

Article

A Quantum Ant Colony Multi-Objective Routing Algorithm in WSN and Its Application in a Manufacturing Environment

Fei Li ^{1,*} , Min Liu ²  and Gaowei Xu ² 

¹ Department of Computer Science, Zhejiang University City College, Hangzhou 310015, China

² College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China

* Correspondence: lif@zucc.edu.cn; Tel.: +86-153-8112-5261

Received: 5 June 2019; Accepted: 26 July 2019; Published: 29 July 2019



Abstract: In many complex manufacturing environments, the running equipment must be monitored by Wireless Sensor Networks (WSNs), which not only requires WSNs to have long service lifetimes, but also to achieve rapid and high-quality transmission of equipment monitoring data to monitoring centers. Traditional routing algorithms in WSNs, such as Basic Ant-Based Routing (BABR) only require the single shortest path, and the BABR algorithm converges slowly, easily falling into a local optimum and leading to premature stagnation of the algorithm. A new WSN routing algorithm, named the Quantum Ant Colony Multi-Objective Routing (QACMOR) can be used for monitoring in such manufacturing environments by introducing quantum computation and a multi-objective fitness function into the routing research algorithm. Concretely, quantum bits are used to represent the node pheromone, and quantum gates are rotated to update the pheromone of the search path. The factors of energy consumption, transmission delay, and network load-balancing degree of the nodes in the search path act as fitness functions to determine the optimal path. Here, a simulation analysis and actual manufacturing environment verify the QACMOR's improvement in performance.

Keywords: wireless sensor network (WSN); energy; ant colony optimization (ACO); routing algorithm; quantum-inspired evolutionary algorithms

1. Introduction

Recent years have seen a worldwide interest in Wireless Sensor Network (WSN) [1] technology, which has been considered one of the most promising technologies in smart manufacturing. Actually, the development tendency of WSN is in accordance with its context of Industry 4.0 [2]. Together with the Industrial Internet, the Internet of Things (IoT) [3], whose kernel is WSN, contributes to the achievement of the connectivity and communication of Cyber-Physical Systems (CPS) [4]. WSN techniques are appropriate for long-term data acquisition for IoT representation in an industrial environment.

WSNs are distinguished from traditional wireless networks by their dissimilar purposes: WSNs are data-centric, while the latter aim for data transmission. In traditional wireless networks, such as Ad hoc and Wireless Local Area Networks (WLANs), the main task is to find the low-latency path between the source node and the destination node, and to improve the utilization of the whole network in order to avoid communication congestion and simultaneously balance network flow. However, in WSNs, a routing method has two main functions: to find the optimal path from the source node to the destination node, and to transmit a data packet along that path. The main aim of network routing improvement is to extend network life and prevent connection errors [5]. The routing method's emphasis is on energy efficiency, because of limited node energy and long lifetime requirements. Meanwhile, since the number of sensor nodes tends to be very large, and these

nodes can only obtain local topological information, a suitable route should be chosen by considering local network information.

Since the network is resource- and power-limited, general wireless communication network routing methods are not well-suited for WSNs, especially in industrial fields in which there is demand for high performance in energy efficiency and longevity. Accordingly, some routing approaches have emerged, such as swarm intelligence-based schemes [5,6]. Social insect colonies, such as those of ants and honeybees [7,8], have complex collective behaviors and decentralized management structures, which are similar to parallel, dynamic, and distributed systems. Researchers have studied ant colony optimization (ACO)-based routing schemes to develop high-performance routing methods [9].

In order to improve the limitations of ACO-based routing methods, such as earlier stagnation and slow astringency, this paper considers the idea of using quantum-inspired evolutionary algorithms (QEAs) [10,11] and ACO together, balancing load, real-time transmission, and energy consumption with a multi-objective fitness function. A novel and efficient routing approach for WSNs, called the Quantum Ant Colony Multi-Objective Routing (QACMOR) algorithm, is proposed accordingly. In QACMOR, some quantum computing mechanisms of QEAs, including the quantum bit (qubit) and the quantum rotation gate, are introduced into ACO. The former represents the node's pheromone, and the latter updates it. QEAs are able to avoid premature convergence with a simple implementation, which has more potential for solving large-scale problems than do other general evolutionary algorithms. In multiple objectives, more attention is paid to computation speed by using the look-up table of the rotation angle of QEAs and setting a time-delay factor in fitness function.

The rest of the paper is organized as follows: Section 2 presents the literature review on WSN routing methods. Section 3 sheds light on ACO-based routing in detail. Section 4 explains the proposed QACMOR approach. Section 5 shows the experimental results of performance evaluation and case study validated in a continuous steel casting production line. Finally, Section 6 discusses conclusions and future work.

2. Literature Review

The routing protocol of WSNs should be devised with properties such as energy efficiency, scalability, robustness, and rapid convergence, compared to that of traditional networks. A large number of routing methods have been proposed. Roughly, they can be divided into four categories through the analysis of relevant literature—that is, data-centric, clustering, geographic location-based, and Quality of Service (QoS)-based routing methods.

Data-centric routing was proposed to reduce the flooding overhead caused by transmitting query and data information. In data-centric routing, data request and collection are based on data attributes, rather than only using local interactions [12,13]. Clustering is the most common technique used for achieving energy-efficient and scalable performance in large-scale sensor networks. Cluster formation is a process whereby sensor nodes decide which cluster head they should associate with among multiple clusters [14,15]. The low-energy adaptive clustering hierarchy (LEACH) [15], a typical cluster-based algorithm, divides a sensor network into a set of clusters, through which energy consumption is balanced and reduced. In geographical routing, the physical location of the sensor node is used to guide the path that a packet takes in the network [16,17]. In some cases of WSN application, a higher-communication QoS is demanded, such as reliability and real-time data transmission. The method in [18,19] can be classified in this category.

