

RESEARCH

Open Access

A hybrid multi-path routing algorithm for industrial wireless mesh networks

Yun Zuo^{1,2}, Zhihao Ling^{1,2*} and Yifeng Yuan^{1,2}

Abstract

Multi-path routing, a routing technique that enables data transmission over multiple paths, is an effective strategy in achieving reliability in wireless sensor networks. However, multi-path routing does not guarantee deterministic transmission. This is because more than one path is available for transferring data from the source node to the destination node. A hybrid multi-path routing algorithm is proposed for industrial wireless mesh networks for improving reliability and determinacy of data transmission, as well as to effectively deal with link failures. The proposed algorithm adopts the enhanced Dijkstra's algorithm for searching the shortest route from the gateway to each end node for first route setup. A virtual pheromone distinct from the regular pheromone is introduced to realize pheromone diffusion and updating. In this way, multiple routes are searched based on the ant colony optimization algorithm. The routes used for data transmission are selected based on their regular pheromone values, facilitating the delivery of data through better routes. Link failures are then handled using route maintenance mechanism. Simulation results demonstrate that the proposed algorithm outperforms traditional algorithms in terms of average end-to-end delay, packet delivery ratio, and routing overhead; moreover, it has a strong capacity to cope with topological changes, thereby making it more suitable for industrial wireless mesh networks.

Keywords: Multi-path routing, Industrial wireless mesh networks, Enhanced Dijkstra's algorithm, Ant colony optimization algorithm

1. Introduction

In recent years, wireless technology has gained great interest in industrial applications. With harsh industrial control environment, however, wireless networks are required to comply with several requirements pertaining to reliable communication. Missing or delaying the process data may severely degrade the quality of control; thus at least two paths should be available from source to destination nodes in the wireless sensor networks (WSNs). WirelessHART is a well-known wireless mesh communication standard that is specifically designed for process measurement and control applications. It requires multi-path routing algorithms to provide reliable and deterministic communication. This standard has been approved by the HART Communication Foundation in September 2007 [1].

Numerous routing algorithms have been proposed for WSNs in the past. Routing algorithm can be divided into three distinct categories, namely, proactive algorithm (including destination sequenced distance vector) [2], reactive algorithm (including ad hoc on-demand distance vector routing) [3], and hybrid algorithm (including zone routing protocol) [4]. In proactive algorithm, nodes attempt to maintain routes to all other nodes at all times, while keeping track of all topological changes. However, this scenario becomes difficult when multiple nodes are present or when the nodes are mobile. In reactive algorithm, nodes only gather routing information when a data session to a new destination starts or when a route in use fails. In general, reactive algorithms are more scalable because they greatly reduce routing overhead. However, such algorithms may suffer from oscillations in performance because they are unprepared for disruptive events. Consequently, according to actual applications, the hybrid algorithm combines the best features of the proactive and reactive algorithms.

* Correspondence: zhling@ecust.edu.cn

¹School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237, China

²Key Laboratory of Advanced Control and Optimization for Chemical Process, (East China University of Science and Technology), Ministry of Education, Shanghai 200237, China

Multi-path routing schemes, which are developed based on the three categories of routing algorithms, can search multiple paths through a single-route discovery process. A source node chooses a path among multiple paths, after which it transfers data packets using the chosen path. If a link on the path is broken, the source node chooses another path and continues to transfer data instead of initiating an additional discovery process. New route discovery process is initiated after all paths have failed, thereby reducing the number of flooding initiations. Recently, several multi-path routing schemes have been proposed, which include multi-path dynamic source routing [5], split multi-path routing [6], and ad hoc on-demand multi-path distance vector (AOMDV) [7].

The abovementioned routing algorithms are known as the shortest path algorithms. Recently, many researchers have considered quality of service (QoS) in routing algorithms for WSNs, which magnifies the difficulties associated with some optimization goals. Gelenbe and Lent [8] proposed an energy-efficient routing algorithm for ad hoc networks based on the cognitive packet network (CPN) [9,10] protocol. The CPN algorithm is a fast, adaptive routing algorithm that uses smart packets for QoS-based path finding to minimize power consumption. QoS-aware multi-path routing algorithms have also been proposed in [11–13], in which QoS routing problems are formulated as different optimization problems (e.g., video quality and energy awareness). However, meeting QoS requirements in WSNs introduces certain overhead to routing protocols in terms of energy consumption, intensive computations, and significantly large storage. Thus, the energy conservation in industrial wireless mesh networks is achieved by the time division multiple access scheduling algorithm. The proposed routing algorithm aims to gain reliable and low-delay communications without high flooding.

In this article, a hybrid multi-path routing algorithm called DAWMNet is proposed for industrial wireless mesh networks, which combines Dijkstra's algorithm and the ant colony optimization (ACO) algorithm. Dijkstra's algorithm achieves the first route setup, whereas the ACO algorithm is responsible for route exploration and maintenance. DAWMNet aims to achieve load balance as well as reliable and deterministic transmission with minimal flooding problem. Simulation results demonstrate that the proposed algorithm outperforms the traditional approaches in terms of average end-to-end delay, packet delivery ratio, and routing overhead in number of packets. Moreover, the proposed algorithm has a strong capacity to cope with topological changes.

The remainder of this article is organized as follows. The network model and some related routing schemes are briefly introduced in Section 2. The proposed algorithm,

DAWMNet, is established in Section 3. The performance evaluation of the proposed algorithm is presented in Section 4. The conclusions and future work are presented in Section 5.

