Clustering algorithms for maximizing the lifetime of wireless sensor networks with energy-harvesting sensors

Pengfei Zhang [a], Gaoxi Xiao [a,*], Hwee-Pink Tan [b]

[a] School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore
[b] Sense and Sense-abilities, Institute for Infocomm Research, Singapore

ARTICLE INFO

ABSTRACT

Motivated by recent developments in wireless sensor networks (WSNs), we present several efficient clustering algorithms for maximizing the lifetime of WSNs, i.e., the duration till a certain percentage of the nodes die. Specifically, an optimization algorithm is proposed for maximizing the lifetime of a single-cluster network, followed by an extension to handle multi-cluster networks. Then we study the joint problem of prolonging network lifetime by introducing energy-harvesting (EH) nodes. An algorithm is proposed for maximizing the network lifetime where EH nodes serve as dedicated relay nodes for cluster heads (CHs). Theoretical analysis and extensive simulation results show that the proposed algorithms can achieve optimal or suboptimal solutions efficiently, and therefore help provide useful benchmarks for various centralized and distributed clustering scheme designs.

1. Introduction

With the developments of low-power and multi-functional sensors, wireless sensor networks (WSNs) [44,4,36,2] composed of sensor nodes with abilities of data sensing/processing and wireless communication have paved the way for a wide variety of practical applications in monitoring, tracking, control, etc.

Since batteries in sensors have finite stored energy and it is generally not convenient to replace or recharge these batteries, a critical issue in WSNs is to achieve high energy efficiency in order to prolong the lifetime of the networks. Extensive researches have been carried out to tackle the problem and many solutions have been proposed, among which include *clustering*-based approaches [2,26]. A clustered WSN is typically composed of a base station (BS) and a certain number of clusters. Each cluster is composed of a cluster head (CH) and some non-cluster head (NCH) nodes. The CH is responsible for receiving data from NCHs,

processing the data and then forwarding the information to the BS, either directly, via other CHs or via one or multiple *relay* nodes. Relay nodes are responsible for forwarding data received from other nodes and may not necessarily be responsible for local sensing. In clustered WSNs, transmitting to a CH nearby rather than to a possibly far away BS helps reduce the energy consumption of NCHs. However, CHs may be heavily burdened since they need to process and transmit the data for the whole cluster. This may shorten the lifespan of CHs, especially in the absence of relay nodes between CHs and BS. Lowering the energy consumption of CHs therefore usually plays a critical role in prolonging the lifetime of clustered WSNs. Since the communication distance largely determines the energy consumption of data transmission, finding a good location for each CH is of critical importance for prolonging network lifetime: an inappropriate CH location may force the CH node to communicate with BS over a long distance and consequently uses up its stored energy quickly.

Clustered WSNs have been extensively studied in recent years [1]. Existing works include energy-efficient schemes and algorithms [21,47,50,51,23,35], enhancement of cluster stability in various network topologies [46,22], MAC

---

* Corresponding author. Tel.: +65 67904552; fax: +65 67933318.
  E-mail addresses: pzhang1@e.ntu.edu.sg (P. Zhang), egxxiao@ntu.edu.sg (G. Xiao), hptan@i2r.a-star.edu.sg (H.-P. Tan).

layer design [48,42,50] and many more. The work on energy-efficient schemes typically adopts two different objectives, namely minimizing the overall energy consumptions [21,47] and maximizing the network lifetime [24,5] respectively. These two parts of work are closely related to but different from each other: the former one works on a minimization problem while the latter one usually works on a min–max problem since the lifetime of a network is usually decided, or at least strongly affected (depending on the definitions of lifetime, as we will discuss in more detail in Section 5 later), by those nodes with shortest lifespan. We term such nodes as the *bottleneck* nodes. In clustered WSNs, as mentioned earlier, the bottleneck nodes are usually, though not always, CHs. In this paper, we focus on designing clustering algorithms to maximize network lifetime.

Existing results on lifetime maximization problem can be largely classified into two categories: centralized methods [33,24,11,5,13,14,49,16,37,40,7,34,18,6] and distributed methods [50,31,28,21,47,9]. Centralized methods typically require knowledge of the sensors' locations to achieve global optimization with respect to certain performance metrics. Distributed methods, on the other hand, make decisions based on local information exchanged between neighboring sensors, thus achieving better scalability. We focus on studying centralized methods in this paper as they can provide a good reference for network pre-planning and serve as a useful benchmark for evaluating the performance of distributed methods.

Parallel to the significantly improved network clustering techniques, another important recent progress is the development of energy harvesting (EH) sensors [41,17,38,19,8]. EH sensors can harvest energy (e.g., solar, kinetic, thermal, etc.) from their environment, converting this energy into electrical energy which is then stored in devices with large numbers of recharge cycles (such as super-capacitors) [41] to achieve virtually infinite lifetime. While the deployment of large-scale WSNs composed solely of EH sensors remain impractical in the near future due to high costs and low achievable duty cycles, an arguably more practical approach is to adopt EH sensor nodes sparsely in WSNs [25,32].

Clustering methods and energy-harvesting techniques come as a natural combination for prolonging the network lifetime: a proper formation of clusters liberates most sensors (especially NCHs) from high energy consumptions, while a carefully planned sparse deployment of EH sensors helps prolong the lifespan of the bottleneck nodes. Since energy harvesting rates are sensitive to the environment [8], it may not be practical to let EH nodes serve as function-critical nodes such as CHs. In this paper, we consider a simple case where EH sensors serve as relay nodes for CHs. By communicating with EH nodes over a shorter distance rather than sending data to BS directly, CHs can have lowered energy consumptions for at least a certain fraction of time. This simple case can provide some useful insights into where EH sensors should be located to maximize network lifetime.

To summarize, in this paper we propose algorithms for maximizing the lifetime of clustered WSNs, with or without EH nodes. Specifically, we assume that a given number of sensors are distributed in a certain area with arbitrary distribution. These sensors can be formed into a given number of clusters under centralized control. Each cluster contains a single CH and a certain number of NCHs. NCHs forward data to their CH. Each CH is responsible for aggregating data from NCHs and forwarding the information to BS, either directly or via a dedicated EH relay node. The EH nodes only serve as relay nodes; they do not collect/process environmental information themselves. Furthermore, we assume that each sensor is equipped with same amount of energy at the beginning and BS is equipped with infinite energy, e.g., through mains power.

The main contributions of this paper are twofold: (i) we propose efficient algorithms for maximizing the network lifetime of both single-and multi-cluster WSNs. Analytical and extensive simulation results demonstrate the fast convergence of our proposed algorithms to optimal or suboptimal solutions and (ii) based on the assumption that the locations of the EH nodes can be adjusted in order to maximize network lifetime, we extend the proposed algorithms to handle the case where EH sensors serve as relay nodes for CHs. Extensive simulation results quantify how much EH nodes may help prolong the network lifetime. Finally, we also briefly discuss on the revision of the proposed algorithms under different definitions of network lifetime.

The rest of the paper is organized as follows: a brief survey of some closely related work is provided in Section 2. In Section 3, we propose algorithms for calculating the optimal locations of CH nodes in single-and multi-cluster networks. Both single-and multi-cluster algorithms are extended to handle the case where EH sensors serve as relay nodes of CHs in Section 4. Brief discussions on the extension of the proposed algorithms under different lifetime definitions are presented in Section 5. In Section 6, extensive simulation results and discussions are presented for verifying the performances of the proposed algorithms and the effects of EH sensors on prolonging network lifetime. Finally, Section 7 concludes the paper and presents several directions for future research.

2. Literature survey

Numerous centralized clustering algorithms for WSNs have been proposed [33,24,11,29,5], typically aiming to reduce the power consumption of CHs to prolong network lifetime. Ning et al. proposed an algorithm which adopted the sequential location-allocation decomposition method to minimize the communication power and achieve high reliability for a large-scale network [33]. Irani et al. proposed heuristic approaches for the CH deployment problem and also studied the effects of the number of clusters [24]. An incremental algorithm was proposed in [11] for efficient placement of CH nodes. Li et al. proposed a clustering scheme based on uncapacitated facility location in which the network lifetime is extended by adding a layer of Super-Cluster-Head nodes to ease the transmission load of the CHs and to balance the load distribution within the network [29]. Aslam et al. proposed a weighted cost function based on the residual energy levels of cluster heads for the

mobile actor to optimally fine-tune its geographical location [5]. These studies, however, mainly focused on (i) minimizing the number and/or the energy consumptions of CHs [33,24,11] or (ii) efficiently utilizing mobile CHs [5].

As aforementioned in Section 1, maximizing network lifetime leads to different optimization problems from those for minimizing overall energy consumptions of networks. Different definitions of network lifetime have been proposed for WSNs [30,12,15], most of which can be classified into two categories: (i) the time until the first $\alpha\%$ of nodes drain out of energy or die and (ii) the time until a certain coverage or connectivity constraint in a certain region cannot be fulfilled [45]. In this paper, we adopt the first definition, and investigate the effects of different values of α. By adopting such a definition, maximizing network lifetime leads to a max–min problem with the objective of making those sensor(s) with shortest lifespan live longest possible.