Routing methods based on swarm intelligence have robust, adaptive, and scalable performance, suitable for autonomous distributed systems [20,21]. Inspired by the foraging principles of honeybees, Saleem et al. [22] proposed a distributed and decentralized routing protocol called the BeeSensor protocol. Camilo et al. [23] studied the application of the ACO metaheuristic to solve the routing problem in WSNs, and came up with an energy-efficient, ant-based routing algorithm (EEABR). Zungeru et al. [24] improved the EEABR algorithm by applying a new scheme to intelligently initialize and update routing tables, reducing the flooding ability of ants for congestion control. In [25],

a self-adaptive routing mechanism is presented to ensure reliability and efficiency during data transmission by adopting the dissemination of a pheromone as a model for dealing with dynamic changes in WSN.

QEAs are based on the concept and principles of quantum computing, such as the quantum bit and the superposition of states. As a kind of evolutionary algorithm, a QEA is also characterized by the representation of the individual, the evaluation function, and population dynamics. Learning from the quantum rotation gate strategy of QEAs, Xing et al. [26] introduced an adaptive evolution mechanism for QoS multicasting in IP/DWDM networks, which allowed each chromosome in a population to update itself to a fitter position according to its own situation.

3. Preliminaries

3.1. Energy Consumption Model

Communication is the activity responsible for the bulk of the energy consumption in WSNs [27]. An energy consumption model used in Reference [27] is applied in this study (see Figure 1).

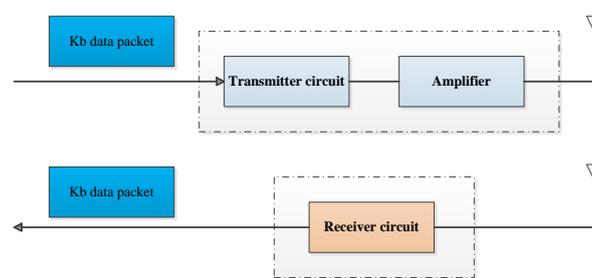


Figure 1. The energy consumption model.

Assumptions are that: the data can reach every node from its neighbors; the data contain information on distance and residual energy; the radio circuit in the sensor has a power control, and can expend the minimum required energy to reach the intended recipients; and radio circuit can be turned off to avoid receiving unintended transmissions. The transmission computation costs and receiving costs for a k -bit message at a certain distance d are shown as follows:

Transmitting

$$E_T(k, d) = E_{elec} \times k + E_{amp} \times k \times d^2 \quad (1)$$

Receiving

$$E_R(k) = E_{elec} \times k + E_{BF} \times k \quad (2)$$

Total energy cost

$$E = E_T + E_R \quad (3)$$

where $E_{elec} = 50nJ/bit$, $E_{amp} = 100pJ/bit/m^2$ for the transmitter amplifier, and $E_{BF} = 5nJ/bit$ when beamforming is used. d represents the distance of two nodes, and k represents the number of message bits.

Thus, by decreasing the communication distance and the volume of data to transmit, energy can be saved.

3.2. Basic Ant-Based Routing (BABR) Algorithm

In ACO, ants exchange data by pheromones, and according to the positive feedback principle, a path with a high density of pheromones has a higher probability of being selected. Such optimization can be adapted to implement basic ant-based routing for WSNs [9,23]:

Step 1: At regular intervals, a forward ant k starts to move from the source node toward the destination. While moving, the identifiers of every visited node are recorded in a list, M_k , and each forward ant avoids traversing a node that has been visited previously.

Step 2: At each node r , a forward ant selects the next hop node in accordance with a certain probability distribution:

$$P_k(r, s) = \begin{cases} \frac{[T(r, s)]^\mu \cdot [E(s)]^\nu}{\sum_{s \notin M_k} [T(r, s)]^\mu \cdot [E(s)]^\nu}, & \text{if } s \notin M_k \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where $P_k(r, s)$ is the probability of individual k that moves from node r to node s , and T is the routing table at each node with the amount of pheromone on the link (r, s) stored. E represents the heuristic information given by $1/(C - e_s)$ (C is the initial energy level of the nodes and e_s is the actual energy level of node s), and μ, ν are weight parameters that signify the importance of pheromones versus heuristics.

Step 3: When a forward ant reaches the destination, a backward ant goes back along the links that the forward ant has visited. Before moving, the amount of pheromones that the ant will drop during the trip is computed:

$$\Delta T_k = \frac{1}{N - Fd_k} \quad (5)$$

where N is the total number of nodes, and Fd_k is the distance traveled by the forward ant k .

Step 4: Whenever a node r receives a backward ant from a neighbor node, the routing table is updated:

$$T_k(r, s) = (1 - \rho)T_k(r, s) + \Delta T_k \quad (6)$$

where ρ is a coefficient, and then $(1 - \rho)$ represents the evaporation of pheromones.

Step 5: Once a backward ant returns to the source node, the next interval is continued.

After several iterations, each node will find the best neighbors to which to send a data packet. While the ability and robustness of the ACO-based method qualify it to find a good solution, it still has the possibility of getting stuck in slow astringency and early stagnation.

4. The QACMOR Routing Method

This section first introduces the basic concepts and rules of QEAs, and then elaborates on the QACMOR algorithm for WSNs routing.

4.1. Mechanisms of QEAs

4.1.1. Basic Elements of QEAs

The memory unit in a classical computer is the bit, which only has two states: "0" or "1", whereas the smallest information unit in QEAs is defined as the qubit [10,11]. A qubit could be in the "0" state, the "1" state, or in a linear superposition of both, which is denoted as $\alpha|0\rangle + \beta|1\rangle$, where $|0\rangle$ and $|1\rangle$ represents the quantum state, and a pair of complex numbers (α, β) is defined with $|\alpha|_2 + |\beta|_2 = 1$, and the value of $|\alpha|^2$ and $|\beta|^2$ indicates the probability of the "0" state and the "1" state, respectively.