2. Network model and challenges

2.1. WirelessHART technology

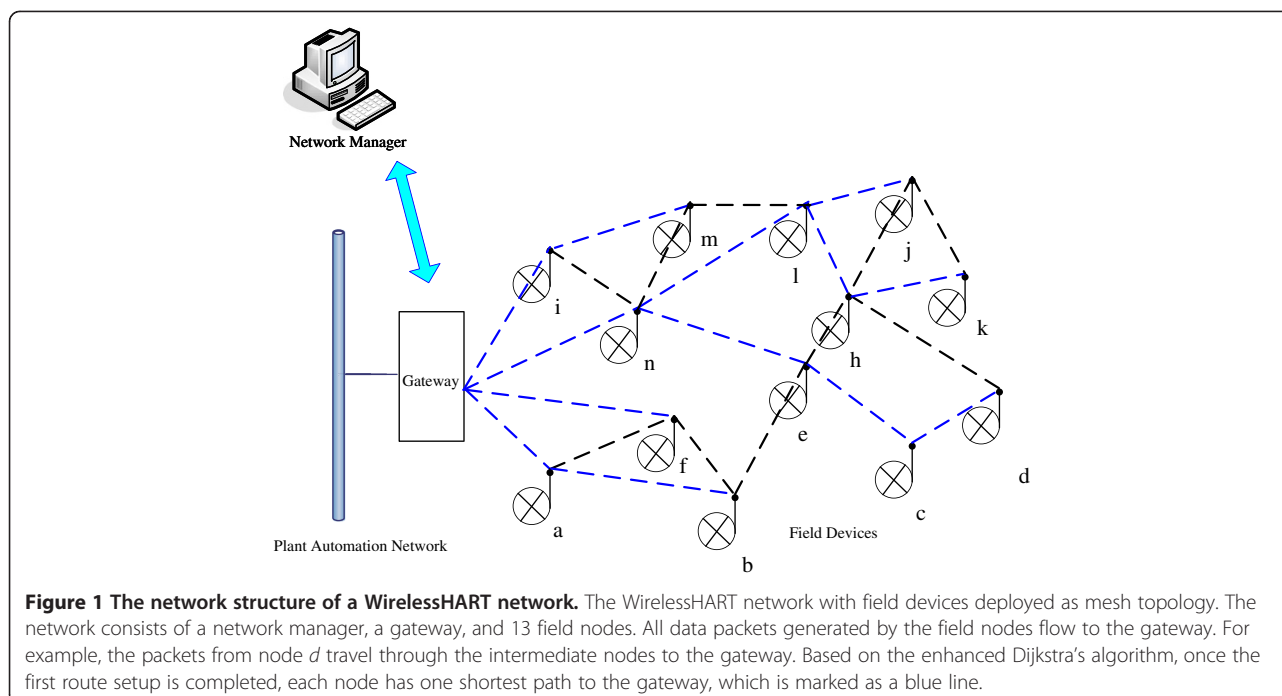
WirelessHART is a secure and robust mesh networking technology, which operates in the 2.4-GHz ISM radio band. It utilizes IEEE 802.15.4 compatible direct-sequence spread spectrum radios with channel hopping on a packet via packet basis. WirelessHART is a user-friendly, reliable, and interoperable wireless mesh sensor protocol used for process measurement and control applications [14]. Figure 1 shows the architecture of the network, which contains the three principal elements listed below.

- (1) WirelessHART field devices are connected to the process or plant equipment.
- (2) The WirelessHART gateway acts as a bridge connecting the network manager to field devices and can also convert one protocol to another. One or more gateways should be present in a WirelessHART network.
- (3) The WirelessHART network manager is the "brain" of the whole network, which is responsible for configuring the network, scheduling communications between nodes, managing routing tables, and monitoring the health status of network. Redundant network managers are supported; however, only a single active network manager is required per WirelessHART network.

2.2. Routing schemes in WirelessHART

The two methods for routing packets in WirelessHART networks are graph routing and source routing. Graph routing requires the network manager to pre-configure the routing table for each potential destination. For redundancy, in a properly configured network, all nodes have at least two paths to destination nodes. When using graphs, nodes only obtain the information about their next-hop node not the global information. In comparison, source routing requires information on network topology, because packets may go to an arbitrary destination without an explicit configuration of intermediate nodes. Graph routing is ideal for both upstream and downstream communications, whereas source routing is only intended for network diagnostics.

However, given that WirelessHART provides little guidance on route discovery and route establishment, this article focuses on the improvements of graph routing to meet demands of industrial applications.



2.3. Related routing protocols

Several solutions to implement multi-path routing scheme in WSNs have been proposed, and some of which are used in this study.

2.3.1. Dijkstra's algorithm

Dijkstra's algorithm [15], first conceived by Edsger Dijkstra in 1956 and published in 1959, is a graph search algorithm that solves the single-source shortest path problem for a graph with non-negative edge path costs, thereby producing a shortest path tree. For a given source vertex in the graph, the algorithm finds the path with the lowest cost (e.g., the shortest path among all vertices) to each other vertices. This algorithm can also be used for finding the shortest path from a single vertex to a single destination vertex by stopping the algorithm once the shortest path to the destination vertex has been determined. Therefore, Dijkstra's algorithm is widely applied in routing algorithms, and is utilized as a subroutine in other graph algorithms to find the shortest routes from one node to all other nodes.

2.3.2. ACO

ACO [16–19] takes inspiration from the foraging behavior of some ant species. These ants mark favorable paths by depositing pheromones on the ground; in this way, other members of the colony can trace the path by simply following the scent of the pheromone. ACO exploits a similar mechanism for solving optimization problems.

ACO is an iterative algorithm in which a number of artificial ants are used at each iteration. Each ant builds a

solution by walking from vertex to vertex on the graph and by avoiding any vertex that has already been visited during its walk. At each step of the solution construction, an ant selects a vertex according to a stochastic mechanism biased by the pheromone: when in vertex *i*, the next vertex is selected stochastically among the previously unvisited ones. In particular, if vertex *j* has not previously been visited, it can be selected with a probability that is proportional to the pheromone associated with edge (*i*, *j*). At the end of the iteration, the pheromone values are modified based on the quality of the solutions constructed by the ants. This is done to encourage ants in future iterations to construct solutions similar to the best ones previously constructed. After completing the iteration, the pheromone values are updated by all the ants that built solutions in the iteration. During the construction of a solution, ants select the next vertex to be visited based on the stochastic mechanism selected.

Numerous routing algorithms based on ants, such as ant-based dynamic routing algorithm [17] and ant routing algorithm (ARA) [18], have been applied to WSN, especially in mobile ad hoc networks (MANETs). These algorithms utilize active path sampling, are robust to agent failures, have multi-path routing capability, and have the ability to resolve data load spreading. However, these algorithms crucially rely on repeated path sampling, thereby leading to significant overhead. AntHocNet [19] is an ACO routing algorithm for MANETs, applies several improvements based on these ant-based algorithms mentioned above. It combines the reactive path setup phase with the proactive path

maintenance, and also makes use of a local repair mechanism.

2.4. Challenges

Nodes in industrial wireless mesh networks typically have low power battery, low memory; moreover, the routing tables of these nodes expand with the network length, and the nodes have no diffusion communication support [20,21]. Routing algorithms for industrial applications must meet the high demands of reliable and deterministic communication. Although Dijkstra's algorithm can find the shortest path from source to destination nodes, much time is spent searching for multiple routes for each node, particularly in large-scale network. Moreover, it has no mechanism to cope with topological changes. Although ant-based protocols can provide multi-path routing and deal with link failures, the continuous generation of small ant packets for gathering routing information can easily lead to excessive overhead; in addition, data may be transmitted along the relatively long paths in the initial stage. Therefore, in this article, a hybrid multi-path routing algorithm that combines the benefits of Dijkstra's algorithm and the ACO algorithm is proposed for industrial wireless mesh networks. This study, which is inspired by the concept of pheromone diffusion in AntHocNet, presents further improvements on route setup and route maintenance mechanisms to achieve reliable, deterministic communication as well as load balance, thus meeting the demands of industrial wireless mesh networks.

