Efficient Approach for Maximizing Lifespan in Wireless Sensor Networks by Using Mobile Sinks

Hoc Thai Nguyen, Linh Van Nguyen, and Hai Xuan Le

Recently, sink mobility has been shown to be highly beneficial in improving network lifetime in wireless sensor networks (WSNs). Numerous studies have exploited mobile sinks (MSs) to collect sensed data in order to improve energy efficiency and reduce WSN operational costs. However, there have been few studies on the effectiveness of MS operation on WSN closed operating cycles. Therefore, it is important to investigate how data is collected and how to plan the trajectory of the MS in order to gather data in time, reduce energy consumption, and improve WSN network lifetime. In this study, we combine two methods, the cluster-head election algorithm and the MS trajectory optimization algorithm, to propose the optimal MS movement strategy. This study aims to provide a closed operating cycle for WSNs, by which the energy consumption and running time of a WSN is minimized during the cluster election and data gathering periods. Furthermore, our flexible MS movement scenarios achieve both a long network lifetime and an optimal MS schedule. The simulation results demonstrate that our proposed algorithm achieves better performance than other well-known algorithms.

Keywords: WSNs, Cluster head election, Optimal trajectory of mobile sink, Salesman problem.

I. Introduction

In recent years, many authors have focused their investigations on wireless sensor network (WSN) lifetime improvement by mobility-based technology. Some studies [1]–[2] tried to change the initial deployment of the network by mobile nodes; some other approaches [3]–[5] utilized MSs to collect data in order to enhance the sensing coverage, connectivity, and network lifetime as well. However, there has been little research work on WSN closed operating cycles to find an optimal solution for improving the network lifetime. Therefore, in this paper, we focus on extending the network lifetime by improving system energy efficiency in a closed cycle, including cluster head election, data collection, and data transmission.

It is known that in order to achieve efficient energy management, clustering is a promising technique by which all cluster member nodes (CNs) transmit their sensing data to a cluster head (CH) node before forwarding them to the MS. Furthermore, the energy resources and traveling time of an MS can also be limited. Therefore, it needs efficient energy use in order to collect the sensing data with minimum energy consumption, and within the predefined time deadline. In this paper, we also consider several problems of mobility-based technology, such as where, when, and how to obtain sensing data while minimizing sensor node energy consumption in a given limited time.

It is proved that a CH will consume much more energy than the normal nodes. Therefore, the choice of the CH is highly critical for balancing the energy consumption among the nodes to increase the WSN network lifetime.

Many studies on CH election in WSNs [6], [7] in order to reduce energy consumption and prolong network lifetime have been performed. Low-energy adaptive clustering hierarchy (LEACH) [8] is one of these well-known clustering protocols.

Manuscript received Sept. 6, 2016; revised Feb. 19, 2017; accepted Mar. 28, 2017.

Hoc Thai Nguyen (corresponding author, nguyenth@hit.bme.hu) is with the Department of Networked Systems & Services, Budapest University of Technology & Economics, Hungary.

Linh Van Nguyen (vlnguyen@ntu.edu.sg) is with the School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore.

Hai Xuan Le (xhaicuwc.edu.vn@gmail.com) is with the Hanoi University of Science and Technology, Vietnam.

https://doi.org/10.4218/etrij.17.0116.0629

The key technology of LEACH is randomized rotation of local CHs to distribute energy among the nodes in the network. One normal node (n) may be a CH in the recent (t mod ($1/p$)) rounds with probability P:

$$P(n) = \begin{cases} \dfrac{p}{1 - p\left(t \bmod (1/p)\right)} & \text{if } n \in Q, \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where p is the desired percentage of CHs, t denotes the current round, and Q indicates the set of nodes that have not been a CH in the last ($1/p$) rounds. As indicated in [9], LEACH can save energy consumption using a time division multiple access schedule. Unfortunately, this clustering protocol still has several shortcomings: (i) A single hop directly between a CH and the base station (BS) will be impossible with a large sensing field, and it may dissipate much energy for data transmission; (ii) it is unreasonable for load balancing when changing the CHs if it does not take into account the residual energy and geographic location of each sensor node; and (iii) it is not certain that CHs will be uniformly distributed throughout the network when a normal node is chosen to be a CH on the basis of a random number. Therefore, energy dissipation becomes inefficient. In order to overcome these shortcomings, some improved versions of LEACH have been proposed. The Impro-LEACH algorithm in [10] is one example that can improve the network lifetime by modifying the LEACH threshold formula. In that study, a node that has higher residual energy (E_r), would have a higher probability ($P(n)$) of becoming a CH. This probability also depends on the number of times this node becomes a CH or vice CH (VCH), indicated by CH_t and VCH_t, respectively. Therefore, the authors in [10] tried to modify the LEACH threshold formula and improve the network lifetime by reducing the frequency of re-clustering. In the steady phase, CHs also collect CNs' information, which helps a CH choose a VCH in a later period of the steady phase. A CH will choose one sensor node that has the largest residual energy in the cluster to take over the CH role, and the selected node is called the VCH. In this way, it is proved that this approach can improve the energy balancing among nodes in the network. However, it still compares a random number with the probability to elect CHs as proposed in LEACH. Therefore, the method proposed in [10] does not overcome the drawbacks of the LEACH method.

Moreover, these methods are still essentially designed for CH election, and do not take into account the combination of energy consumption in each sensor node and the energy balance expenditure among nodes in the network. To eliminate this problem, we propose an efficient algorithm for choosing: (i) The number of clusters in a network, which guarantees energy balance among nodes; (ii) The best location of a single CH for each cluster, where the CH not only can receive data from all CNs with the minimum energy consumption, but also guarantees the energy balance among nodes in the network.

We also consider scenarios related to MS trajectory in mobile wireless sensor networks. Herein, an optimal trajectory of the MS is obtained when both minimizing energy consumption and the constraint time in data gathering are met. As proposed in [4], an MS is utilized to gather sensed data in the network. During data gathering, the residual energy information of the CHs is also collected by the MS for scheduling its next movement strategy. However, the MS always moves towards energy-rich areas, thereby causing huge data delivery latency from those areas where more events occur. Another interesting idea for the data collecting schedule is proposed in [11]. In that study, Tashtarian and others proposed an energy-efficient data collection algorithm that collects data in reporting by using the mixed integer linear programming mathematical model. However, if the MS has not enough time to visit all CHs in the network, the transmission ranges of all CHs must increase to create two or more "range overlapped" areas. Therefore, the MS can save its traveling time by moving to these range overlapped areas to collect data. Unfortunately, the energy consumption of the CHs increases exponentially as their radio transmission ranges grow. The long radio transmission range may disrupt communications and increase the data transmission delay. Moreover, it is not reasonable when all CH transmission ranges increase at the same level while they have different residual energy. Therefore, in our approach the CH transmission ranges are increased only after maximizing the speed of the MS, and the transmission range of each CH will be increased on the basis of its residual energy.