A qubit with the size of n can be represented as the following, which has 2^n kinds of states:

$$\left(\begin{array}{c|c|c|c|c} \alpha_1 & \alpha_2 & \dots & \alpha_i & \dots & \alpha_n \\ \beta_1 & \beta_2 & \dots & \beta_i & \dots & \beta_n \end{array} \right) \quad (7)$$

For example, a quantum individual with three qubits is given like this:

$$\left(\begin{array}{c|c|c} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & \frac{\sqrt{3}}{2} \end{array} \right) \quad (8)$$

It can also be represented as:

$$\frac{1}{4}|000\rangle + \frac{\sqrt{3}}{4}|001\rangle - \frac{1}{4}|010\rangle - \frac{\sqrt{3}}{4}|011\rangle + \frac{1}{4}|100\rangle + \frac{\sqrt{3}}{4}|101\rangle - \frac{1}{4}|110\rangle - \frac{\sqrt{3}}{4}|111\rangle \quad (9)$$

which means that the probabilities of the states $|000\rangle$, $|001\rangle$, $|010\rangle$, $|011\rangle$, $|100\rangle$, $|101\rangle$, $|110\rangle$, and $|111\rangle$ are $1/16$, $3/16$, $1/16$, $3/16$, $1/16$, $3/16$, $1/16$, and $1/16$, separately.

Commonly, ξ ($\xi \in (-\pi, \pi]$) denotes the phase of the qubit, and the i^{th} bit phase is $\xi_i = \arctan(\beta_i/\alpha_i)$. The position of ξ_i in coordination is given in Figure 2.

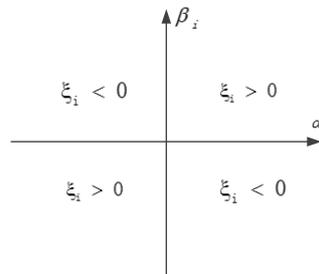


Figure 2. The position of ξ_i in coordination.

4.1.2. The Updating of Qubit in QEAs

In QEAs, the quantum rotation gate updates the qubit. The following formula represents a qubit which rotates θ_i degrees from the original vector, $(\alpha_i \ \beta_i)^T$ to $(\alpha'_i \ \beta'_i)^T$

$$\begin{bmatrix} \alpha'_i \\ \beta'_i \end{bmatrix} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} \quad (10)$$

θ_i is the rotation degree according to the following formula:

$$\theta_i = \Delta\theta \times s(\alpha_i, \beta_i) \quad (11)$$

$$\Delta\theta = 5 \times \exp(-t/t_{\max}) \quad (12)$$

In Formulas (11) and (12), $\Delta\theta$ represents the rotation step, controlling the rotation speed; t represents the current number of iterations; and t_{\max} represents the predefined maximal times of calculation determined by the scale of the problem. The function $s(\alpha_i, \beta_i)$ defines the direction:

$$s(\alpha_i, \beta_i) = (d_{\text{ibest}}/d_{\text{inow}})(\xi_{\text{ibest}} - \xi_{\text{inow}}) \quad (13)$$

where

$$\begin{aligned} d_{\text{inow}} &= \beta_{\text{inow}}/\alpha_{\text{inow}} \\ d_{\text{ibest}} &= \beta_{\text{ibest}}/\alpha_{\text{ibest}} \\ \xi_{\text{ibest}} &= \arctan(\beta_{\text{ibest}}/\alpha_{\text{ibest}}) \\ \xi_{\text{inow}} &= \arctan(\beta_{\text{inow}}/\alpha_{\text{inow}}) \end{aligned} \quad (14)$$

In Formula (14), α_{inow} , β_{inow} , α_{ibest} , β_{ibest} are the probability of the i^{th} qubit of the current and optimal solution, respectively. Finally, if $s(\alpha_i, \beta_i) < 0$, the θ_i rotates clockwise—otherwise, it rotates counterclockwise.

4.2. The QEAs in QACMOR

4.2.1. Representing the Pheromone with Qubit

In QACMOR, a qubit represents the pheromone for a population with the size of m individuals—that is, $Q = (q_1, q_2, \dots, q_j, \dots, q_m)$, $j = 1, 2, \dots, m$, and

$$q_j = \left(\begin{array}{c|c|c|c} \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \beta_1 & \beta_2 & \dots & \beta_n \end{array} \right) \quad (15)$$

where n is the number of qubits, $|\alpha_i|_2 + |\beta_i|_2 = 1, i = 1, 2, \dots, n$.

4.2.2. Updating the Pheromone with Quantum Rotation Gate

In QACMOR, similarly to Formulas (10) and (11), the quantum rotation gate G acting on the i^{th} bit of the j^{th} individual q_j of solution Q is described as follows:

$$\begin{bmatrix} \alpha'_{ji} \\ \beta'_{ji} \end{bmatrix} = G \begin{bmatrix} \alpha_{ji} \\ \beta_{ji} \end{bmatrix} \tag{16}$$

$$G = \begin{pmatrix} \cos \theta_{ji} & -\sin \theta_{ji} \\ \sin \theta_{ji} & \cos \theta_{ji} \end{pmatrix} \tag{17}$$

$$\theta_{ji} = \Delta\theta_{ji} \times s(\alpha_{ji}, \beta_{ji}) \tag{18}$$

where $i = 1, 2, \dots, n, (\alpha'_{ji}, \beta'_{ji})^T$ represents the updated bit, θ_{ji} is the rotation angle, $\Delta\theta_{ji}$ signifies the magnitude of the rotation angle, and $s(\alpha_{ji}, \beta_{ji})$ is a function of α_{ji} and β_{ji} , and controls the direction of rotation. For the computation speed, the look-up table was applied to compute the rotation angle as shown in Table 1, which includes all feasible solutions. $f(\cdot)$ denotes the fitness function as Formula (20); x_{ji} and b_i represent the i^{th} bit of the j^{th} individual of the current solution x and the best solution b , respectively. The schematic diagram in Figure 3 shows the rotation gate polar plot for a qubit individual.

