Hindawi
Journal of Sensors
Volume 2021, Article ID 5511745, 13 pages
https://doi.org/10.1155/2021/5511745

Research Article

Sensor Duty Cycle for Prolonging Network Lifetime Using Quantum Clone Grey Wolf Optimization Algorithm in Industrial Wireless Sensor Networks

Yang Liu, Jing Xiao, Chaoqun Li, Hu Qin, and Jie Zhou

College of Information Science and Technology, Shihezi University, Shihezi 832000, China

Correspondence should be addressed to Jie Zhou; jiezhou@shzu.edu.cn

Received 3 February 2021; Revised 19 February 2021; Accepted 11 March 2021; Published 27 March 2021

Academic Editor: Bin Gao

The application of industrial wireless sensor networks (IWSNs) frequently appears in modern industry, and it is usually to deploy a large quantity of sensor nodes in the monitoring area. This way of deployment improves the robustness of the IWSNs but introduces many redundant nodes, thereby increasing unnecessary overhead. The purpose of this paper is to increase the lifetime of IWSNs without changing the physical facilities and ensuring the coverage of sensors as much as possible. Therefore, we propose a quantum clone grey wolf optimization (QCGWO) algorithm, design a sensor duty cycle model (SDCM) based on real factory conditions, and use the QCGWO to optimize the SDCM. Specifically, QCGWO combines the concept of quantum computing and the clone operation for avoiding the algorithm from falling into a local optimum. Subsequently, we compare the proposed algorithm with the genetic algorithm (GA) and simulated annealing (SA) algorithm. The experimental results suggest that the lifetime of the IWSNs based on QCGWO is longer than that of GA and SA, and the convergence speed of QCGWO is also faster than that of GA and SA. In comparison with the traditional IWSN working mode, our model and algorithm can effectively prolong the lifetime of IWSNs, thus greatly reducing the maintenance cost without replacing sensor nodes in actual industrial production.

1. Introduction

As industrial wireless sensor networks (IWSNs) have more and more applications in the factories, the way to prolong the lifetime of IWSNs without changing the physical facilities has become a hot issue [1, 2]. The content of this paper includes the design of a sensor duty cycle model (SDCM) for IWSN modeling and a novel group intelligence algorithm based on grey wolf optimization for optimizing the SDCM. By using the SDCM, we can conveniently increase the lifetime of the IWSNs in the factory, thus avoiding the very time-consuming, labor-intensive, and sometimes impossible operations in real life [3]. In addition, the use of an artificial intelligence algorithm to optimize the established model can effectively prolong the lifetime of the IWSNs, thereby reducing the maintenance cost of the IWSNs and increasing the benefit of the factory [4, 5]. Furthermore, making full use of existing devices can reduce the generation of discarded equipment and protect the environment on the basis of reducing resources.

In this research, we investigate the IWSNs frequently used in factories, such as chemical sensors that monitor the content of harmful gases, pressure sensors in industrial production, and ultrasonic sensors in the field of industrial automation. We find that these IWSNs are basically placed by using the traditional wide-spreading method and then periodically control some sensors to enter the sleep state for saving energy [6]. This approach has two disadvantages, one is that it cannot meet the requirements of full coverage, another is it cannot minimize the energy consumption of the sensors.

The goals of this article are to increase the lifetime of IWSNs and reduce the cost of factory replacement of sensor devices. The above goals are motivated by the actual needs in the factory [7, 8]. In general, the purpose of using wireless sensors in factories is to monitor the production environment for ensuring safe production conditions. However, to achieve full coverage of the monitoring in the production workshop, the factory has to place redundant sensors to ensure the performance of the IWSNs, which causes a lot of waste of sensor energy and speeds up the replacement of sensors [9]. In this case, it is necessary to propose a method that can effectively utilize redundant sensors for reducing the number of sensor replacements.

In the issue of improving the lifetime of IWSNs, it is necessary to ensure high coverage of targets first and then perform a sensor node duty cycle [10–12]. When establishing the SDCM, we comprehensively consider the sensor's monitoring range and working time, then give a mathematically measurable lifetime of the IWSNs. Therefore, we can use an artificial intelligence algorithm to optimize the lifetime of IWSNs through the SDCM. The designed model has been verified by a series of simulation experiments. For a given set of IWSN data, it can be input into the SDCM by the sensor's ability of working time and coverage, then use our proposed algorithm to optimize the SDCM for a longer lifetime of the IWSNs.

The innovations of this research are as follows: (i) The sleep mode of industrial wireless sensor nodes is modeled, and the sensor duty cycle model is proposed. Through the SDCM, the lifetime of IWSNs can be prolonged by using an intelligent algorithm, thereby effectively reducing the factory's maintenance costs for IWSNs. (ii) A novel GWO-based intelligent algorithm is proposed, which uses the quantum probability amplitude in quantum computing and the clone concept in biology to avoid falling into local optimum, thereby increasing the usability of the algorithm. In addition, the performance of the proposed algorithm has been compared with GA and SA.

The paper's structure can be expressed as follows. Relative researches on the duty cycle of sensors are given in Section 2. Subsequently, Section 3 shows the evaluation method of IWSN lifetime and the establishment of the SDCM. In Section 4, in order to obtain the optimal IWSN lifetime, we introduce a novel group intelligent algorithm based on grey wolf optimization. Section 5 presents the performance of the proposed model and algorithm through simulation experiments and makes discussion. Finally, in Section 6, the conclusion part is given.

2. Related Work

The current research on the duty cycle of IWSNs can be divided into three types. The first and most used one is to design a routing protocol for reducing the unnecessary communication overhead of sensors; the second type uses artificial intelligence methods to process sensor data for obtaining a suitable mode of duty cycle; the third type applies mathematical approaches to model IWSNs, then optimizes the duty cycle of sensor nodes.

Firstly, a proper routing protocol can reduce communications of sensors in IWSNs. In [13], the authors use a threefold method to improve the lifetime of the IWSNs by adjusting the duty cycle process of sensor nodes, then switch between the active mode and the sleep mode according to the trust value obtained by the nodes. On the other hand, to better improve the service quality of IWSNs, the paper [14] proposes an AODV routing protocol for surplus energy, which realizes the reduction of energy consumption of IWSNs through cross-layer design. Similar to the paper [18], another cross-layer routing method is also proposed. In [15], the authors adjust the wake-up and sleep of nodes in the forwarding stage through the cooperation of routing and MAC layer. Then, the paper [16] proposes a routing protocol for anycast. Each sensor node decides how to transmit data based on its local information and dynamically changes the node's duty cycle status. What is more, for the purpose of solving the problem of excessive energy consumption of the nodes around the sink node in the IWSNs, the paper [17] proposes a method based on the path optimization of the sink node and establishes a corresponding energy consumption model. From other perspectives, the paper [18] uses multihop communication to reduce the long-distance communication overhead of nodes, proposes a routing protocol for clustering nodes, and uses a multihop simulated annealing algorithm to select intermediate nodes. In [19], the authors propose a perceptual routing protocol including network scheduling and task cycle, which helps sensor nodes to continuously monitor. However, the above methods for improving the duty cycle model by using routing protocol do not consider the way of placement for wireless sensors in the real environment. They only reduce the energy consumption of each sensor node but ignore the premise that there is a large quantity of redundant nodes in the IWSNs.

Secondly, artificial intelligence technology can also effectively improve the duty cycle of sensor nodes. In [20], the authors use reinforcement learning to maximize the sensing quality of the sensor nodes, then perform duty cycle based on the available energy, and use the collected energy to make the nodes continuously adapt to the changing environment. With the same purpose of prolonging lifetime of the IWSNs, the paper [21] expresses the position distribution of sensors as an optimization problem and then proposes a cuckoo algorithm to solve the problem, thereby obtaining the optimal position of sensor nodes. In [22], the authors use Q-learning technology and linear regression function to design a MAC protocol. The protocol takes the relationship between load conditions and performance into account and makes up for the disadvantage of Q-learning, and it can do low-latency sensor scheduling. Although these methods can improve the lifetime of the IWSNs, they do not consider the convergence speed of the algorithm, and they also fail to make good use of redundant nodes in the IWSNs.