The existing work on algorithm design for maximizing network lifetime can be roughly classified into three categories:

- Static clustering-based algorithms which do not change the clusters once they are formed up [13,14,49]. Such existing work, however, all study on different network topologies compared with ours, e.g., tree-based [13], chain-based [14], multiple-hop intra-cluster topologies [49], etc.
- Dynamic clustering algorithms, which re-select CHs periodically, adopting the same objective as, yet different network models from, ours. For example, a multilevel intra-cluster topology is studied in [16]. Some other works [37,40] introduce heterogeneity into the network design, where CHs are equipped with more energy compared with NCHs; or introduce multiple (more than two) layers in the network topology, e.g., a super-CH layer to reduce the burden of CHs [7,34].
- Dynamic clustering algorithms adopting the same network model and design objective as ours. They include: LEACH [21], EECS [50], HEED [47], MOECS [6], AEEC [9], and pLEACH [18]. Amongst these algorithms, LEACH was first proposed and widely serves as a benchmark for comparison with other algorithms. MOECS is found to have longer lifetime compared with EECS and HEED [6,50]. AEEC and pLEACH are recently proposed schemes that perform better than LEACH [9,18]. However, no comparison between them and MOECS has been performed.

In this paper, we consider the problem of maximizing network lifetime in static clustering. Since there are no existing algorithms designed for the same network model and design objective, we extend our algorithm to dynamic clustering for comparison. Specifically, we benchmark the extended algorithm with LEACH, MOECS, AEEC and pLEACH. As will be shown in detail later in Section 6, the extended algorithm, though not optimized for dynamic clustering at the first place, manages to outperform all the existing methods.

There have also been some studies on clustering in WSNs with EH nodes [43,27,3,18], typically with the assumption that the network was solely composed of EH sensors which have infinite lifetime. Islam et al. considered a hybrid WSN which comprised both battery-powered and EH nodes [25]. However, they let EH nodes served as CHs with a higher probability than that of the battery-powered nodes. To the best of our knowledge, there is no existing work for maximizing network lifetime where EH nodes serve as relay nodes for CHs.

3. Clustering algorithms for maximizing network lifetime in WSNs

Throughout this paper, without loss of generality, we assume that N_s sensor nodes with known locations and the same amount of initial energy are deployed in 2-D space, and a single BS with infinite energy is located at $(0,0)$. The N_s sensor nodes will be partitioned into N_c clusters, each comprising one CH. We also assume TDMA-based communications where each TDMA frame comprises N_s slots: the NCH nodes will transmit to the respective CH nodes in the first N_s–N_c slots; the CH nodes then forward the received data to the BS in the subsequent N_c slots. As in [21], we consider the Friis free-space propagation model where the transmission power is proportional to the square of the distance. Note that the proposed algorithms can be easily extended to other propagation models, e.g., multi-path fading model [39]. The notations used in the paper are shown in Table 1.

In Section 3.1, we propose an algorithm for finding the optimal location of the CH in a single-clustered WSN; this is then extended to multi-cluster networks in Section 3.2.

3.1. Cluster head selection algorithm $\left(N_c = 1, \alpha = \frac{1}{N_s}\right)$

We describe our proposed cluster head selection algorithm for a single-cluster network. Amongst the N_s nodes, one node will be selected as the CH, which is assumed to be able to fuse information from all NCHs. Our single-cluster network model is depicted in Fig. 1 and the objective of the algorithm design is to maximize the time until the first node dies (i.e., $\alpha = \frac{1}{N_s}$).

Denoting by NCH_f the NCH node that is farthest away from the CH, and the distance between nodes A and B by $d_{A,B}$, the power required for CH and NCH_f is shown below:

$$P_{CH} = E_{elec}r(N_s - 1) + E_{DA}rN_s + E_{elec}r + E_{amp}d_{CH,BS}^2 r \qquad (1)$$

$$P_{NCH_f} = E_{elec}r + E_{amp}d_{CH,NCH_f}^2 r \qquad (2)$$

Intuitively, we know that either the CH or the NCH_f will die first since they consume more energy than any other NCHs. Hence the problem is to find the optimal location for the CH, (x^*, y^*), which minimizes $\max(P_{CH}, P_{NCH_f})$. It is easy to see that at the optimal location, we shall have $P_{CH} = P_{NCH_f}$.

This problem is closely related to the well-known *weighted smallest circle problem* [20], which is defined as follows: denote the location of node i as $p_i = (x_i, y_i)$ and its positive weight as w_i, $i = 1, \ldots, n$. For any point $p = (x, y)$, let d_{p,p_i} be the distance between p and p_i and

$$H(p) = \max_{1 \leqslant i \leqslant n} w_i d_{p,p_i}.$$

Find $p^* = (x^*, y^*)$ such that H is minimized.

Table 1
Notations used throughout this paper.

Notation	Description	Value
N_s	Number of sensors in the network	$\{100, 150, 200\}$
N_c	Number of clusters in the network	$\{1, 2, \ldots, 10\}$
$N_{s,j}$	Number of sensor nodes in cluster j	N.A.
i	Node index	$\{i = 1, 2, \ldots, N_s\}$
j	Cluster index	$\{j = 1, 2, \ldots, N_c\}$
k	Index of iteration	N.A.
r	Data transmission rate in bits/s	2.5×10^5 bits/s
E_{elec}	Energy required for processing each bit of data	5×10^{-8} J/bit
E_{DA}	Energy consumption for data aggregation	5×10^{-9} J/bit/signal
E_{amp}	Coefficient of energy consumption by transmission amplifier	10^{-10} J/bit/m^2
E_s	Energy stored in battery for each sensor	0.5 J
CH_j	The CH node for cluster j	N.A.
$NCH_{f,j}$	The NCH node that is farthest away from the CH_j in cluster j	N.A.
(x_i, y_i)	Coordinates for node i	N.A.
$d_{A,B}(k)$	Distance between nodes A and B in iteration k	N.A.
$\hat{t}$	Lifetime for CH	N.A.
$P_{EH_j,h}$	Harvesting rate for EH node in cluster j	N.A.
$P_{EH_j,c}$	Energy consumption rate for EH node in cluster j	N.A.
$P_{CH,off}$	Energy consumption rate for CH when EH is not working	N.A.
$P_{CH,on}$	Energy consumption rate for CH when EH is working	N.A.
t_{EH}	Time duration when EH node works	N.A.
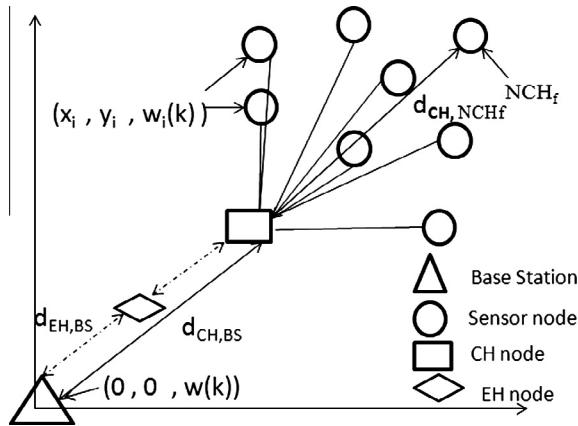
Fig. 1. The single-cluster WSN model.

To tackle the weighted smallest circle problem, Hearn and Vijay proposed a simple yet efficient optimization algorithm [20]. The main idea of the algorithm is to start with a minimum circle covering at least two nodes and then iteratively find among all the nodes outside the circle (if any) the one weighted-farthest away from the circle center, and update the circle to cover it. In each iteration, it is ensured that, for those nodes on the boundary of the existing circle and the weighted-farthest away node, the newly built circle is the minimum one which covers all these nodes by going through at least two or three of them on the boundary. It can be proved that the algorithm converges to the optimal solution.

By ignoring the item $E_{elec}r$ and $E_{elec}r(N_s - 1) + E_{DA}rN_s + E_{elec}r$, which are constants, in Eqs. (1) and (2) respectively, we have that the energy consumption of each NCH is proportional to the square of its distance to CH and the energy consumption of CH is largely decided by the distance from BS to CH. Finding the best location of CH is equivalent to finding p^* for all the NCHs and the BS, if (and only if) the weight of each NCH and the BS is properly assigned such that when H in the minimum weighted circle problem is minimized, $\max(P_{CH}, P_{NCH_f})$ is also minimized.

Since the energy consumption of each NCH (minus a constant) is proportional to its distance to CH, we conveniently let $w_i = 1$ for $i = 1, 2, \ldots, N_s$. For BS, denote its weight as w_0. To take into account the energy consumption by CH for data processing and aggregation, we shall have

$$w_0 = \sqrt{\frac{E_{elec}(N_s - 1) + E_{DA}N_s}{E_{amp}d_{CH,BS}^2} + 1} \tag{3}$$

By doing so, when $w_0 d_{CH,BS} = d_{CH,NCH_f}$, we have $P_{CH} = P_{NCH_f}$.

The value of d_{CH,NCH_f} remains unknown until the optimal solution of p^* is found, which prohibits finding the best solution of w_0 at the beginning. In our algorithm, we initialize $w_0 = 1$ and update its value iteratively. As we will see later, the iterative calculations converge.