In the literature, some authors utilized the traveling salesman problem (TSP) to find the optimal trajectory of the MS [12], [13]. In these works, a node may be visited more than once before all other nodes are visited. However, in our model, we keep the original TSP technique to find the optimal trajectory, but the time cost of the MS is total time for moving and collecting data. In this paper, we investigate the problem of controlled mobility to find optimal movement strategy for the MS, which can reduce the total communication cost within a predefined reporting time for each data transmission round.

The contributions of this paper are highlighted as follows:

- We performed a detailed review of the literature about CH election, and utilizing mobility technology in WSNs, which constitute the main features of this study.
- Unlike previous solutions for CH election (CHE), our proposed CHE algorithm tries to find the best candidate node to become a CH for each geographical region, which has both high residual energy and low communication cost. This enables us to investigate the effect of residual energy and

https://doi.org/10.4218/etrij.17.0116.0629

balanced energy among nodes on network lifetime improvement. The average improvement in network lifetime of CHE versus the Impro-LEACH and LEACH algorithms is up to 10.64% and 26.2%, respectively.

- In this paper, we introduce some solutions to find the optimal trajectory of the MS. The optimal movement strategy (OMS) algorithm exhibits better performance than conventional strategies. The simulation results show that our proposed OMS algorithm can improve the lifespan of the network compared with random, static, and fixed-trajectory scenarios.

The remainder of the paper is organized as follows: In Section II, we describe the system model and its basic assumptions. Section III describes the proposed algorithm. The performance of our algorithm is analyzed and compared to the traditional algorithms in Section IV. Finally, in Section V, we conclude this paper and discuss some directions for future work.

II. System Model and Assumptions

1. Basic Assumptions

Before presenting our algorithms, we first specify the general assumptions about the WSN model we used in this study.

- We consider a network with N homogeneous sensor nodes randomly distributed. They have a limited initial energy E_0 and are stationary after deployment.
- An MS node can travel freely in the sensing field with unconstrained energy and storage capacity. During the moving time, the MS does not receive any data packets, because the CHs transmit their data only when they can communicate directly with the MS. We assume that the MSs periodically returns to the support center for recharging. Therefore, the energy consumption of the MS operation and movement does not affect the network lifetime.
- The geographic position of the network nodes is known to the MS and used to find the optimal trajectory of the MS.
- In this study, we define the network lifetime of a sensor network as the number of rounds until a certain percentage (ϕ = 85%) of the network nodes run out of power.
- CHs transmit their buffered data to the MS during a specified interval time called the reporting time ξ.
- To evaluate the network lifetime, round (T_{Γ}) is a cycle in which the MS successfully receives sensed data from all nodes in the network:

$$T_{\Gamma} = \varsigma + \sum_{i=1}^{N} \tau_i + \sum_{i=1}^{M} t_i + \sigma(t), \qquad (2)$$

where ς indicates the cluster election time for M areas, τ_i is the data collection time from sensor node S_i to its corresponding cluster head node CH_i, t_i indicates the period of time the MS stops at CH_i until all buffered data can be collected, and σ is the total traveling time of the MS.

- Each sensor node generates a data sample of an m-bit data packet every round and needs to transmit it to its CH.

2. Network Model

Figure 1 briefly presents a network structure with N = 30 sensor nodes deployed randomly in the area of interest A. For clustering purposes, area A is divided into M = 6 equal parts $\{A_1, A_2, \ldots, A_M\}$. The number of sub-areas (M) depends on the size of the monitored area and the node density (ρ), which guarantees that there is no need to send more than one MS to a CH for data gathering.

Our objective is to find the best candidate nodes for CH election, which can minimize the communication cost and balance the energy consumption among all sensor nodes in the network. On the other hand, we also find the optimal movement strategy for the MS, which helps the MS to collect sensing data from all CHs with minimum energy expenditure and within a desired time deadline.

3. Energy Model

Similar to [14], we use a simple model for the radio hardware energy dissipation to calculate the energy consumption. According to whether the distance d between source node and the destination node is shorter or longer than a threshold distance $d_0 = \sqrt{\varepsilon_{\text{fs}} / \varepsilon_{\text{mp}}}$, a free space or multi-path fading channel model will be used, respectively. ε_{fs} and ε_{mp} denote the free space and multi-path transmitter amplifier model, respectively. The amount of energy E_{TX} will be consumed at

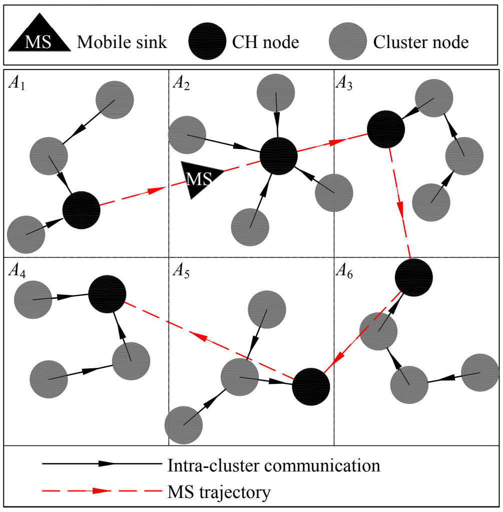


Fig. 1. Structure of the wireless sensor network.

the source node, if it transmits an m-bit data packet over a distance d to the destination node. It can be calculated by

$$E_{\mathrm{TX}} = mE_{\mathrm{elec}} + E_{\mathrm{amp}}(m,d) = mE_{\mathrm{elec}} + m\varepsilon d^{\chi}, \tag{3}$$

where $\begin{cases} \varepsilon = \varepsilon_{\mathrm{fs}};\, \chi = 2 & \text{if } d < d_0, \\ \varepsilon = \varepsilon_{\mathrm{mp}};\, \chi = 4 & \text{otherwise.} \end{cases}$

E_{elec} is the electronic energy, which depends on the digital coding, modulation, filtering, and spreading of the signal. $E_{\mathrm{amp}}(m, d)$ is the energy needed by the radio amplifier circuit to send m bits d meters. To receive an m-bit data packet, the destination node also has to expend E_{RX} amount of energy, calculated as

$$E_{\mathrm{RX}} = mE_{\mathrm{elec}}. \tag{4}$$

The load $L_k(t)$ of node S_k during round t is the total power that the node consumes to receive and transmit data in that round.

$$L_k(t) = E_{\mathrm{TX}}(t) + E_{\mathrm{RX}}(t). \tag{5}$$

The lifespan of one sensor node T_k refers to the time when its residual energy is less than a threshold (θ). Thus, we have

$$E_0 - \sum_{t=1}^{T_k} L_k(t) \le \theta, \{k = 1, \ldots, N\}. \tag{6}$$

It is clear that the higher the load $L_k(t)$, the shorter the lifetime of node S_k. Hence, minimizing the network load and improving the load balancing among the WSN nodes are prerequisites for achieving the maximum network lifetime.