Table 1. The look-up table of the quantum-inspired evolutionary algorithm (QEA) rotation angle [28].

x_i	b_i	$f(x) > f(b)$	$\Delta\theta_i$	$s(\alpha_i, \beta_i)$			
				$\alpha_i \beta_i > 0$	$\alpha_i \beta_i < 0$	$\alpha_i = 0$	$\beta_i = 0$
0	0	False	0	0	0	0	0
0	0	True	0	0	0	0	0
0	1	False	0	0	0	0	0
0	1	True	0.05π	+1	-1	0	± 1
1	0	False	0.01π	+1	-1	0	± 1
1	0	True	0.025π	-1	+1	± 1	0
1	1	False	0.005π	-1	+1	± 1	0
1	1	True	0.025π	-1	+1	± 1	0

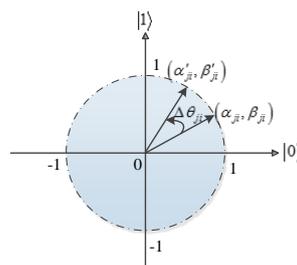


Figure 3. The polar plot of the rotation gate for a qubit individual.

Additionally, a conventional binary solution is significantly important for performance evaluation, and can be obtained by observing the qubits. For example, it is assumed that $x_i (i = 1, 2, \dots, n)$ is a certain bit of the binary individual x , then α_i of the qubit individual is compared with a random number $w (0 < w < 1)$. If $|\alpha_i|^2 > w$, then set the value of x_i to be "0", otherwise set the value of x_i to be "1". Therefore, for $Q = (q_1, q_2, \dots, q_j, \dots, q_m), j = 1, 2, \dots, m$, its binary solution is $P = (p_1, p_2, \dots, p_j, \dots, p_m)$, while $p_j (j = 1, 2, \dots, m)$ is a n -length binary individual, and then every element of p_j (for example, p_{ji}) is determined by comparing α_{ji} of q_j with $w, 0 < w < 1$.

4.3. The QACMOR Algorithm

The flowchart of the proposed approach is shown in Figure 4. The basic algorithm of QACMOR can be described as follows:

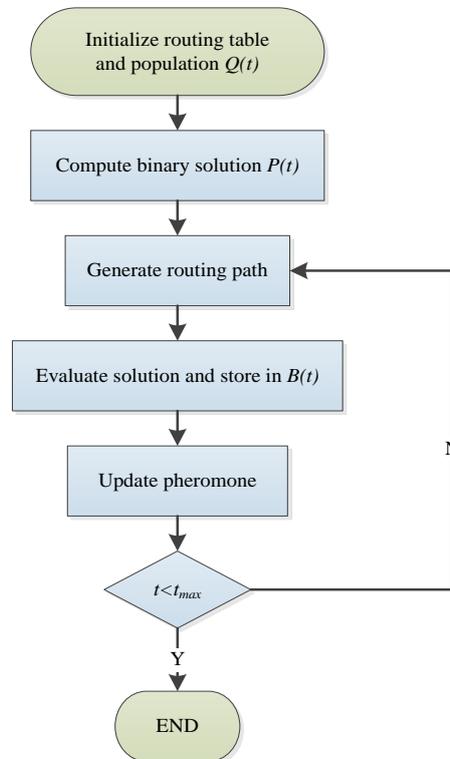


Figure 4. The process of the Quantum Ant Colony Multi-Objective Routing (QACMOR) approach.

Step 1: The initialization step. Add every node and its neighbor nodes into the routing table. A forward ant is generated from source nodes which carry the information of source nodes, sink nodes, and passing nodes. The population is represented as $Q(t) = (q_1^t, q_2^t, \dots, q_j^t, \dots, q_m^t)$ with the size of m individuals, where $q_j^t (j = 1, 2, \dots, m)$ is the j^{th} individual in the t^{th} iteration. The representation is shown as:

$$q_j^t = \left(\begin{array}{c|c|c} \alpha_{j1}^t & \dots & \alpha_{jn}^t \\ \beta_{j1}^t & \dots & \beta_{jn}^t \end{array} \right) \quad (19)$$

where n is the number of qubits. Initialize $\alpha_{ji}, \beta_{ji} (i = 1, 2, \dots, n)$ with $1/\sqrt{2}$. The maximum iterations are represented as t_{max} , and the initial value of the current iterations t is 0.

Step 2: Compute the binary solution $P(t)$. $P(t) = (p_1^t, p_2^t, \dots, p_j^t, \dots, p_m^t)$, $p_j^t (j = 1, 2, \dots, m)$ is a binary individual with n -length. The probable solution is obtained by measurement of $Q(t)$. The value of element p_{ji} in p_j is determined by comparing α_{ji} of q_j with w , $0 < w < 1$.

Step 3: Generate the routing path. Assign m individuals into the source nodes at random. We used the state transition rule to generate the routing path of these individuals. In each step of the decision, an individual positioned on node r moves to the node s in line with Equations (4)–(6).