The proposed algorithm is presented in detail as Algorithm 1. For simplicity of notation, we let $w(k)$ denote the weight of BS (i.e., w_0 above) in the kth iteration, and $d(k)$ the distance between NCH_f and CH in the kth iteration. Note that only $w(k)$ needs to be updated in each iteration, while for all the other sensors, their weights remain as 1 throughout the calculation.

Algorithm 1. Single Cluster Algorithm

- **Initialization**
  Input set of sensor locations
  $A = (x_1, y_1), (x_2, y_2) \ldots (x_{N_s}, y_{N_s})$. Let $(x_0, y_0) = (0,0)$ be the coordinates for the BS, and set $w(1) = 1$.
- **Step 1**
  Let $k = 1$. Run the weighted smallest circle algorithm [20] to find the center of the circle, record

the coordinate as $(x(1), y(1))$, and record $d(1)$.
- **Step 2**
  Let $k = k + 1$. Set $w(k)$ according to Eq. (3) and run the weighted smallest circle algorithm to find the center for the kth iteration. Record the coordinate as $(x(k), y(k))$; record $d(k)$ and update $d_{CH,BS}(k)$.
- **Step 3**
  If $|d(k) - d(k-1)| \leqslant \epsilon$, go to step 4; otherwise, go to step 2.
- **Step 4**
  Having obtained the optimal CH position, (x^*, y^*), we select the nearest sensor as CH.

It can be proved that the algorithm converges.

Lemma 1. *Given $w(k+1) \geqslant w(k)$, we have $d(k+1) \geqslant d(k)$.*

Proof. Refer to Appendix A. □

Theorem 1. *If $w(k+1) \geqslant w(k)$, then either the algorithm terminates, i.e., $d(k+1) = d(k)$, or $d_{CH,BS}(k+1) \leqslant d_{CH,BS}(k)$.*

Proof. Refer to Appendix B. □

Theorem 2. *$d(k)$ will converge within finite steps of k.*

Proof. Refer to Appendix C. □

Remark 1. We select the sensor nearest to the optimal position derived from our algorithm as the CH node. Although this cannot guarantee to achieve optimal solution, we use extensive simulations to demonstrate its near-optimality in Section 6.

Remark 2. We derived our weight expression based on the idea that $P_{CH} = P_{NCH_f}$. We prove the convergence of our algorithm. It can be shown that in most of deployment cases after we run this algorithm, we could get $P_{CH} = P_{NCH_f}$. Furthermore, our algorithm can also be used to deal with the case when NCHs are deployed quite close to BS, i.e., even we put CH at BS, we still have $P_{CH} > P_{NCH_f}$. The convergence of our algorithm is not affected according to our proof.

3.2. Cluster formation algorithm $\left(N_c > 1, \alpha = \frac{1}{N_s}\right)$

In a large-scale network, it may be beneficial to partition it into N_c clusters ($N_c > 1$), where N_c CH nodes are able to fuse information from NCHs and forward the information to the BS. Such an N_c-cluster network model is depicted in Fig. 2. Our problem is defined as follows: Given N_c, how do we partition a network with N_s nodes into N_c clusters and determine the respective CH nodes, denoted by $CH_1, CH_2, \ldots, CH_{N_c}$, to maximize the overall network lifetime, i.e., the time till the first node dies?
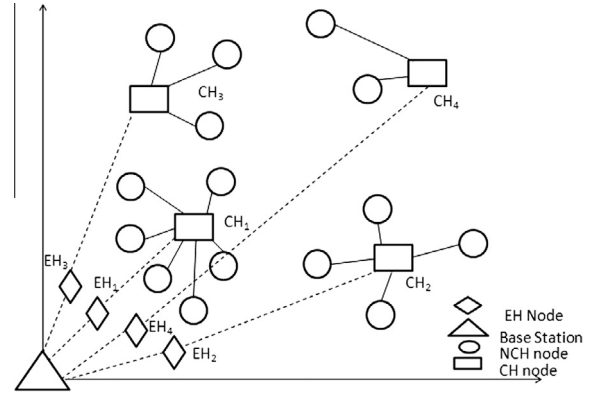

Fig. 2. Our proposed multiple cluster WSN model.

We begin by considering the case where N_c CH nodes have been given, and the task is to join NCHs into these N_c clusters with maximized network lifetime. Denote the CH nodes as CH_j, $1 \leqslant j \leqslant N_c$. The power consumption rate of CH_j is given as follows:

$$P_{CH_j} = E_{elec}r(N_{s,j} - 1) + E_{DA}rN_{s,j} + E_{elec}r + E_{amp}d_{CH_j,BS}^2 r \qquad (4)$$

The main idea is to join one NCH into a certain cluster in each iteration until all the sensors have joined the clusters. In each iteration, let us denote by NCH_j^* the sensor that is closest to CH_j among all the sensors which have not joined any cluster yet and $P_{NCH_j^*}$ the corresponding power consumption rate if the sensor does join cluster j. Obviously, if any sensor is to join cluster j in this iteration, it should be NCH_j^* since it leads to the smallest increase in $NCH_{f,j}$. We repeat this process for all CHs, determine $j^* = \arg\min_j P_j$, where $P_j = \max\left(P_{NCH_j^*}, P_{CH_j}\right)$ where P_{CH_j} is calculated based on the assumption that one more sensor is joining cluster j, and assign $NCH_{j^*}^*$ to CH_{j^*}. This process is repeated until all NCHs join the clusters.

We now describe the multi-cluster formation algorithm. Randomly select N_c nodes as initial CHs and use the algorithm above to join all the other sensors into these clusters. Denote the network lifetime as $L(0)$. We then apply our cluster head selection algorithm as described in Section 3.1 to select new CHs in each cluster independently; repeat the cluster formation process described above, and denote the network lifetime as $L(1)$. Repeat the cluster formation and cluster head selection algorithms iteratively until $|L(k) - L(k-1)| < \epsilon$. The proposed algorithm for an N_c-cluster network is shown in Algorithm 2. We illustrate the algorithm for a simple example of $N_c = 2$ in Figs. 3–5.

Algorithm 2. N_c Cluster Algorithm, $N_c > 1$

- **Initialization**
  Input the set of sensor locations
  $A = (x_1, y_1), (x_2, y_2) \cdots (x_{N_s}, y_{N_s})$. Let $(x_0, y_0) = (0,0)$ be the coordinates for the BS. Randomly select N_c CHs from the existing N_s sensors, labeled as $CH_1, CH_2, \ldots, CH_{N_c}$. Set iteration number $k = 0$.
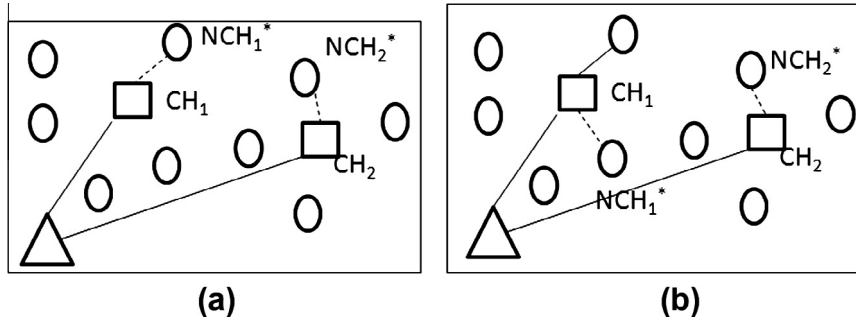
(continued on next page)

Fig. 3. Step-by-step illustration of the cluster formation algorithm (a) CH_1 and CH_2 are randomly selected from existing sensors to serve as CHs, NCH_1^* and NCH_2^* are closest NCHs from CH_1 and CH_2 respectively. If NCH_1^* joins CH_1, we denote $P_1 = \max\left(P_{CH_1}, P_{NCH_1^*}\right)$. If NCH_2^* joins CH_2, we denote $P_2 = \max\left(P_{CH_2}, P_{NCH_2^*}\right)$. The dotted line is used to show the closest NCH for each CH. (b) If $P_1 < P_2$, NCH_1^* in Fig. 3(a) joins CH_1. We use a solid line to show the connection. Then we find new closest NCHs from both CHs, still denoted as NCH_1^* and NCH_2^*, which are connected to their respective CHs through dotted lines.
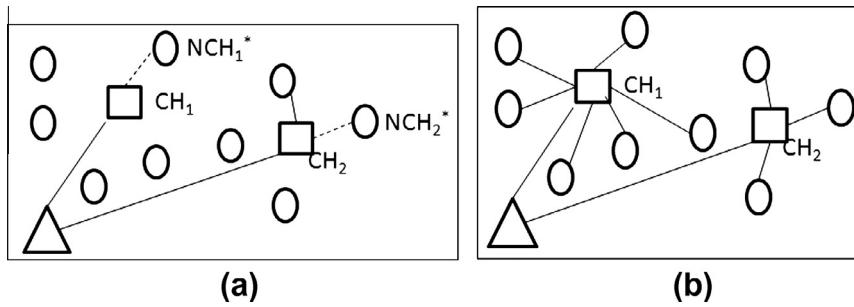