III. Proposed Approach

1. Cluster Heads Election Algorithm

In this subsection, we focus on the CH election, which helps to save sensor node energy power and prolong the network lifetime. To do this, the CHs should be spread evenly [15]. Therefore, in this study, the sensing field was divided into M equal regions A_i, $\{1 \le i \le M\}$. These regions are called clusters, and the CH election is performed within each geographical region. For simplicity, without loss of generality, we focus on CH election for N_k sensor nodes randomly deployed in region A_k. To begin the CH election procedure, every node broadcasts a "HELLO" message to the network by a controlled flooding method. The format of the "HELLO" is shown in Fig. 2.

Node ID (S_k)	Location (x_k, y_k)	Total length (l_k)	Residual energy (E_k)

Fig. 2. Format of the "HELLO" message broadcast by the sensor nodes.

After receiving the "HELLO" message from source node S_k, one sensor node S_c can know whether it is inside or outside the S_k's region by comparing its location with S_k's position. If it is outside the region, it will simply drop the message. If it is inside, the prior values (PV_k, PV_c) will be compared with each other (these parameters described later in (8)). If $PV_c \le PV_k$, S_c will not broadcast its "HELLO" message and only re-broadcast S_k's "HELLO" message. Otherwise, S_k's "HELLO" message will be dropped and S_c will broadcast its own "HELLO" message to the region. As a result, one sensor node knows the ID and location of its CH, whose prior value is the highest in the region. After becoming a CH node, it will announce itself as a CH by broadcasting a CH announcement message (CHAM) to the network. This CHAM contains the CH node's ID and location. After receiving these CHAMs, one normal sensor node will choose the closest CH (S_k) as its CH node. This enables some sensor nodes, which are in other regions but closer to the CH S_k than the CH in their regions, to join S_k's cluster in order to reduce their energy consumption. The shortest path routing from each cluster node to its CH will be found for data transmission purposes. In order to gain efficiency in using the CH election method, we state the following Definition:

Definition 1: Assuming that N_i sensor nodes are deployed randomly over a sub-region of interest A_i. E_k indicates the residual energy of node S_k. l_k is the total length of the shortest path routing from node S_k to all ($N_i - 1$) nodes in A_i. The energy expenditure of one cluster can be minimal if and only if the candidate node S_k has

$$\begin{cases} l_k = \min\{l_1, \ldots, l_{N_i}\}, \\ E_k = \max\{E_1, \ldots, E_{N_i}\}. \end{cases} \tag{7}$$

In practical applications, it is not easy to find one candidate node that satisfies (7) in a large area with a large number of CNs. Therefore, in this study, one candidate node will be a CH if it has a maximum priori value, given as

$$PV_k = \max\left\{\alpha \frac{1}{l_k} + \beta \frac{E_k(t)}{E_0}\right\}, k = 1, \ldots, N_i, \tag{8}$$

where α, β are positive real numbers such that $\alpha + \beta = 1$. These are the weighting factors, which are found heuristically in the course of the optimization. Note that the relative importance of the objectives depends on these heuristic constants α, β. In our numerical tests, we ran the CHE algorithm 1,000 times for each set of values (α, β). The most profitable values of (α, β), which are the best balance between l_k and $E_k(t)$, were chosen for the subsequent simulations.

The CH election procedures for each area can be described in more detail by following the steps of CHE algorithm. In Fig. 3, we illustrate an example CH election process for sub-area A_1. Figure 3 depicts the sensor deployment of $N_1 = 6$ sensor nodes in the area A_1 with the percentage residual energy.

https://doi.org/10.4218/etrij.17.0116.0629

Algorithm 1. CHE algorithm

```
Step 1. Compute the shortest path $\Re^{s}_{k\lambda}$ from a single source node
    (node $S_k$) to $(N_i - 1)$ normal nodes in $A_i$ where $k = \{1, \dots, N_i\}$; $\lambda = \{1, \dots, N_i - 1\}$ $\lambda \neq k$.
Step 2. Calculate the total length of the shortest path from node $S_k$ to
    $(N_i - 1)$ normal nodes in the sub-region of interest.
    $l_k = \sum_{\lambda=1}^{N_i-1} \Re^{s}_{k\lambda}, k = \{1, \dots, L\}, \lambda \neq k$
Step 3. Calculate the priori value $PV_k$ for each node
    $PV_k = \left\{\alpha \frac{1}{l_k} + \beta \frac{E_k(t)}{E_0}\right\}$.
Step 4. A normal node $S_k$ becomes a CH for the sub-area $(A_i)$ if $PV_k = \max\{PV_1, PV_2, \dots, PV_{Ni}\}$.
Step 5. Repeat Steps 1through 4 until CH election for all sub-areas in
    the network.
```

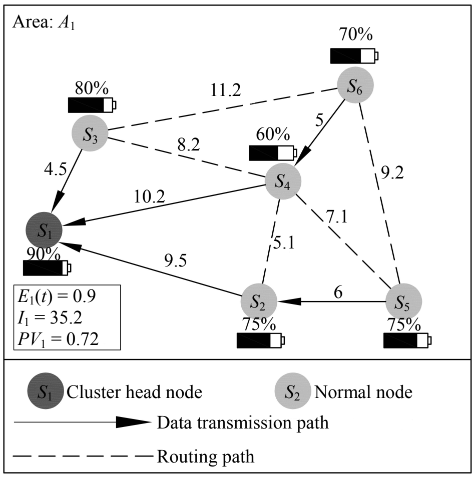


Fig. 3. Cluster communication mechanism if node S_1 becomes A_1's CH node.

Table 1. Example for cluster-head election.