Step 4: Evaluate the solution and store the best solutions in $B(t)$. The evaluation function of the routing tree is shown as follows:

$$f(t) = \frac{1}{[Z_1(t)]^{C_1} [Z_2(t)]^{C_2} [E_r(t)]^{C_3} [\sigma_r(t)]^{C_4}} \quad (20)$$

$$Z_1(t) = \sum Kd_{rs}^\lambda, (r, s) \in Tree(t) \quad (21)$$

$$Z_2(t) = \max Fd_k(t) \quad (22)$$

$$F(t) = \max f(n), (n = 0, 1, \dots, t) \quad (23)$$

where $C_1, C_2, C_3,$ and C_4 are weight parameters, and $E_r(t)$ and $\sigma_r(t)$ are factors which describe the network load balance, and respectively represent the average value and standard deviation of the load for node r . $Z_1(t)$ is the energy consumption factor, K is an array which indicates the total number of leaf nodes extended from each node in the routing tree, λ is a parameter with a value from 2 to 4 which generally approaches 4, d_{rs} is the distance of link (r,s) , $Tree(t)$ denotes the routing tree, $Z_2(t)$ is the time-delay factor, and Fd_k is the distance traveled by the forward ant k .

After the sink node receives forward ant packages, evaluate the solution by Equations (20)–(23), and then save in $B(t)$.

Step 5: Update the pheromone according to the rules of the quantum rotation gate, after receiving back the ant.

Step 6: If the current iterations are less than the maximum iterations, return to Step 3.

It should be noted that QACMOR is an evolutionary algorithm rather than a quantum algorithm, in spite of the fact that the proposed approach is based on quantum computing mechanisms. In QACMOR, some problems in basic ACO can be tackled. The representation of qubit introduces the probability research method, making the balance between exploration and exploitation easier than the conventional ACO algorithm, and adjustment of the magnitude of the rotation angle can make convergence speeds faster. Exploring the unused nodes by using heuristic information, as Formula (4) shows, updating the local pheromone according to Formula (5) and (6) in Step 2, and updating the global pheromone with the quantum rotation gate will generate population diversity, preventing the algorithm from becoming trapped in local convergence or premature stagnation.

5. Experimental Results

5.1. Performance Evaluation

Routing is a crucial process to consider in WSNs when dealing with multiple performance metrics, since routing decisions can impact network lifetime, packet delivery rates, and end-to-end packet delays [29]. Different performance metrics can be used for comparing different routing algorithms in WSNs. The main metrics considered in this paper to validate the performance of the proposed algorithm are as follows:

- (1) General property, such as communication distance, energy consumption, and hops.
- (2) Convergence rate, that is, the number of iterations needed to find an approximation to a fixed point.
- (3) Network lifetime, that is, the duration up to the time when data can no longer be forwarded due to the depletion of energy of the sensor nodes.

Sensor nodes are assigned at random. Figure 5 shows an instance in which the network range was 1000 m^2 , and the total number of nodes was 50. Each link between a node and its accessible neighbors was denoted by a dotted line. Figure 5 shows the optimal path obtained by QACMOR, shown as a solid red line. Source nodes were numbered 16, 21, 22, 24, 30, 47, and 50, and the sink node was numbered 1. Notice that the value of t_{max} should be greater than the number of iterations for the algorithm to converge.

Three groups of experiments were conducted on a MATLAB simulation platform. Table 2 lists the values of parameters used in this simulation.

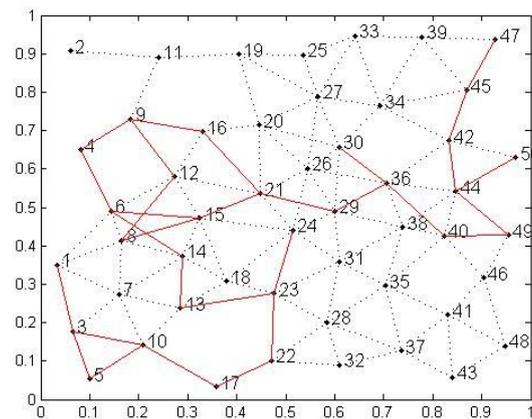


Figure 5. Optimal path in network topology.

Table 2. Parameter-setting in experiments.

Item	Experiment 1	Experiment 2	Experiment 3
Number of nodes	10, 20, 30, . . . , 100	50	50
Network range	1000 m ²	1000 m ²	5000 m ²
Initial energy	/	/	0.5 J
C_1	0.5	0.5	0.5
C_2	0.1	0.1	0.1
C_3	0.1	0.1	0.1
C_4	0.1	0.1	0.1
t_{max}	400	400	400

In the first experiment, a comparison of the value of $F(t)$ in cases in which the number of nodes ranges from 10 to 100 was conducted between two algorithms, that is, BABR and QACMOR. The curve lines in Figure 6 show that the values of $F(t)$ for the two algorithms are same at the beginning, and descend as the number of nodes increases. Compared with BABR, the curve of QACMOR has a more sluggish downtrend. The reason for this is that QACMOR takes more properties into account, including energy efficiency, load balance, and time delay.

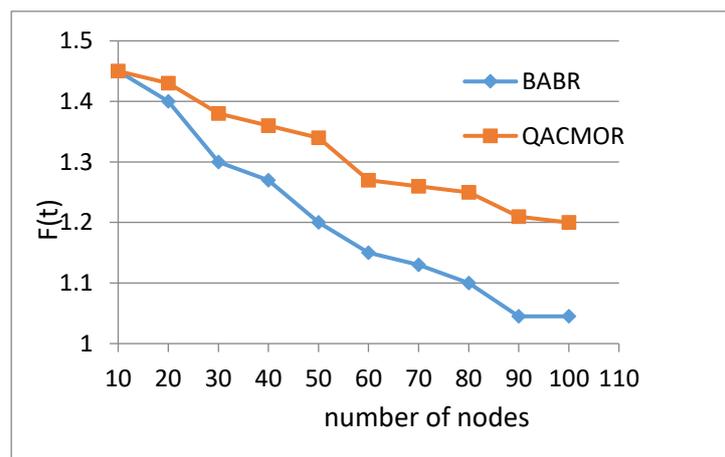


Figure 6. Value of optimal route vs. number of nodes.

