with WOA

QWOA is applied to solve the aforementioned problems with population size set to 30, maximum iterations to 500, and quantum rotation angle set to rand $\times 2\pi$. The comparison results are presented in .

The calculation methods for minimum value improvement, average value improvement, and success rate improvement in are respectively: (WOA minimum - QWOA minimum) / WOA minimum, (WOA average - QWOA average) / WOA average, and QWOA success rate - WOA success rate.

The comparison between QWOA and WOA for solving JSP shows significant improvements in optimization success rate. For FT06, LA01, LA06, and LA11 problems, QWOA' s success rates reach 93.3%, 100%, and 100% respectively, with an average improvement of 59.2% over WOA. Notably, QWOA converges to known optimal solutions for the LA31 problem that WOA, GWO, and CS cannot achieve, and nearly reaches known optimal solutions for the LA16 problem.

QWOA also substantially improves minimum and average values, outperforming WOA, GWO, and CS with average improvement rates of 10.5% for minimum values and 9.7% for average values. [Figure 3: see original paper] compares the convergence curves of QWOA with WOA, CS, and GWO, showing that QWOA requires fewer iterations to converge to optimal values, achieves higher convergence precision, and possesses stronger capability to escape local optima for global search.

Considering the variance in optimization performance (not shown due to space limitations), the overall results demonstrate that the quantum computation and optimization-enhanced WOA exhibits higher convergence accuracy, fewer convergence iterations, and stronger global search capability when solving job-shop scheduling optimization problems.

---

5 Conclusion

The Whale Optimization Algorithm (WOA) is an emerging swarm intelligence algorithm primarily applied to continuous optimization problems, with limited application to discrete combinatorial optimization problems like job-shop scheduling (JSP). This paper validates the feasibility of WOA for JSP and identifies its limitations in this domain. To overcome these deficiencies, a Quantum Whale Optimization Algorithm (QWOA) is proposed, incorporating computational complexity analysis, convergence proof, and experimental simulation.

Comparative analysis reveals that the proposed QWOA algorithm demonstrates superior performance in solving JSP problems, characterized by fewer iterations, higher convergence precision, and stronger global search capability. Future work will extend WOA to multi-objective combinatorial optimization problems and further enhance its application performance.

References

[1] Cheng T C E, Peng B, Lü Z. A hybrid evolutionary algorithm to solve the job shop scheduling problem [J]. Annals of Operations Research, 2016, 242(2): 223-237.
[2] Mirjalili S, Lewis A. The whale optimization algorithm [J]. Advances in Engineering Software, 2016, 95: 51-67.
[3] Trivedi I N, Bhoye M, Bhesdadiya R H, et al. An emission constraint environment dispatch problem solution with microgrid using whale optimization algorithm [C]// Proc of Power Systems Conference. 2017.
[4] Prakash D B, Lakshminarayana C. Optimal siting of capacitors in radial distribution network using whale optimization algorithm [J]. Alexandria Engineering Journal, 2016.
[5] Dao T K, Pan T S, Pan J S. A multi-objective optimal mobile robot

path planning based on whale optimization algorithm [C]// Proc of IEEE International Conference on Signal Processing. 2017: 337-342.
[6] Kaveh A, Ghazaan M I. Enhanced whale optimization algorithm for sizing optimization of skeletal structures [J]. Mechanics Based Design of Structures & Machines, 2017, 45: 345-362.
[7] Oliva D, Aziz M A E, Hassanien A E. Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm [J]. Applied Energy, 2017, 200: 141-154.
[8] Jangir P, Trivedi I N, Jangir N, et al. A novel adaptive whale optimization algorithm for global optimization [J]. Indian Journal of Science & Technology, 2016, 9(38).
[9] Ling Y, Zhou Y, Luo Q. Lévy Flight trajectory-based whale optimization algorithm for global optimization [J]. IEEE Access, 2017, 5(99): 6168-6186.
[10] 钟明辉，龙文．一种随机调整控制参数的鲸鱼优化算法 [J]．科学技术与工程, 2017, 17(12): 68-73.
[11] 刘竹松，李生．正余混沌双弦鲸鱼优化算法 [J]．计算机工程与应用, 2017.
[12] Cheng T C E, Peng B, Lü Z. A hybrid evolutionary algorithm to solve the job shop scheduling problem [J]. Annals of Operations Research, 2016, 242(2): 223-237.
[13] Dey S, Saha I, Bhattacharyya S, et al. Multi-level thresholding using quantum inspired meta-heuristics [J]. Knowledge-Based Systems, 2014, 67(3): 373-400.
[14] 高洪元，刁鸣．量子群智能及其在通信技术中的应用 [M]．北京：电子工业出版社, 2016.
[15] Fang W, Sun J, Chen H, et al. A decentralized quantum-inspired particle swarm optimization algorithm with cellular structured population [J]. Information Sciences, 2016, 330(C): 19-48.
[16] 姚远远，叶春明．求解作业车间调度问题的改进混合灰狼优化算法 [J/OL]．计算机应用研究, 2018, 35(5) (2017-06-14). http://kns.cnki.net/kcms/detail/51.1196.TP.20170614.1317.050.html.
[17] Mirjalili S. How effective is the Grey Wolf optimizer in training multi-layer perceptrons [J]. Applied Intelligence, 2015, 43(1): 150-161.
[18] Li X, Yin M. Modified cuckoo search algorithm with self adaptive parameter method [J]. Information Sciences, 2015, 298(C): 80-97.

Note: Figure translations are in progress. See original paper for figures.