3. The hybrid multi-path routing algorithm

The hybrid multi-path routing algorithm for industrial wireless mesh network, hereby known as DAWMNet, combines the enhanced Dijkstra's algorithm and ACO algorithm. The enhanced Dijkstra's algorithm achieves first route setup, whereas the ACO algorithm is responsible for route exploration and maintenance. After the initialization of a network, the network manager applies the enhanced Dijkstra's algorithm to calculate the shortest path to each end node. Afterwards, the remaining routes are searched by using ACO in the route exploration stage, which is executed concurrently with data transmission. The route maintenance scheme can respond effectively to some topological changes in a timely manner. The hybrid architecture not only improves efficiency in route setup, it also makes efforts on redundant routes exploration and link failure maintenance. Routing information is stored in the routing tables and data transmission is directed by the information from these routing tables.

3.1. Routing tables

In industrial wireless mesh networks, all nodes maintain a series of tables that control the communications performed by the nodes. These tables collect statistics on

communications, such as neighbor table and graph table. In the proposed algorithm, pheromone tables are added as extension tables for all nodes.

- (1) *Neighbor table*. This refers to a list of nodes that a node is able to communicate with directly. The neighbor table of each node is populated after the initial period of a network.
- (2) *Graph table*. Graphs in networks are used to route messages from their source to their destination nodes. Each node does not know the entire route, but only knows the next hop destination legal for propagating a packet. Data transmission is based on the route with graph ID listed in graph table. The regular pheromones of routes in graph table are used as the index for route selection.
- (3) *Pheromone table*. This records pheromone information about routes from node i to the gateway g through intermediate node j . The pheromone table includes information of the intermediate nodes on this route and a regular pheromone value R_{ij}^g or a virtual pheromone value V_{ij}^g . Each route cannot have both a regular pheromone value and a virtual pheromone value simultaneously. The regular pheromone is utilized to estimate the goodness of routes and to determine which route can be used for data transmission. The virtual pheromone indicates the routes that can possibly relay the data, and then guides ants to sample these possible routes in networks [22]. The virtual pheromone is obtained through information bootstrapping using the goodness values reported by neighbor nodes during route exploration. In route exploration and maintenance, the bootstrapped value is stored temporarily in pheromone table. The goodness of route is expressed as the inverse of the cost of a route, which depends on metrics. In this study, the shortest routes are preferred, thus the distance between two nodes is an important metric. The cost of a route is calculated using the following equation:

$$c_i^j = d_i^j \quad (1)$$

where c_i^j is the cost of route from nodes i to j , d_i^j is the distance between nodes i and j .

3.2. First route setup by enhanced Dijkstra's algorithm

In networks, all end nodes must send field data to the gateway. The gateway then exchanges management messages with all the end nodes. After the initialization of the network, the enhanced Dijkstra's algorithm is adopted for first route setup with low calculation complexity and low overhead. The operation is executed by the network manager rather than allowing each node to launch its own

search process. In addition, the enhanced Dijkstra's algorithm utilized Fibonacci heap [23] to implement Dijkstra's algorithm to increase searching efficiency.

An industrial wireless mesh network can be represented as a simple connected graph, $G = (V, E)$, where V and E denote the set vertices and the set of edges, respectively. The simple weight of each edge is represented by the distance between two nodes, as the length of the edge. g indicates the gateway in networks, and $d(v)$ is the distance between g and node v . The enhanced Dijkstra's algorithm is presented in subsection The enhanced Dijkstra's algorithm, it works as follows:

During algorithm operation, each vertex is either in unlabeled or labeled state. Initially, $d(g) = 0$ and $d(v) = \infty$, the states of all vertices are unlabeled (lines 1–4). The Fibonacci heap stores all the vertices, and $d(v)$ is the key of each vertex (line 5). The vertex v with the minimum key is extracted from the heap, and the state of v is converted into labeled (lines 6–8). For each edge (v, u) , if the state of u is unlabeled and $d(v) + d(v, u)$ is smaller than $d(u)$, then $d(u)$ is replaced by $d(v) + d(v, u)$ and the key of u is modified accordingly (lines 9–14). This algorithm is performed repeatedly until the heap is empty. In this way, the shortest routes from the gateway to each node are discovered.

3.2.1. The enhanced Dijkstra's algorithm

Enhanced Dijkstra(G, g, v)

1. for each vertex v in G
2. distance[v] = infinite; parent[v] = NULL;
state[v] = unlabeled;
3. for gateway g in G
4. distance[g] = 0;
5. insert all the vertexes into distance Fibonacci heap Q ;
6. while (! $Q.empty()$)
7. $v = \text{Extract-Min}(Q)$;
8. state[v] = labeled;
9. for each edge (v, u)
10. if (state[u] = unlabeled)
11. if (distance[u] > distance[v] + distance[v, u])
12. distance[u] = distance[v] + distance[v, u];
13. parent[u] = v ;
14. Decrease-Key(u)

Once the first shortest route from node i to the gateway is found, the pheromone tables of all nodes along this route and the graph table of node i are populated. The pheromone on this route is a regular pheromone R_{ij}^g , and this route is added to graph table for data transmission.

$$R_{ij}^g \leftarrow \gamma R_{ij}^g + (1 - \gamma)(c_i^g)^{-1}, \quad \gamma \in [0, 1] \quad (2)$$

where γ is a parameter regulating the speed of adaptation of the pheromone values to new values, which is in accordance

with the ant sampling in route exploration stage; γ is maintained at $0.7j$ indicates the intermediate node on this route; and c_i^g is defined as shown in Equation (1).

When the search process is completed, the network manager holds the information of the shortest routes for each node. Afterwards, the network manager sends the related routing information to the nodes in order to populate their respective graph tables. From the information in the graph tables, the nodes can identify the graph ID, the destination node, and the next-hop node. This method is called target direction query. Here, if a person asks for directions to a particular place, then other person will give him some directions; then, at the next crossing, this person will ask another person for help. This method decreases the memory capacity of each node, thus avoiding redundant multi-hop search process. In Figure 1, the routes that contain regular pheromones are marked as blue lines. Each node has one shortest route to the gateway for sending data packets.

3.3. Route exploration by ACO

Multiple redundant routes for data transmission in networks are realized in this stage. This process extends available routing information to a mesh of multiple routes on the basis of first route setup period. Route exploration consists of two sub-processes, namely, pheromone diffusion and pheromone updating. Pheromone diffusion aims to spread the available pheromone information over the network by using periodic "keep-alive" messages. Pheromone updating aims to control and update pheromone information through route sampling by using ants.

