IEEE Access
Multidisciplinary Rapid Review Open Access Journal

Received June 28, 2019, accepted August 6, 2019, date of publication August 12, 2019, date of current version August 28, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2934889

Energy-Efficient Load Balancing Ant Based Routing Algorithm for Wireless Sensor Networks

XINLU LI[1], BRIAN KEEGAN[2], (Senior Member, IEEE), FREDRICK MTENZI[2], THOMAS WEISE[1], AND MING TAN[1]

[1]Department of Computer Science, Institute of Applied Optimization, Hefei University, Hefei 230601, China
[2]School of Computing, Technological University Dublin, Dublin, D08 X622 Ireland

Corresponding author: Ming Tan (tanming@hfuu.edu.cn)

This work was supported in part by the Science Foundation for Research Development of Hefei University under Grant 18ZR06ZDA and 18-19RC25, in part by the Anhui Natural Science Research Project under Grant KJ2019A0835, and in part by the National Natural Science Foundation of China under Grant 6167335.

ABSTRACT Wireless Sensor Networks (WSNs) are a type of self-organizing networks with limited energy supply and communication ability. One of the most crucial issues in WSNs is to use an energy-efficient routing protocol to prolong the network lifetime. We therefore propose the novel Energy-Efficient Load Balancing Ant-based Routing Algorithm (EBAR) for WSNs. EBAR adopts a pseudo-random route discovery algorithm and an improved pheromone trail update scheme to balance the energy consumption of the sensor nodes. It uses an efficient heuristic update algorithm based on a greedy expected energy cost metric to optimize the route establishment. Finally, in order to reduce the energy consumption caused by the control overhead, EBAR utilizes an energy-based opportunistic broadcast scheme. We simulate WSNs in different application scenarios to evaluate EBAR with respect to performance metrics such as energy consumption, energy efficiency, and predicted network lifetime. The results of this comprehensive study show that EBAR provides a significant improvement in comparison to the state-of-the-art approaches EEABR, SensorAnt, and IACO.

INDEX TERMS Ant colony optimization, energy efficiency, load balancing, routing algorithm, wireless sensor networks.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) are self-managed network systems, which consist of numerous distributed sensor nodes [1], [2]. The purpose of a WSN is either to monitor the conditions, such as the temperature and humidity, of an environment or to detect the motion of mobile targets, such as wildlife or the spread of fire [3], [4]. The nodes of WSNs usually have inherent limitations in terms of the available energy supply as well as their computing and communication abilities. Energy-efficiency is therefore a very important feature of a routing protocol for WSNs, needed to extend the network lifetime and to improve the communication performance and therefore receives much research attention.

The associate editor coordinating the review of this article and approving it for publication was Abderrezak Rachedi.

Ant Colony Optimization (ACO) [5]–[8] is a Swarm Intelligence based algorithm inspired by the foraging behavior of ant colonies, which has distributed, self-organized and positive-feedback characteristics. The basic idea behind the ACO algorithm for routing is to use these very same characteristics to discover good routes from the source to the destination nodes [9]. First, ACO-based routing protocols were developed for wired networks [10] and then for ad-hoc networks (MANETs) [11]. Zhang *et al.* [12] found that the protocols available in literature [11], [13] were not suitable for sensor networks and then presented a seminal framework for ant-based WSN routing protocols. Chen et al. presented the E-D ANTS [14] protocol, which aims to find a path with minimum energy consumption and to maximize the network lifetime. We recently proposed a hierarchical Ant Colony based Clustering Routing (ACCR) [15] protocol for large-scale WSNs. Within ACCR, a clustering algorithm is applied to estimate the average energy consumption to further improve the routing quality. We have continued our research on this topic and here can contribute the following significant improvements:

1) A novel pseudo-random proportional routing discovery algorithm is developed, which speeds up the algorithm convergence while maintaining a balanced energy load. This new algorithm combines the following three novel components into one protocol.
2) An improved pheromone trail update method, based on the energy levels of the sensor nodes and the lengths of the paths between them, is proposed to prolong the network lifetime.
3) An effective heuristic strategy is presented to greedily reinforce routes with low energy cost in the route discovery phase based on an expected cost metric.
4) An opportunistic broadcast scheme is developed to replace flooding for transmitting control packets. Consequently, the energy consumption is reduced further.

The rest of this paper is organized as follows: Section II provides a comprehensive overview of the state-of-the-art of WSN routing protocols. In Section III, we introduce our Energy-Efficient Load Balancing Ant-based Routing Algorithm (EBAR). In Section IV, a comprehensive experimental study is conducted to provide an in-depth analysis of the performance of our approach. Finally, we conclude the paper with a summary and an outlook on future work in Section V.

II. RELATED WORK

The primary concern of routing algorithm design for WSNs is energy efficiency to maximize the network lifetime. In order to achieve this goal, it is more important to route the data traffic such that the energy consumption is balanced among the nodes in proportion to their available energy instead of minimizing the absolute consumed energy [16], [17]. Over the past few years, several ACO-based routing algorithms were proposed to satisfy the inherent limitations of WSNs regarding available energy supply and both computing and communication abilities. Generally, ant-based routing algorithms work in a distributed manner [18] where the sensor nodes periodically release "forward ants" to discover possible routes toward a sink node pro-actively, which then send back "backward ants" to update the route information.

The Energy-Efficient Ant-Based Routing Algorithm (EEABR) defined in [19] is an improved version of Ant Routing [12]. EEABR considers the energy factors of the wireless sensor nodes on top of the ACO mechanism to extend the network lifetime. It also applies a novel distance-based strategy during the pheromone updated triggered by the backward ants. However, it does not consider the energy balance of the entire network.

Sun *et al.* [20] introduce an improved heuristic function in ACO and consider the distances and residual energy of the nodes to find the optimal path of data transmission. In [21], a mechanism for WSN routing, which can be more effective regarding the criteria of route length, end-to-end delay, and node energy, is presented. This method uses an ant colony-based routing algorithm and local inquiries to find optimal routes. Both of the above algorithms [20], [21] pay more attention to the path length and residual energy whilst ignoring the effects of other energy statistics (e.g., average energy, minimum energy), which could lead to an imbalance in the energy dissipation of the entire network.

Okdem and Karaboga [22] present an ACO-based routing protocol for WSNs consisting of stationary nodes. It provides an effective multi-path data transmission method to achieve reliable communication in the case of node faults, while considering the energy levels of the nodes. However, the algorithm assumes that the network is static and cannot be applied in scenarios with multiple sink nodes.

Saleh *et al.* [23] propose a self-optimizing algorithm (SensorAnt) using ACO to discover the best route to achieve balanced energy consumption. This method uses several routing metrics including the residual energy, the number of hops, and the average energy of the nodes on the route and in the whole network. SensorAnt improves the performance compared to EEABR in terms of energy efficiency. However, it requires a significant amount of overhead energy for its operation. This energy consumption can be considered as waste and has a significant impact on overall performance of the network.

Gurav and Nene [24] utilize centralized processing and propose an ACO approach for optimal route discovery. After defining the network topology, the sink nodes search the optimal path using ACO, and then sensor nodes use that path to communicate with each other. However, this scheme requires prior knowledge of the entire network topology.

A different approach to data aggregation was proposed by Weise *et al.* [25], [26], who attempted to synthesize efficient distributed aggregation formulas via Genetic Programming, which should lead to fast convergence and hence fewer messages. This approach did not consider energy efficiently directly but hoped to improve it indirectly.

Recently, combinations of ACO-based routing algorithms and other optimization algorithms have been proposed in order to overcome issues such as the loop problem and uneven network clustering. The routing technique proposed in [27] combines an Artificial Immune System (AIS) and ACO, with the goal to balance energy dissipation to maximize network lifetime. The proposed protocol utilizes ACO to discovery the optimum path from the sensor nodes to the sink and uses the AIS to solve the packet loop problem and to control route direction. Ghosh *et al.* [28] proposed a cluster-based and chain-based routing algorithm (FCM-ACO) to improve the data fusion performance and extend the network lifetime. Under FCM-ACO, Fuzzy C Means clustering [29] is used to divide the network into several clusters. The cluster heads are then linked together in a near-optimal chain formed by ACO. CB-RACO [30] combines ACO with the computationally cheap and distributed community detection technique Label Propagation (LP). LP creates communities

TABLE 1. Comparison of Swarm Intelligence (SI) based routing protocols for WSNs.

Routing protocol	Basic Algorithm	Limitations
E-D ANTS [14]	ACO	difficulties to scale to large topologies
EEABR [19]	ACO	does not consider the energy balance of the entire network
IACO [20], FACOR [21]	ACO+Fuzzy	lead to an imbalance in the energy dissipation of the entire network
ACO-RC [22]	ACO	assumes the network is static, cannot be applied in scenarios with multiple sink nodes
SensorAnt [23]	ACO	requires a significant amount of overhead energy
Protocol [24]	ACO	requires prior knowledge of the entire network topology
Protocols [25], [26]	ACO	do not consider energy efficiency directly but hoped to improve it indirectly

in WSNs and balances the energy consumption by routing data inside communities through swarm intelligence. Kaur and Mahajan [31] proposed an ACO-PSO hybrid routing algorithm to improve the data distribution in an energy efficiency way. The algorithm first clusters the nodes based on their remaining energy, then the hybrid ACO-PSO based data aggregation will improve the inter-cluster data aggregation further. Moreover, fusing the information of several sensing sources can be very beneficial to enhance the efficiency of the routing algorithm [32], [33]. As a result, decision fusion strategies based on swarm intelligence techniques [2], [34], [35] have been designed to address energy consumption problems. We summarize the limitations of the algorithms most closely related to our approach in Table 1.