Finally, there are some researches to optimize sensor duty cycle from other aspects. In [23], the authors consider the asymmetry of the asynchronous duty cycle, obtain the upper and lower limits of the node's duty cycle, and use block design to establish the duty cycle model. The paper [21] uses empirical data to find the noise relationship between the maximum discovery time and the duty cycle by analyzing the error of the proposed model, and it concludes that the same duty cycle value in an asymmetric scenario can achieve low latency. However, it only considers how to use a node duty cycle to obtain low-latency information transmission and does not pay attention to prolonging the lifetime of the IWSNs.

The previous researches only solve the problem of reducing the communication overhead of each sensor node, thereby achieving the purpose of increasing the lifetime of the IWSNs. They do not make good use of the large quantity of redundant nodes and do not consider solving the problem from the entire network. We can go a step further on the basis of the previous work. On the premise of meeting the requirements of industrial production, we use duty cycle to shut down redundant nodes and reduce the communication overhead of each node, hence the maximum of the lifetime in the IWSNs.

In this paper, we start from the entire wireless sensor network and model the real factory IWSNs. Particularly, our model not only considers the coverage capability of the sensor nodes but also makes the lifetime of the sensor nodes measurable, which is more convenient for subsequent optimization. To reduce sensor communications, we design a novel heuristic optimization algorithm, which can effectively improve the convergence performance and avoid falling into the local optimum, so that the lifetime of the IWSNs can be prolonged.

3. System Model

3.1. Problem Description. In the real factory scene, the target coverage of IWSNs can be divided into two types. One is full coverage, which means every target being covered and monitored by at least one sensor node at every moment. The other is to improve coverage rate, which is often used in environments where full coverage is impossible. In most cases, the placement of factory sensors is redundant, which aims to decrease the occurrence of accidents. Redundant placement often meets the requirement of full coverage. Under the premise of redundant nodes, we can perform duty cycle operation on sensor nodes for saving energy, thereby prolonging the lifetime of the IWSNs.

In this paper, with the aim of facilitating the modeling of the sensor node duty cycle problem from a mathematical perspective, we propose a concept of measurable sensor lifetime, which can be explained as follows: general industrial sensors have their service lifetime, different types of sensors have different values, and even the same type of sensors will have different lifetime values. However, to make the sensors produced by factories more competitive in the market, manufacturers often give the theoretical lifetime of the sensors. Working under more severe conditions than usual, the theoretical lifetime can be obtained by converting the working lifetime with a certain calculation formula. Therefore, with the theoretical lifetime of the sensors, we can divide the lifetime and mathematically model it through the operation of the duty cycle.

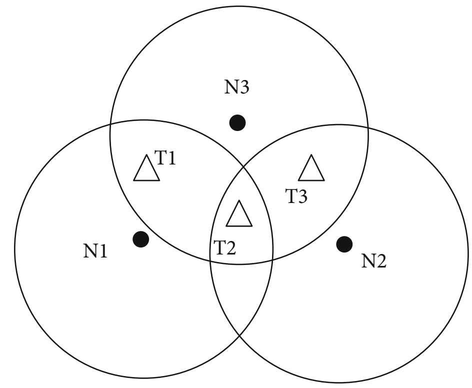

Figure 1: Coverage relationship between sensors and targets.

To better understand the sensor duty cycle model (SDCM), we use an example to illustrate it. In Figure 1, three targets are monitored by three sensor nodes with circular monitoring radii, and the monitor requirement is full coverage. The sensor nodes **N**1, **N**2, and **N**3 are, respectively, represented by points located at the center of the circle, the monitored targets **T**1, **T**2, and **T**3 are symbolized by triangles, and the circle stands for the monitoring radius of the sensor node.

According to Figure 1, we can know that **N**1 covers **T**1 and **T**2, **N**2 covers **T**2 and **T**3, and **N**3 covers **T**1, **T**2, and **T** 3. Assume that the lifetime of each sensor node can be divided into two rounds. If the duty cycle operation is not performed, the total working time of the entire IWSNs is two rounds, which are as follows: the first round, we turn on **N**1, **N**2, and **N**3, and the second round, we also make **N** 1, **N**2, and **N**3 in active mode. However, under the premise of ensuring full coverage, if we divide the sensors into different coverage sets and only turn on one coverage set in each round, we can prolong the lifetime of the IWSNs through duty cycle operation. Specifically, in the above case, **N**1 and **N**2 can form a coverage set, and **N**3 can be another one. In the first and second rounds, we can only switch on **N**1 and **N**2 and turn off **N**3 for saving energy. At the end of the second round, the energy of **N**1 and **N**2 is exhausted; in the third and fourth rounds, we turn on the **N**3 node. At the end of the fourth round, the energy of the three sensor nodes has been exhausted. Obviously, the lifetime of the IWSNs has increased from the previous two rounds to four rounds through the duty cycle operation while ensuring the full coverage of the targets.

In modern IWSNs, with the increase of redundant nodes and the improvement of sensor coverage, the duty cycle operation of sensor nodes plays a more and more important role. Subsequently, we establish a mathematical model for using an artificial intelligence algorithm to optimize the SDCM.

3.2. Sensor Duty Cycle Model. Suppose there are **X** sensor nodes and **K** monitoring targets in the IWSNs. In the SDCM, in order to ensure that the sensors complete full coverage of the targets, we first create a matrix **S** for expressing the coverage relationship between sensors and targets, which can be shown as

$$S = \begin{bmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,K-1} & s_{1,K} \\ s_{2,1} & s_{2,2} & \cdots & s_{2,K-1} & s_{2,K} \\ \vdots & & s_{x,k} & & \vdots \\ s_{X-1,1} & s_{X-1,2} & \cdots & s_{X-1,K-1} & s_{X-1,K} \\ s_{X,1} & s_{X,2} & \cdots & s_{X,K-1} & s_{X,K} \end{bmatrix} (s_{x,k} \in \{0, 1\}), \quad (1)$$

where $s_{x,k}$ represents the monitoring relationship between the x_{th} sensor and the k_{th} target. $s_{x,k} = 0$ means that the sensor cannot detect the target node, and $s_{x,k} = 1$ means that the target node is within the monitoring range of the sensor.

However, a coverage relationship matrix is not enough for establishing the SDCM, it is also necessary to create a duty cycle sequence matrix. Combining the previously described concept of measurable lifetime, we divide the theoretical lifetime of each sensor into **N** rounds. It is not difficult to conclude that the maximum life of the entire wireless sensor network is **XN** rounds, where **X** is the number of sensors. The duty cycle sequence matrix can be expressed as

$$T = \begin{bmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,X-1} & t_{1,X} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,X-1} & t_{2,X} \\ \vdots & & t_{i,x} & & \vdots \\ t_{XN-1,1} & t_{XN-1,2} & \cdots & t_{XN-1,X-1} & t_{XN-1,X} \\ t_{XN,1} & t_{XN,2} & \cdots & t_{XN,X-1} & t_{XN,X} \end{bmatrix} (t_{i,x} \in \{0, 1\}), \quad (2)$$

where $t_{i,x} = 1$ indicates that, in the i_{th} round, the x_{th} sensor node is in the active state, and $t_{i,x} = 0$ denotes that the x_{th} sensor node is in the sleep state in the i_{th} round.