In this study, pheromone diffusion proceeds as soon as the shortest routes are discovered; it is then terminated when the regular pheromone information of all routes is populated through the process of pheromone updating. This process forms a cycle of route exploration, and the next cycle is triggered by topological changes or the timer.

3.3.1. "Keep-alive" message

In industrial wireless mesh networks, each node exchanges its information with neighbors using "keep-alive" messages. The original intention of this message is to probe quiescent links, discover new neighbors, and maintain time synchronization. "Keep-alive" messages should be sent no more than once per minute to avoid power consumption and network flooding due to frequent messages.

In this study, "keep-alive" message is expanded to contain the routing information for pheromone diffusion. This message plays a key role during pheromone diffusion, and as such, some properties need adjustment. In the route exploration period, once a node obtains new information of routes to the gateway, the node sends "keep-alive" messages to its neighbors. The time interval

of the transmission of these messages must be more than 1 s. Each message carries information about five routes to the gateway at most. As this step stops, the properties of “keep-alive” message are reset to their default configuration.

3.3.2. Pheromone diffusion

Pheromone diffusion spreads out existing regular pheromone information, so that all the pheromone information pointing towards the gateway becomes available in the network. In this way, multi-path routing is achieved.

During the process of pheromone diffusion, when node i receives a “keep-alive” message containing new route information from node j , then node i can derive the reported pheromone value w_j^g , whose definition is provided in Equation (3). This reported pheromone value indicates the goodness of the route from j to the gateway g and the locally maintained estimate of the cost c_i^j equals to the distance between nodes i and j . The resultant value of these two factors is called bootstrapped value B_{ij}^g , which is then used to update virtual pheromones or regular pheromones [24]. The exact formula of the bootstrapped value is given in Equation (4).

$$w_j^g = R_{jk}^g \cdot \frac{R_{jk}^g}{\max_{k \in N_j} (R_{jk}^g)} \quad (3)$$

$$B_{ij}^g = \left[(w_j^g)^{-1} + (1 - \gamma)^{-1} c_i^j \right]^{-1} \quad (4)$$

where N_j is the set of the one-hop neighbors of node j , these neighbors provide pheromone information through “keep-alive” messages. The $(1 - \gamma)$ parameter is in accordance with the definition of regular pheromone in Equation (2).

In Figure 1, the first shortest route from node c to the gateway g is marked as a blue dotted line. Along this route, all nodes have regular pheromones in pheromone tables, such as node b . In the process of pheromone diffusion, node e may receive a “keep-alive” message from node b . This message informs node e that it may reach the gateway over node a . Then, node e derives the bootstrapped value of the route over node b to the gateway.

3.3.3. Pheromone updating

When a node receives a “keep-alive” message about route pheromone, the derived bootstrapped value initiates the pheromone updating of routes. There are two possibilities cases about pheromone updating: routes without regular pheromone and routes with regular pheromone. Corresponding to these two cases, the bootstrapped value can be used for route exploration and route updating, respectively.

Route exploration If only a bootstrapped value B_{ij}^g is present instead of a regular pheromone value in the pheromone table of node i , B_{ij}^g indicates a possible new route. However, this new route has never been sampled completely, so it is potentially unreliable. Thus, node i updates its pheromone information as follows: $V_{ij}^g = B_{ij}^g$. Once a virtual pheromone is assigned to a route, proactive forward ants are sent out by the source node, which is a typical mode of operation in the ACO algorithm.

Proactive forward ants are unicast rather than broadcast toward the destination. During the advancing process, a proactive forward ant takes a new routing decision at the intermediate node m using the probability P_{im}^g as defined in Equation (5). There are regular pheromones or virtual pheromones on routes from node i to the gateway. Thus, P_{im}^g is determined by the actual situation of route pheromone. That a proactive ant arrives at a node which provides no routing information about the gateway suggests the route does not exist. Afterward, the forward ant is discarded. Conversely, the network manager records the route taken by the proactive forward ant once it arrives at the gateway. In this way, the virtual pheromone is investigated, and the regular pheromone is assigned on this route. Thus, other routes from i to g which contain regular pheromone may be involved in pheromone updating. As the number of routes being sampled increases, the available routes for data transmission also increases to a full mesh.

$$P_{im}^g = \frac{(\psi_{im}^g)^{\beta_1}}{\sum_{j \in N_i} (\psi_{ij}^g)^{\beta_1}}, \psi \in \begin{cases} R & \text{regular} \\ V & \text{virtual} \end{cases} \quad (5)$$

where P_{im}^g is the likelihood of choosing the next-hop node m to the gateway during the advancing process; ψ responds to a regular or virtual pheromone value, but never to both simultaneously; N_i is the set of one-hop neighbors of node i to the gateway, which states pheromone information using “keep-alive” message; and β_1 is a protocol configuration parameter used for new route exploration. By adapting the β_1 parameter, ant forwarding can be less or greedier with respect to the best available routes. By setting β_1 at a high value, ants concentrate on the best routes. By setting β_1 at a low value, ants spread over multiple routes, while considering the low-quality routes at the same time. By utilizing multiple routes in ant forwarding, the throughput is improved because the load is spread more evenly over the available network resources. In this study, β_1 is kept at a relatively low value of 5, which is set for a quick exploration of all the routes in the network. This approach to choose the next-hop node during the advancing process is similar to the sensible routing policies analyzed in [25].

Route updating If node i has a regular pheromone value in its pheromone table for the route from i to the gateway through j , then this route may have been sampled by ants in the past or is considered the shortest route. Therefore, this route is reliable. B_{ij}^g is treated as an update of the goodness estimate of this reliable route and is used to replace R_{ij}^g when B_{ij}^g is better than

R_{ij}^g . In this way, pheromones on current routes are kept up-to-date. At the same time, the graph tables need to be updated as well. Pheromone updating does not last as long as the whole network lifetime. A loop is terminated when all routes in the network have regular pheromones.

The flow chart of DAWMNet is showed in Figure 2.