Essentially, the design of an ant-based routing algorithm lies in the probabilistic strategy of routing discovery, the update of the pheromone trail, and the heuristic function. The algorithms previously discussed are all focused on the improvement of the latter two components, i.e., the metaheuristic part of the protocol. They introduce energy-related or other application-specified factors into the positive feedback mechanism of ACO. In doing so, they guide ants to follow the more attractive paths to the sink node, then find a near-optimal route. These algorithms have proven to be effective and efficient for WSNs [18], [36], [37].

However, in the early stages of route discovery, it is difficult for ants to find a good path to the sink node quickly. Especially in large-scale WSNs with a topology of high density, it is difficult to find a satisfactory feasible path from a large number of chaotically emerging paths in a short period of time. Furthermore, at the beginning of the path search, the pheromones are initialized to a low uniform value. The probability distributions will not be updated until the first ant arrives at the sink and traverses back to the source. Hence, the ants do not receive guidance during that phase, so they may end up constructing (and reinforcing) low-quality paths. In our research, we found no ant-based routing algorithm which properly addresses these important issues.

Applying a heuristic routing mechanism raises the question for convergence, i.e., whether the algorithm will actually produce a stable route and whether this route will be optimal with respect to the performance metrics. Extensive research work has been invested on the analysis of the ACO heuristic on static problems [38]–[41]. The domain of routing in WSNs itself is much less tractable for theoretical analysis, since it poses a dynamic problem: the energy of the nodes and the topology of the network change while the routing is ongoing. To the best of our knowledge, no work has thus been contributed on this topic.

Recently, Lissovoi and Witt [42], [43] investigated the convergence of ACO on dynamic path finding problems and proved that appropriate parameters can enhance the global route discovery ability of the algorithm and improve the convergence speed considerably. However, these parameters setting depend strongly on the available computation time and the scale of the problem (e.g., the scale of the network). In summary, while no theoretical results for heuristic routing in (dynamic) WSNs are available, good convergence behavior has been proven for ACO, the basis of our work, even in dynamic scenarios.

In this paper, we introduce a novel probabilistic route discovery procedure, an improved global pheromone update scheme, and an innovative local heuristic strategy. We present an Energy-Efficient Load Balancing Ant-based Routing Algorithm (EBAR) for Wireless Sensor Networks which overcomes the limitations of the existing approaches.

III. PROTOCOL SPECIFICATION

A. NETWORK MODEL

In this paper, we describe WSNs as undirected weighted graphs. The vertices are the sensor nodes (including the sink) and are denoted by positive integer numbers. If two sensor nodes are in communication range, they are connected by an edge. A source node periodically collects the sensing data from its surroundings and sends the data to the next hop until the it reaches the sink node. The goal is to find an energy-efficient route such that the energy consumption is balanced and the lifetime of the network is prolonged. We assume that:

1) All nodes are isomorphic, i.e., each node is equipped with the same energy, computing power and storage memory.
2) Links are symmetric. If the node's transmitting power is known, nodes can calculate the approximate distance of senders according to the Received Signal Strength Indication (RSSI).
3) Depending on the recipient's distance, the node can adjust its transmitter power level.

TABLE 2. Overview of the notation (alphabetically sorted), containing the control parameters either of standard ACO or our protocol as well as state variables and where they are used.

parameter	meaning and use
$\alpha \geq 0$	parameter controlling importance of pheromone τ_{ij}: higher $\Rightarrow$ more important; (1)
$b_{ij} \in [0,1]$	probability to forward heuristic update message from node i to node j; (6)
$\beta \geq 0$	parameter controlling importance of heuristic desirability η_{ij}: higher $\Rightarrow$ more important; (1)
$C_{ij} \in [0,1]$	link quality between nodes i and j: larger values denote better links; (5), (6)
$\Delta\tau$	total pheromone update value distributed by backward ant; (7), (9), Alg. 4
$\Delta\tau_{ij}$	pheromone update at node i for link (i,j); (8), (9), Alg. 4
$E_i \geq 0$	remaining energy at node i; (5)
$E_{ij} \geq 0$	approximation of the energy cost required by node i to transmit a unit of data to node $j \in N_i$; (4)
$E_{ij}^d \geq 0$	inverse of η_{ij}: approximated energy cost to send a message from node i to sink d over node j; (4), Alg. 3
$\eta_{ij} \geq 0$	local heuristic desirability: edges (i,j) with higher desirability are more likely to be used; (1), Algo. 3, and (4)
$\lambda \in [0,1]$	control parameter adjusting q_0 over time, larger $\lambda \Rightarrow q_0$ decreases quickly $\Rightarrow$ more exploration, smaller λ $\Rightarrow$ more exploitation for a longer time; (3)
M	the list of nodes that the forward ant has already visited; (2)
N_i	the neighbors of node i, i.e., all nodes in communication range of i
$p_{ij} \in [0,1]$	probability to take edge (i,j) in exploration; (2)
$q_0 \in [0,1]$	control parameter tuning between exploitation (big q_0) and exploration (small q_0); (1), 2 – controlled, in turn, by parameter λ; (3)
$\rho \in (0,1)$	pheromone evaporation coefficient: higher values $\Rightarrow$ impact of pheromone update increases; (8)
$\tau_{ij} \geq 0$	the global pheromone trail: edges (i,j) with higher pheromone level are more likely to be used; (1), (8), Alg. 4
$\xi \in (0,1)$	a control coefficient, which scales $\Delta\tau_{ij}$ in (9)

4) We use the First Radio Model [44] as energy consumption model. In this model, the radio consumes a certain amount of energy to run the transmitter or receiver circuitry and the energy loss due to channel transmission depends on the transmission distance.

Based on these assumptions, the transmitter power level can be adjusted to use the minimum energy required to reach the intended next hop receiver. Thus, the energy consumption per unit information transmission depends on the choice of the next hop node, i.e., the selected route. An overview of the notation used in the further text is given in Table 2.

B. THE PROTOCOL OVERVIEW

Informally, our improved Energy-Efficient Load Balancing Ant-based Routing Algorithm (EBAR), works as follows:

1) At the beginning of the route discovery phase, the local heuristic update procedure in Section III-D is launched to update the expected energy cost greedily.
2) At variable intervals, a forward ant is launched from the source node toward the sink node.
3) Each forward ant generates a route by choosing the next node according to a probabilistic state transition rule according to (1) and (2) in Section III-C.
4) Once the forward ant reaches the sink node, a backward ant is created, which moves back along the route that the forward ant had previously traversed.
5) The backward ant will modify the amount of pheromone on its path by applying the pheromone updating rule according to (7), (8), and (9) in Section III-E.
6) Once a backward ant arrives at the source node, the next iteration will begin after a specified interval.

Algorithm 1 ACO-based Routing Pseudocode

```
Initialization of network and parameters
update the local heuristic value (Algorithm 3)
while ¬ termination conditions do
    forward ant is released to search route towards sink
    node and collects and updates statistic along the route
    (Algorithm 2)
    backward ant returns and updates pheromone trails
    (Algorithm 4)
end while
```

This process of our routing algorithm can be summarized briefly as Algorithm 1 and its details are described in the following sections.

C. ROUTE DISCOVERY

The first novelty introduced in our algorithm is a probabilistic route discovery scheme based on the original Ant Colony System (ACS) [5] approach. When traveling towards a sink node, a forward ant located at node i will choose its next hop j at variable intervals according to (1).

$$j = \begin{cases} \arg\max_{r \in N_i}\{\tau_{ir}^{\alpha} \cdot \eta_{ir}^{\beta}\} & \text{if } q \leq q_0 \text{ (exploitation)} \\ \text{draw } j \text{ according to } p_{ij} & \text{otherwise (exploration)} \end{cases} \tag{1}$$

Algorithm 2 Route Discovery Pseudocode

forward ants are released from source nodes to sink node
while sink node is not reached **do**
  update statistic and visited node list
  sample a random number $0 \leq q \leq 1$
  if $q \leq q_0$ **then**
    select the best link according to (1)
  else
    select the next hop node according to (2)
  end if
end while

Here, α and β are two parameters that control the relative importance of the global pheromone trail τ_{ij} and the local heuristic desirability η_{ij} of link (i, j), respectively. p_{ij} is a probability distribution given in (2).