To obtain the monitoring matrix between the x_{th} sensor node and the k_{th} monitored target in each round, we need to multiply the matrix **T** and the matrix **C**. The reason for this approach is that **T** is a duty cycle sequence matrix with **XN** rows and **X** columns, and **S** is a sensor coverage matrix with **X** rows and **K** columns. To obtain the monitoring relationship matrix **TS** between the sensor node and the monitored target, it is necessary to multiply the **T** matrix by the **S** matrix to obtain a matrix of **XN** rows and **K** columns. The rows of the **TS** represent the monitoring relationship of the sensor to the target in the round. If the elements in the i_{th} row are all 1, $i \in (1, XN)$, it means that the sensor network has completed full coverage in the i_{th} round and reached the set goal. If there is 0 in the i_{th} row, the task fails in the i_{th} round. The monitoring matrix can be shown as

$$TS = \begin{bmatrix} \sum_{x=1}^{X} t_{1,x}s_{x,1} & \sum_{x=1}^{X} t_{1,x}s_{x,2} & \cdots & \sum_{x=1}^{X} t_{1,x}s_{x,K-1} & \sum_{x=1}^{X} t_{1,x}s_{x,K} \\ \sum_{x=1}^{X} t_{2,x}s_{x,1} & \sum_{x=1}^{X} t_{2,x}s_{x,2} & \cdots & \sum_{x=1}^{X} t_{2,x}s_{x,K-1} & \sum_{x=1}^{X} t_{2,x}s_{x,K} \\ \vdots & & \sum_{x=1}^{X} t_{i,x}s_{x,k} & & \vdots \\ \sum_{x=1}^{X} t_{XN-1,x}s_{x,1} & \sum_{x=1}^{X} t_{XN-1,x}s_{x,2} & \cdots & \sum_{x=1}^{X} t_{XN-1,x}s_{x,K-1} & \sum_{x=1}^{X} t_{XN-1,x}s_{x,K} \\ \sum_{x=1}^{X} t_{XN,x}s_{x,1} & \sum_{x=1}^{X} t_{XN,x}s_{x,2} & \cdots & \sum_{x=1}^{X} t_{XN,x}s_{x,K-1} & \sum_{x=1}^{X} t_{XN,x}s_{x,K} \end{bmatrix}. \quad (3)$$

In equation (3), $\sum_{x=1}^{X} t_{i,x}s_{x,k} = 0$ means that the k_{th} monitored target in the i_{th} round is not monitored by any sensor, and $\sum_{x=1}^{X} t_{i,x}s_{x,k} = p(p > 0)$ represents that the k_{th} monitored target in the i_{th} round is monitored by **p** sensors. The requirement of full coverage shown in the matrix **TS** is that the elements in a row are all positive numbers.

To calculate the lifetime of the IWSNs, we define the fitness function **first-zero** to represent the number of rows where the first element 0 appears in the matrix **TS** and define the restriction condition. Therefore, the SDCM can be expressed as a fitness function (4) and restriction condition (5).

$$f(T) = \text{first\_zero}(TS) - 1, \tag{4}$$

$$\sum_{x=1}^{XN} t_{i,x} \leq N, \quad x = 1 \cdots X, \tag{5}$$

where **T** represents the duty cycle sequence matrix, and **N** denotes the lifetime cycle number of each sensor. The reason of the fitness function (4) is because under the requirement of sensor node full coverage, only the elements in the row of matrix **TS** are all 1, which means that the task is completed. Since the coverage cannot appear gaps, the lifetime value of IWSNs is the number of rows where the first zero element appears minus 1. Equation (5) shows the working lifetime of each sensor does not exceed **N** rounds.

Subsequently, due to the complexity of the sensor duty cycle increases exponentially with the number of sensor nodes and working lifetime, we decide to use a novel heuristic algorithm to solve this problem.

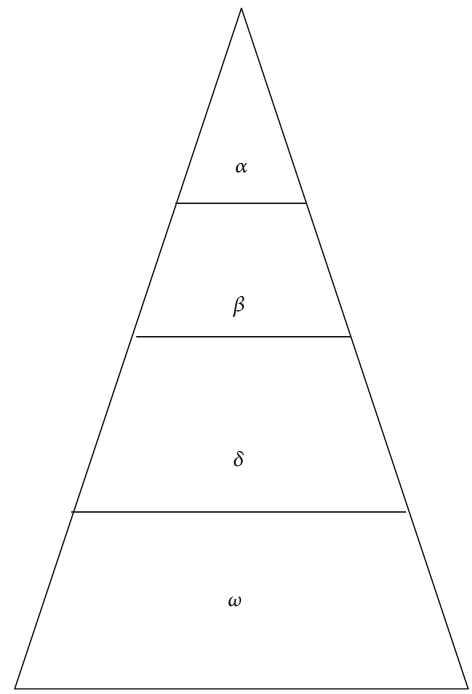

Figure 2: Diagram of wolf population level.

4. QCGWO-Based Duty Cycle in IWSNs to Maximize Network Lifetime

In IWSNs, obtaining the longest network lifetime of sensors in the duty cycle problem is obviously an NP-difficult problem. For the purpose of gaining the optimal solution of the sensor duty cycle, we design a heuristic optimization algorithm based on the gray wolf optimization (GWO) algorithm. The GWO is a group intelligent optimization algorithm proposed in 2014, and it has the characteristics of simple operation and few parameters. However, the traditional GWO is prone to falling into the local optimum, and its global search ability is difficult to control. To overcome these disadvantages, we combine the concept of quantum computing and clone with GWO, then propose a quantum clone gray wolf optimization (QCGWO) algorithm.

The description of the process for QCGWO are problem coding, population initialization, fitness calculation and class division, update wolf population and algorithm parameters, quantum probability amplitude and quantum revolving gate, clone expansion, and termination operation.

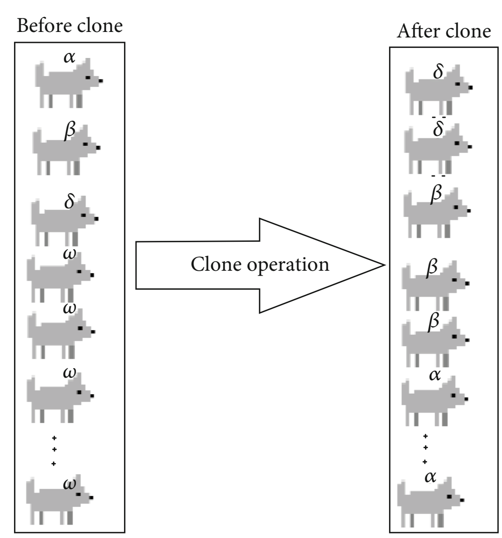

Figure 3: Clone operation.

4.1. Problem Coding. In the sensor duty cycle problem, the key is to control the sensor to turn on at an appropriate time, so that redundant nodes can be fully utilized and the energy consumption of the sensor network can be reduced. Assuming that different sensors in IWSNs can be divided into the same number of lifetime rounds, since the opening and the closing are a group of Boolean values, we decide to use binary coding in the duty cycle problem. Zero means that the sensor is turned off in this round; otherwise, 1 means that the sensor is turned on in this round. There are two core matrices in the above-mentioned sensor duty cycle problem, one is the sensor coverage matrix **S**, and the other is the sensor duty cycle sequence matrix **T**. The coverage matrix **S** is generated only once in the algorithm, and subsequent duty cycle operations are based on it. Each wolf in QCGWO carries a sequence matrix **T**, and the matrix **T** is optimized through the proposed algorithm until the algorithm ends and an optimal solution is obtained. To enhance the understanding of the sequence matrix **T**, we gave an example for illustrating. Suppose that the working lifetime of each sensor is 2 rounds and there are 2 sensors in IWSNs, then the rows of the matrix **T** are $2 \times 2 = 4$, and the columns are 2. The encoding of matrix **T** can be expressed as

$$T = \begin{bmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{6}$$

In (6), the sum of 1 in each column is 2, which means that the lifetime of the sensor is 2 rounds. The first column indicates that the first sensor turns on in the second and third rounds, and the second column represents that the second sensor turns on in the first and second rounds.