Fig. 4. Step-by-step illustration of the cluster formation algorithm (a) Assume $P_1 > P_2$ in Fig. 3 and a node joins CH_2, shown as the solid line. We find new closest NCHs from both CHs, which are connected to their respective CHs via dotted lines. (b) We repeat the above process for each NCH one by one until all NCHs join the corresponding CHs. We denote the whole network lifetime as L_0.
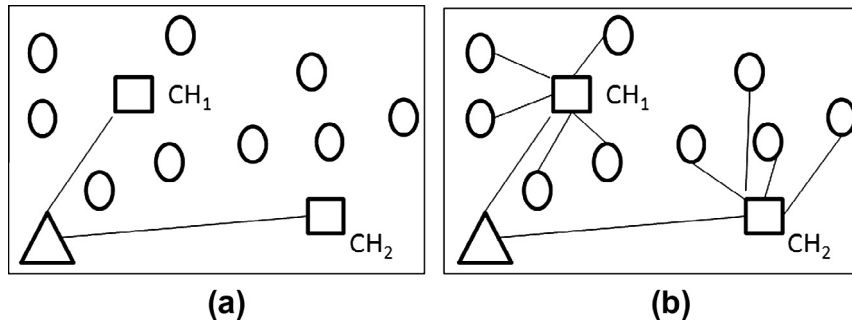

Fig. 5. Cluster head selection + cluster formation (a) We run Algorithm 1 for each cluster independently to find the new CHs, denoted as CH_1 and CH_2. (b) Cluster formation by using steps shown in Fig. 3 and Fig. 4. Denote the network lifetime as L_1.

- **Step 1**
  For each CH_j, determine P_j if the closest unassigned NCH, denoted by NCH_j^*, is added to it. Find $j^* = \arg\min_j P_j$ and add $NCH_{j^*}^*$ to cluster j^*. Repeat until all NCH nodes are assigned to clusters. Denote the network lifetime as $L(k)$.
- **Step 2**
  Use Algorithm 1 to find the new CH position independently for each cluster, and increment k.

- **Step 3**
  Repeat the process in Step 1 until all the nodes join the clusters. We record the lifetime of the network as $L(k)$.
- **Step 4**
  Compare $L(k)$ with $L(k-1)$: if $|L(k) - L(k-1)| > \epsilon$, then goto Step 2; otherwise, terminate the algorithm.

It can be easily proven that Algorithm 2 converges. Specifically, whenever step 2 is executed, the network lifetime either remains unchanged or is increased (when a better candidate is found). The lifetime may be further improved in step 3 when every NCH joins the selected CH to maximize lifetime. In short, the overall network lifetime increases progressively until a local optima is found.

Remark 3. Starting from a set of randomly selected CHs, Algorithm 2 can only ensure achieving a local optimal solution. Running the algorithm from a large enough number of different initial sets of CHs helps improve the likelihood that global optimality or sub-optimality is achieved. We demonstrate in Section 6 that our algorithm's performance is close to optima in most cases.

3.3. Dynamic clustering algorithm $\left(N_c \geqslant 1, \alpha = \frac{1}{N_s}\right)$

As discussed in Section 2, since there are no existing results with the same network model and objective function as ours, we extend our algorithm to dynamic clustering for comparison. Specifically, we assume, as in most existing results, that re-clustering is carried out at the beginning of each round of sensing. Note that, for simplification, we neglect any possible energy consumption involved in cluster formation and re-formation. To allow the proposed algorithm to handle dynamic clustering, the minimum change needed is to handle the case where different sensors may have different initial energies. Specifically, the changes we make are as follows:

- Recall that in the cluster head selection algorithm presented in Section 3.1, we assign weight 1 to sensors and w_0 to BS according to Eq. (3) and iteratively update w_0 in the kth iteration (denoted as $w(k)$). This is based on the assumption that all the sensors are equipped with same energy. Now we consider the case where every sensor is equipped with energy E_i, which may be different for different i. We need to derive the new weight expression for sensors such that at the optimal location of CH, $\min\left(\frac{E_{CH}}{P_{CH}}, \frac{E_{NCH_f}}{P_{NCH_f}}\right)$ is maximized. It is easy to see that at the optimal CH position, we have $\frac{E_{CH}}{P_{CH}} = \frac{E_{NCH_f}}{P_{NCH_f}}$. By adopting the same approach as shown in Section 3.1, we have:

$$w_0 = \sqrt{\frac{E_{elec}(N_s - 1) + E_{DA}N_s}{E_{amp}d_{CH,BS}^2} + 1} \qquad (5)$$

$$w_i = \sqrt{\frac{E_{CH}}{E_i}}. \qquad (6)$$

By doing so, when $w_0 d_{CH,BS} = w_{NCH_f} d_{CH,NCH_f}$, we have $\frac{E_{CH}}{P_{CH}} = \frac{E_{NCH_f}}{P_{NCH_f}}$. Once again, only w_0 needs to be updated in each iteration. Hereafter we always use $w(k)$ to denote the value of w_0 in the kth iteration.

As E_{CH} is not known in advance, a convenient option is to let $E_{CH} = \frac{\sum_{i=1}^{N_s} E_i}{N_s}$ though setting E_{CH} at any other

positive values would hardly affect the algorithm efficiency. After the CH selection algorithm finds the optimal CH position, among sensors that have energy larger than the average energy of sensors in the cluster, the one closest to the optimal position is chosen as CH. In this way, if the sensor closest to the optimal position has low residual energy, its energy can be conserved by not serving as CH. The convergence of the algorithm can also be proven using the theorems shown in Section 3.1.

- Recall that for the cluster formation algorithm shown in Section 3.2, we determine $j^* = \arg \min_j P_j$ based on the assumption that the initial energy is the same in different sensors. When this is not true, both E_i and P_i have to be considered in assigning sensors into clusters to optimize lifetime. Specifically, in each iteration, we still add one sensor into one of the clusters: given N_c CHs, for the closest NCHs from these CHs, we determine $j^* = \arg \max_j L_j$, where $L_j = \min\left(E_{NCH_j^*} - P_{NCH_j^*} T, E_{CH_j} - P_{CH_j} T\right)$. T is a constant, which is the duration of each time slot in the TDMA schedule. Then we add $NCH_{j^*}^*$ into CH_{j^*}, and repeat this process until all NCHs join the respective CHs. By adopting an approach similar to that in Section 3.2, the convergences of the modified cluster formation algorithm can be proven.

Note that the extended algorithm is not optimized for dynamic clustering. Instead, it is the algorithm for static clustering with some minimum changes. As we will see later in Section 6, however, the extended algorithm nevertheless outperforms all the existing ones.

4. Algorithms for maximizing lifetime of WSNs with EH sensors $\left(\alpha = \frac{1}{N_s}\right)$

In this section, we assume the availability of EH nodes that can harvest and store energy from the environment, and study how they can be exploited in a clustered WSN to maximize network lifetime. In general, each EH node can operate in three different ways: (i) serves as relay for CH, i.e., receives the data from CH and forwards it to BS; (ii) serves as CH, i.e., receives, aggregates information from all NCHs and forwards to BS whenever it has available energy; or (iii) serves as a relay between NCHs and CH node. In this paper, we consider the case where N_c EH nodes are available as depicted in Fig. 2, and focus on Case (i), where each EH node, EH_j, serves as a dedicated relay node for CH_j, leaving the remaining cases to future work. Our objective is to study the joint placement of CH and EH nodes to maximize the lifetime of WSNs.

The main idea of the proposed algorithm is that, for each cluster j, we find out the *relationship* between the best location of CH_j and the best location of EH_j. Then by treating EH_j as if it were the BS and assigning a proper weight to it, we find the best location of CH_j and correspondingly, the best location of EH_j as well.

We begin with a single cluster WSN as shown in Fig. 1, and let E_s and $\hat{t}$ denote the initial energy and the lifetime of the CH node respectively. Over the duration of $\hat{t}$, the amount of energy that the EH node (assumed to have zero initial energy) can harvest is given by $P_{EH,h}\hat{t}$. Since it

consumes energy at a rate of $P_{EH,c}$, it can remain active and relay data for the CH for a duration of t_{EH}, where

$$t_{EH} = \min\left(\frac{P_{EH,h}\hat{t}}{P_{EH,c}}, \frac{E_s}{P_{CH,on}}\right),$$

and the latter term is the lifetime of the CH node when it transmits via the EH node.