We initialize τ to values drawn uniformly randomly from the range (0, 1), i.e., to small, non-zero values for the start of the route discovery. η is calculated by Algorithm 3 before the route discovery procedure. N_i is the set of neighbours of node i, i.e., the set of nodes within the communication radius of sensor node i. q is a random number uniformly distributed in [0, 1] and $q_0 \in [0, 1]$ is a control parameter tuning between route exploitation and exploration. In an exploration step, j becomes a random variable following the probability distribution p_{ij} given in (2).

$$p_{ij} = \begin{cases} \dfrac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_{r \in N_i} \tau_{ir}^{\alpha} \cdot \eta_{ir}^{\beta}} & \text{if } j \in N_i \text{ and } j \notin M \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Here, M is the list of nodes that the forward ant has already visited, which it carries along its path and p_{ij} is computed as the normalized sum of the product of the pheromone τ_{ij} with the heuristic value η_{ij}. This allows us to take into account the state of the link by making η_{ij} a function of the energy of the nodes i and j. The pheromone update method and the heuristic value are described in Sections III-E and III-D, respectively. Algorithm 2 illustrates the structure of the routing discovery procedure. According to this method, a forward ant will either select the optimal neighbour node (exploitation) or a random one (exploration) as next hop. In (1), exploitation implies that the forward ant has the ability to utilize a priori and accumulated knowledge, while exploration means that the ant will pay more attention to exploring new paths toward a sink node. This allows for a learning process and increases the convergence speed for the route discovery in order to achieve a better path in early route discovery phases.

The parameter q_0 controls the balance between the routing discovery experience gathered so far and the exploration of unvisited links. It can be adapted depending on routing statistics. For instance, it can be increased to bias for route exploitation in the initial phase and decreased in order to favor route exploration. In order to automatically tune from exploitation in the early phase towards more exploration at the later stage of the route discovery, we define the control parameter q_0 as follows.

$$q_0 = e^{-\lambda k} \text{ with } k \geq 0 \text{ and } \lambda \in [0, 1) \quad (3)$$

Here, λ is a constant and k is the zero-based index of the current route discovery iteration. This way, the control coefficient for exploitation and exploration will change over time. The larger k is, the smaller q_0 will be, allowing for exploitation at the beginning. This will speed up the convergence rate at the initial phase in order to quickly find a good route while, at the same time, avoiding converging too fast to a local optimum.

D. LOCAL HEURISTIC UPDATE

In the ACO framework, heuristic values represent a priori information about the problem or run-time information provided by a source different from the ants [5]. The heuristic information is usually defined as a function of the residual energy or the average energy of neighbor nodes in ant-based routing [3], [18], [36], [45]. Making routing decisions on these functions aims to increase the energy efficiency of the WSNs

At the very beginning of the route discovery phase, the pheromones are usually initialized to a very small constant value or to random values of low variance. The heuristic values η_{ij} will thus dominate the routing decision making process defined in (1) and (2). If the simple features from [3], [18], [36], [45] are used as heuristics by the forward ant, this may result in an overall bad path quality [46].

We use a greedy expected energy cost approach as heuristic function, which enables the forward ant to build reasonably effective paths from the very beginning. Based on the approximated energy cost E_{ij} to send a message from node i to a neighboring node $j \in N_i$, we can also approximate the expected energy cost E_{ij}^{d} required by i to transmit a unit of data to a sink node d through j.

Then, we can define the heuristic value η_{ij} as the minimum expected energy cost in an observation window:

$$\frac{1}{\eta_{ij}} = E_{ij}^{d} = E_{ij} + \min_{k \in N_j} \{E_{jk}^{d}\} \quad (4)$$

The expected energy cost E_{ij}^{d} from the current node to the sink is 0 if it is sink node, otherwise, the cost is obtained using a greedy manner by (4).

Based on the assumptions and energy consumption model in Section III-A, we broadcast the expected energy cost from the sink node to the source node. This can be achieved by flooding, but this would cause additional control overhead, network congestion, and extra energy consumption, particularly in a large-scale network.

In order to reduce the impact of the control overhead on performance, we adopt the opportunistic broadcast algorithm presented in our previous work [46]. In this method, a sensor node uses an opportunistic strategy to select a single next node to re-broadcast the energy-related information. We give

Algorithm 3 Local Heuristic Update Pseudocode

```
while ¬ termination conditions do
    if sink node then
        expected energy cost E_ij^d equals 0
        broadcast E_ij^d
    else
        update E_ij^d, local heuristic value according to (4)
        re-broadcast E_ij^d according to (5) and (6)
    end if
end while
```

a goodness function to measure the link quality C_{ij} between node i and its neighbour node j.

$$C_{ij} = \frac{E_i - E_{min}}{E_{max} - E_{min}}, \quad j \in N_i \tag{5}$$

Here, E_{min} and E_{max} are the minimum and maximum residual energy over the complete neighborhood of node i.

When node i receives a broadcast packet, one of the neighbour nodes j will be selected according to distribution b given below to re-broadcast the packet.

$$b_{ij} = \frac{C_{ij}}{\sum_{k \in N_i} C_{ik}} \tag{6}$$

Thus, the opportunistic scheme may select different nodes at each time to re-broadcast the control message instead of flooding. The process of local heuristic update is briefly summarized in Algorithm 3.

E. GLOBAL PHEROMONE UPDATE

A global pheromone update procedure will be applied when the forward ant arrives at the sink node. The global pheromone includes information about the routes taken by the forward ants. After arriving at the sink node, the forward ant will be converted into a backward ant. The backward ant inherits all the statistic information about the route gathered by the forward ant, including path length, energy levels, and the visited node list M. The sink node calculates the amount $\Delta\tau$ of pheromone value attached to this path according to the route statistics as follows:

$$\Delta\tau = \frac{E_{min} \cdot E_{avg}}{e^{E_{init}} \cdot F_{ant}} \tag{7}$$

E_{min}, E_{avg} and E_{init} represent the minimum residual energy, average residual energy and initial energy in the discovered path respectively, F_{ant} denotes the length of the path, and the exponential function is applied to E_{init} to bring $\Delta\tau$ into the right scale. The backward ant carries $\Delta\tau$ at the start of its journey following the reverse path. Each node in the path then updates the global pheromone trail, i.e., the routing table, according to (8).

$$\tau_{ij} = (1-\rho) \cdot \tau_{ij} + \rho \cdot \Delta\tau_{ij} \tag{8}$$

Here, $\rho \in (0, 1)$ is the pheromone evaporation coefficient and $1-\rho$ hence is the pheromone residue factor. $\Delta\tau_{ij}$ is the increment of pheromone value attached to the link (i, j). It is computed at a node i according to:

Algorithm 4 Global Pheromone Update Pseudocode

```
calculate the amount of pheromone value Δτ according to (7)
sink node creates backward ant toward source node
while source node is not reached at each node i receiving the backward ant do
    select the next hop following the reverse path
    calculates the pheromone increment Δτ_ij according to (9)
    updates the pheromone trail τ_ij (i.e., its routing table) according to (8)
end while
```

$$\Delta\tau_{ij} = \begin{cases} \xi \cdot \frac{E_j}{E_{init} \cdot B_{ant}} \cdot \Delta\tau & \text{if } i, j \in M \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

Here, $\xi \in (0, 1)$ is a control coefficient, which scales $\Delta\tau_{ij}$ to the interval $(0, 1)$. E_j is the residual energy of node j from which the backward ant came. B_{ant} denotes the path length from a sink node to the current node i. The pheromone value is thus a function of both the energy levels and the length of the path. As a result:

1) Shorter Paths (less hops) will get a larger pheromone increment.
2) If the minimum energy level is high, this means the weakest node on the path has more energy. Hence, we actually can (and will) route more traffic over this path.
3) A path with higher average energy will attract more data flow.
4) The closer a node is to the sink, the more pheromones are obtained.

The process of global pheromone update by the backward ant is summarized as Algorithm 4.

IV. PERFORMANCE ANALYSIS

In this section we evaluate the performance of the proposed algorithms based on a comprehensive set of experiments. Among the related works, SensorAnt [23] and EEABR [19] are the most representative algorithms, which have common features of energy efficient routing protocols. Moreover, the algorithm given in [20], here named IACO, is a recent addition to the state-of-the-art. We therefore choose these algorithms for comparison to evaluate the performance of our proposed method. In addition, in order to assess the performance of the route discovery scheme in the proposed routing algorithm, we test two variants of EBAR, namely EBAR-R and EBAR-P. In EBAR-R, we just adopt the random-proportional rule from (2) to discover routes, while in EBAR-P, we adopt the pseudo-random-proportional rule given in (1) and (2). This allows us to evaluate whether the pseudo-random proportional rule inspired by ACS can further improve the energy efficiency. For the experiments, we developed a simulation environment on the basis of NS-2 [47], [48] (Version 2.35).

A. EVALUATION METRICS

We use the following performance metrics to evaluate the results obtained from our simulations:

The ***Energy Consumption*** is the sum of the consumed energy of all the sensor nodes in the network during the period of the experiment in Joules (J). We estimate the node energy consumption based on the energy model proposed in [44].

The ***Throughput*** of the network is the sum of the throughput from all destinations. The throughput of a destination is the number of messages it receives per second.

The ***Control Overhead*** is the number of control messages divided by the total amount of messages that have been transmitted. The control packets in our algorithm are the broadcast messages from Section III-D as well as the forward and backward ants from Section III-C. Control overhead can cause network congestion and extra energy dissipation.

The ***Energy Efficiency*** denotes the ratio between the total size of the non-overhead data packet received by the sink node and the total energy consumed in network (Kbits/J). A routing algorithm with high Energy Efficiency transmits more sensor data per consumed energy unit.

The ***Energy Standard Deviation*** is the average variance between energy consumed on all nodes. It describes the energy consumption distribution of the networks. A low standard deviation indicates that the nodes tend to consume the same amount of energy, while a higher standard deviation indicates that the communication load is unevenly distributed among the nodes. An energy-efficient protocol should not just reduce the overall energy consumption, but also avoid overly depleting the battery of some specific nodes. With the Energy Standard Deviation, we can measure how good a protocol is in maintaining an overall balanced energy distribution in a network.