Node ID	Node location	$E_i(t)$ (%)	l_i	Priori value
S_1	**(16, 10)**	**90**	**35.2**	**0.726**
S_2	(25, 7)	75	33.8	0.606
S_3	(18, 14)	80	36.1	0.646
S_4	(26, 12)	60	35.6	0.486
S_5	(31, 7)	75	40.0	0.605
S_6	(29, 16)	70	40.7	0.565

In this example, we set ($\alpha = 0.2$; $\beta = 0.8$). The priori values of all sensor nodes were calculated, and are given in Table 1.

In this example, with the highest priori value, node S_1 is chosen as the CH, and the shortest path routing of this cluster is described in Fig. 3.

By analyzing the simulation results, we can state several key features of our proposed CHE algorithm: First, uniform distribution of CHs in each geographic region causes more efficient energy use and balanced load in the network. Second, the number of CHs increases with the size of the monitored area. Therefore, it can reduce the packet drop rate and prevents the black hole problem. The last and the most important feature is that one normal node can be elected as a CH depending not only on its residual energy but also on the cluster's communication cost.

2. Optimizing the Trajectory of the Mobile Sink

In this section, we present the MS trajectory optimization (MSTO) algorithm, whose basic idea is to investigate the path with the smallest time spent on traveling and gathering data. To facilitate the MSTO algorithm, we first introduce the following theorem, which helps to construct the infrastructure of the MSTO algorithm.

Theorem 1: Let $K_{\min}$ denote the number of MSs used in data collection; ξ_0 is a threshold of reporting time, and R indicates the data transmission rate of the MS. It will be an efficient data gathering scheme without loss of sensed data if and only if

$$\frac{1}{K_{\min}}\left(\frac{\Re(t)}{v(t)} + \frac{mN}{R}\right) \leq \xi_0, \tag{9}$$

where $\Re(t)$ is the total shortest path length of M CHs in the network and $v(t)$ is the velocity of the MS at current round t.

Proof. In order to collect data effectively, all monitored data should be collected in time to avoid the buffer overflow problem occurring at CHs. Thus, the reporting time at every cycle time $\xi(t)$ must remain static at the desired value ξ_0. Its value at instant round t can be calculated by

$$\xi(t) = \sum_{i=1}^{M} t_i + \sigma(t). \tag{10}$$

If cluster of CH_i has V_i CNs and R indicates the data transmission rate between a CH node and the MS, the time spent by the MS for data collection at CH_i will be

$$\underset{(1 \leq i \leq M)}{t_i} = \frac{mV_i}{R}, \tag{11}$$

where m denotes the data packet size generated by each sensor node in one round. Total traveling time $\sigma(t)$ of the MS at the current round t can be computed as

$$\sigma(t) = \frac{\Re(t)}{v(t)}. \tag{12}$$

By inserting (12) and (13) into (11), the reporting time can be rewritten as

$$\xi(t) = \frac{mN}{R} + \frac{\Re(t)}{v(t)}. \tag{13}$$

Thus, the total time spent by $K_{\min}$ MSs is given as

$$\xi(t) = \frac{1}{K_{\min}}\left(\frac{\Re(t)}{v(t)} + \frac{mN}{R}\right) \le \xi_0. \qquad (14)$$

■

If ξ_0 is high enough, a single MS with constant velocity can harvest all the captured data from M CHs in order to reduce the communication cost. Unfortunately, in practical applications, this is not enough time for data collection. In this situation, we have two scenarios to solve this problem: (i) Scenario 1. Change the speed of a single MS; (ii) Scenario 2. Utilize more than one MS to collect data.

The problem here is how to reduce the total length of the MS's trajectory in order to minimize the execution time and the energy dissipation for data collection [16]. With these definitions, the optimal trajectory of the MS can be formulated as follows.

Objective:

$$\max\left\{\frac{1}{K_{\min}}\left(\frac{\Re(t)}{v(t)} + \frac{mN}{R}\right)\right\} \le \xi_0. \qquad (15)$$

Subject to:

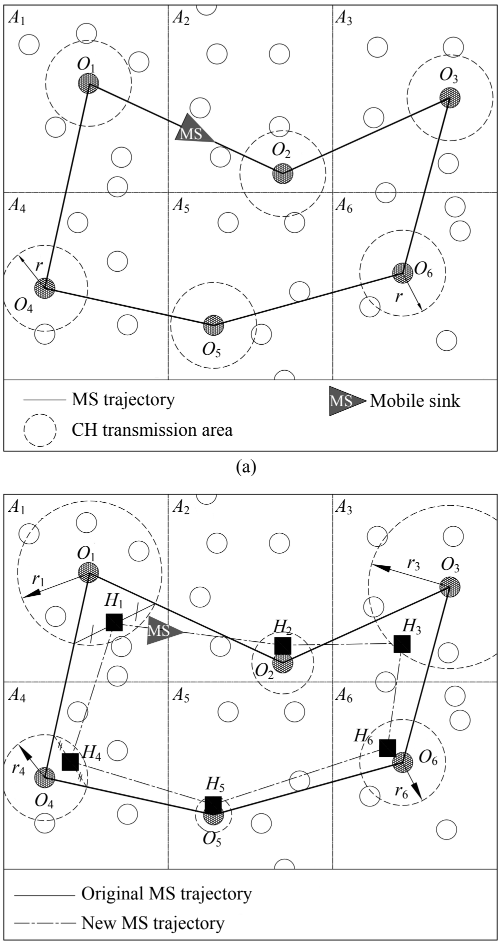
$$\begin{aligned} &C1: 1 \le K_{\min} \le M \\ &C2: 0 < t_i \le \xi_0, \text{for all } i \text{ with } 1 \le i \le M \qquad (16) \\ &C3: \Re(t) = \{CH_1, \ldots, CH_M\}. \end{aligned}$$

Constraint ($C1$) ensures that there is no need to send more than one MS to one CH at the same time for data gathering. Constraint ($C2$) restricts the data collection time from one CH to the MS. Finally, ($C3$) is the shortest path routing between M CHs in the current round (t). According to (15), the velocity of the MS or the number of MSs will be calculated depending on the chosen scenario.

Scenario 1. In this scenario, a single MS ($K_{\min} = 1$) moves along the shortest path of all CHs to collect sensed data. Its velocity $v(t)$ will now vary according to the change of the shortest path length $\Re(t)$:

$$v(t) \ge \frac{\Re(t)R}{\xi_0 R - mN}. \qquad (17)$$

However, if $\Re(t)$ is too long and $v(t)$ may be greater than a predefined speed $v_{\max}$, the speed of the MS will be kept at $v_{\max}$ in these cases. The data transmission range of some CHs can then be changed in order to reduce the total length of the shortest path of CHs. Unfortunately, the larger a CH's radio transmission range, the longer the process of data packets toward their final destinations. Therefore, the energy dissipation by each CH with the radio transmission range grows exponentially as given in [17]. The problem here is determining which CHs must increase their transmission ranges. To answer this question, we present a solution that can find the best data collection places for the MS after changing the transmission range of each CH. This provides better results than the approach proposed in [11], which may lead to an imbalanced energy dissipation of the sensor nodes in a WSN when the transmission range of all CHs are increased equally.