4.2. Population Initialization. In the QCGWA, every wolf is a potential optimal solution. Due to each wolf carries a two-dimensional matrix **T**, the population adopts a three-dimensional encoding method. First, it is necessary to calculate the distance between the sensor node and the monitored target; then, the sensor coverage matrix **S** is generated according to the monitoring range of the sensor. In addition, because of the particularity of the sensor duty cycle problem, we need to ensure that each sensor is turned on to its maximum number of lifetime rounds; hence, the initial wolf population is required to meet certain restrictions. Supposing that there are **X** sensor nodes in IWSNs and the maximum lifetime of each sensor node is **N** rounds, then the initialized individual in the population can be represented by

$$\text{individual} = \begin{bmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,X-1} & p_{1,X} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,X-1} & p_{2,X} \\ \vdots & & p_{i,x} & & \vdots \\ p_{XN-1,1} & p_{XN-1,2} & \cdots & p_{XN-1,X-1} & p_{XN-1,X} \\ p_{XN,1} & p_{XN,2} & \cdots & p_{XN,X-1} & p_{XN,X} \end{bmatrix} \left(p_{i,x} \in \{0, 1\}\right), \tag{7}$$

$$\sum_{i=1}^{XN} p_{i,m} = N, \quad m \in \{1, 2, \cdots, X\}. \tag{8}$$

In (7) and (8), **p** is a binary bit of the individual, and it indicates the on state of the sensor, 0 means the sensor is off, and 1 means the sensor is on.

4.3. Fitness Calculation and Class Division. The proposal of QCGWO is based on the characteristics of gray wolves in nature. Gray wolves are a group of animals, and they have a strict social hierarchy in the population, as shown in Figure 2. There are differences in quantity and work duty of wolves in different classes. Specifically, the wolf with the highest rank is called the Alpha wolf or the dominant wolf. What is more, the Alpha wolf is not necessarily the strongest wolf in the population, but the optimum at managing the group, and its role is to make decisions about group activities. The wolves in the second class are Beta wolves, which help Alpha wolves make decisions and are potential candidates for Alpha wolves. The wolves in the third class are called Delta wolves, and they usually obey the command of Alpha wolves and Beta wolves. At last, the Omega wolves are in the lowest level; they have the lowest status but help maintain the overall combat ability of the wolves.

In QCGWO, it is necessary to calculate the fitness of each individual in the wolf group for obtaining different levels of wolves. After initializing the population, we can calculate the fitness of each individual according to equation (4), and then, the individual with the highest fitness is divided into Alpha wolves, and the remaining individuals are divided into Beta wolves, Delta wolves, and Omega wolves according to their fitness.

4.4. Update Wolf Population and Algorithm Parameters. In the update mechanism of QCGWO, the process of encircling prey by wolves is imitated; specifically, the QCGWO considers the location of the wolves and the location of the prey. Each update operation of the population is carried out according to the position of the prey, so that the search for the solution space of the problem is realized. In the sensor node duty cycle problem, the prey refers to the individual with the highest fitness. Subsequently, the update process can be represented by

$$L = \left| C * O_{\text{prey}}(\text{gen}) - O(\text{gen}) \right|, \tag{9}$$

$$O(\text{gen} + 1) = O_{\text{prey}}(\text{gen}) - A * L. \tag{10}$$

In (9) and (10), **O** and O_{prey} represent the current positions of the wolf and the prey, respectively, **gen** is the current iteration number, and **L** stands for the distance between the wolf and the prey. **A** and **C** are two vector coefficients, they can be expressed as

$$\begin{aligned} A &= 2a * \text{rand}_1 - a, \\ C &= 2 * \text{rand}_2, \end{aligned} \tag{11}$$

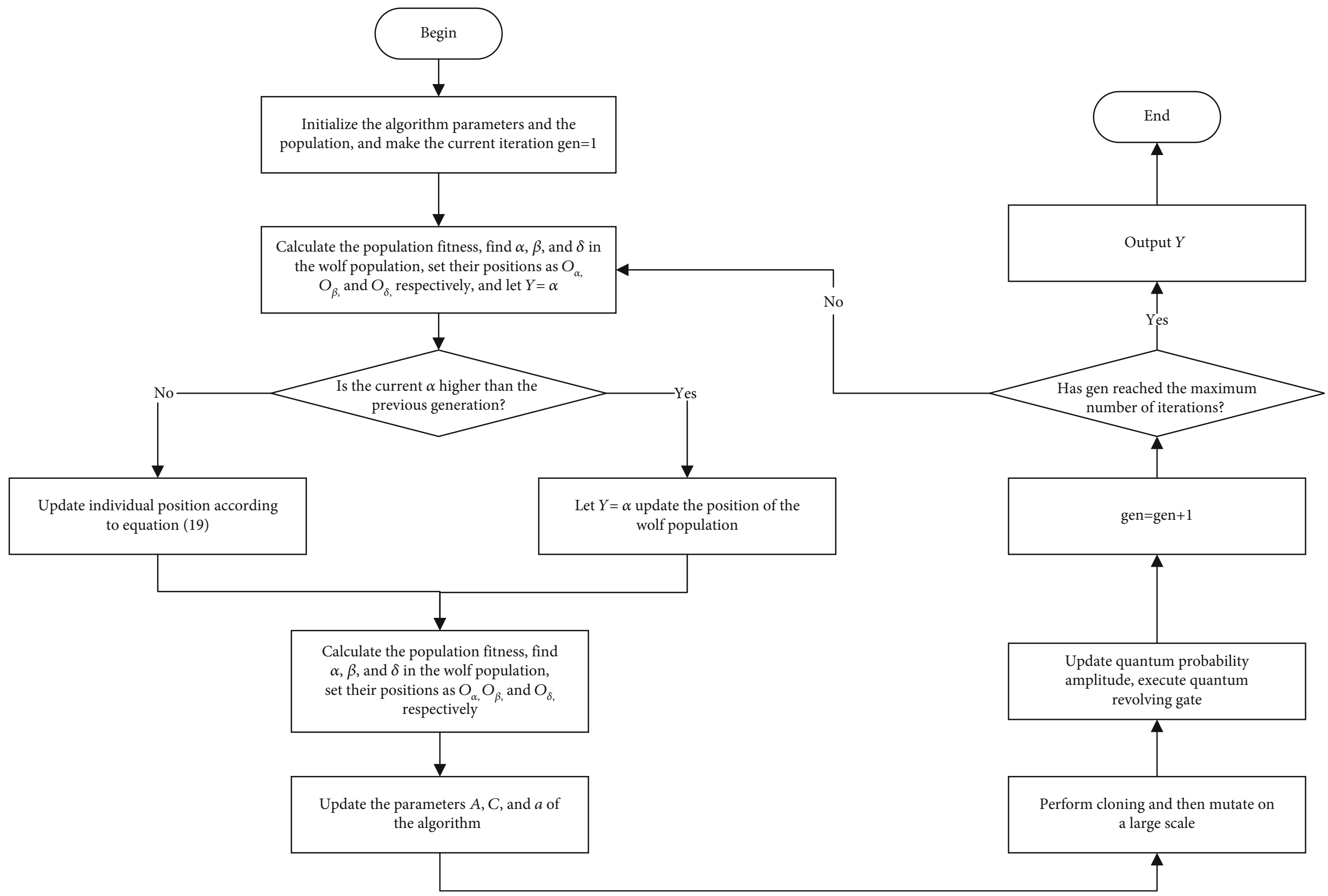

Figure 4: Steps of QCGWO.