During this period, the energy depleted in the CH node is given by $P_{CH,on}t_{EH}$. Subsequently, the EH node is inactive, and the CH node transmits directly to BS for a duration of $\frac{E_s - P_{CH,on}t_{EH}}{P_{CH,off}}$ until its energy is depleted. Accordingly, we obtain the expression for $\hat{t}$ as:

$$\hat{t} = \begin{cases} \frac{P_{EH,h}\hat{t}}{P_{EH,c}} + \frac{E_s P_{EH,c} - P_{CH,on}P_{EH,h}\hat{t}}{P_{CH,off}P_{EH,c}}, & \hat{t} < \frac{E_s P_{EH,c}}{P_{EH,h}P_{CH,on}}; \\ \frac{E_s}{P_{CH,on}}, & \text{otherwise}. \end{cases} \quad (7)$$

• **Case 1:** $\hat{t} < \frac{E_s P_{EH,c}}{P_{EH,h}P_{CH,on}}$

According to Eq. (7), we have

$$\hat{t} = \frac{E_s P_{EH,c}}{P_{EH,c}P_{CH,off} + P_{CH,on}P_{EH,h} - P_{EH,h}P_{CH,off}} \quad (8)$$

We express $P_{CH,off}$, $P_{CH,on}$ and $P_{EH,c}$ in terms of $d_{CH,BS}$ and $d_{CH,EH}$ in a similar way as that in Eq. (1), and they are given as follows:

$$P_{EH,c} = E_{elec}r + E_{amp}d_{EH,BS}^2 r \quad (9)$$

$$P_{CH,on} = E_{elec}rN_s + E_{DA}rN_s + E_{amp}d_{CH,EH}^2 r \quad (10)$$

$$P_{CH,off} = E_{elec}rN_s + E_{DA}rN_s + E_{amp}d_{CH,BS}^2 r \quad (11)$$

According to the rule of triangularity, if EH is not located on the line connecting CH and BS, we have $d_{CH,EH} \geqslant d_{CH,BS} - d_{EH,BS}$. Accordingly, Eq. (10) becomes:

$$P_{CH,on} \geqslant E_{elec}rN_s + E_{DA}rN_s + E_{amp}(d_{CH,BS} - d_{EH,BS})^2 r \quad (12)$$

According to Eq. (12), to keep $P_{CH,on}$ low, the EH node should be placed on the line connecting CH and BS. Substituting Eqs. (9), (11) and (12) into (8), we obtain

$$\hat{t} = \frac{E_s}{E_{elec}rN_s + E_{DA}rN_s + E_{amp}d_{CH,BS}^2 r - f(d_{EH,BS})} \quad (13)$$

where $f(d_{EH,BS}) = \frac{(2d_{CH,BS} - d_{EH,BS})P_{EH,c}}{d_{EH,BS} + \frac{E_{elec}}{E_{amp}d_{EH,BS}}}$.

In order to maximize $\hat{t}$, according to Eq. (13), $f(d_{EH,BS})$ should be maximized. Since $\hat{t} < \frac{E_s P_{EH,c}}{P_{EH,h}P_{CH,on}}$, substituting Eq. (8), it can be easily deduced that $P_{EH,h} < P_{EH,c}$. Applying Eq. (9), we have

$$P_{EH,h} \leqslant E_{elec}r + E_{amp}rd_{EH,BS}^2 \Rightarrow d_{EH,BS} > \sqrt{\frac{P_{EH,h} - E_{elec}r}{E_{amp}r}}$$

If we denote $d_{EH,BS}^* = \sqrt{\frac{P_{EH,h} - E_{elec}r}{E_{amp}r}}$, and assume that $P_{EH,h} \gg E_{elec}r$, then $d_{EH,BS}^* > \sqrt{\frac{E_{elec}}{E_{amp}}}$. Considering $f(d_{EH,BS})$, we note that when $d_{EH,BS} = d_{EH,BS}^*$, $\frac{2d_{CH,BS}}{d_{EH,BS} + \frac{E_{elec}}{E_{amp}d_{EH,BS}}}$ is maximized while

$\frac{d_{EH,BS}}{d_{EH,BS} + \frac{E_{elec}}{E_{amp}d_{EH,BS}}}$ is minimized. Thus, $f(d_{EH,BS})$, and hence $\hat{t}$, is maximized when $d_{EH,BS} = d_{EH,BS}^*$.

• **Case 2:** $\hat{t} \geqslant \frac{E_s P_{EH,c}}{P_{EH,h}P_{CH,on}}$

According to Eq. (7), we have:

$$\hat{t} = \frac{E_s}{P_{CH,on}} \quad (14)$$

Taking Eq. (12) into Eq. (14), we have

$$\hat{t} = \frac{E_s}{E_{elec}rN_s + E_{DA}rN_s + E_{amp}(d_{CH,BS} - d_{EH,BS})^2 r} \quad (15)$$

Since $\hat{t} \geqslant \frac{E_s P_{EH,c}}{P_{EH,h}P_{CH,on}}$, applying Eqs. (14) and (9), we have $P_{EH,h} \geqslant P_{EH,c}$ and $d_{EH,BS} \leqslant d_{EH,BS}^*$. According to Eq. (15), $\hat{t}$ can be maximized if $d_{EH,BS} = d_{EH,BS}^*$.

In summary, $\hat{t}$ is maximized when the EH node is placed on the line joining the BS and CH nodes such that $d_{EH,BS} = d_{EH,BS}^*$. At this position, we have that for the EH sensor, the energy consumption rate equals its energy harvesting rate. The corresponding maximum lifetime can be expressed as:

$$\hat{t}_{max} = \frac{E_s}{E_{elec}rN_s + E_{DA}rN_s + E_{amp}r\left(d_{CH,BS} - d_{EH,BS}^*\right)^2}$$

Next, to determine the CH position, we use the same approach as shown in Section 3.1 by letting $P_{CH} = P_{NCH_f}$, where

$$P_{CH} = P_{CH,off}\frac{\hat{t} - t_{EH}}{\hat{t}} + P_{CH,on}\frac{t_{EH}}{\hat{t}},$$

and P_{NCH_f} is given in Eq. (2). We apply Algorithm 1 to determine the optimal CH position, (x^*, y^*), with the following expression for the weight for EH:

$$w(k) = \sqrt{\frac{E_{elec}(N_s - 1) + E_{DA}N_s}{(E_{amp}d_{CH,BS}(k-1)^2)} + \frac{\left(d_{CH,BS}(k-1) - d_{EH,BS}^*\right)^2}{d_{CH,BS}(k-1)^2}}, \quad (16)$$

and the corresponding location of the EH node, (x_{EH}, y_{EH}), is then given by:

$$(x_{EH}, y_{EH}) = \left(d_{EH,BS}^*\frac{x^*}{\sqrt{x^{*2} + y^{*2}}}, d_{EH,BS}^*\frac{y^*}{\sqrt{x^{*2} + y^{*2}}}\right) \quad (17)$$

The algorithm for a single cluster with EH node is presented in Algorithm 3. As those in Section 3.1, we can prove that $d_{CH,NCH_f}(k), d_{CH,BS}(k)$ and $w(k)$ converge and the efficiency of convergence is also verified through simulations.

Algorithm 3. Single Cluster Algorithm with EH node

• **Initialization**
  Input set of sensor locations
  $A = (x_1, y_1), (x_2, y_2) \ldots (x_{N_s}, y_{N_s})$. Let $(x_0, y_0) = (0,0)$ which represents the coordinates for the BS, and set $w(1) = 1$;
• **Step 1**
  Run Algorithm 1 using the weight expression in Eq. (16) instead of Eq. (3).

- **Step 2**
  Record the CH position from the algorithm as (x^*, y^*) (not from existing sensors). The EH node position (x_{EH}, y_{EH}) is then given by Eq. (17).
- **Step 3**
  We select the sensor that is closest to (x^*, y^*) as CH and place the EH node at position (x_{EH}, y_{EH}).

Remark 4. We note that although the EH node is placed at (x_{EH}, y_{EH}), which is computed based on (x^*, y^*). For simplicity, we let it remain in this position when the node nearest to (x^*, y^*) is chosen as the CH. The optimal location of the EH sensor, if needed, can be computed as on the straight line between BS and CH with its energy consumption rate equaling its energy harvesting rate, which is typically quite close to (x_{EH}, y_{EH}) in a cluster with a reasonably high density of sensors.

The extension to the multi-cluster case is obtained by introducing the cluster index, j, to the corresponding expressions for $d^*_{EH,BS}$ and $w(k)$ and applying Algorithm 2 in Section 3.2. The corresponding procedure is given in Algorithm 4.

Algorithm 4. Multiple Cluster Algorithm with EH nodes

- **Initialization**
  Input set of sensor locations
  $A = (x_1, y_1), (x_2, y_2) \dots (x_{N_s}, y_{N_s})$. Let $(x_0, y_0) = (0, 0)$ which represents the coordinates for the BS.
- **Step 1**
  Run Algorithm 2 for the network using weight expression as shown in Eq. (16) for EH_j.
- **Step 2**
  Record the position for CH_j from the algorithm as $\left(x^*_{CH_j}, y^*_{CH_j}\right)$ where $1 \leqslant j \leqslant N_c$ (not from existing sensors). Position for EH_j can be calculated as
  $$\left(d^*_{EH_j,BS} \frac{x^*_{CH_j}}{\sqrt{x^*_{CH_j}{}^2 + y^*_{CH_j}{}^2}}, d^*_{EH_j,BS} \frac{y^*_{CH_j}}{\sqrt{x^*_{CH_j}{}^2 + y^*_{CH_j}{}^2}}\right).$$
- **Step 3**
  Select the sensor in cluster j that is closest from $\left(x^*_{CH_j}, y^*_{CH_j}\right)$ to serve as CH for cluster j, other sensors remain as NCHs. N_c EH nodes are deployed at the positions shown in step 2.