The ***Network Lifetime Prediction*** is defined as the difference of the total initial energy E of the network and the sum of the average $\bar{\mu}$ of the consumed energy μ of the nodes and the standard deviation σ of their energy levels:

$$\text{Lifetime Prediction} = E - (\bar{\mu} + \sigma) \tag{10}$$

The basic motivation behind this definition is that an algorithm should try to maximize the average remaining energy levels of nodes with a minimal standard deviation.

The ***Route Setup Time*** represents the time spent by a protocol to discover the initial effective routes from source nodes to sink nodes. It is used to measure the time until the first effective route is discovered.

We design a series of dynamic networks of different sizes to analyze these metrics and to evaluate the performance of route discovery scheme and local heuristic update strategy of our proposed routing algorithm.

B. SIMULATION MODEL AND PARAMETERS

To better understand the difference between the routing algorithms in our experiment, we evaluated each algorithm using the same two typical applications of WSNs mentioned at the beginning of our introduction: data collection and target tracking. For simplicity, we consider data collection in a static network, while target tracking takes place in a dynamic network. In each scenario, the nodes are all equipped with the same radio device and transmission power, resulting in symmetric links between them, and they are unaware about their location coordinates.

In the data collection application, all sensor nodes are randomly deployed to monitor a static event source, collect the relevant sensor data, and then transfer them to the sink node periodically. All nodes, including the sink node, are fixed and the topology of the network does not change in any significant way.

In the target tracking application, a sensor node in the vicinity of a moving target generates a sequence of events. As the target moves out of the range of that node, the node stops generating the events and another node takes over. Here, we assume that the source node moves randomly in the monitored area. Hence, paths may break and need to be replaced by new paths so that the event information can be delivered.

In both static and dynamic scenarios, there is one sink node, and its location is fixed. Table 3 lists the simulation parameters. In both scenarios, 100 sensor nodes are randomly placed in a square area of size 1000 m^2. Without loss of generality, the sink node is deployed in the corner of the area in order to have longer paths. The data traffic is generated by 30 constant bit rate (CBR) sources, each sending one 64-byte packet per second. In the MAC layer, we adopt the popular IEEE 802.11 protocol suite [49]. Since the parameters in ACO greatly influence the performance of the algorithms, we apply the best practice from [10] and we set the values 1, 5, and 0.5 for α, β, and ρ, respectively. We set the constant λ from (3) to 0.1 and the control coefficient ξ from (9) to 0.9.

C. SIMULATION RESULTS

All results for the static and dynamic network are summarized by Tables 4 and 5, respectively. In the static network, there is one sink node and the other sensor nodes collect sensing data. In the dynamic network, 10% of the mobile source nodes detect moving events and send the data to the single sink node. The results show the superiority of EBAR-P with regard to other protocols in all scenarios.

1) ENERGY CONSUMPTION

Figures 1a and 1b present the overall results with respect to the average energy consumption in the static data collection application and the dynamic target tracking application, respectively. It is clear that sensor nodes consume more energy as the simulation time progresses regardless of the routing protocol used. The results show the EBAR-P

TABLE 3. Simulation Parameters.

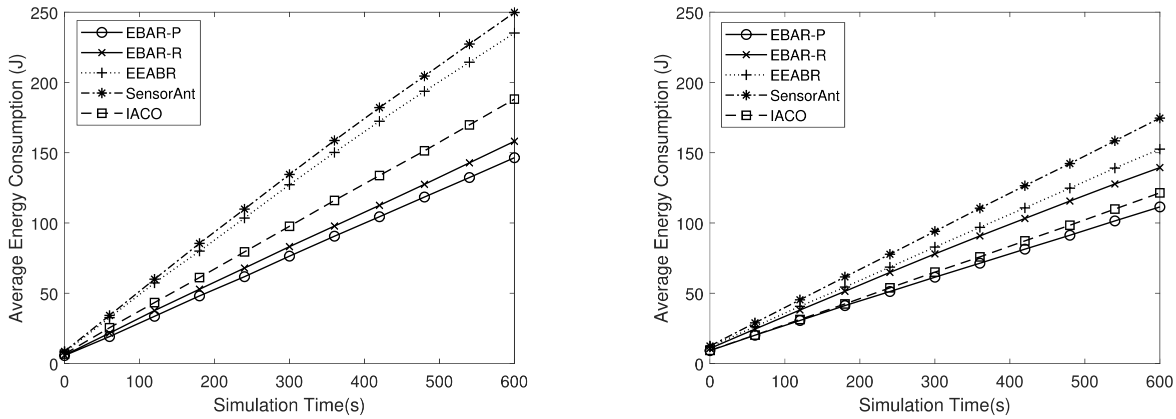
Parameter name	Scenario1 Value	Scenario2 Value
Dimension of topology	1000 m^2	1000 m^2
Number of nodes	100	100
Nodes placement	Random	Random
Mobility	Static	Random waypoint
Traffic	Many-to-One CBR	Point-to-Point CBR
Simulation time	600s	600s
MAC layer	IEEE 802.11	IEEE 802.11
Propagation	Two way	Two way
Initial energy	1000 J	1000 J
α	1	1
β	5	5
ρ	0.5	0.5
ξ	0.9	0.9

TABLE 4. Results for the static network with 100 nodes (best result highlighted in bold face).

Metric	EBAR-P	EBAR-R	EEABR	SensorAnt	IACO
Energy	**146.40**	158.20	235.20	249.70	188.20
Throughput	**1.68**	1.47	0.77	1.09	0.79
Overhead	**0.36**	0.36	0.47	0.47	0.44
Efficiency	**5.86**	3.87	1.97	2.68	4.69
Deviation	4.03	**2.90**	14.63	8.54	8.02
Lifetime	**0.70**	0.68	0.54	0.60	0.59
Setup Time	4.0	**3.80**	10.60	14.60	11.30

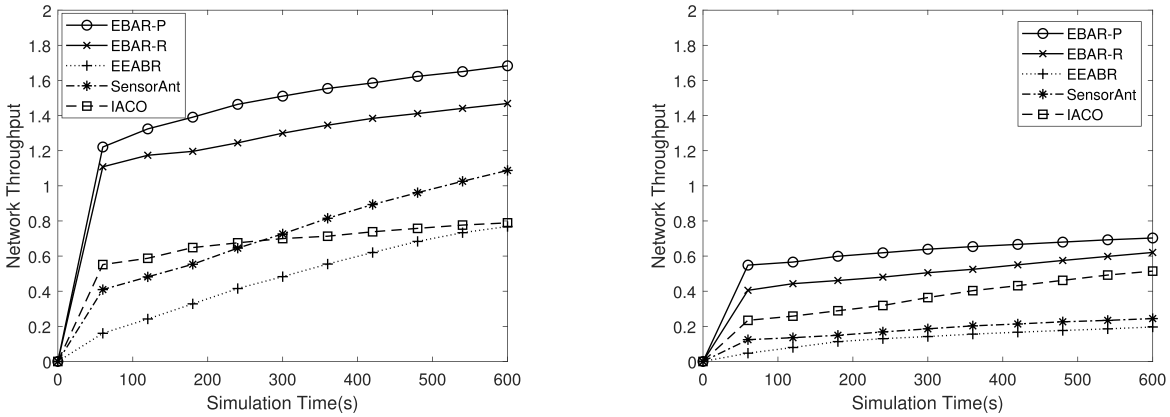
TABLE 5. Results for the dynamic network with 100 nodes (best result highlighted in bold face).

Metric	EBAR-P	EBAR-R	EEABR	SensorAnt	IACO
Energy	**111.40**	139.40	152.60	174.60	121.40
Throughput	**0.70**	0.51	0.20	0.24	0.62
Overhead	**0.47**	0.50	0.60	0.56	0.51
Efficiency	**3.96**	2.61	0.66	0.76	2.93
Deviation	2.36	**1.77**	3.11	2.66	3.93
Lifetime	**0.80**	0.77	0.78	0.74	0.78
Setup Time	**3.80**	4.20	29.40	36.20	37.00



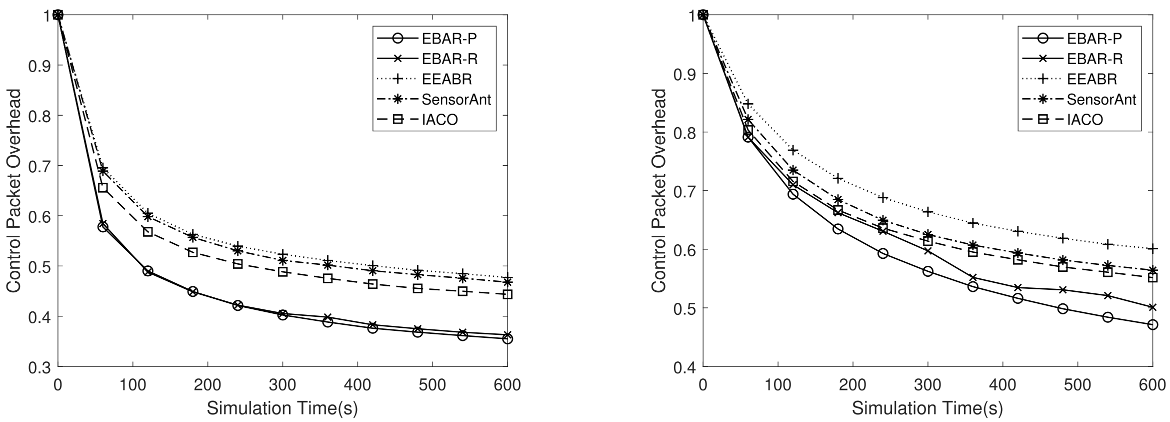
(a) Static Network (data collection) (b) Dynamic Network (target tracking)

FIGURE 1. Comparison of average energy consumption (J) of the network running different routing algorithms.