(a)

(b)

Fig. 4. Sink movement strategy for Scenario 1: (a) sink movement strategy for $\Re(t) \le \Re_{\max}$ and (b) sink movement strategy for $\Re(t) > \Re_{\max}$.

For simplicity, we consider a sensing field with six CHs as depicted in Fig. 4. As shown in Fig. 4(a), if the total length of the MS's trajectory ($\Re(t) = O_1O_2 + O_2O_3 + O_3O_6 + O_6O_5 + O_5O_4 + O_4O_1$, where O_i denotes the best place for the MS to gather data from CH_i), is less than the longest length ($\Re_{\max} = \xi_0 * v_{\max}$), the MS needs only to change its speed in order to collect the data in time. Otherwise, the communication range of each CH will be changed in order to decrease the MS's trajectory. In this case, two parameters are recomputed: the new transmission range of each CH and the new optimal trajectory of the MS.

A. Evaluating CH Transmission Range

Let us assume that $E_i(t)$ is the residual energy of CH_i with

N_i CNs at the current round (t), and E_{DA} is the energy consumption for data aggregation. The maximum distance that CH_i can transmit $N_i m$ bits to the MS is

$$d^i_{\max} = \sqrt[\chi]{\frac{E_i(t) - \left[\left(2N_i - 1\right)E_{\mathrm{elec}} + N_i m E_{\mathrm{DA}}\right]}{N_i m \varepsilon}}. \tag{18}$$

The transmission range (tr_i) of the CH_i will be increased by

$$tr_i = \lambda d^i_{\max}, \tag{19}$$

where λ is a coefficient that increases from 0 to 1. According to (19), the increase in radio transmission range of each CH depends on its residual energy. Thus, it guarantees the energy balance among sensor nodes in the network.

B. Evaluating the Best Data Collection Places for the MS

If $\mathfrak{R}(t) > \mathfrak{R}_{\max}$, the coefficient λ will be increased from 0 to 1 in order to increase the CHs transmission ranges until $\mathfrak{R}(t) = \mathfrak{R}_{\max}$. For each increase in step, the trajectory of the MS will be recomputed, and the best data collection places for the MS now are H_i, $i = 1, \ldots, 6$. The change in the MS's trajectory is shown in Fig. 4(b). After changing the CHs' transmission ranges, the total length of the MS's trajectory now can be calculated as follows. $\mathfrak{R}'(t) = H_1H_2 + H_2H_3 + H_3H_6 + H_6H_5 + H_5H_4 + H_4H_1$. It is easily proven that $\mathfrak{R}'(t) \le \mathfrak{R}_{\max}$, and if $\mathfrak{R}'(t) = \mathfrak{R}_{\max}$, the real transmission range of the CH_i will be $r_i' = O_iH_i$.

Scenario 2. In this scenario, the velocity of the MSs is kept constant at a low level, and the number of MSs needed for data gathering is $K_{\min}$:

$$\frac{1}{K_{\min}}\left(\frac{\sum_{i=1}^{K_{\min}} \mathfrak{R}_i(t)}{v(t)} + \frac{mN}{R}\right) \le \xi_0, \tag{20}$$

where $\mathfrak{R}_i(t)$ is traveled path length of the MS ith.

Hence, the main problem is how to schedule the movements of $K_{\min}$ MSs, which start and end at a single depot, to visit all CHs within the desired time deadline. We assume that this single depot is the network control center (NCC), which can monitor and control all operations in the system, including number, velocities, and trajectories of the MSs in order to avoid conflicts among them. In this study, the NCC is located at the center of the network field. Similar to the multiple traveling salesman problem (mTSP) [18], each CH must be visited exactly once by an MS in each round.

The strategy of each MS is optimized by minimizing the total cost traveled and within the reporting time. For simplicity without loss of generality, the mTSP in this case can be determined on a graph $G = (V, E)$, where V is the set of M CHs and E is the set of edges. Let $C = (t_{ij})$ denotes a cost (execution time for data propagation and the MS's traveling from node i to node j) matrix associated with E. We define the following binary variable:

$$x_{ij} = \begin{cases} 1 & \text{if edge } (i, j) \text{ is used on the tour,} \\ 0 & \text{otherwise.} \end{cases} \tag{21}$$

Then, a general scheme of the assignment-based directed integer linear programming formulation of the mTSP for one MS(kth) can be given as

$$\max\left\{\sum_{i=1}^{M_{k-1}} \sum_{j=i+1}^{M_k} t_{ij}x_{ij}\right\} \le \xi_0. \tag{22}$$

M_k is total number of CHs is visited by MS kth; subject to:

$$\sum_{i=1}^{M_k} x_{ij} = 1, j = 1, \ldots, M_k, \tag{23}$$

$$\sum_{i=1}^{M_{k-1}} \sum_{j=i+1}^{M_k} x_{ij} = M_k, \tag{24}$$

$$\sum_{i=1}^{K_{\min}} M_i = M, \tag{25}$$

$$x_{ij} \in \{0,1\}, \forall(i, j) \in E, \tag{26}$$

$$0 < t_{ij} \le \xi_0. \tag{27}$$

Algorithm 2. MSTO algorithm

```
1: Start tour $T_k$ for $MS_k$ by setting $T^k_{ex} = 0$ ($T^k_{ex}$ is the best
   improvement so far for total execution time of $MS_k$).
2: Estimate the $T^k_{ex}$ if choosing the closest ($CH_c$) to the NCC, which
   is unvisited by an MS, for the next stop of the tour. $T^k_{ex} = T^k_{ex} + t_p$
   $+ t_c + (d_{p,c}/v)$, where $d_{p,c}$ is the distance between $CH_p$ and $CH_c$, $v$ is
   the velocity of the MS, and $tp$, $tc$ are the total time spent by an MS
   to collect data from $CH_p$ and $CH_c$, respectively.
   a) If $T^k_{ex} < \xi_0$, reject the $CH_c$ and continue estimating with the
      further CH. If all choices (($M - 1$) choices) are exhausted without
      profit, return to Step 1 with a better starting point.
   b) Otherwise, accept the $CH_c$, add it to the tour, and mark this CH
      as a visited point of the MS.
3: Repeat Step 2 in order to increase the length of the tour, as long as
   the execution time criterion $T^k_{ex} \le \xi_0$ are satisfied.
4: Check if all CHs are visited. If not, increase the number of MSs
   $k = k + 1$ and return to Step 1. If yes, terminate the algorithm.
```

Constraint (23) forces every CH to be visited once. Constraints (24) and (25) imply that no more than $K_{\min}$ MSs are employed for data collection. Constraint (24) ensures that it is not necessary to use more than one MS to collect data at one CH.