Similar to that for Algorithm 2 in Section 3.2, it can be easily proven that Algorithm 4 converges.

For dynamic clustering schemes, the algorithms can be extended similarly as that in Section 3.3 to handle the case where different sensors have different initial energies. Specifically, for the cluster head selection algorithm, we assign a weight $w_i = \sqrt{\frac{E_{CH}}{E_i}}$ to each sensor. When the optimal position for CH has been found, we select the closest sensor from this position with energy larger than the average energy in the cluster to serve as CH. For the cluster formation

Table 2
Effect of node density on the network lifetime (Case II, $N_s = \{50, 100, 150, 200\}$, $N_c = 1$).

Scheme $\vert N_s$	50	100	150	200
Brute force	48.65	32.41	23.91	18.98
Our algorithm	48.65	32.41	23.91	18.98

Table 3
Comparison of network lifetime for various cases ($N_s = 100$, $N_c = 1$).

	Case I	Case II	Case III
Brute force	17.99	32.41	35.05
Our algorithm	17.99	32.41	33.88

algorithm, we adopt the same procedure as that in Section 3.3.

5. Algorithms for maximizing lifetime of WSNs with EH sensors $\left(\frac{1}{N_s} < \alpha < 1\right)$

Some networks may still function well even when a portion of nodes have died [30]. Therefore, we also consider the case when the network lifetime is the time duration until a portion of nodes (defined as α%) die. In this section, we generalize our previous algorithms to the case where $\frac{1}{N_s} < \alpha < 1$. Our problem can be stated as follows: for an N_c-cluster WSN with EH nodes, how do we form the clusters and place the CH and EH nodes to maximize the overall network lifetime, i.e., the time until αN_s nodes die?

The intuition behind the algorithm is similar to that for the cluster formation algorithm shown in Section 3.2. We begin by assuming that N_c CH nodes have been assigned and the task is to form N_c clusters (i.e., assign NCHs to clusters) with the maximized lifetime. The lifetime is defined as the time until $N_s \alpha$ nodes die. In other words, the lifetime is the time duration that $N_s(1 - \alpha)$ nodes can function. As in Section 3.2, we assign one NCH to a cluster in every iteration. The only difference is that the process is repeated only until $N_s(1 - \alpha)$ closest nodes are covered. By doing this, we maximize the time that $N_s(1 - \alpha)$ nodes can function, thus maximizing the network lifetime.

We now describe the multi-cluster formation algorithm for the general lifetime definition. We randomly select N_c nodes as initial CHs and use the algorithm above to join $N_s(1 - \alpha)$ closest nodes into these clusters. Denote the network lifetime as $L(0)$. We then apply our cluster head selection algorithm as described in Section 3.1 to select new CHs in each cluster independently; repeat the cluster formation process described above, and denote the network lifetime as $L(1)$. Repeat the cluster formation and cluster head selection algorithms iteratively until $|L(k) - L(k-1)| < \epsilon$.

Our proposed algorithm is shown in Algorithm 5. Again, it can be easily proven that Algorithm 5 converges.

Algorithm 5. Multiple Cluster Algorithm with EH node for general α

- **Initialization**

  Input set of sensor locations
  $A = (x_1, y_1), (x_2, y_2) \ldots (x_{N_s}, y_{N_s})$. Let $(x_0, y_0) = (0,0)$ be the coordinates for the BS. Input $0 < \alpha < 1$. Randomly select N_c CHs from the existing N_s sensors, labeled as $CH_1, CH_2, \ldots, CH_{N_c}$. Set iteration number $k = 0$.

- **Step 1**

  For each CH_j, determine P_j if the closest unassigned NCH, denoted by NCH_j^*, is added to it. Find $j^* = \arg\min_j P_j$ and add $NCH_{j^*}^*$ to cluster j^*. Repeat until $N_s(1 - \alpha)$ nodes are assigned to clusters. Denote the network lifetime as $L(k)$.

- **Step 2**

  Use Algorithm 3 to find the new CH position for each cluster, and increment k.

- **Step 3**

  Repeat the process in Step 1 until $N_s(1 - \alpha)$ nodes are assigned to the clusters. We record the lifetime of the network as $L(k)$.

- **Step 4**

  Compare $L(k)$ with $L(k-1)$: if $|L(k) - L(k-1)| > \epsilon$ then goto Step 2; otherwise, we terminate the algorithm. The jth EH position is then given by

$$\left(d_{EH_j,BS}^* \frac{x_{CH_j}^*}{\sqrt{x_{CH_j}^{*\,2} + y_{CH_j}^{*\,2}}}, d_{EH_j,BS}^* \frac{y_{CH_j}^*}{\sqrt{x_{CH_j}^{*\,2} + y_{CH_j}^{*\,2}}} \right).$$

6. Simulation results

We demonstrate the performance of our proposed algorithms through simulations using Qualnet for a 2-D network with a base station deployed at $(0,0)$ and $N_s = \{50, 100, 150, 200\}$ nodes randomly distributed over a square region, each initially equipped with 0.5 J of energy and has a data transmission rate of 2000 bits per packet. We consider three square regions of network deployment, where the coordinates of the vertices are as follows: **Case I** $(200, 200)$, $(200, 300)$, $(300, 200)$ and $(300, 300)$; **Case II** $(100, 100)$, $(100, 200)$, $(200, 100)$, $(200, 200)$; **Case III** $(0,0)$, $(0, 300)$, $(300, 0)$, $(300, 300)$. Although Cases I and II have the same area, nodes are located closer to the BS in Case II than Case I. Cases I and III differ in terms of node density. By using these three cases, we can determine the influences of proximity to the BS and node density on the performance of the proposed algorithms.

The performance metric used in this paper is the number of rounds the network could operate, where a round is a TDMA frame composed of N_s times slots, one for each sensor node. After cluster formation and placement of EH nodes (where applicable), we repeat the TDMA frame until $\alpha\%$ of nodes dies. In Section 6.1, we verify that our clustering algorithms (with complexity $O(N_s)$) can achieve near-optimal network lifetime for the case of $\alpha = \frac{1}{N_s}$. We do so by bench-marking against a brute force method, where the optimal CH position is obtained through exhaustive search (with complexity $O(N_s^2)$). We also characterize the multi-cluster algorithms in terms of the optimal number of clusters for a given network. In Section 6.2, we ana-

lyze the effects of EH nodes, specifically the energy harvesting rates, on the network lifetime. In Section 6.3, we study the performance of our proposed clustering algorithms for general α.

6.1. Performance of clustering algorithms without EH nodes

- **$N_c = 1$**

  The effects of node density and proximity to the BS on the algorithm performance are shown in Tables 2 and 3 respectively. The results, obtained by averaging over 150 runs, show that our algorithm can achieve performance very close (up to 3.34%) to the optimal case (brute force).

  We then observe the rate of convergence for our algorithm, which is shown in Fig. 6. The results clearly show fast convergence (usually within 10 iterations). Due to the length limit, hereafter we only present results for Cases I and II with $N_s = \{100, 150\}$. Note that our conclusions hold for all the other cases.

- **$N_c > 1$, Cases I, II, $N_s = \{100, 150\}$**

  We compare the network lifetime for $N_c = \{2, 3\}$ in Table 4. Once again, our proposed algorithm can achieve performance close to that of the brute force method with lower time complexity – this gain becomes significant as N_c increases.

  We then compare the performance of the extended algorithm vs the existing dynamic clustering algorithms. For fair comparison with LEACH, MOECS, AEEC and pLEACH, we adopt the same optimal number of clusters shown in LEACH [21]. We run simulations with 25 differ-
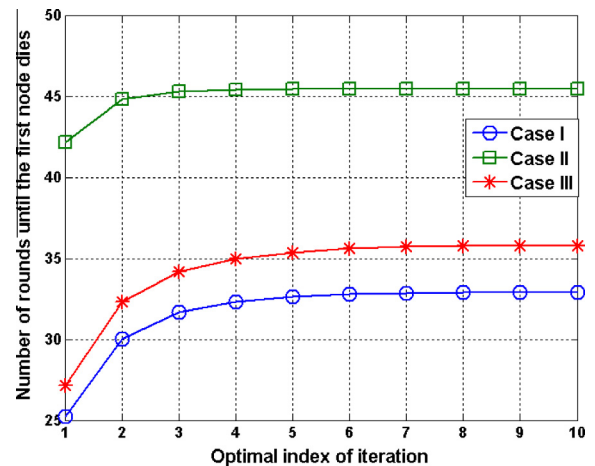

Fig. 6. Comparison of convergence efficiency of our algorithm for all cases ($N_s = 100$).

Table 4
Comparison between our method and brute force method ($N_c = \{2, 3\}$, $N_s = \{100, 150\}$).

	(N_s, N_c)			
	$(100, 2)$	$(150, 2)$	$(100, 3)$	$(150, 3)$
Brute force	49.28	39.72	60.42	51.20
Our algorithm	49.28	39.72	58.68	50.83

ent network configurations for both Case I and Case II from 100 to 150 nodes in steps of 10. The results are plotted in Fig. 7(a) and (b) for both Case I and Case II.