(a) Static Network (data collection) (b) Dynamic Network (target tracking)

FIGURE 2. Comparison of throughput of the network running different routing algorithms.



(a) Static Network (data collection) (b) Dynamic Network (target tracking)

FIGURE 3. Comparison of control packet overhead of the network running different routing algorithms.

performs better than other protocols, which means it will consume the least energy during the both application scenarios. In the dynamic network, IACO can outperform EBAR-R, while EBAR-P remains surperior. One reason for the slightly worse performance of EBAR-R here may be that it adopts random proportional route discovery in (2), and as such dissipates more energy in the route exploration procedure.

In the static network application, the Energy Consumption of EBAR-P is reduced by approximately 7.5% compared to EBAR-R, 37.8% to EEABR, 41.4% to SensorAnt and 22.2% to IACO. In the dynamic application, the reductions are 20.1%, 29.6%, 36.2%, and 8.2%, respectively.

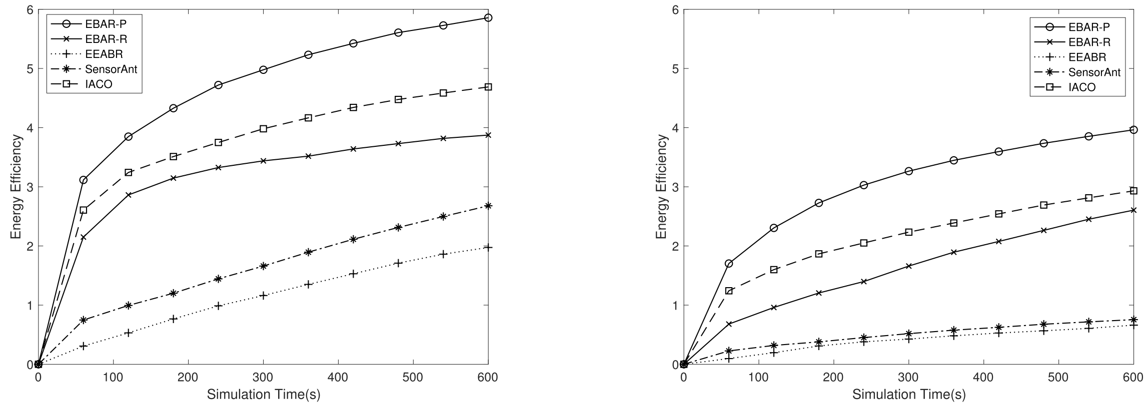
2) THROUGHPUT

Figure 2 demonstrates the impact of the proposed algorithms on the average throughput in the static network (Figure 2a) and the dynamic network (Figure 2b). The results obtained show that EBAR-P and EBAR-R have better performance than the other three protocols. In the initial stages of both application scenarios, due to the probabilistic route discovery, the throughput is low. Moreover, the First Route Setup Times of the EBAR protocols are better than for the other protocols (see Tables 4, 5 and Figure 7 later on), giving them an early advantage in throughput.

One may notice that the throughput tends to stabilize after a while. Our EBAR algorithms prefer routes with high energy levels and fewer hops according to Section III-E. Hence the source node can effectively transmit more sensing data to sink node. In addition, due to the opportunistic broadcast scheme from Section III-D, there are fewer control packets in EBAR-P and EBAR-R compared to the other protocols. Therefore, more of the limited bandwidth remains for the packets with the actual sensor data. The number of packets transmitted through EBAR-P is approximately 209% of the packets transmitted through EEABR, 155% of SensorAnt, and 213% of IACO in a static network application.

3) CONTROL OVERHEAD

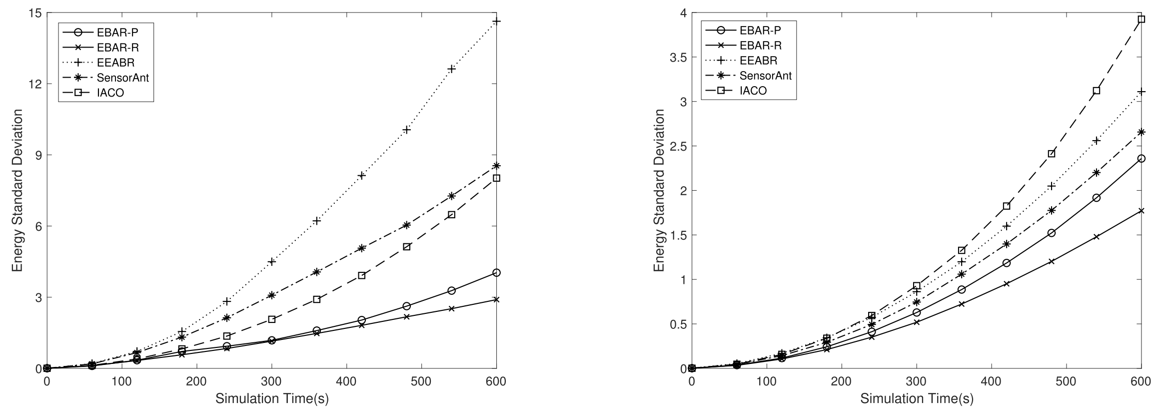
High control packet overhead can cause network congestion and extra energy consumption. The routing control overhead of the different protocols in our study is illustrated in Figure 3. Figure 3a shows that the control overhead results of EBAR-P



(a) Static Network (data collection)

(b) Dynamic Network (target tracking)

FIGURE 4. **Comparison of energy efficiency of the network running different routing algorithms.**



(a) Static Network (data collection)

(b) Dynamic Network (target tracking)

FIGURE 5. **Comparison of energy standard deviation of the network running different routing algorithms.**

and EBAR-R are highly similar and always lower than those of other routing algorithm in the static network. Our proposed routing algorithm uses an opportunistic strategy to select a single node to re-broadcast control message instead of flooding. Due to this opportunistic broadcast scheme defined (5) and (6), the control packet overhead of our EBAR algorithm is reduced in average by approximately 24% compared to EEABR, 24% to SensorAnt and 20% to IACO.

The overhead of EBAR-R exhibits some very small fluctuations in the dynamic network. The reason is that the movement and change of sensor nodes lead to more collisions. In general, the control packet overhead observed in both EBAR-P and EBAR-R is less than that observed in EEABR, IACO, and SensorAnt. Correspondingly, a lower energy consumption can be expected, which we have already seen in Figure 1.

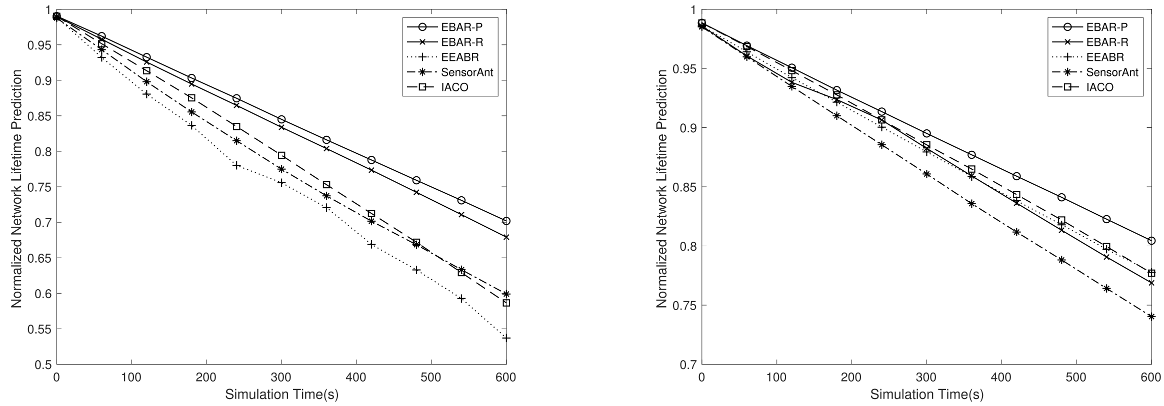
4) ENERGY EFFICIENCY

Figure 4 reports the energy efficiency of routing protocols. EBAR-P and EBAR-R present a significant improvement over other compared protocols. As we can already assume from the energy consumption, throughput, and overhead metrics, EBAR-P and EBAR-R can indeed deliver more sensing data to the sink node at the same energy consumption. Specifically, in the static network application, the Energy Efficiency in EBAR-P is approximately 2 times better than EEABR, 1.2 times better than SensorAnt and 0.3 times better than IACO, while the improvements are 5, 4.3 and 0.4, respectively in a dynamic application.

In both scenarios, EBAR-P demonstrates better Energy Efficiency over EBAR-R. Indeed, one of primary advantages of the EBAR-P is to transmit more valid sensing data with a small amount of energy. Therefore, the results support the claim that energy efficiency may be enhanced when using the pseudo random route discovery scheme, improved global pheromone and local heuristic update with a greedy expected energy cost as given in (4).