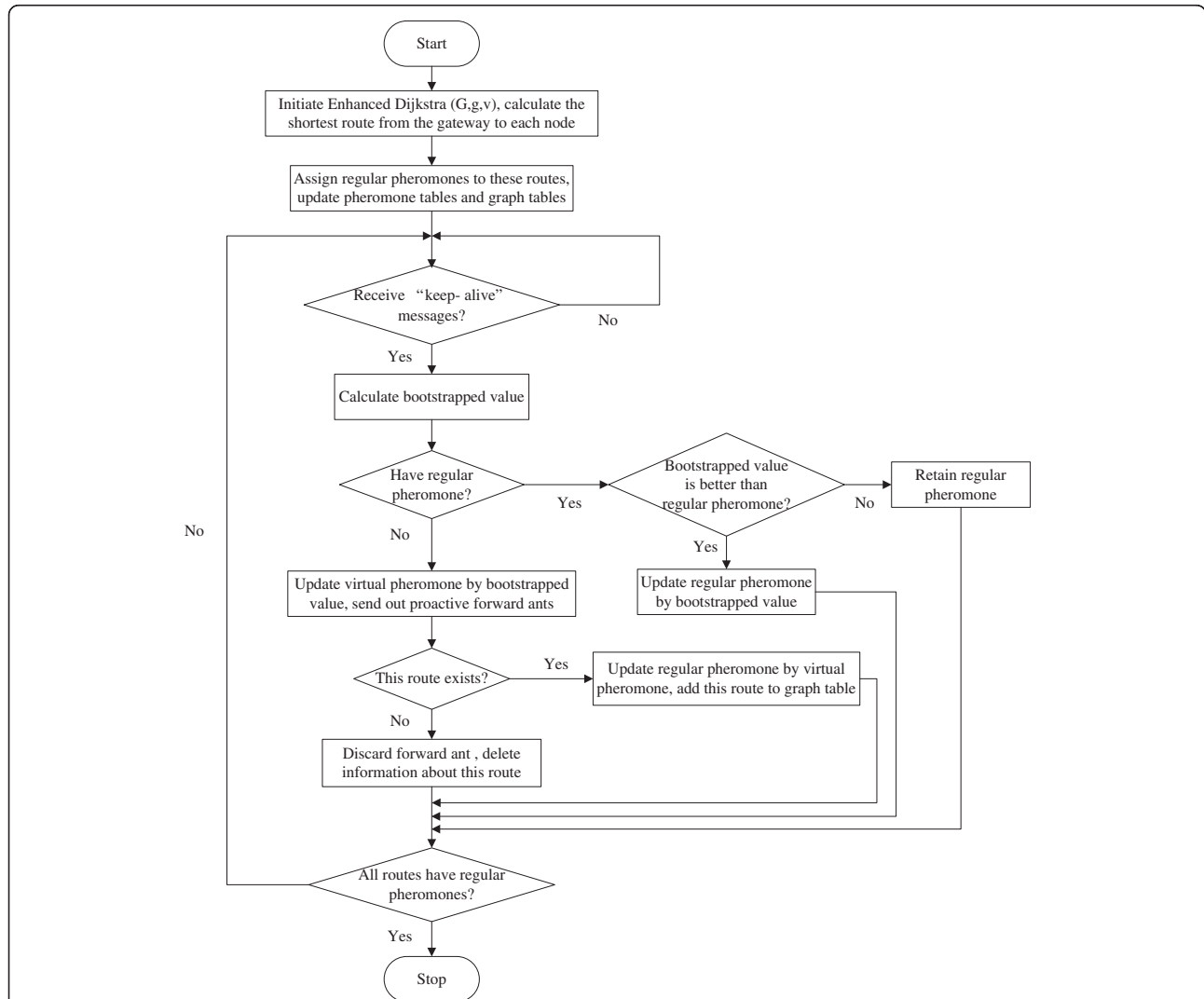
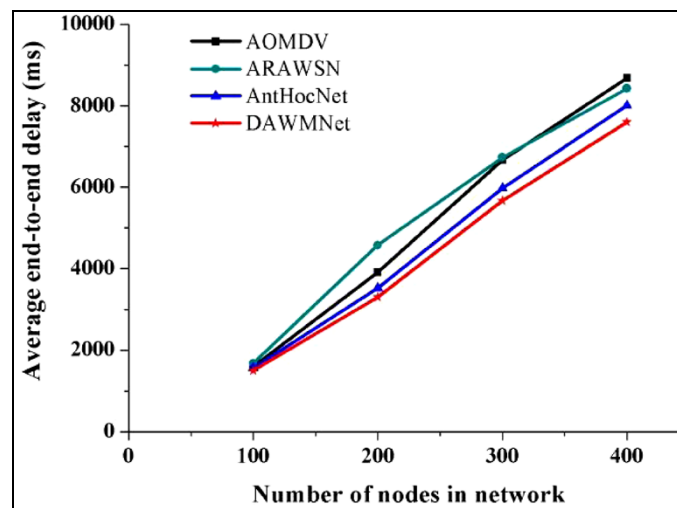
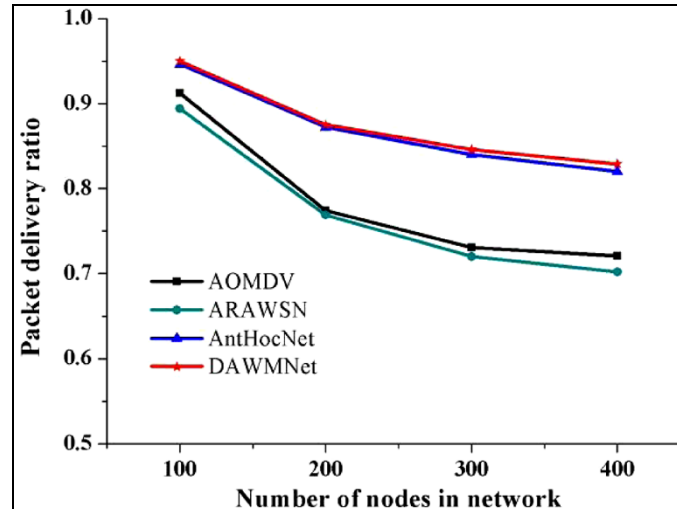