We find that our algorithm achieves the highest lifetime compared with the other approaches. Specifically, for Case II with 100 sensors, our algorithm improves the lifetime by 33.03% compared with LEACH and 3.31% compared with pLEACH. The improvement is less significant compared with pLEACH because pLEACH divides the area into several sectors in order to minimize the transmission distance from sensors to CHs as well as to balance the cluster size. Our work uses a simpler way without dividing the area into several sectors, and yet can still outperform pLEACH. For Case I with 100 sensors, our algorithm improves the lifetime by 56.84% compared with LEACH and 18.4% compared with pLEACH. For Case II, we find that with the increase in the number of sensors, the lifetime does not always increase. This is because the optimal number of

clusters varies with the number of sensors [21], thus affecting the network lifetime.

Next, we investigate the performance of our proposed algorithm for $N_c = \{1, 2, \ldots, 10\}$ as shown in Fig. 8. For Case II, network lifetime is maximized with 5 and 6 clusters when $N_s = 100$ and 150 respectively; for Case I, network lifetime is maximized where there are 4 and 5 clusters respectively. The differences in the optimal numbers of clusters for these two cases are due to the fact that sensors in Case I are farther from BS than those in Case II. Consequently, the optimal number of clusters becomes lower to have fewer CHs which have to communicate with BS over long distances. In other words, when CHs are far away from BS, the distance between BS and CH has stronger impact on network lifetime than the number of sensors in each cluster.

It is well-known that the overall energy consumption is a convex function of the number of clusters (e.g., [21]). It is interesting, though not a big surprise to see that the network lifetime appears to be a concave function of the number of
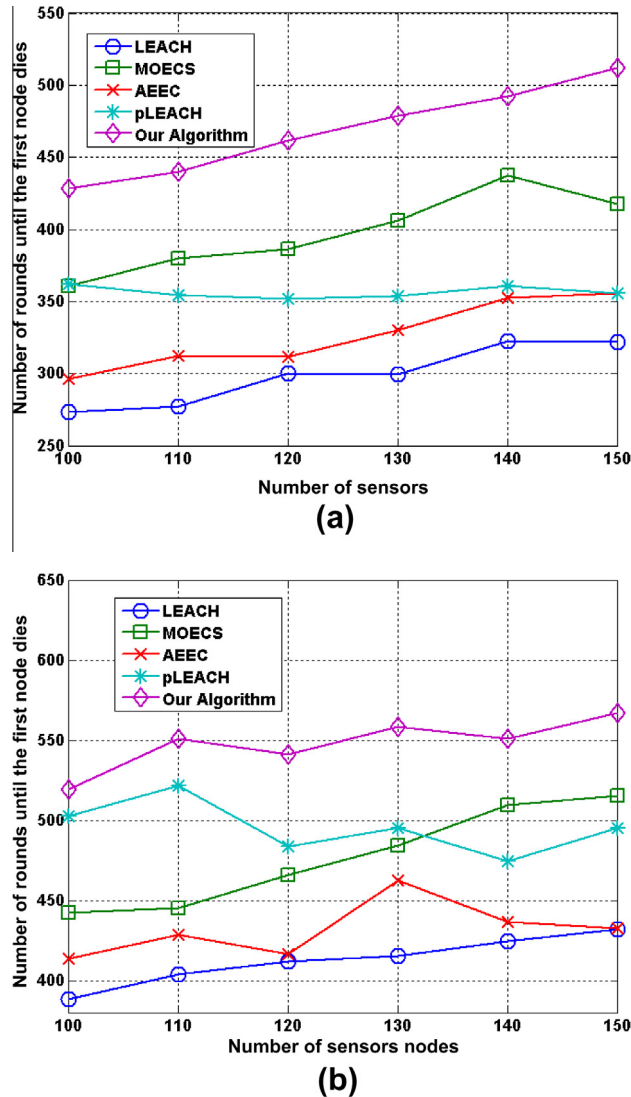

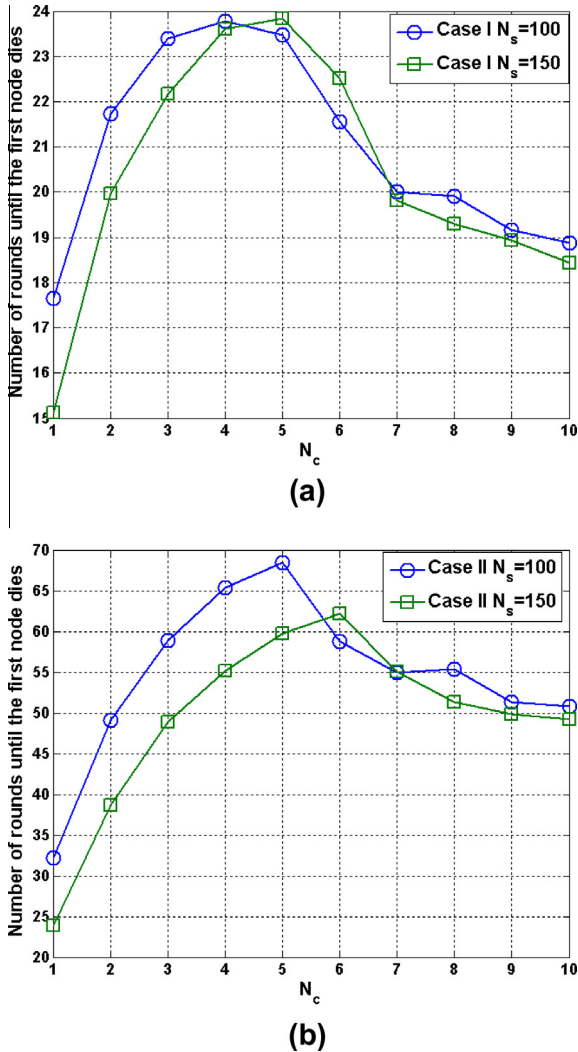**Fig. 7.** Comparison between our proposed algorithm with selected algorithms for (a) Case I and (b) Case II.

Fig. 8. Network lifetime vs number of clusters for our proposed algorithm ($N_c = \{1, 2, \ldots, 10\}$).

clusters: when N_c is very small (say, $N_c = 1$ in the extreme case), the CH needs to receive data from all NCHs, and therefore network lifetime is determined by this CH. When N_c is very large, e.g., $N_c = N_s$ in the extreme case, all NCH nodes will transmit directly to the BS, and hence the network lifetime is determined by the NCH which is farthest from the BS. Therefore, as N_c increases, the energy consumption rate of the CH node is reduced (i.e., network lifetime increases) initially due to the reduction in cluster size; however, with further increase in N_c, the energy consumption rates in some CH nodes increase because the distance between these CHs and BS increases, which reduces the network lifetime.

6.2. Performance of joint EH placement and clustering algorithm

Next, we study the performance of our proposed joint EH placement and clustering algorithm, where each CH node is served by a unique EH node, for the range of EH rates between 0.03 W and 0.09 W, in steps of 0.02 W. The

energy harvesting rate is assumed according to the previous literature [10,41], typically tens of milliwatts.

- $N_c = 1$

We quantify the impact of the EH node on the network lifetime in Fig. 9. For a single cluster network, introducing an EH node with a charging rate of 30 mW results in a gain of 8–13% in network lifetime for different cases. Certainly the improvements depend on many factors including energy harvesting rate, network scale, and more. As expected, for a given node density, the network lifetime decreases when the network is farther from the BS; for a given deployment, the network lifetime is reduced when the node density increases.

- $N_c > 1$

We investigate the performance of the proposed algorithm for Case II with $N_c = \{1, 2, \ldots, 10\}$, as shown in Fig. 10. Similar to that in Section 6.1, the network lifetime is maximized with 5 and 6 clusters when $N_s = 100$ and 150 respectively. The optimal number of clusters for each case remains invariant for energy harvesting rate varying from 30 mW to 90 mW. This can be explained as follows: in our simulations, we adopt relatively high data transmission rate of 2.5×10^5 b/s and typical harvesting rates of 30–90 mW. As a result, the harvesting rate of EH is much lower than the energy consumption rate of CH. Hence the optimal location of EH is typically much closer to BS than the CH. For example, for Case II, $N_c = 5$ and $N_s = 100$, the optimal location of EH is 26.46 m away from BS while the closest CH from BS is 151.43 m away. Since the optimal number of clusters is largely determined by distance between CH and BS, the existence of EH nodes therefore does not change the optimal number of clusters.