5) ENERGY STANDARD DEVIATION

The energy standard deviation indicates the average variance of the energy consumed on all nodes in the network. An energy-efficient routing algorithm should maximize the



(a) Static Network (data collection)

(b) Dynamic Network (target tracking)

FIGURE 6. **Comparison of normalized network lifetime prediction running different routing algorithms.**

average remaining energy levels of the nodes *but* with a small standard deviation so as to prolong the network lifetime. In Figure 5, we can see that, in terms of the the energy standard deviation, EBAR-R performs well in both the static and dynamic scenario. Although EBAR-P and EBAR-R both incorporate an energy-related factor in pheromone and heuristic update ((7), (8), (9), and (4)), they use different rules in the route discovery procedure. In EBAR-P, the forward ant prefers selecting the route with maximum weight $\tau \cdot \eta$ in the initial phase according to the pseudo random selection given in (1). EBAR-R utilizes the random probability selection from (2), where the forward ant has a higher probability of choosing a path which looks better in terms of both the pheromone trail and heuristic value, instead of selecting the maximum weight. This helps to balance the consumption of energy in the network and avoids a rapid deterioration of the optimal path. This is why EBAR-R is superior to EBAR-P with respect to the energy standard deviation.

The pseudo random selection in EBAR-P, on the other hand, is more suitable for rapid establishment of a first effective route to the sink node. In order to limit the impact of this scheme on the energy standard deviation, we define the control coefficient in (3) to constrain the exploitation ability and avoid a fast local convergence in the route discovery procedure.

6) NETWORK LIFETIME PREDICTION

Based on Figures 1 and 5, we can already conclude that EEABR and SensorAnt have lower expectation in term of network lifetime than EBAR-P and EBAR-R. Figure 6 illustrates the normalized network lifetime prediction of these routing algorithms. Under EBAR-P it is improved by approximately 31% compared to EEABR, 17% to SensorAnt, 20% to IACO in the data collection application, while enhanced by 3.5% compared to EEABR, 8.7% to SensorAnt, 3.6% to IACO in the target tracking application. The results imply that the performance improvement of EBAR-P in a static network is even more saliently better than in dynamic network.

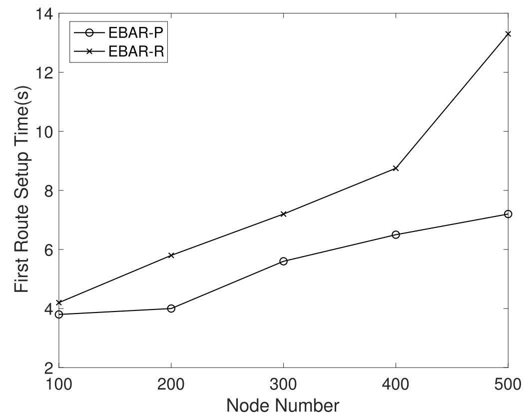


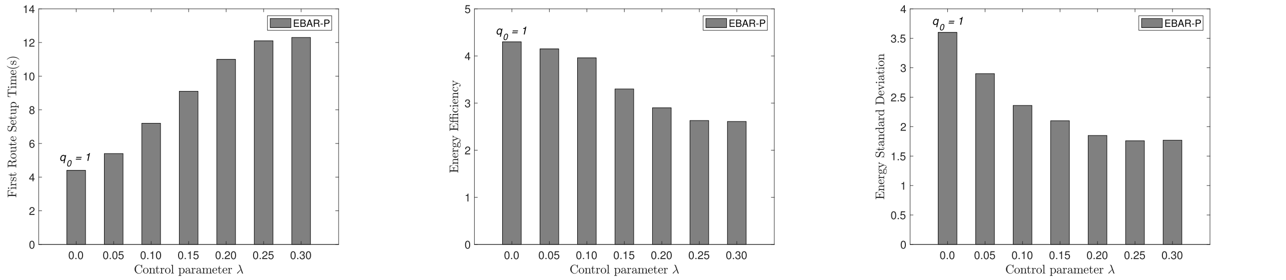
FIGURE 7. **Impact of network scale on First Route Setup Time in dynamic network.**

It is noteworthy that EBAR-R does not demonstrate better predicted lifetime than EBAR-P, although it achieves the lowest energy standard deviation. The reason is that the average energy consumption has a significant impact on the network lifetime. As seen in Figure 1, EBAR-P consumes the least amount of energy in these routing algorithms.

7) ROUTE SETUP TIME

The Route Setup Time represents the time that passes until the first effective route to the sink node is established. We utilize this metric to measure the impact of the pseudo random route discovery algorithm and the heuristic strategy based on greedy expected energy cost in our proposed routing algorithm.

Tables 4 and 5 show the results of Route Setup Time both in the static and dynamic network with 100 sensor nodes, respectively. In both cases, EBAR-P and EBAR-R can discover an effective route in a shorter period of time than the other simulated routing algorithms. This difference becomes most salient in the dynamic case. The reason of this performance gap is that our heuristic strategy guides the



(a) Impact of control parameter λ on First Route Setup

(b) Impact of control parameter λ on Energy Efficiency

(c) Impact of control parameter λ on Energy Standard Deviation

FIGURE 8. Impact of control parameter λ on First Route Setup Time, Energy Efficiency and Energy Standard Deviation in dynamic network.

artificial ant to travel trough the link with lowest expected energy cost, so it finds the sink node faster in the network initialization phase, where there is no pheromone prompt. There is no significant difference between EBAR-P and EBAR-R with respect to the Route Setup Time. To further assess the difference, we simulate the proposed routing algorithms for different network densities by letting the number of nodes range from 100 to 500. Here, we only consider the dynamic network.

Figure 7 illustrates the impact of the network scale on this performance metric. The setup time of EBAR-P is always less than that of EBAR-R and with increasing network scale, this difference becomes more apparent. Specifically, an improvement of 85% is obtained compared with EBAR-R when the network has 500 sensor nodes. We have thus confirmed in Figure 7 that the pseudo random routing discovery algorithm can speed up the convergence rate of routing, especially in a large-scale network.

D. IMPACT OF PARAMETER ON PERFORMANCE

Our routing discovery mechanism has several parameters, as shown in Table 2. α and β from (1) and (2) are two parameters to control the relative importance of the pheromone and heuristic value. ρ from (8) is the pheromone evaporation coefficient. We use ξ to scale the increment of pheromone value in (9). Furthermore, we introduce λ in (3) to tune q_0 in (1) in the pseudo-random-proportional route discovery rule to balance exploitation to exploration in the routing discovery procedure.

These parameters have an important impact on the performance of the proposed ant-based routing algorithm. α, β and ρ are derived from best practices for ACO. As suggested in [10], [11], [20], [23], we set them to 1, 5, and 0.5, respectively. The study of the impact of various parameters on the behavior of ACO algorithms has been an important subject [5], [6], [50].

In EBAR, we first proposed a pseudo-random proportional routing discovery algorithm to speed up the routing algorithm convergence, which is discussed in Section III-C. Here, the control parameter $q_0 \in [0, 1]$ has a significant impact on the route selection. A larger value of q_0 indicates that the routing algorithm relies more on exploitation, while a smaller q_0 emphasizes on exploration.

To evaluate the impact of q_0 and λ on the performance of the routing algorithm, we leave α, β and ρ at their default settings, and vary λ from 0 to 0.30 with step of 0.05. We only consider the dynamic network with 500 sensor nodes running the EBAR-P variant.

Figure 8 demonstrates the impact of the control parameter λ in (3) on the First Route Setup Time, Energy Efficiency and Energy Standard Deviation. We start our investigation with $\lambda = 0$, which leads to $q_0 = 1$, i.e., forces the algorithm to always pick the node with the best heuristic value and follow the route with the lowest energy requirement. In scenarios on the other end of the scale, with $\lambda > 0.25$, q_0 will approach zero quickly. These settings strongly favor exploration over exploitation and the algorithm behaves similar EBAR-R, from which we know that it is outperformed by our EBAR-P.

The time to find the first effective path to the sink node is gradually increasing as λ becomes larger (Figure 8a). For small values of λ, q_0 will approach 0 very slowly, so that forward ants will more often greedily pick the node with lowest energy requirement for routing, which tends to be closer to the sink according to (1). Hence they are likely to establish the first effective route quickly. When λ is greater than 0.25, the setup time change is not significantly anymore. This complies with our findings in Figure 7, where EBAR-P is faster than *EBAR-R*.

Figure 8b shows that smaller values of λ result in better Energy Efficiency. This is partly due to the algorithm's ability to establish an effective path faster, hence it can transmit more sensor data in a limited simulation runtime. Here, the settings with $\lambda \in \{0, 0.05\}$ perform even slightly better than those with our default value $\lambda = 0.1$. These slight gains in (mean) Energy Efficiency, however, come at a high cost: the Energy Standard Deviation (Figure 8c) increases very significantly. In other words, a protocol that constructs routes only greedily is not robust and likely to converge to a locally optimal solution. Our default setting with $\lambda = 0.1$ presents a good trade-off, as it offers a much smaller Energy Standard Deviation at a minimally decreased Energy Efficiency, i.e., is both efficient and robust. Nevertheless, the performance indicators behave well over the range of reasonable control parameter settings and degenerate gracefully if bad settings are picked.