Figure 5 illustrates the MSs' trajectories in three sample rounds (t_1, t_2, and t_3). In round t_1 (see Fig. 5(a)), the distances between MSs are rather far; therefore, four MSs are utilized to collect data from all CHs in order to finish this work within ξ_0. In rounds t_2 and t_3, as shown in Figs. 5(b) and 5(c), these

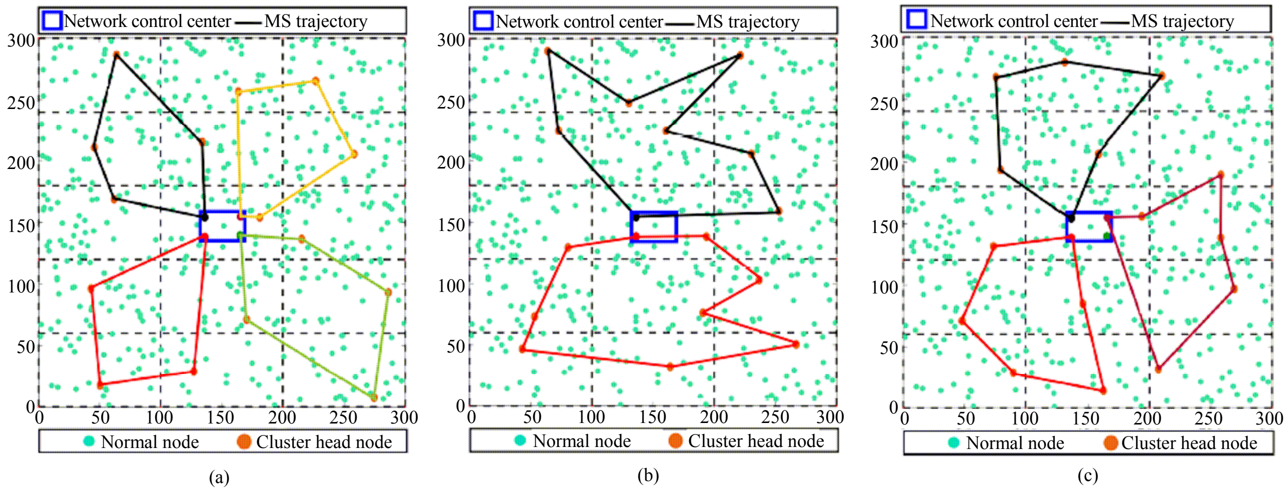


Fig. 5. Sink movement strategy for Scenario 2: (a) optimal trajectories for four MSs by OMS algorithm in round t_1, (b) optimal trajectories for two MSs by and OMS algorithm in round t_2, and (c) optimal trajectories for three MSs by OMS algorithm in round t_3.

distances are closer and the number of MSs is smaller at two and three MSs, respectively. We now present an outline of the basic heuristic algorithm to find the optimal trajectory of the MS. The procedures of the MSTO algorithm are explained in Algorithm 2.

3. Optimal Movement Strategy for Mobile Sink

In this section, we provide details of the optimal movement strategy (OMS) for an MS. The pseudo code of the OMS is presented in Algorithm 3.

Algorithm 3. Optimal Movement Strategy for MS

```
1: While min {$E_k(t) \geq \theta$} do
2: CHE by the CHE algorithm.
3: CHs broadcast announcements and one normal node will join the
   cluster whose CH is nearest to its position.
4: According to the shortest path routing, normal nodes will transmit
   or forward data to their CHs.
       $E_k(t) = E_0 - L_k(t),\ k = 1, \ldots, N.$
5: Compute the minimal spanning tree routing between $M$ CHs in the
   sensing field.
6: Choose the scenario on the basis of the total shortest path length,
   and the requirement of reporting time:
   • Scenario 1: Set $K_{\min} = 1$ and calculate the velocity of the MS by
       (17).
   • Scenario 2: Set the velocity of the MSs to a constant and find
       $K_{\min}$ by (21).
7: Find the optimal trajectory of the MS(s) by the MSTO algorithm.
8: After collecting sensed data from all CHs, increase the number of
   rounds for estimating the network lifetime $t = t + 1$; go back to
   Step 1.
9: End While
```

Table 2. Simulation parameters.

Parameter	Value
Node deployment	Random and uniform
# initial energy (E_0)	0.1 J
E_{elec}	50 nJ/b
ε_{fs}	10 pJ/b/m^2
ε_{mp}	0.0013 pJ/b/m^4
Maximum speed of MS ($v_{\max}$)	25 m/s
MSs' velocity in Scenario 2: ($v2$)	15 m/s
Packet length (m)	4,000 bits
Transmission range (tr)	30 m
Data transmission rate (R)	250 kb/s
Reporting time (ξ_0)	60 s

IV. Performance Evaluation and Discussion

In this section, we provide simulation results of our algorithm performed in a MATLAB environment.

1. Simulation Environment

In our simulations, the parameter settings are summarized in Table 2. One normal node is called a "dead node" if its residual energy is less than $\theta = 0.1$ mJ.

2. Numerical Results and Discussion

A. Cluster Head Election Performance Analysis and Experimental Comparison

For using the CHE algorithm efficiently, we conducted some experimental studies in a static network. $N = 150$ sensor nodes

https://doi.org/10.4218/etrij.17.0116.0629

Table 3. Network lifetime with different values of (α, β).