Table 1: The experimental conditions in Figure 5.

	Sensors	Targets	Maximum lifetime	Monitoring radius (m)
Figure 5(a)	40	12	10	150
Figure 5(b)	50	15	10	150
Figure 5(c)	60	20	10	150
Figure 5(d)	100	30	10	150

where **a** is the convergence factor that decreases with the number of iterations from 2 to 0, and rand_1 and rand_2 are random numbers in [0,1].

In QCGWO, the approximate position of the prey is in the middle of Alpha, Beta, and Delta wolves. Subsequently, all wolves in the population surround the estimated position. The movement process of wolves can be expressed as

$$L_\alpha = |C_1 * O_\alpha - O|, \tag{12}$$

$$L_\beta = |C_2 * O_\beta - O|, \tag{13}$$

$$L_\delta = |C_3 * O_\delta - O|, \tag{14}$$

$$O_1 = O_\alpha - A_1 * L_\alpha, \tag{15}$$

Table 2: The experimental conditions in Figure 6.

	Sensors	Targets	Maximum lifetime	Monitoring radius (m)
Figure 6(a)	30	10	15	180
Figure 6(b)	45	15	15	180
Figure 6(c)	60	20	15	180
Figure 6(d)	75	25	15	180

Table 3: The experimental conditions in Figure 7.

	Sensors	Targets	Maximum lifetime	Monitoring radius (m)
Figure 7(a)	30	10	20	130
Figure 7(b)	50	15	20	130
Figure 7(c)	70	25	25	130
Figure 7(d)	100	30	25	130

$$O_2 = O_\beta - A_2 * L_\beta, \tag{16}$$

$$O_3 = O_\delta - A_3 * L_\delta. \tag{17}$$

In equations (12)–(17), O_1, O_2, and O_3 represent the positions of Alpha, Beta, and Delta wolves, respectively. L_α,

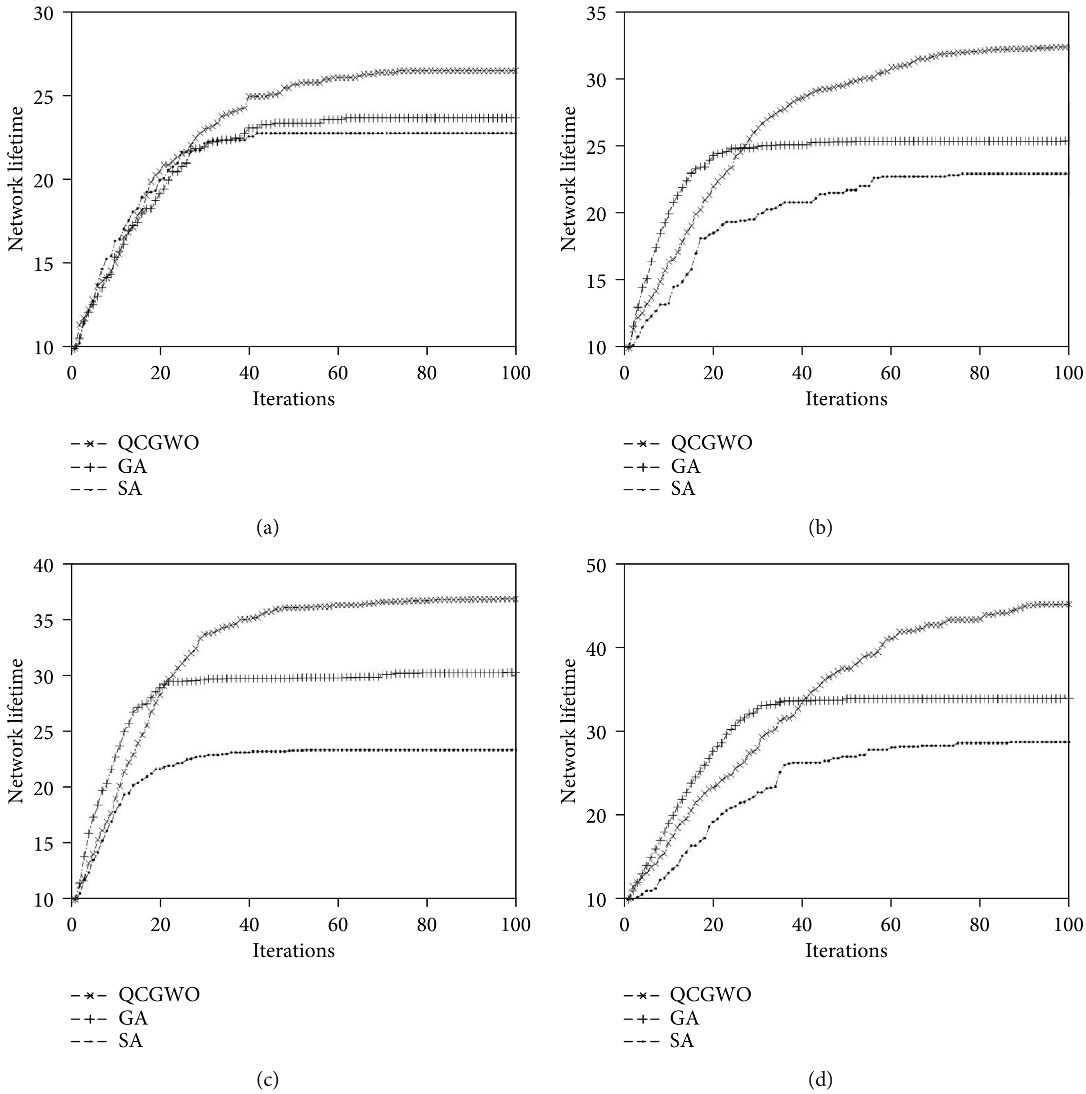

Figure 5: Network lifetime in comparison of the three algorithms: (a) 40 sensors and 12 targets; (b) 50 sensors and 15 targets; (c) 60 sensors and 20 targets; (d) 100 sensors and 30 targets.

L_β, and L_δ denote the distance from the current individual to Alpha, Beta, and Delta wolves, respectively. What is more, the process of wolves rounding up prey is shown in

$$O_{\text{gen}+1} = \frac{[O_1(\text{gen}) + O_2(\text{gen}) + O_3(\text{gen})]}{3}. \tag{18}$$

In (18), **gen** represents the current number of iterations.

4.5. Quantum Probability Amplitude and Quantum Revolving Gate. QCGWO uses the natural parallel mechanism of quantum computing to update the population. Compared with the traditional gray wolf optimization algorithm, the addition of quantum probability amplitude greatly enhances the global parallel search capability of QCGWO, which is a big difference between QCGWO and traditional GWO. By combining qubits and quantum superposition states, the convergence speed can be effectively improved when solving the problem of a large-scale sensor duty cycle. Furthermore, QCGWO represents each wolf in the population with a set of binary quantum probability amplitude bits. In the initial wolf population, all qubit probability amplitudes on each wolf are produced by logistic chaotic mapping, which is shown in

$$q_{z+1} = bq_z(1 - q_z), \quad z = 0, 1, \cdots, X. \tag{19}$$

In (19), $b = 4$ indicates that the mapping is in a chaotic state, $\mathbf{q}$ represents the generated quantum probability amplitude, and $\mathbf{X}$ is the number of sensors.