V. CONCLUSION

WSNs are used in many application areas such as data collection and target tracking. Due to resource limitations of sensor nodes, it is necessary to design an energy-efficient routing algorithm. In this paper, we presented an in-depth study of energy-efficient routing for WSNs and propose the novel energy-efficient load balancing ant-based routing protocol (EBAR). The goals of EBAR are to balance the energy consumption, prolong the network lifetime, and to speed up the convergence of route discovery under the constraints of the limited energy supply. In the design of EBAR, we adopt a pseudo random algorithm to discover routes, which not only accelerates the search of an effective route, but also considers the balance of energy consumption. In the pheromone trail update procedure, we now also consider energy levels and path length information, with the result of prolonging the network lifetime. Furthermore, we presented an effective heuristic strategy based on the greedy expected energy cost, which further speeds up the initial route establishment. Finally, in order to reduce the impact of control overhead on the overall performance, an energy-based opportunistic broadcast scheme is adopted.

Our extensive experimental study based on simulations shows that EBAR has better performance in comparison to the state-of-the-art protocols, EEABR, SensorAnt, and IACO, according to all relevant metrics. We are in the process of implementing the proposed algorithm on a physical testbed to study the impact of varying real-world conditions on the performance of these algorithms. These practical results will then be used to adjust and fine-tune the simulation scenarios, which, in turn, can then be used to quickly and efficiently develop even better routing protocols.

As we described in Section III-A, we assume that the routing algorithm runs under an ideal condition, i.e., all nodes are isomorphic and the links are symmetric. An interesting direction that we will pursue in our future work is adapting our EBAR algorithm to heterogeneous networks, where these conditions may not hold. If the links are non-symmetric, the predicted energy cost E_{ij} for sending a message from node i to j may be different from E_{ji}. In this case, the pheromone trail η and heuristic value τ in (1) and (2) have to be defined for both directions. If the nodes may differ in hardware, it is necessary to normalize the remaining energy metric in (4), (5), and (9) with the energy consumption and battery capacity. We will generalize EBAR along these lines and verify the new algorithm experimentally.

REFERENCES

[1] D. S. Deif and Y. Gadallah, "An ant colony optimization approach for the deployment of reliable wireless sensor networks," *IEEE Access*, vol. 5, pp. 10744–10756, 2017.

[2] X. Cheng, D. Ciuonzo, and P. S. Rossi, "Multi-bit decentralized detection of a weak signal in wireless sensor networks with a Rao test," in *Proc. IEEE 23rd Int. Conf. Digit. Signal Process. (DSP)*, Shanghai, China, Nov. 2018, pp. 1–5.

[3] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 551–591, May 2013.

[4] B. Sun, C. Gui, Y. Song, and H. Chen, "A novel network coding and multi-path routing approach for wireless sensor network," *Wireless Pers. Commun.*, vol. 77, no. 1, pp. 87–99, 2014.

[5] M. Dorigo and M. Birattari, "Ant colony optimization," in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds. Boston, MA, USA: Springer, 2011, pp. 36–39.

[6] M. Dorigo and T. Stützle, "The ant colony optimization metaheuristic: Algorithms, applications, and advances," in *Handbook of Metaheuristics*, F. Glover and G. A. Kochenberger, Eds. Boston, MA, USA: Springer, 2003, pp. 250–285.

[7] A. K. Kar, "Bio inspired computing—A review of algorithms and scope of applications," *Expert Syst. Appl.*, vol. 59, no. 15, pp. 20–32, Oct. 2016.

[8] T. Weise, R. Chiong, J. Lassig, K. Tang, S. Tsutsui, W. Chen, Z. Michalewicz, and X. Yao, "Benchmarking optimization algorithms: An open source framework for the traveling salesman problem," *IEEE Comput. Intell. Mag.*, vol. 9, no. 3, pp. 40–52, Aug. 2014.

[9] X. Liu, "Routing protocols based on ant colony optimization in wireless sensor networks: A survey," *IEEE Access*, vol. 5, pp. 26303–26317, 2017.

[10] T. Liao, K. Socha, M. A. M. de Oca, and T. Stützle, and M. Dorigo, "Ant colony optimization for mixed-variable optimization problems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 503–518, Aug. 2014.

[11] G. A. Di Caro, "Ant colony optimization and its application to adaptive routing in telecommunication networks," Ph.D. dissertation, Dept. Institut Recherches Interdisciplinaires Développements Intell. Artificielle (IRIDIA), Université Libre de Bruxelles, Bruxelles, Belgium, Sep. 2004. [Online]. Available: http://people.idsia.ch/ gianni/Papers/thesis.pdf

[12] Y. Zhang, L. D. Kuhn, and M. P. J. Fromherz, "Improvements on Ant routing for sensor networks," in *Proc. Int. Workshop Ant Colony Optim. Swarm Intell.* Lecture Notes in Computer Science, vol. 3172, M. Dorigo, M. Birattari, C. Blum, L. M. Gambardella, F. Mondada, and T. Stützle, Eds. Berlin, Germany: Springer, Sep. 2004, pp. 154–165.

[13] G. D. Caro, F. Ducatelle, and L. M. Gambardella, "Ant HocNet: An ant-based hybrid routing algorithm for mobile ad hoc networks," in *Proc. Int. Conf. Parallel Problem Solving From Nature* Lecture Notes in Computer Science, vol. 3242, X. Yao, E. K. Burke, J. A. Lozano, J. Smith, J. J. Merelo-Guervós, J. A. Bullinaria, J. E. Rowe, P. Tiňo, A. Kabán, and H.-P. Schwefel, Eds. Berlin, Germany: Springer, 2004, pp. 461–470.

[14] Y.-F. Wen, Y.-Q. Chen, and M. Pan, "Adaptive ant-based routing in wireless sensor networks using energy* delay metrics," *J. Zhejiang Univ.-Sci. A*, vol. 9, no. 4, pp. 531–538, Apr. 2008.

[15] X. Li, B. Keegan, and F. Mtenzi, "Ant colony clustering routing protocol for optimization of large-scale wireless sensor networks," in *Proc. 14th Inf. Technol. Telecommun. Conf. (IT&T)*, C. Muntean and P. Pathak, Eds. Dublin, Ireland: National College of Ireland, Oct. 2015, pp. 2–9.

[16] J.-H. Chang and L. Tassiulas, "Maximum lifetime routing in wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 12, no. 4, pp. 609–619, Aug. 2004.

[17] J. Habibi, A. G. Aghdam, and A. Ghrayeb, "A framework for evaluating the best achievable performance by distributed lifetime-efficient routing schemes in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 6, pp. 3231–3246, Jun. 2015.

[18] T. Gui, C. Ma, F. Wang, and D. E. Wilkins, "Survey on swarm intelligence based routing protocols for wireless sensor networks: An extensive study," in *Proc. IEEE Int. Conf. Ind. Technol. (ICIT)*, Taipei, Taiwan, Mar. 2016, pp. 1944–1949.

[19] T. Camilo, C. Carreto, J. S. Silva, and F. Boavida, "An energy-efficient ant-based routing algorithm for wireless sensor networks," in *Proc. Int. Workshop Ant Colony Optim. Swarm Intell.* Lecture Notes in Computer Science, vol. 4150, M. Dorigo, L. M. Gambardella, M. Birattari, A. Martinoli, R. Poli, and T. Stützle, Eds. Berlin, Germany: Springer, 2006, pp. 49–59.

[20] Y. Sun, W. Dong, and Y. Chen, "An improved routing algorithm based on ant colony optimization in wireless sensor networks," *IEEE Commun. Lett.*, vol. 21, no. 6, pp. 1317–1320, Jun. 2017.

[21] E. Amiri, H. Keshavarz, M. Alizadeh, M. Zamani, and T. Khodadadi, "Energy efficient routing in wireless sensor networks based on fuzzy ant colony optimization," *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 7, 2014, Art. no. 768936.

[22] S. Okdem and D. Karaboga, "Routing in wireless sensor networks using an ant colony optimization (ACO) router chip," *Sensors*, vol. 9, no. 2, pp. 909–921, Feb. 2009.

[23] A. M. S. Saleh, B. M. Ali, M. F. A. Rasid, and A. Ismail, "A self-optimizing scheme for energy balanced routing in wireless sensor networks using sensorant," *Sensors*, vol. 12, no. 8, pp. 11307–11333, Aug. 2012.

[24] A. A. Gurav and M. J. Nene, "Multiple optimal path identification using ant colony optimisation in wireless sensor network," *Int. J. Dec. Mobile Netw.*, vol. 5, no. 5, pp. 119–128, Oct. 2013.

[25] T. Weise, M. Zapf, and K. Geihs, "Evolving proactive aggregation protocols," in *Proc. Eur. Conf. Genetic Program.* Lecture Notes in Computer Science, vol. 4971, M. O'Neill, L. Vanneschi, S. M. Gustafson, A. I. Esparcia-Alcázar, I. de Falco, A. D. Cioppa, and E. Tarantino, Eds. Berlin, Germany: Springer-Verlag, 2008, pp. 254–265.