α	β	# of rounds	α	β	# of rounds
0	1.00	1,008	0.55	0.45	1,528
0.10	0.90	1,167	**0.60**	**0.40**	**1,602**
0.15	0.85	1,195	0.65	0.35	1,554
0.20	0.80	1,248	0.70	0.30	1,536
0.25	0.75	1,281	0.75	0.25	1,568
0.30	0.70	1,316	0.80	0.20	1,521
0.35	0.65	1,363	0.85	0.15	1,430
0.40	0.60	1,405	0.90	0.10	1,312
0.45	0.55	1,491	0.95	0.05	1,217
0.50	0.50	1,512	1	0	831

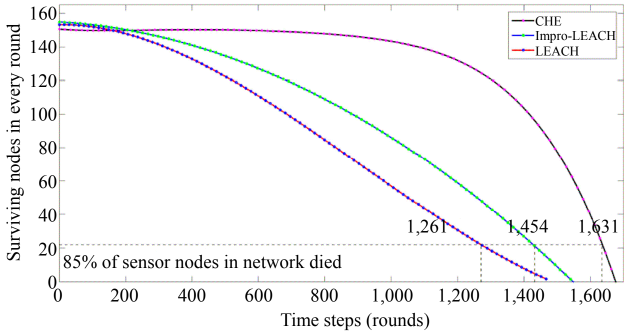


Fig. 6. Network lifetime comparison.

were deployed in an area of 300 × 300 (m^2). A single static BS was located at B_1(150, 350). Herein, for comparison between methods, after collecting data from the sensor nodes, each CH node transferred its data to the BS directly.

We first ran simulations with different values of (α, β) to find the best profit weighting factors (α, β). The simulation results, are given in Table 3, indicated that the set ($\alpha = 0.6$, $\beta = 0.4$) is the best solution with the highest network lifetime. Therefore, we used these values of (α, β) in the subsequent simulations.

The better performance of the CHE algorithm was evaluated and compared with the method proposed in [8] and [10], when changing the network size. It is proved that LEACH can prolong the network lifetime when compared with direct transmission and minimum-transmission-energy routing. However, as can be seen in Fig. 6, the network lifetime of the LEACH algorithm is limited to t = 1,261 (rounds), whereas for the CHE algorithm, this number is t = 1,631 (rounds). Therefore, our proposed CHE algorithm can improve the network lifetime up to 29% compared with the LEACH algorithm. Meanwhile, the Impro-LEACH proposed in [10] has better results with 1,454 (rounds) than LEACH algorithm. With 177 more rounds, CHE can prolong the network lifetime up to 12% compared with Impro-LEACH. However, it is worth to noting here that our algorithm has more balanced energy consumption among all sensor nodes, and the network continues to work with a high number of live nodes to the last round.

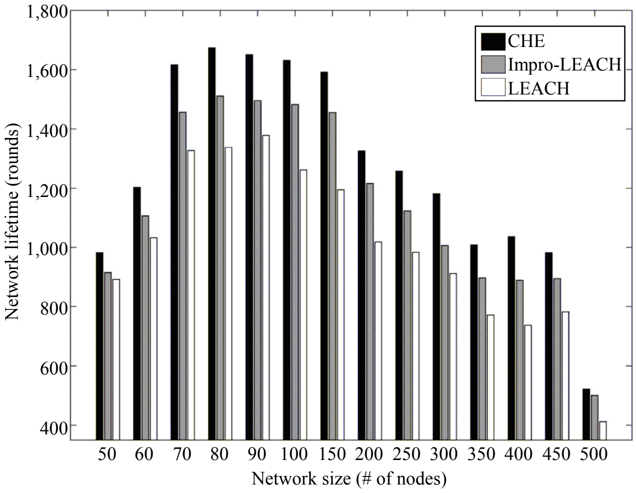


Fig. 7. Network lifetime comparison with different network sizes.

In the LEACH and Impro-LEACH algorithms, the number of dead nodes increases over time. For evaluating the CHE, Impro-LEACH, and LEACH algorithms, we conducted experimental studies with different numbers of sensor nodes (varying from 50 to 500), which were deployed randomly in the sensing field of 300 × 300 m^2.

The network lifetime improvement L_i in these experimental studies can be computed as (28)

$$L_i = \left(1 - \frac{\text{Lifetime by LEACH or Impro-LEACH}}{\text{Lifetime by CHE}}\right) 100\%. \tag{28}$$

The results of the network lifetime comparison are summarized in Fig. 7. It can be seen that the CHE algorithm has a longer lifespan than the LEACH and Impro-LEACH algorithms in most of the cases. The improvement achieved in network lifetime by the CHE algorithm is 26.2% and 10.64% compared with the LEACH and Impro-LEACH algorithms, respectively.

B. Performance Analysis of the Mobile Sink Trajectory Optimization Algorithm

In this subsection, we evaluate the performance of two scenarios of the OMS algorithm (we define scenario 1 as OMS1 and scenario 2 as OMS2) by comparing them with two other conventional strategies: (i) static sink, where a stationary sink node is located at the center of the network B_0(150,150); (ii) random moving strategy, where an MS moves randomly in the sensing field; and (iii) MSs moving along the boundary of the network as proposed in [19].

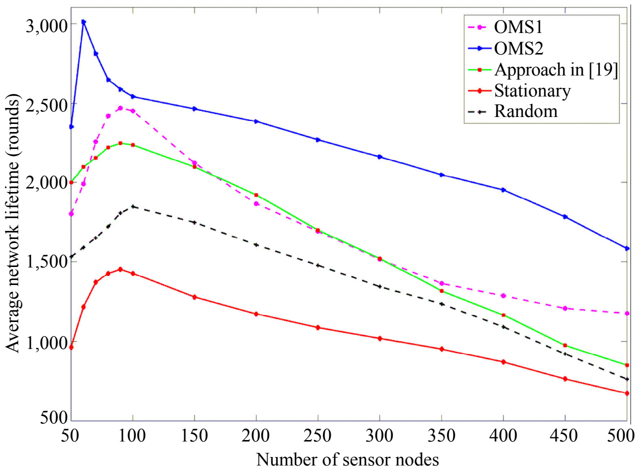


Fig. 8. Network lifetime variation with different node densities.

Figure 8 depicts the comparison of the network lifetime between these five schemes. It is clear that with increasing node density, the network lifetime of each scheme decreases, because the more sensors, the more data reported to the BS. However, a small number of sensor nodes deployed over a larger geographic area means a longer distance between sensor nodes, thus reducing the network lifetime. As shown in Fig. 8, our proposed schemes (OMS1 and OMS2) have longer lifetimes than the three other strategies because with the controlled mobility, OMS1 and OMS2 avoid the hot-spot problem completely, which is the main reason for short network lifetime in the stationary scheme and fixed-trajectory attempt. In the random scheme, the MS randomly moves in the sensing field, therefore, it cannot guarantee balanced energy consumption among sensor nodes in the network.

The OMS2 scheme provides better performance, and the network lifetime in this scheme is always higher than that in the OMS1 scheme. However, it is rather expensive to equip more than one MS in the network. Thus, each scenario of the OMS algorithm should be chosen depending on the specific application requirement.