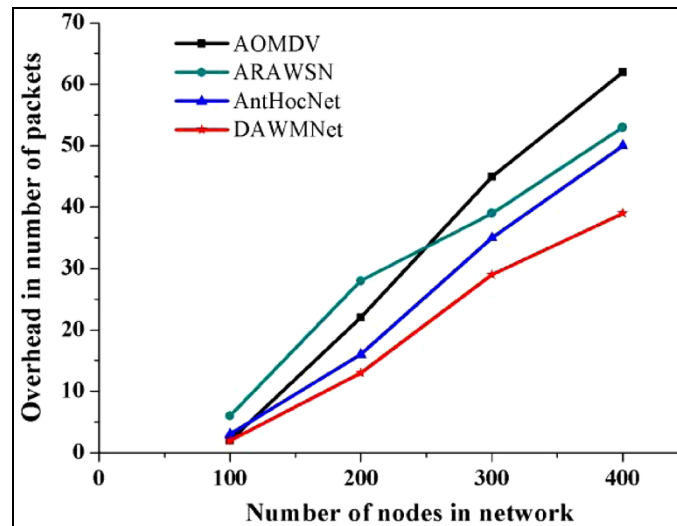
Figure 2 The flowchart of the hybrid multi-path routing algorithm DAWMNet. After the initialization of a network, the network manager adopts the enhanced Dijkstra's algorithm to calculate the shortest path to each end node. Once the first shortest route of each node is found, the pheromone tables of all nodes along this route and graph table of this node are populated. Route exploration consists of two sub-processes, namely, pheromone diffusion and pheromone updating. During the process of pheromone diffusion, a node can derive the bootstrapped value. If only a bootstrapped value is present instead of a regular pheromone value for a route, the bootstrapped value is converted to a virtual pheromone. Once a virtual pheromone is assigned to a route, proactive forward ants are sent out to sample this route. That a proactive ant arrives at a node that provides no routing information about the gateway suggests the route does not exist. Afterward, the forward ant is discarded. Conversely, the network manager records the route taken by the proactive forward ant once it arrives at the gateway. In this way, the regular pheromone is assigned on this route. If there is a regular pheromone value for a route, the bootstrapped value is treated as an update of the goodness estimate of this route and is used to replace the regular pheromone on this route when the bootstrapped value is better than the regular pheromone. In this way, pheromones on current routes are kept up-to-date. In this study, pheromone diffusion proceeds as soon as the shortest routes are discovered; it is then terminated when the regular pheromone information of all routes is populated through the process of pheromone updating.



(a) Average end-to-end delay



(b) Packet delivery ratio



(c) Overhead in number of packets

Figure 3 (See legend on next page.)

(See figure on previous page.)

Figure 3 Performances of all the algorithms in simulation situation 1. The proposed hybrid multi-path routing algorithm DAWMNet is compared in terms of its properties with existing multi-path routing algorithms, namely, AntHocNet, AOMDV, and ARAWSN. In the simulation, 100, 200, 300, and 400 nodes are randomly distributed in a square area with side length of 10 units. Two nodes can communicate with each other if the distance between them is less or equal to $\sqrt{2}$ units. The gateway is placed at the origin for all kinds of networks. A total of ten different topologies are simulated, and the average results are presented for each random network size. The number of nodes in the network is indicated on the x-axis. **(a)** Average end-to-end delay, **(b)** Packet delivery ratio, and **(c)** Overhead in number of packets to assess these algorithms.

3.4. Route selection for data transmission

When all the routes have regular pheromones, then the cycle of route exploration ends. However, for deterministic transmission, not all explored routes are used for data transmission. In this study, the network manager reserves top five routes for each node based on the pheromone values of routes. If node i does not have five routes, then all discovered routes are chosen for data transmission. Each of these top five routes is denoted by a graph ID, a regular pheromone value, and information regarding the next-hop node along this route. The information on these routes for data transmission is stored in graph tables of nodes along these routes.

Prior to each data transmission, node i chooses the route whose graph ID is k from the graph table using the probability P_i^k , as shown in Equation (6). The pheromone value of this route is represented by R_i^k , while N_i^{Id} is the number of the graph IDs stored in the graph table of node i . The parameter β_2 controls the forwarding of data packet and is set at 5. Setting a high β_2 indicates that the data are spread thoroughly over the routes if several routes have similar qualities. However, if one path is clearly better than other paths, then the better path will always be preferred, whereas the data are spread out over multiple paths when β_2 is low.

$$P_i^k = \frac{(R_i^k)^{\beta_2}}{\sum_{k=1}^{N_i^{Id}} (R_i^k)^{\beta_2}} \quad (6)$$

The number of routes used for data transmission is defined beforehand due to the fact that the quality of the

Table 1 Packet delivery ratio (%) of different algorithms with topological changes

Network size	100 nodes		200 nodes		300 nodes		400 nodes	
	10	20	10	20	10	20	10	20
AOMDV	71.2	59.2	72.4	63.9	72.6	72.0	71.7	70.8
ARAWSN	72.8	61.1	73.9	65.7	70.9	70.1	70.4	69.7
AntHocNet	75.1	62.7	82.7	71.1	83.0	82.1	81.5	80.4
DAWMNet	75.9	63.0	83.2	71.3	83.5	82.4	81.9	80.6

Table 1 shows the packet delivery ratio referring to the simulation situation 2. In this situation, 100 to 400 nodes are randomly deployed in a square area with side length 10 units, respectively. 10 and 20 of randomly selected nodes are turned off power. The links along these nodes are in failure.

routes varies with each node. In this way, data transmission can be dispersive for load balance. Therefore, selecting the five best routes and setting β_2 as five guarantees deterministic transmission, and balances the link load in networks, thereby transmitting the data through relatively better routes. Consequently, these five routes can undertake data transmission equally.

3.5. Route maintenance for topological changes

Topological changes may occur in industrial fields due to the faulty nodes or imminent external environmental factors. Any change of network topology, such as joining of new nodes, moving of nodes, and node failure, can trigger route maintenance. To adapt to these, the network manager can make corresponding processes based on different topological changes.

3.5.1. Joining of new nodes

When a node wants to join a network, it must apply to join the network according to the information from the "Advertise" messages sent by other nodes in the network. After succeeding in joining the network, the new node establishes connections with other nodes if the distance between them is within communication range. This node can obtain the route information through pheromone diffusion, which is conducted by its neighbors. Afterwards, the node sends out proactive forward ants for route sampling. The network manager selects several routes to transmit data for this node according to pheromone values of sampled routes.

3.5.2. Nodes moving

When the existing node moves, "keep-alive" messages are sent to notify the neighbors of this moving node. In this case, the network manager decides whether or not to start route exploration based on the specific conditions. If the node moves a short distance, then there is no change of the hops between nodes. Therefore, starting a route exploration is not necessary; instead, the network manager updates the pheromone values of the routes based on the moving node. This is because the distance between the node and its neighbors has changed. Otherwise, if the routes along this moving node are broken down, then the network manager considers the maintenance mechanisms below.

Table 2 Average end-to-end delay (ms) of different algorithms with topological changes

Network size	100 nodes		200 nodes		300 nodes		400 nodes	
	10	20	10	20	10	20	10	20
AOMDV	1611.6	1585.4	3965.9	4007.3	6682.8	6774.1	8683.5	8752.2
ARAWSN	1703.9	1648.1	4523.7	4675.1	6756.4	6836.5	8436.1	8516.3
AntHocNet	1583.4	1556.3	3508.1	3622.7	6083.5	6179.2	8047.7	8092.9
DAWMNet	1541.5	1492.2	3314.2	3401.2	5732.3	5839.6	7685.2	7712.8

Table 2 shows the average end-to-end delay referring to the simulation situation 2. In this situation, 100 to 400 nodes are randomly deployed in a square area with side length 10 units, respectively. 10 and 20 of randomly selected nodes are turned off power. The links along these nodes are in failure.