[26] T. Weise, K. Geihs, and P. A. Baer, "Genetic programming for proactive aggregation protocols," in *Proc. Genetic Program. Proactive Aggregation Protocols* Lecture Notes in Computer Science, vol. 4431, B. Beliczy ski, A. Dzieli ski, M. Iwanowski, and B. Ribeiro, Eds. Berlin, Germany: Springer-Verlag, 2007, pp. 167–173.

[27] S. K. Leabi and T. Y. Abdalla, "Energy efficient routing protocol for maximizing the lifetime in WSNs using ant colony algorithm and artificial immune system," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 3, pp. 100–108, 2016.

[28] S. Ghosh, S. Mondal, and U. Biswas, "Fuzzy C means based hierarchical routing protocol in WSN with ant colony optimization," in *Proc. 2nd Int. Conf. Appl. Theor. Comput. Commun. Technol. (ICATccT)*, Jul. 2016, pp. 348–354.

[29] R. Krishnapuram and J. M. Keller, "The possibilistic C-means algorithm: Insights and recommendations," *IEEE Trans. Fuzzy Syst.*, vol. 4, no. 3, pp. 385–393, Aug. 1996.

[30] V. Rosset, M. A. Paulo, J. G. Cespedes, and M. C. V. Nascimento, "Enhancing the reliability on data delivery and energy efficiency by combining swarm intelligence and community detection in large-scale WSNs," *Expert Syst. Appl.*, vol. 78, pp. 89–102, Jul. 2017.

[31] S. Kaur and R. Mahajan, "Hybrid meta-heuristic optimization based energy efficient protocol for wireless sensor networks," *Egyptian Inform. J.*, vol. 19, no. 3, pp. 145–150, Nov. 2018.

[32] P. S. Rossi, D. Ciuonzo, and T. Ekman, "HMM-based decision fusion in wireless sensor networks with noncoherent multiple access," *IEEE Commun. Lett.*, vol. 19, no. 5, pp. 871–874, May 2015.

[33] D. Ciuonzo, G. Romano, and P. S. Rossi, "Channel-aware decision fusion in distributed MIMO wireless sensor networks: Decode-and-fuse vs. Decode-then-fuse," *IEEE Trans. Wireless Commun.*, vol. 11, no. 8, pp. 2976–2985, Aug. 2012.

[34] D. Ciuonzo, G. Romano, and P. S. Rossi, "Optimality of received energy in decision fusion over Rayleigh fading diversity MAC with non-identical sensors," *IEEE Trans. Signal Process.*, vol. 61, no. 1, pp. 22–27, Jan. 2012.

[35] M. Najimi, A. Ebrahimzadeh, S. M. H. Andargoli, and A. Fallahi, "A novel sensing nodes and decision node selection method for energy efficiency of cooperative spectrum sensing in cognitive sensor networks," *IEEE Sensors J.*, vol. 13, no. 5, pp. 1610–1621, May 2013.

[36] A. M. Zungeru, L.-M. Ang, and K. P. Seng, "Classical and swarm intelligence based routing protocols for wireless sensor networks: A survey and comparison," *J. Netw. Comput. Appl.*, vol. 35, no. 5, pp. 1508–1536, Sep. 2012.

[37] H. Zhang, X. Wang, P. Memarmoshrefi, and D. Hogrefe, "A survey of ant colony optimization based routing protocols for mobile ad hoc networks," *IEEE Access*, vol. 5, pp. 24139–24161, 2017.

[38] T. Stützle and M. Dorigo, "A short convergence proof for a class of ant colony optimization algorithms," *IEEE Trans. Evol. Comput.*, vol. 6, no. 4, pp. 358–365, Aug. 2002.

[39] F. Neumann and C. Witt, "Runtime analysis of a simple ant colony optimization algorithm," *Algorithmica*, vol. 54, no. 2, pp. 243–255, 2009.

[40] B. Doerr and D. Johannsen, "Refined runtime analysis of a basic ant colony optimization algorithm," in *Proc. IEEE Congr. Evol. Comput.*, Singapore, Sep. 2007, pp. 501–507.

[41] Y. Zhou, "Runtime analysis of an ant colony optimization algorithm for TSP instances," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 1083–1092, Oct. 2009.

[42] A. Lissovoi and C. Witt, "Runtime analysis of ant colony optimization on dynamic shortest path problems," *Theor. Comput. Sci.*, vol. 561, pp. 73–85, Jan. 2015.

[43] A. Lissovoi and C. Witt, "A runtime analysis of parallel evolutionary algorithms in dynamic optimization," *Algorithmica*, vol. 78, no. 2, pp. 641–659, 2017.

[44] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.

[45] M. Saleem, G. A. Di Caro, and M. Farooq, "Swarm intelligence based routing protocol for wireless sensor networks: Survey and future directions," *Inf. Sci.*, vol. 181, no. 20, pp. 4597–4624, Oct. 2011.

[46] X. Li, B. Keegan, and F. Mtenzi, "Clustering opportunistic ant-based routing protocol for wireless sensor networks," in *Proc. 7th Int. Conf. Comput. Eng. Netw.*, vol. 299. Trieste, Italy: Sissa Medialab, 2017, pp. 1–10.

[47] NS2. (2016). *The Network Simulator*. Accessed: Mar. 28, 2019. [Online]. Available: http://www.isi.edu/nsnam/ns/

[48] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu, "Advances in network simulation," *Computer*, vol. 33, no. 5, pp. 59–67, May 2000.

[49] S. McCann. (Mar. 2019). *Official IEEE 802.11 Working Group Project Timelines–2019-03-26*. Accessed: Mar. 28, 2019. [Online]. Available: http://grouper.ieee.org/groups/802/11/Reports/802.11_Timelines.htm

[50] T. Stützle, M. López-Ibáñez, P. Pellegrini, M. Maur, M. M. de Oca, M. Birattari, and M. Dorigo, "Parameter adaptation in ant colony optimization," in *Autonomous Search*. Berlin, Germany: Springer, 2011, pp. 191–215.

XINLU LI received the M.S. degree in computer science from Anhui University, Anhui, China, in 2009, and Ph.D. degree in computer science from Technological University Dublin (TU Dublin), Dublin, Ireland, in 2019.

Since 2010, he has been a Lecturer with the Computer Science Department, Hefei University, Anhui. From 2014 to 2018, he was a Teaching Assistant with the School of Computing, TU Dublin. His research interest includes energy-efficient routing protocol design for large-scale wireless sensor networks based on swarm intelligence algorithms.

BRIAN KEEGAN (M'04–SM'19) was born in Dublin, Ireland. He received the B.Eng degree, the M.Phil. degree in electronic engineering, and the Ph.D. degree in wireless communications engineering from the Dublin Institute of Technology (DIT), in 2002, 2005, and 2010, respectively.

Prior to working with the School of Computer Science, he worked with Sun Microsystems (Oracle), Dublin, as part of the Solaris ON-PIT Group. He was with Cisco Systems, San Jose, as part of the WNBU Group. He is currently a Senior Lecturer with the School of Computer Science, Technological University Dublin. He teaches undergraduate and postgraduate courses and is supervising Ph.D. students. Teaching duties include network engineering, secure systems development, and systems administration. His research interests include computer networks, WLAN, Wireless mesh, routing, VoIP, network programming, computer security, software-defined networking, system administration, the Internet of Things, and STEM/STEAM.

FREDRICK MTENZI is currently the Team Leader of the Ubiquitous Computing Research Group (UCRG), School of Computing, Dublin Institute of Technology. He is also supervising postgraduate students at master's and Ph.D. levels. He has authored or coauthored over 60 scientific publications in algorithm design, energy aware mobile ad hoc networks, computer games, computer education, and security. He has organized and chaired peer-reviewed international conferences in computer games, including AI, animation, mobile, educational and serious games, and Internet technology and secured transactions. His research interests include the design and implementation of energy-aware routing algorithms for mobile ad hoc networks (MANET), security issues in MANET and e-healthcare, application of mobile computing to healthcare, high-performance e-commerce, knowledge management, and development of culturally sensitive educational models (Ireland, China, and Tanzania).

Dr. Mtenzi is also a Professional Member of the Association for Computing Machinery (ACM), the IEEE Technical Committee, and the Information Systems Security Association (ISSA). He is also a member of the Editorial Board of the International Journal for Digital Society (IJDS), the Special Issue of the *Journal of Computer Science* and the *International Journal of Internet Technology and Secured Transactions* (IJITST).

THOMAS WEISE received the Diplom Informatiker (equivalent to M.Sc. degree in computer science) from the Chemnitz University of Technology, in 2005, and the Ph.D. degree in computer science from the University of Kassel, in 2009. In 2009, he joined the University of Science and Technology of China (USTC) as a Postdoctor and subsequently became an Associate Professor at the USTC-Birmingham Joint Research Institute in Intelligent Computation and Its Applications (UBRI), USTC. In 2016, he joined Hefei University as a Full Professor to found the Institute of Applied Optimization (IAO) at the Faculty of Computer Science and Technology. He has more than 80 scientific publications in international peer-reviewed journals and conferences. His book *Global Optimization Algorithms – Theory and Application* has been cited 840 times. He has acted as a Reviewer, an Editor, or a Programme Committee Member at 70 different venues. He is a member of the Editorial Board of the *Applied Soft Computing* journal.

MING TAN has been a Professor with the Computer Science Department, Hefei University, Anhui, China, since 2010. His research interests include network control, multimedia technology, and network programming. He has more than 20 scientific publications in international peer-reviewed journals and conferences.