V. Conclusions

In this study, we examined the optimization problem for maximizing network lifetime in WSNs by CH election and finding the optimal trajectory of the MS. Our investigation was based on the number and location of CHs in each round to optimize the travel paths of the MSs. The experimental results demonstrate that our proposed CHE algorithm can greatly extend the network lifetime by 10.64% to 26.20% compared with the Impro-LEACH and LEACH algorithms, respectively. Moreover, in optimizing the MS's trajectory, our proposed OMS algorithm with two scenarios adapts well to the different network sizes, and improves the network lifespan and exhibits better performance than the random, stationary, and fixed-trajectory schemes.

References

[1] C.H. Liu, K.F. Ssu, and W.T. Wang, "A Moving Algorithm for Non-uniform Deployment in Mobile Sensor Networks," *Int. J. Autonomous Adaptive Commun. Syst.*, vol. 4, no. 3, 2011, pp. 271–290.

[2] M.C. Akewar and N.V.T. Nileshsingh, "A Study of Wireless Mobile Sensor Network Deployment," *Int. J. Comput. Wireless Commun.*, vol. 2, no. 4, Aug. 2012, pp. 533–541.

[3] M. Vecchio et al., "DEEP: Density-Based Proactive Data Dissemination Protocol for Wireless Sensor Networks with Uncontrolled Sink Mobility," *Comput. Commun.*, vol. 33, no. 8, May 2010, pp. 929–939.

[4] B. Nazir and H. Hasbullah, "Mobile Sink Based Routing Protocol (MSRP) for Prolonging Network Lifetime in Clustered Wireless Sensor Network," *Int. Conf. Comput. Applicat. Ind. Electron.*, Kuala Lumpur, Malaysia, Dec. 5–8, 2010, pp. 624–629.

[5] S. Go and J.W. Chong, "Improved TOA-Based Localization Method with BS Selection Scheme for Wireless Sensor Networks," *ETRI J.*, vol. 37, no. 4, Aug. 2015, pp. 707–716.

[6] Y.F. Huang et al., "Lifetime Performance of an Energy Efficient Clustering Algorithm for Cluster-Based Wireless Sensor Networks," *Int. Symp. Parallel Distrib. Process. Applicat.*, Niagara Fall, Canada, Aug. 28–Sept. 1, 2007, pp. 455–464.

[7] S. Ci, M. Guizani, and H. Sharif, "Adaptive Clustering in Wireless Sensor Networks by Mining Sensor Energy Data," *Comput. Commun.*, vol. 30, no. 14, Oct. 2007, pp. 2968–2975.

[8] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, Maui, HI, USA, Jan. 4–7, 2000, pp. 1–10.

[9] S. Kosunalp et al., "Practical Implementation and Stability Analysis of ALOHA-Q for Wireless Sensor Networks," *ETRI J.*, vol. 38, no. 5, Oct. 2016, pp. 911–921.

[10] F. Zhao, Y. Xu, and R. Li, "Improved LEACH Routing Communication Protocol for a Wireless Sensor Network," *Int. J. Distrib. Sensor Netw.*, vol. 8, no. 12, Dec. 2012, pp. 1–6.

[11] F. Tashtarian, M.H.Y. Moghaddam, and S. Effati, "Energy Efficient Data Gathering Algorithm in Hierarchical Wireless Sensor Networks with Mobile Sink," *Int. eConf. Comput. Knowl. Eng.*, Mashhad, Iran, Oct. 18–19, 2012, pp. 232–237.

[12] W. Aioffi, G. Mateus, and F. Quintao, "Optimization Issues and Algorithms for Wireless Sensor Networks with Mobile Sink," *Int. Netw. Optimization Conf.*, Spa, Belgium, Apr. 22–25, 2007, pp. 1–6.

[13] Y. Gu et al., "Partitioning Based Mobile Element Scheduling in

https://doi.org/10.4218/etrij.17.0116.0629

Wireless Sensor Networks," *IEEE Commun. Soc. Sensor Ad Hoc Commun. Netw.*, Santa Clara, CA, USA, Sept. 26–29, pp. 386–395.

[14] 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, Oct. 2002, pp. 660–670.

[15] A.R. Chalak, S. Misra, and M.S. Obaidat, "A Cluster-Head Selection Algorithm for Wireless Sensor Networks," *IEEE Int. Conf. Electron., Circuits, Syst.*, Athenes, Greece, Dec. 12–15, 2010, pp. 130–133.

[16] A.W. Khan, "A Comprehensive Study of Data Collection Schemes Using Mobile Sinks in Wireless Sensor Networks," *Sensors*, vol. 14, no. 2, 2014, pp. 2510–2548.

[17] J. Deng et al., "Optimal Transmission Range for Wireless Ad Hoc Networks Based on Energy Efficiency," *IEEE Trans. Commun.*, vol. 55, no. 7, July 2007, pp. 1439–1439.

[18] G. Dantzig, R. Fulkerson, and S. Johnson, "Solution of a Large-Scale Traveling-Salesman Problem," *J. Operations Res. Soc. America*, vol. 2, no. 4, 1954, pp. 393–410.

[19] J. Wang et al., "An Energy Efficient Distance-Aware Routing Algorithm with Multiple Mobile Sinks for Wireless Sensor Networks," *Sensors*, vol. 14, no. 8, 2014, pp. 15163–15181.

Hoc Thai Nguyen received his BS degree in electrical and electronic engineering from Hanoi Agriculture University, Hanoi, Vietnam in 2006, and MA degree in automation from Vietnam National University of Agriculture, Hanoi in 2010. He is currently pursuing PhD degree in infocommunication technologies at the Budapest University of Technology and Economics, Hungary. His research interests include wireless sensor networks, machine learning, and artificial neural networks.

Linh Van Nguyen received his BE and ME degrees in electrical engineering and automation from the Vietnam National University of Agriculture, Hanoi, in 2006 and 2010, respectively, and his PhD degree in electrical engineering and robotics from the University of Technology Sydney, Ultimo NSW, Australia, in 2015. His current research interests include spatial adaptive sampling methods, Gaussian processes, statistical learning algorithms for static and mobile sensor networks and robotics, sensors and signal processing, and infrastructure robotics.

Hai Xuan Le received his MA degree in electrical and electronics engineering from the Vietnam National University of Agriculture, Hanoi, in 2010. He is pursuing his PhD degree at the Hanoi University of Science and Technology, Vietnam. His main research interests are non-linear adaptive control, fuzzy systems – neural networks, wireless sensor networks, microcontroller applications, and programmable logic controller.