- (1) When other routes are still available in the pheromone table for replacement, then the substitute routes are added to the graph table.
- (2) When no route is available in the pheromone table for replacement, then the route exploration is restarted (the steps of the route exploration are shown in Section 3.3.2). One link failure may affect the pheromone values of other routes; thus, the scope of maintenance depends on the specific conditions.

3.5.3. Node failure

When a node receives the “unconnected message” from its neighbor node or when a neighbor node does not exchange the message exceeding the “Time keep-alive interval,” then the node should send the command “Alarm graph route failed” to the network manager. Afterwards, the network manager informs the other nodes about the route of the failure node, so that other nodes can keep track of this topological change. The route maintenance for link failure is similar to the two maintenance mechanisms presented for nodes moving. In this study, if there is only one route available for data transmission of an end node, the network manager must restart route exploration for this node.

4. Performance evaluations

4.1. Simulation environment

The performance of the proposed algorithm is evaluated by conducting a simulation in OMNeT++ on a Pentium 4 desktop with 2-GB memory. The data rate in the network at the physical layer is 250 kbps, and the packet length is 1 kb. Each simulation runs for 200 s. In addition, γ , which is mentioned in Equations (2) and (4), is maintained at 0.7, while β_1 and β_2 , mentioned respectively in Equations (5) and (6), are both set at 5.

In the simulation, 100, 200, 300, and 400 nodes are randomly distributed in a square area with side length of 10 units. Two nodes can communicate with each other if the distance between them is less or equal to $\sqrt{2}$ units. The gateway is placed at the origin for all kinds of networks. For each number of nodes, a total of ten different

topologies are simulated, and the average results are presented for each network size.

The properties of DAWMNet are compared with existing multi-path algorithms, namely, AOMDV [7], AntHocNet [19], and ARAWSN [26]. AOMDV is considered the standard multi-path routing algorithm for wireless mesh networks, whereas ARWSN is the multi-path routing algorithm for WSN based on ARA.

4.2. Evaluation criteria

The most important performance metrics in assessing the proposed algorithm are given below.

- (1) *End-to-end delay*: this refers to cumulative statistical measure of the delays taken by packets traveling from their source nodes to their destination nodes.
- (2) *Packet delivery ratio*: this is the ratio between the number of packets successfully received by the gateway and the number of packets sent out by the end nodes.
- (3) *Overhead in number of packets*: this refers to the relationship between the total control packets transmitted by the nodes and the total delivered data packets flowing to the gateway.

4.3. Simulation results and analysis

Two kinds of situations are examined in the simulation. In situation 1, all nodes in the networks are functioning properly, whereas in situation 2, 10, and 20 randomly selected nodes are turned off separately in each network size. The links along these nodes are broken down.

4.3.1. Simulation situation 1

In this situation, all nodes work well and generate packets that flow to the gateway. The three metrics mentioned above are taken in to account for comparison. The performances of all the algorithms are shown in Figure 3. Based on the results shown in Figure 3a, DAWMNet has better performance in average end-to-end delay than the other three approaches when the number of nodes is increased. DAWMNet and AntHocNet achieve better performances

in terms of packet delivery ratio compared with the two other algorithms in Figure 3b. Moreover, the difference in performances among all the algorithms increases with increasing network size. In particular, the packet delivery ratio in the DAWMNet algorithm is maintained at 80% even when the number of nodes reaches 400. This finding indicates that DAWMNet guarantees reliable communication in industrial wireless mesh networks. DAWMNet also yields lower overhead than the three other algorithms with increasing network size in Figure 3c.

Unlike AntHocNet, DAWMNet adopts the first route setup mechanism, the stop scheme in route exploration, and the route selection mechanism. With these improvements, data transmission is dispersed on relatively better routes instead of being centralized on the best route with lower overhead. In addition, the route maintenance scheme in DAWMNet can make corresponding processes based on different topological changes. These processes are operated only when link failure occurs; these are not repeatedly executing. Furthermore, each route is sampled by ants, thus achieving reliable communication. AOMDV and ARAWSN are based on control packets for route establishment; thus, it is difficult to guarantee that the optimum routes are used for data transmission. Moreover, routing overhead is inevitable for AOMDV and ARAWSN. Therefore, DAWMNet shows better performance than the other three algorithms in terms of average end-to-end delay, packet delivery ratio, and routing overhead.

4.3.2. Simulation situation 2

The control environment is harsh in the industrial fields. Thus, the links between nodes may fail because of imminent external environmental factors or faulty nodes. Among the three kinds of topological changes presented in Section 3.5, the most severe form is the link failure caused by node failure. To verify the effectiveness of route maintenance of the proposed algorithm for topological changes, 10 and 20 randomly selected nodes are turned off separately in each network size. Therefore, the routes along these nodes are broken down. In this simulation, the average end-to-end delay and packet delivery ratio are taken into account for comparison.

DAWMNet and AntHocNet have similar performances in terms of packet delivery ratio compared with the two other algorithms with the same failure nodes, as shown in Table 1. With the route maintenance mechanism, DAWMNet ensures the reliability of routes and avoids unnecessary restarting route exploration. Table 2 shows the average end-to-end delay of the different algorithms. With the same failure nodes, DAWMNet shows the best performance among all the algorithms. As shown in Table 1, the packet delivery ratio decreases with increasing number of failure nodes; moreover, the gap of delivery ratio between the two types of topological changes becomes

narrow with increasing network size. Rebuilding the routes after link failures is difficult because not many alternative routes are available in small-scale networks. However, with the increasing size of networks, the effect of such situation wears off because numerous substitute routes can easily be found in large-scale networks. Thus, the average end-to-end delay of 20 failure nodes is smaller than that with 10 failure nodes in a 100-node network; in comparison, the average end-to-end delay with 20 failure nodes is larger than that with 10 failure nodes in other network sizes (Table 2).

Multiple route exploration in DAWMNet is based on the shortest route setup, thereby ensuring that data packets are transmitted over optimum routes. In addition, the network manager carries out different route maintenance schemes based on topological changes. Therefore, even with the failure of some nodes, DAWMNet still achieves better performance in terms of packet delivery ratio and average end-to-end delay compared with the three other algorithms.

In sum, DAWMNet outperforms the three other algorithms in terms of average end-to-end delay and delivered packet ratio without causing much overhead. DAWMNet is more scalable because its performance advantage increases with number of nodes; moreover, it can cope with topological changes. Therefore, this proposed approach achieves reliable and deterministic transmission with low flooding problem, ensures balanced loading, and has the ability to cope with topological changes.