After all the wolves in the population are updated, in order to enhance the population diversity, the quantum revolving gate needs to be updated according to the qubits

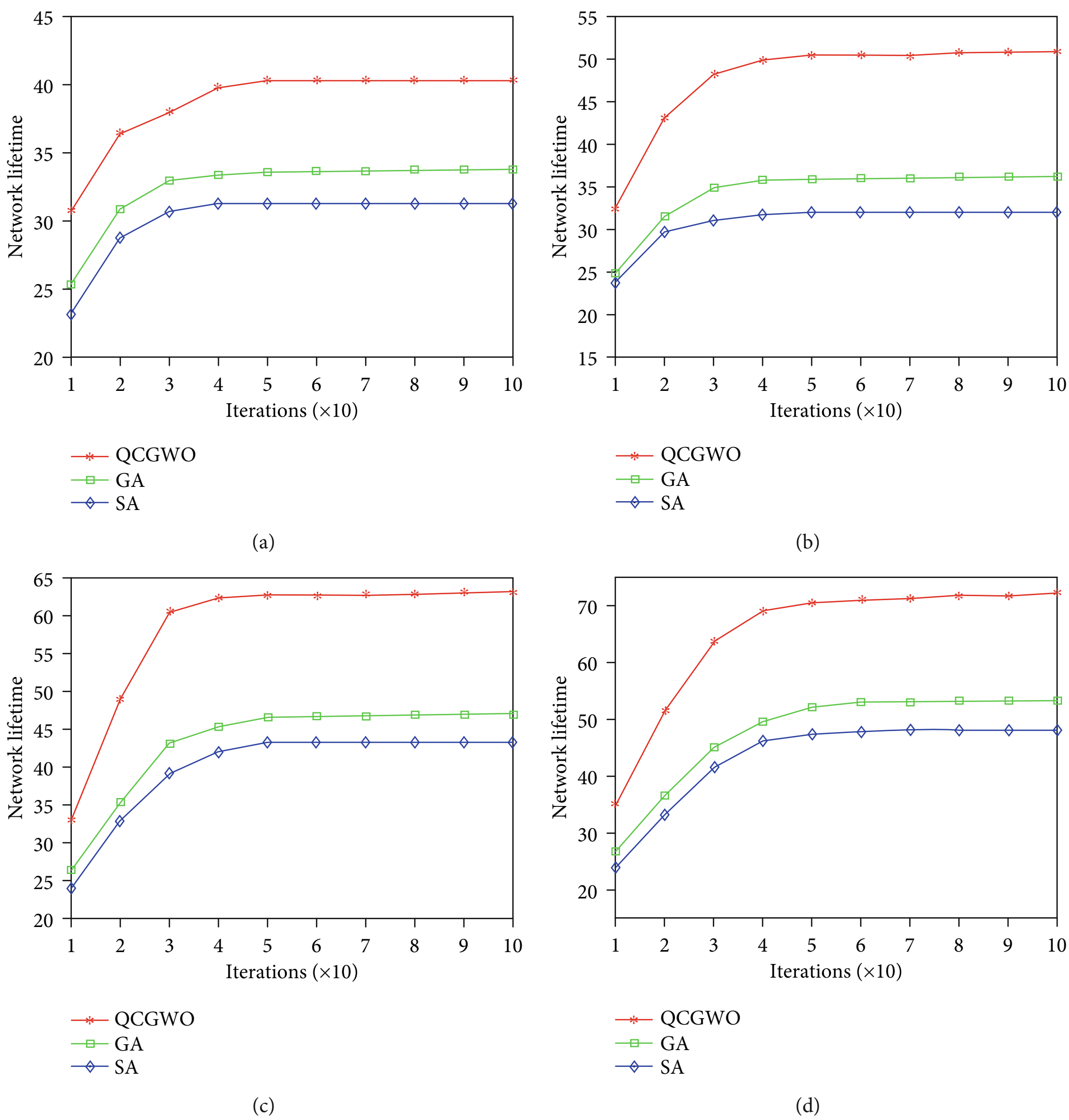

Figure 6: Network lifetime in comparison of the three algorithms: (a) 30 sensors and 10 targets; (b) 45 sensors and 15 targets; (c) 60 sensors and 20 targets; (d) 75 sensors and 25 targets.

on the Alpha wolves in the current population. The update process can be shown as

$$\begin{bmatrix} {y_1}^{\text{after}} \\ {y_2}^{\text{after}} \end{bmatrix} = \begin{bmatrix} \cos \varepsilon & -\sin \varepsilon \\ \sin \varepsilon & \cos \varepsilon \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}. \tag{20}$$

In (20), y_1 and y_2 represent the quantum probability amplitude before revolving, and ε is the angle of quantum revolving.

4.6. Clonal Expansion. The purpose of clonal expansion is to maximize the preservation of individuals with high adaptability, which can obviously increase the convergence performance of the QCGWO. At the same time, under the effect of the quantum probability amplitude, the application of the clone operator will not reduce the algorithm's global search ability, and the combination of clone and quantum effectively improves the performance of QCGWO. In addition, the selection of the clone parent is based on the fitness of the individuals in the current population. The higher the fitness, the more likely the individual is to be selected. For the same purpose of increasing the diversity of the population, the traditional clone operation is updated in QCGWO, and the cloned population is optimized through multilevel cloning. The specific cloning operation can be expressed as Figure 3.

4.7. Termination Operation. In each iteration, QCGWO will repeat the above process. If the QCGWO reaches the specified number of iterations, it will be terminated.

4.8. Steps of the Algorithm. The detailed process of QCGWO is shown below.

Step 1. Initialize the parameters in QCGWO, randomly initialize the positions of the sensors and the targets, then