The aim of the second experiment was to estimate the convergence property by observing the optimal value $F(t)$ of QACMOR and BABR when the number of iterations grows. As iterations grow, it can be seen in Figure 7 that the value of $F(t)$ tends to be stable. In addition, we notice that QACMOR begins to converge at nearly 200 iterations, while it takes approximately 350 iterations for BABR. This demonstrates that QACMOR has a faster convergence rate than BABR.

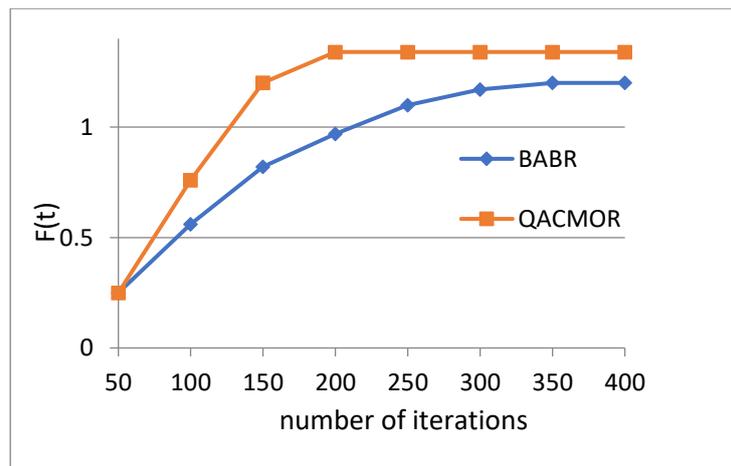


Figure 7. Value of optimal route vs. iterations.

The third experiment evaluates the network lifetime of QACMOR. The experiment is performed on the condition that the number of dead nodes grows. In Figure 8, the x-axis denotes the number of dead nodes, and the y-axis represents the lifetime of the network. It can be seen that the value of lifetime for QACMOR is consistently higher than the same value for BABR, the gap becomes bigger with increasing dead nodes, and maintains a fixed value after 35 nodes, nearly 900 h.

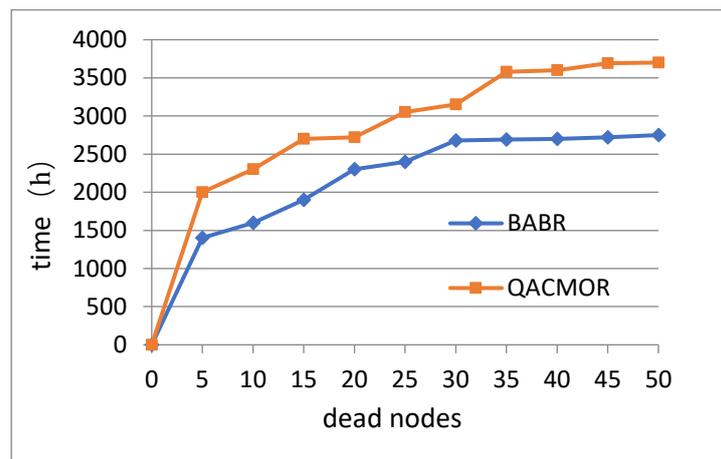


Figure 8. Time vs. number of dead nodes.

The above experimental results indicate that the QACMOR algorithm is capable of use as an efficient and reliable solution for routing, with balanced energy consumption and an improved network lifetime.

5.2. Case Study

In this section, a case study of a maintenance, repair, and overhaul (MRO) system for a steel manufacturing enterprise is illustrated, in order to evaluate the practicality of QACMOR.

Some situations requiring WSNs, such as continuous steel casting lines, present unique characteristics, mainly due to their harsh industrial environment. In the case of a casting line, this is at high temperature and full of powder, dust, and noise. The installation site of the sensor nodes and sink makes it inconvenient to charge or replace the power supply. Therefore, network longevity should be considered. It is important to build routing algorithms which can be adapted to monitor equipment conditions and prolong the WSN lifetime as much as possible. Another major challenge in the harsh environment is insufficient QoS in WSNs, such as delay, bandwidth, and packet loss.

The role of the online monitoring system is to obtain the status information of equipment, including temperature, pressure, and revolving speed. The system architecture is depicted in Figure 9. The complex structure of the continuous casting line made it difficult to install and deploy a reliable cable network, while a WSN had the ability to overcome the field wiring problem. In the WSN, these field data were sent to the Advanced RISC Machines (ARM)-based gateway for data collection, fusion, and processing. Then, the data were sent to the server. At the server, the collected data were imported into a database for further analysis and diagnosis of potential faults by the MRO system.

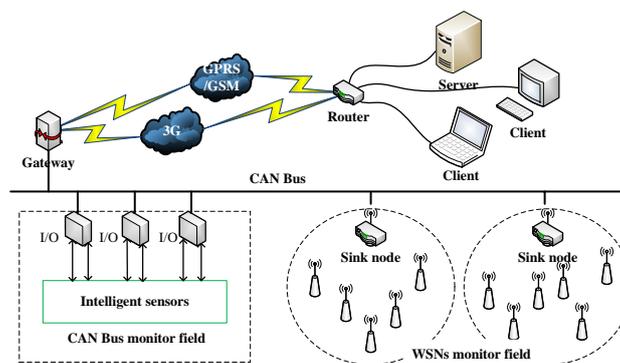


Figure 9. Online monitoring system architecture.

Sensor nodes distributed within the continuous casting line constitute the system's perception layer. Figure 10 shows the installation site of three frame-offset wireless sensors on a segment. In this section, we chose one segment as the test object. Specifically, in one segment, 24 temperature sensors were used to collect information about the working status of the hydro-cylinders; 24 pressure sensors were installed to collect information about the bearings; eight revolving speed sensors were embedded to collect information about the rollers; and three frame-offset wireless sensors were installed onto each segment to monitor the displacement of the segment's frame.