5. Conclusions

In this article, a hybrid multi-path routing algorithm for industrial wireless mesh networks is proposed. The proposed algorithm is an improvement of the ACO routing protocol combined with the enhanced Dijkstra's algorithm. The hybrid algorithm enhances efficiency in the route setup and improves redundant route exploration and failure link maintenance. The simulation results demonstrate that the proposed algorithm outperforms the traditional algorithms in terms of average end-to-end delay, packet delivery ratio, and routing overhead. Moreover, the proposed algorithm can cope with topological changes. Therefore, this algorithm achieves reliable and deterministic communication, ensures balanced loading, and has the ability to cope with topological changes without high flooding.

Several possible lines of work can be developed further. The advantages of this approach in small-scale network are not that evident. Thus, improvements in the proposed algorithm should be made to allow it to adapt to different network scales. And the proposed algorithm will be improved to combine with the TDMA scheduling scheme to collectively conserve energy consumption in further research.

Competing interests

The authors declare that they have no competing interests.

Acknowledgments

This study was supported by the National High Technology Research and Development Program of China (863 Program) under Grant no. 2011AA040103 and a grant from the Shanghai Leading Academic Discipline Project, China (no.B504).

Received: 27 July 2011 Accepted: 4 February 2013

Published: 20 March 2013

References

1. HART Communication Foundation, *HART7 Overview and Summary (Technical Documentation)* (Austin, 2007)
2. CE Perkins, P Bhagwat, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, in *ACM SIGCOMM Computer Communication Review*, 1994. 24 (4), 234–244
3. CE Perkins, EM Royer, Ad hoc on-demand distance vector routing, in *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications (WMCSA' 99)*, (New Orleans). 25–26 February 1999, pp. 90–100
4. ZJ Haas, MR Pearlman, The zone routing protocol: a hybrid framework for routing in ad hoc networks, in *Ad Hoc Networking*, ed. by C.E. Perkins (Reading, MA, Addison-Wesley, 2001), pp. 221–254
5. A Nasipuri, R Castaneda, SR Das, Performance of multipath routing for on-demand protocols in mobile ad hoc networks. *Mob. Netw. Appl.* **6**(4), 339–349 (2001)
6. SJ Lee, M Gerla, Split multipath routing with maximally disjoint paths in ad hoc networks, in *Proceedings of IEEE International Conference on Communications, ICC 2001*. vol. 10, Helsinki, 11–14, June 2001, pp. 3201–3205
7. MK Marina, SR Das, On-demand multipath distance vector routing in ad hoc networks, in *Proceedings of Ninth International Conference on Network Protocols, ICNP 2001*. California, 11–14 November 2001, pp. 14–23
8. E Gelenbe, R Lent, Power-aware ad hoc cognitive packet networks. *Ad Hoc Netw.* **2**(3), 205–216 (2004)
9. E Gelenbe, Cognitive Packet Network. US Patent **6**, 804–201B1 (2004)
10. E Gelenbe, R Lent, ZG Xu, Measurement and performance of a cognitive packet networks. *Comput. Netw.* **37**(6), 691–701 (2001)
11. Q Qu, Y Pei, JW Modestino, XS Tian, B Wang, Cross-layer QoS control for video communications over wireless ad hoc networks. *EURASIP J. Wirel. Commun. Netw.* **2005**(5), 743–756 (2005)
12. AB Bagula, KG Mazandu, Energy constrained multipath routing in wireless sensor networks, in *Lecture Notes in Computer Science (LNCS)*, vol 5061, ed. by FE Sandnes et al. (UIC, 2008), pp. 453–467
13. SR Heikalabad, H Rasouli, F Nematy, N Rahmani, QEMPAR: QoS and energy aware multi-path routing algorithm for real-time applications in wireless sensor networks. *Int. J. Comput. Sci. Issues* **8**(1), 466–471 (2011)
14. HART Communication Foundation, Network Management Specification. Technical Documentation, Austin, 2007
15. EW Dijkstra, A note on two problems in connexion with graphs. *Numer. Math.* **1**(1), 269–271 (1959)
16. M Dorigo, M Birattari, T Stutzle, Ant colony optimization-artificial ants as a computational intelligence technique. *IEEE Comput. Intell. Mag.* **4**(1), 28–39 (2006)
17. Y Lu, GZ Zhao, FJ Su, Adaptive ant-based dynamic routing algorithm, in *Proceedings of 5th World Congress on Intelligent Control and Automation (WCICA 2004)*, vol. 3, Hangzhou, 15–19 June 2004, pp. 2694–2697
18. M Gunes, U. Sorges, I Bouazizi, ARA—the ant-colony based routing algorithm for MANETS, in *Proceedings of the International Conference on Parallel Processing Workshops (ICPPW02)*, Vancouver, 20–23 August 2002, pp. 79–85
19. F Ducatelle, GA Di Caro, LM Gambardella, Ant agents for hybrid multipath routing in mobile ad hoc networks, in *Proceedings of the Second Annual Conference on Wireless On-demand Network Systems and Services (WONS 2005)*, St Moritz, 19–21 January 2005, pp. 44–53
20. E Gelenbe, Steps toward self-aware networks. *Commun. ACM* **52**(7), 66–75 (2009)
21. G Koh, D Oh, H Woo, A graph-based approach to compute multiple paths in mobile ad hoc networks, in *Lecture Notes in Computer Science (LNCS)*, vol. 2713 (171), ed. by C-W Chung et al. (HSL, 2003), pp. 323–331

22. F Ducatelle, GA Di Caro, LM Gambardella, Using ant agents to combine reactive and proactive strategies for routing in mobile ad hoc networks. *Int. J. Comput. Intell. Appl. (Special Issue on Nature-Inspired Approaches to Networks and Telecommunications)* **5**(2), 169–184 (2005)
23. ML Fredman, RE Tarjan, Fibonacci heaps and their uses in improved network optimization algorithms. *J. ACM* **34**(3), 596–615 (1987)
24. RS Sutton, AG Barto, *Reinforcement Learning: An Introduction* (MIT Press, Cambridge, MA, 1998)
25. E Gelenbe, Sensible decisions based on QoS. *Comput. Manag. Sci.* **1**(1), 1–14 (2003)
26. HW Liang, WM Chen, S Li, T Mei, QH Meng, ACO-based routing algorithm for wireless sensor networks (ARAWSN). *Chin. J. Sensors Actuators* **20**(11), 2450–2455 (2007)

doi:10.1186/1687-1499-2013-82

Cite this article as: Zuo et al.: A hybrid multi-path routing algorithm for industrial wireless mesh networks. *EURASIP Journal on Wireless Communications and Networking* 2013 **2013**:82.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com