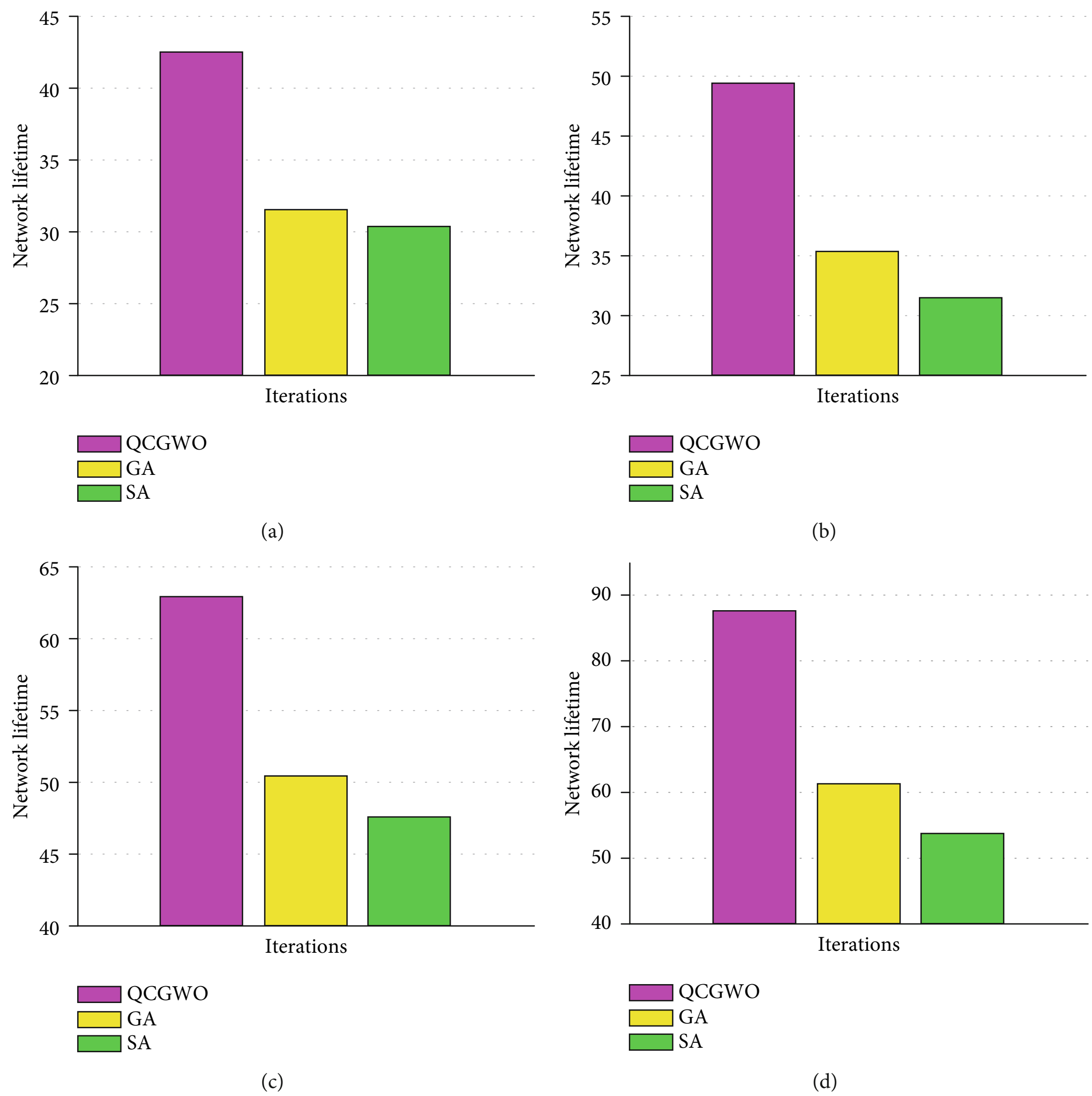

Figure 7: Network lifetime in comparison of the three algorithms: (a) 30 sensors and 10 targets; (b) 50 sensors and 15 targets; (c) 70 sensors and 25 targets; (d) 100 sensors and 30 targets.

generate the sensor coverage matrix **S**, and randomly generate the initial population. The initial quantum probability amplitude is 0.5, and the initial iteration **gen** = 1.

Step 2. Calculate the fitness of each wolf in the population, find **α**, **β**, and **δ** in the current population, and set their positions as O_α, O_β, and O_δ, respectively.

Step 3. Update the positions of all individuals according to equation (18).

Step 4. Calculate the fitness of wolves in the population, find **α**, **β**, and **δ** in the current population, and set their positions as O_α, O_β, and O_δ, respectively.

Step 5. Update the parameters **a**, **A**, and **C** of the algorithm.

Step 6. Sort the fitness, and select the individuals with the highest fitness as the parent to perform the clone operation.

Step 7. Perform large-scale mutation operations on the clonal population. The mutation process uses the quantum probability amplitude.

Step 8. Calculate the fitness of the cloned population, and use the cloned population as the population for the next iteration process.

Step 9. Update the quantum probability amplitude according to the individual with the highest fitness, and execute the quantum revolving gate.

Step 10. **gen** = **gen** + 1, if the maximum number of iterations is reached, terminate the algorithm, otherwise go to step 3.

Table 4: Solution quality comparison.

	The percentage of QCGWO's solution better than that of GA	The percentage of QCGWO's solution better than that of SA
Lifetime in Figure 5(a)	11.81%	16.23%
Lifetime in Figure 5(b)	27.46%	41.19%
Lifetime in Figure 5(c)	21.45%	57.71%
Lifetime in Figure 5(d)	33.04%	57.14%

Table 5: The comparison of convergence speed in Figure 6.

The specified lifetime rounds	The number of iterations of QCGWO	The number of iterations of GA	The number of iterations of SA
30 rounds in Figure 6(a)	10	17	27
30 rounds in Figure 6(b)	8	18	22
40 rounds in Figure 6(c)	14	26	33
40 rounds in Figure 6(d)	14	24	28

The algorithm flow chart of QCGWO is shown in Figure 4.

5. Results and Discussion

The QCGWO method we propose on solving the sensor duty cycle problem will take a series of experiments, and QCGWO has been compared with GA and SA for proving its effectiveness. The comparison for the algorithms is carried out under the conditions of different quantity of sensors, monitored targets, different maximum sensor lifetime, and monitoring area radius. In addition, all test cases are completed on the machine with a R7 4800H 2.9 GHz CPU, and the fitness used in the algorithms is calculated according to formula (4).

With the purpose of enabling the three algorithms to be compared under the same experimental conditions, we uniformly define the parameters commonly used in the sensor duty cycle problem in the IWSNs. The number of iterations is set to 100 generations, and the population size is 40. The monitoring area of IWSNs is set as a square area with a side length of 200, and the coordinates of the sensors and the target nodes are randomly generated in the area. In QCGWO, the initial value of the quantum probability amplitude is set to 0.5, the probability of the quantum revolving gate is set to 0.05, and the mutation rate of the clone operation is 0.3. In GA, the mutation rate of the population is 0.1, and the crossover method is two-point crossover. In SA, the initial temperature is 200, the annealing method is exponentially decreased and the annealing factor is 0.95.

Tables 1–3 show the experimental conditions of Figures 5–7, respectively.

In Figures 5(a)–5(d), the convergence speed of the three algorithms is shown. Specifically, Figure 5(a) indicates that QCGWO converged faster than GA and SA, and QCGWO has maintained a fast convergence rate during the iterative process. In contrast, SA fell into premature convergence in the 40th iteration, and GA also fell into premature convergence in the 60th iteration, so they are unable to find the optimal solution. In Figure 5(b), the maximum network lifetime obtained by QCGWO is 32.56 rounds. However, the optimal solution obtained by GA is 25.08 rounds, and the optimal solution obtained by SA is 23.14 rounds. The solutions of QCGWO are 28% and 39% higher than GA and SA, respectively. In Figure 5(c), GA and SA fell into the local optimum in about 30 iterations, while QCGWO effectively jumped out of the local optimum by its good global search ability. What is more, in Figure 5(d), QCGWO has maintained rapid convergence speed until the 90th generation finds the optimal solution 45.68; however, due to the premature convergence and weak ability for jumping out of the local optimum, the highest solutions obtained by GA and SA are 33.43 and 28.59, respectively, which are lower than the network lifetime of QCGWO. The comparison of the solution quality of the three algorithms in Figure 5 is shown in Table 4.

According to Table 4, it is obvious that the quality of the solution obtained by using QCGWO is better than that of GA and SA. Particularly, when the scale of IWSNs expands, the advantages of QCGWO become more prominent.

Figures 6(a)–6(d) show the trend of the three algorithms more clearly in the form of line charts. According to Figure 6(a), in the 10th generation, QCGWO's network lifetime value is already higher than GA and SA. Since then, QCGWO has maintained a leading position and found the optimal solution 40.10. Subsequently, Figure 6(b) shows that GA and SA fall into premature convergence in the 40th generation, which leads to the local optimal solutions of 36.47 and 27.89, respectively, while QCGWO effectively obtains the optimal solution 51.66. In Figures 6(c) and 6(d), during the early iterations, QCGWO, GA, and SA all converged very quickly, but it is obvious that both GA and SA have fallen into local convergence. Therefore, both GA and SA only got local optimal solutions, while QCGWO obtained better solutions than them. In general, QCGWO has better performance than GA and SA, and the convergence speed of the three algorithms in Figure 6 can be shown as Table 5.

According to Table 5, we can find that in the same experimental conditions, the number of iterations of QCGWO for reaching the specified number of lifetime rounds is always less than that of GA and SA, which proves the good convergence performance of QCGWO.

With the purpose of making the network lifetimes obtained by the three algorithms more obvious, Figures 7(a)–7(d) use bar charts to display the data. In Figure 7(a), the maximum network lifetime values obtained by QCGWO, GA, and SA are 42.40, 31.50, and 30.30, respectively. In Figures 7(c) and 7(d), the network lifetime obtained by QCGWO is also the highest, with values of 49.35, 62.90, and 87.67, respectively. Moreover, the values obtained by GA are 35.30, 50.40, and 61.33, respectively, and the solutions obtained by SA are 31.50, 47.60, and 53.67, respectively. Therefore, under the specified experimental conditions, QCGWO always performed better than GA and SA.