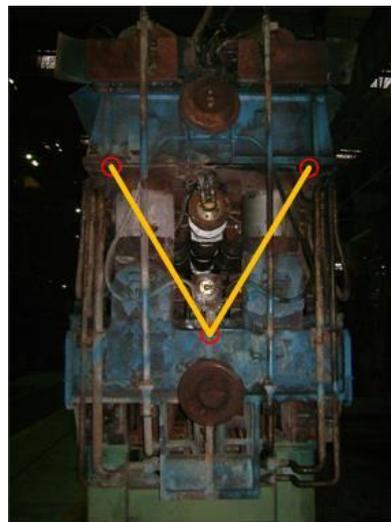


Figure 10. Diagram of three frame-offset wireless sensors.

We conducted a test on one segment in a real casting-shop environment to compare the network lifetime of three algorithms—that is, BABB, AODV, and QACMOR, and verify the running practicality of QACMOR. In this test, the total number of nodes is 60, with one sink node and 59 sensor nodes for one segment. The parameter settings are listed in Table 3, which shows the same weight value (C_1 , C_2 , C_3 , C_4) as that listed in Table 2 in Experiment 3 in Section 5.1. As in that experiment, the comparison of the whole network lifetime was made by observing the number of dead nodes. Results in Figure 11

indicate that in terms of time elapsed before first node death or total network lifetime, QACMOR still has an advantage over BABR, even in harsh working conditions.

Table 3. Parameter-setting in case study.

Item	Value
Number of nodes	60
Network range	300 m × 280 m
Initial energy	0.5 J
C_1	0.5
C_2	0.1
C_3	0.1
C_4	0.1
t_{max}	500

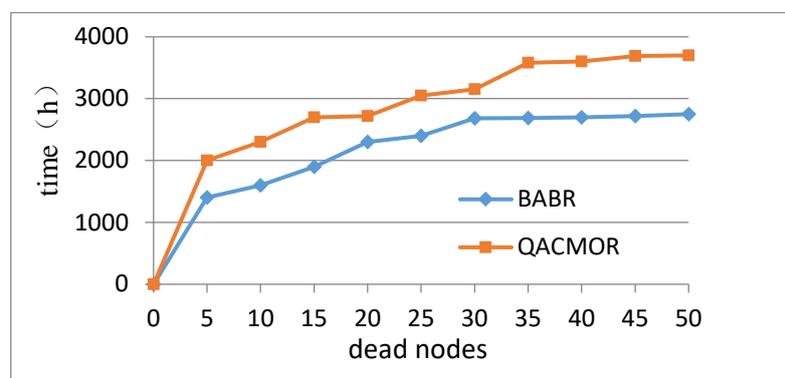


Figure 11. Comparison of network lifetimes in a continuous steel casting line.

6. Conclusions and Future Work

ACO-based routing has been used widely in WSNs. To improve convergence performance and save energy consumption in basic ACO routing methods, quantum computing mechanisms were introduced in the QACMOR method. This paper studied two performance metrics: convergence rate and network lifetime, with reference to the features of industrial continuous steel casting production. Simulation results indicated that the algorithm proposed can rapidly obtain the optimal path with a fast convergence rate, and prolong the network lifetime. A WSN, based on the proposed QACMOR algorithm, was also deployed in an MRO system for a steel manufacturing enterprise. Physical WSN deployment and experiments showed that the proposed QACMOR algorithm is reliable in such applications, after consideration of packet loss based on our previous work [21,30]. In future work, focus and attention should be given to the potential synergies between WSNs and other existing and emerging technologies, such as Cloud Computing and Big Data, so as to improve their overall performance and efficiency.

Author Contributions: F.L. was responsible for methodology, validation and original draft writing, and M.L. guided the whole process, and also reviewed and edited the manuscript, and G.X. worked in software and experimental data collection.

Funding: This research was funded by the Scientific Research Projects of the National Natural Science Foundation of China (Grant no. 61573257), Hangzhou Municipal Science and Technology Bureau of Social Development and Scientific Research Projects (No. 20150533B16), and by the Scientific Research Project of Zhejiang Education Department (No. Y201432791).