6. Conclusions

The purpose of this paper is to prolong the lifetime of the IWSNs. Therefore, we modeled the industrial sensor network in the real factory, proposed a quantum clone gray wolf optimization (QCGWO) algorithm, designed the sensor duty cycle model from a different perspective compared with the previous works, and proposed a concept of measurable sensor lifetime. The algorithm we proposed has the advantages of high solution accuracy, strong convergence performance, and strong global search ability. What is more, the QCGWO learns from the traditional gray wolf optimization algorithm (GWO), but we have achieved important innovations of combining the GWO with some current popular technologies, including quantum operator and clone operator, thereby effectively making up for the weakness of GWO that is easy to fall into local optimum.

The effectiveness of the proposed model has been verified by different experimental conditions in Section 5, and the results suggested that the proposed model can achieve a longer network lifetime. In addition, in order to prove the advantages of the proposed algorithm in solving the sensor duty cycle problem, we compare QCGWO with GA and SA. The results show that the QCGWO is more competitive than GA and SA in improving the lifetime of IWSNs. Finally, we would like to highlight that the proposed model and the QCGWO can successfully solve the sensor duty cycle problem, and the approach proposed in this paper provides a new perspective for prolonging the network lifetime in IWSNs.

Data Availability

The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Disclosure

The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

This paper was funded by the Corps Innovative Talents Plan, grant number 2020CB001, the project of Youth and Middle-aged Scientific and Technological Innovation Leading Talents Program of the Corps, grant number 2018CB006, the China Postdoctoral Science Foundation, grant number 220531, the Funding Project for High Level Talents Research in Shihezi University, grant number RCZK2018C38, and the Project of Shihezi University, grant number ZZZC201915B.

References

[1] E. H. Houssein, M. R. Saad, F. A. Hashim, H. Shaban, and M. Hassaballah, "Levy flight distribution: a new metaheuristic algorithm for solving engineering optimization problems," *Engineering Applications of Artificial Intelligence*, vol. 94, p. 18, 2020.

[2] B. Houtan and H. Zarrabi, "Obstacle-aware fuzzy-based localization of wireless chargers in wireless sensor networks," *Canadian Journal of Electrical and Computer Engineering*, vol. 43, no. 1, pp. 17–24, 2020.

[3] E. G. Pardo, J. Blanco-Linares, D. Velazquez, and F. Serradilla, "Optimization of a steam reforming plant modeled with artificial neural networks," *Electronics*, vol. 9, no. 11, 2020.

[4] S. Najjar-Ghabel, L. Farzinvash, and S. N. Razavi, "Mobile sink-based data gathering in wireless sensor networks with obstacles using artificial intelligence algorithms," *Ad Hoc Networks*, vol. 106, p. 12, 2020.

[5] G. S. Gandhi, K. Vikas, V. Ratnam, and K. S. Babu, "Grid clustering and fuzzy reinforcement-learning based energy-efficient data aggregation scheme for distributed WSN," *IET Communications*, vol. 14, no. 16, pp. 2840–2848, 2020.

[6] H. Chen, Y. Qin, K. Lin et al., "PWEND: proactive wakeup based energy-efficient neighbor discovery for mobile sensor networks," *Ad Hoc Networks*, vol. 107, p. 102247, 2020.

[7] A. Janarthanan and D. Kumar, "Localization based evolutionary routing (LOBER) for efficient aggregation in wireless multimedia sensor networks," *Computers, Materials and Continua*, vol. 30, no. 3, pp. 895–912, 2019.

[8] D. M. Gao, S. Zhang, F. Q. Zhang, X. J. Fan, and J. C. Zhang, "Maximum data generation rate routing protocol based on data flow controlling technology for rechargeable wireless sensor networks," *Computers, Materials & Continua*, vol. 59, no. 2, pp. 649–667, 2019.

[9] Q. W. Zhang, D. Z. Li, Y. Fei, J. K. Zhang, Y. Chen, and F. Tong, "RDCPF: a redundancy-based duty-cycling pipelined-forwarding MAC for linear sensor networks," *Sensors*, vol. 20, no. 19, 2020.

[10] M. A. Panhwar, D. Z. Liang, K. A. Memon, S. A. Khuhro, M. A. K. Abbasi, and Z. A. Noor-ul-Ain, "Energy-efficient routing optimization algorithm in WBANs for patient monitoring," *Journal of Ambient Intelligence and Humanized Computing*, 2020.

[11] J. Z. Xie, B. J. Zhang, and C. P. Zhang, "A novel relay node placement and energy efficient routing method for

heterogeneous wireless sensor networks," *IEEE Access*, vol. 8, pp. 202439–202444, 2020.

[12] M. A. Panhwar, Z. Deng, S. A. Khuhro, and D. N. Hakro, "Distance based energy optimization through improved fitness function of genetic algorithm in wireless sensor network," *Studies in Informatics and Control*, vol. 27, no. 4, pp. 461–468, 2018.

[13] A. Latha, S. Prasanna, S. Hemalatha, and B. Sivakumar, "A harmonized trust assisted energy efficient data aggregation scheme for distributed sensor networks," *Cognitive Systems Research*, vol. 56, pp. 14–22, 2019.

[14] U. Patil, A. V. Kulkarni, R. Menon, and M. Venkatesan, "A novel AEB-AODV based AADITHYA cross layer design hibernation algorithm for energy optimization in WSN," *Wireless Personal Communications*, pp. 1–21, 2020.

[15] I. Jemili, D. Ghrab, A. Belghith, and M. Mosbah, "Cross-layer adaptive multipath routing for multimedia wireless sensor networks under duty cycle mode," *Ad Hoc Networks*, vol. 109, p. 102292, 2020.

[16] S. Chen, L. Zhang, Y. Tang et al., "Indoor temperature monitoring using wireless sensor networks: a SMAC application in smart cities," *Sustainable Cities and Society*, vol. 61, article 102333, 2020.

[17] H. Wang, K. Li, and W. Pedrycz, "A routing algorithm based on simulated annealing algorithm for maximising wireless sensor networks lifetime with a sink node," *International Journal of Bio-Inspired Computation*, vol. 15, no. 4, pp. 264–275, 2020.

[18] A. S. Kirsan, U. H. Al Rasyid, I. Syarif, and D. N. Purnamasari, "Energy efficiency optimization for intermediate node selection using MhSA-LEACH: multi-hop simulated annealing in wireless sensor network," *EMITTER International Journal of Engineering Technology*, vol. 8, no. 1, pp. 1–18, 2020.

[19] H. L. Wang, G. B. Zhou, L. Bhatia, Z. C. Zhu, W. Li, and J. A. McCann, "Energy-neutral and QoS-aware protocol in wireless sensor networks for health monitoring of hoisting systems," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 8, pp. 5543–5553, 2020.

[20] F. Fraternali, B. Balaji, Y. Agarwal, and R. K. Gupta, "ACES," *ACM Transactions on Sensor Networks (TOSN)*, vol. 16, no. 4, pp. 1–31, 2020.

[21] S. Tahmasebi, M. Safi, S. Zolfi, M. R. Maghsoudi, H. R. Faragardi, and H. Fotouhi, "Cuckoo-PC: an evolutionary synchronization-aware placement of SDN controllers for optimizing the network performance in WSNs," *Sensors*, vol. 20, no. 11, article 3231, 2020.

[22] H. Y. Huang, K. T. Kim, and H. Y. Youn, "Determining node duty cycle using Q-learning and linear regression for WSN," *Frontiers of Computer Science*, vol. 15, no. 1, pp. 1–7, 2020.

[23] D. Passos, C. O. de Sousa, and C. Albuquerque, "An NDT model for block designs operating under asymmetrical duty cycling," *IEEE Wireless Communications Letters*, vol. 9, no. 12, pp. 2116–2120, 2020.