Acknowledgments: The research work is partially supported by Hangzhou Key Laboratory for IoT Technology & Application, and Zhejiang Engineering Laboratory Intelligent Plant Factory.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Xie, X.; Huang, Y.; Niu, C.; Li, C.; Wang, H. Distributed data mining based on deep neural network for wireless sensor network. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 157453–157457.
2. Li, X.; Li, D.; Wan, J.; Vasilakos, A.V.; Lai, C.-F.; Wang, S. A review of industrial wireless networks in the context of Industry 4.0. *Wirel. Netw.* **2017**, *23*, 1–19. [[CrossRef](#)]
3. Distefano, S.; Banerjee, N.; Puliafito, A. Smart objects, infrastructures, and services in the internet of things. *Int. J. Distrib. Sens. Netw.* **2016**, *12*, 8642512. [[CrossRef](#)]
4. Liu, K.; Lee, V.C.S.; Ng, J.K.-Y.; Chen, J.; Son, S.H. Temporal data dissemination in vehicular cyber physical systems. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 2419–2431. [[CrossRef](#)]
5. Çelik, F.; Zengin, A.; Tuncel, S. A survey on swarm intelligence based routing protocols in wireless sensor networks. *Int. J. Phys. Sci.* **2010**, *5*, 2118–2126.
6. Kang, Q.; Zhou, M.; An, J.; Wu, Q. Swarm intelligence approaches to optimal power flow problem with distributed generator failures in power networks. *IEEE Trans. Autom. Sci. Eng.* **2013**, *10*, 343–353. [[CrossRef](#)]
7. Pan, Q.-K. An effective co-evolutionary artificial bee colony algorithm for steelmaking-continuous casting scheduling. *Eur. J. Oper. Res.* **2016**, *250*, 702–714. [[CrossRef](#)]
8. Zaheeruddin, D.K.L.; Pathak, A. Energy-aware bee colony approach to extend lifespan of wireless sensor network. *Aust. J. Multi-Discip. Eng.* **2017**, *13*, 1–18.
9. Liu, X. Routing protocols based on ant colony optimization in wireless sensor networks: a survey. *IEEE Access* **2017**, *5*, 26303–26317. [[CrossRef](#)]
10. Han, K.H.; Kim, J.H. Quantum-inspired evolutionary algorithms with a new termination criterion, H/sub /spl epsi// gate, and two-phase scheme. *IEEE Trans. Evol. Comput.* **2004**, *8*, 156–169. [[CrossRef](#)]
11. Kim, J.-H.; Han, J.-H.; Kim, Y.-H.; Choi, S.-H.; Kim, E.-S. Preference-based solution selection algorithm for evolutionary multiobjective optimization. *IEEE Trans. Evol. Comput.* **2012**, *16*, 20–34. [[CrossRef](#)]
12. Albano, M.; Chessa, S.; Nidito, F.; Pelagatti, S. Dealing with nonuniformity in data centric storage for wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **2010**, *22*, 1398–1406. [[CrossRef](#)]
13. Chakchouk, N.; Hamdaoui, B.; Frikha, M. WCDS-DCR: an energy-efficient data-centric routing scheme for wireless sensor networks. *Wirel. Commun. Mob. Comput.* **2012**, *12*, 195–205. [[CrossRef](#)]
14. Lu, H.; Li, J.; Guizani, M. Secure and Efficient data transmission for cluster-based wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 750–761.
15. Salim, A.; Osamy, W.; Khedr, A.M. IBLEACH: intra-balanced LEACH protocol for wireless sensor networks. *Wirel. Networks* **2014**, *20*, 1515–1525. [[CrossRef](#)]
16. Zhang, H.; Shen, H. Energy-efficient beaconless geographic routing in wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **2010**, *21*, 881–896. [[CrossRef](#)]
17. Ghaffari, A.; Rahmani, A.M.; Khademzadeh, A. Energy-efficient and QoS-aware geographic routing protocol for wireless sensor networks. *IEICE Electron. Express* **2011**, *8*, 582–588. [[CrossRef](#)]
18. Cheng, L.; Niu, J.; Cao, J.; Das, S.K.; Gu, Y. QoS aware geographic opportunistic routing in wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 1864–1875. [[CrossRef](#)]
19. Liu, M.; Xu, S.; Sun, S. An agent-assisted QoS-based routing algorithm for wireless sensor networks. *J. Netw. Comput. Appl.* **2012**, *35*, 29–36. [[CrossRef](#)]
20. Zungeru, A.M.; Ang, L.-M.; Seng, K.P. Classical and swarm intelligence based routing protocols for wireless sensor networks: A survey and comparison. *J. Netw. Comput. Appl.* **2012**, *35*, 1508–1536. [[CrossRef](#)]
21. Wang, J.; Cao, Y.; Li, B.; Kim, H.-J.; Lee, S. Particle swarm optimization based clustering algorithm with mobile sink for WSNs. *Futur. Gener. Comput. Syst.* **2017**, *76*, 452–457. [[CrossRef](#)]
22. Saleem, M.; Ullah, I.; Farooq, M. BeeSensor: An energy-efficient and scalable routing protocol for wireless sensor networks. *Inf. Sci.* **2012**, *200*, 38–56. [[CrossRef](#)]
23. Camilo, T.; Carreto, C.; Silva, J.S.; Boavida, F. *An Energy-Efficient Ant-Based Routing Algorithm for Wireless Sensor Networks*; Springer Science and Business Media LLC: Berlin, Germany, 2006; Volume 4150, pp. 49–59.
24. Zungeru, A.M.; Seng, K.P.; Ang, L.-M.; Chia, W.C. Energy efficiency performance improvements for ant-based routing algorithm in wireless sensor networks. *J. Sensors* **2013**, *2013*, 1–17. [[CrossRef](#)]
25. Saleh, A.M.S.; Ali, B.M.; Mohamad, H.; Rasid, M.F.A.; Ismail, A. RRSEB: A reliable routing scheme for energy-balancing using a self-adaptive method in wireless sensor networks. *TIIS* **2013**, *7*, 1585–1609.

26. Xing, H.; Ji, Y.; Bai, L.; Liu, X.; Qu, Z.; Wang, X. An adaptive-evolution-based quantum-inspired evolutionary algorithm for QoS multicasting in IP/DWDM networks. *Comput. Commun.* **2009**, *32*, 1086–1094. [[CrossRef](#)]
27. Heinzelman, W.R.; Chandrakasan, A.; Balakrishnan, H. Energy-efficient communication protocol for wireless microsensor networks. In Proceedings of the 33rd annual Hawaii international conference on system sciences, Maui, HI, USA, 7 January 2000.
28. Wang, L.; Niu, Q.; Fei, M. A novel quantum ant colony optimization algorithm. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2007; Volume 4688, pp. 277–286.
29. Cobo, L.; Quintero, A.; Pierre, S. Ant-based routing for wireless multimedia sensor networks using multiple qos metrics. *Comput. Netw.* **2007**, *54*, 2991–3010. [[CrossRef](#)]
30. Zhang, F.; Liu, M.; Zhou, Z.; Shen, W. An IoT-based online monitoring system for continuous steel casting. *IEEE Internet Things J.* **2016**, *3*, 1355–1363. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).