Note that for the case while CHs have a lower energy consumption rate or EH has a much higher energy harvesting rate, we can expect the optimal number of clusters to increase. For the extreme case where EH nodes have an infinite energy harvesting rate, the optimal location of EH is to stay at the same location of its CH and the optimal
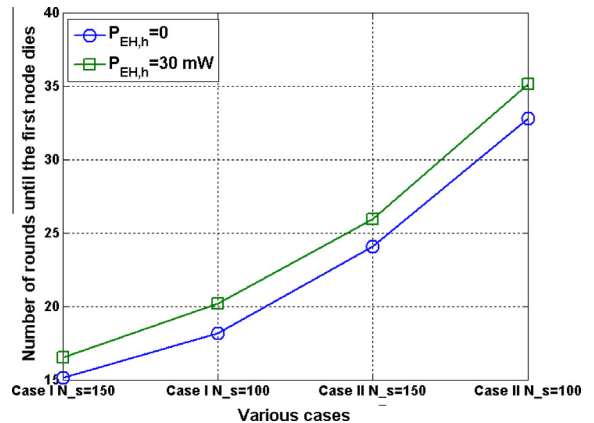

Fig. 9. Effects of using EH node on the network lifetime ($N_c = 1$, $P_{EH,h} = 0.03$ W).
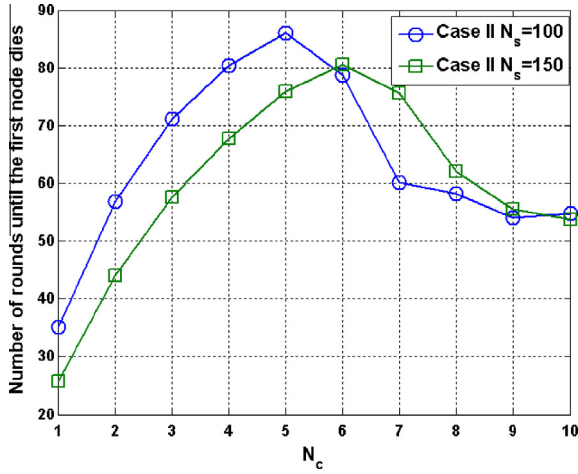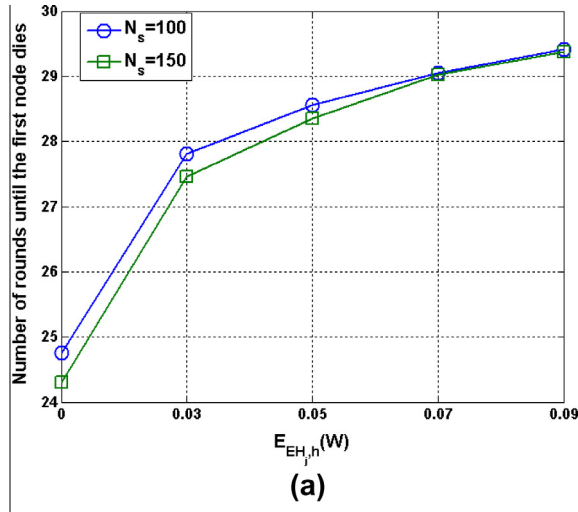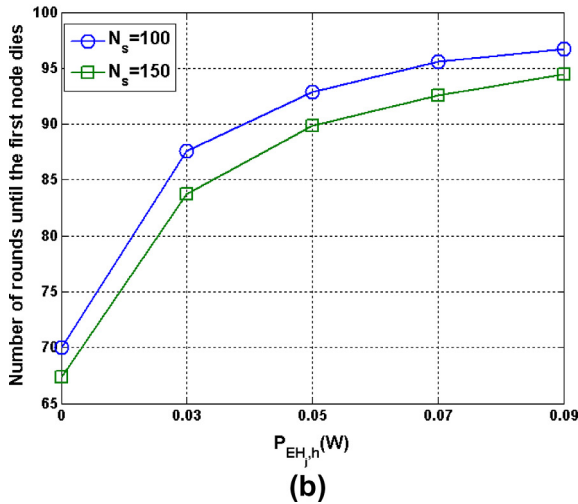
Fig. 10. Network lifetime vs number of clusters for our proposed algorithm ($N_c = \{1,2,\ldots,10\}$, $P_{EH,h} = 0.03$ W).
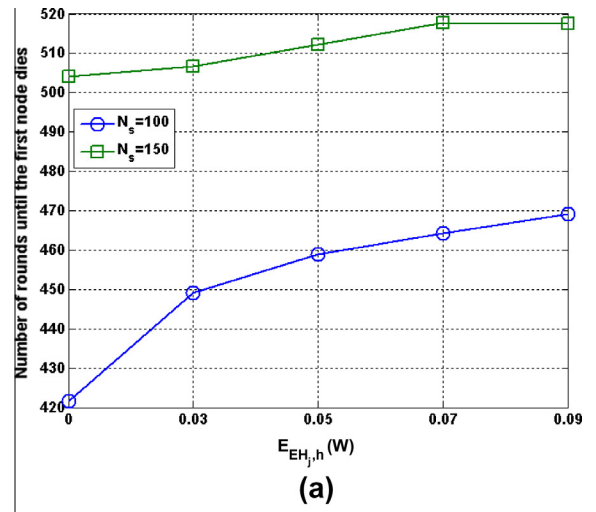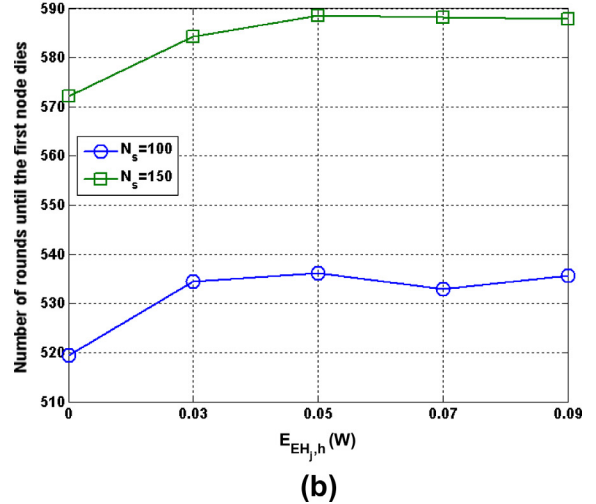
number of clusters equals the number of sensors. This unrealistic case nevertheless illustrates the trend of the optimal number of clusters when the EH harvesting rate goes up.

For Case II, with N_c being set to the optimal number of clusters for $N_s = 100$ and 150 respectively, we quantify the gain in network lifetime achievable at various energy harvesting rates in Fig. 11. The result is average value over simulations of 30 different network topologies with 100 simulations over each network topology. We observe that the network lifetime increases with energy harvesting rate. Specifically, in Case II, with the existence of EH nodes with a harvesting rate of 30mW, the lifetime of the network increases from 70.05 to 87.58, leading to a 25.02% increase. Note that this increase is achieved when CHs are transmitting at a relatively high bit rate. If CHs are transmitting at a lower bit rate, the increase will be more significant.

For a given number of clusters, a continuous increase in the EH sensor harvesting rate leads to a less and less significant further improvement in network lifetime when EH moves towards CH. The lifetime will finally reach a con-

Fig. 11. Effects of energy harvesting rate on network lifetime for different cases with optimal number of clusters for (a) Case I and (b) Case II.

Fig. 12. Effects of energy harvesting rate on network lifetime using dynamic clustering algorithm for (a) Case I and (b) Case II.
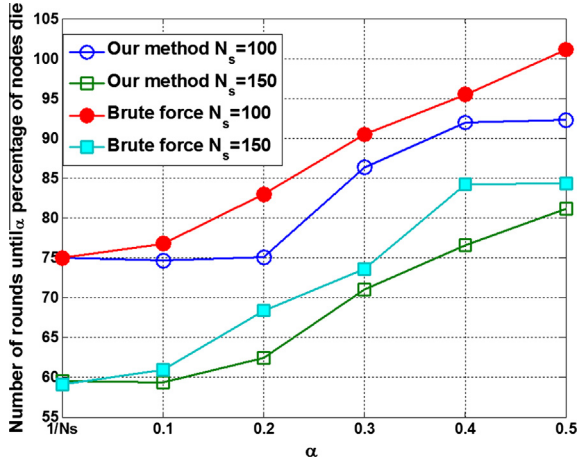
Fig. 13. Comparison between our algorithm with brute force method at various α (Case II, $N_c = 3$, $N_s = \{100, 150\}$).

stant value where further increasing harvesting rate does not help. By then the cluster lifetime is decided by the distance between CH and the farthest NCH, which cannot be arbitrarily shortened by the existence of the EH node.

We finally evaluate the effects of energy harvesting rate on the network lifetime using the dynamic clustering algorithm. The results are shown in Fig. 12(a) and (b) for both Case I and Case II when the number of clusters is still set to be the optimal number of clusters of LEACH. For both Case I and Case II, we find that the lifetime is improved significantly when the harvesting rate is increased from 0 to 0.03 W. When the harvesting rate continues to increase, once again we find that the network lifetime is less significantly further improved, for the same reason given earlier.

6.3. Algorithm performance for different lifetime definitions

Lastly, we illustrate the performance of our proposed joint EH placement and clustering algorithms for different values of α ranging from $\{0.1, 0.2, \ldots, 0.5\}$ for Case II, with $N_c = 3$. We compare the network lifetime with the brute force (optimal) method for clustering, and plot the results in Fig. 13, with the first point on the x-axis corresponding to $\alpha = \frac{1}{N_s}$.

As expected, the network lifetime increases as α (i.e., the permissible number of dead nodes) increases. We also observe that our algorithm is able to achieve performance sufficiently close to the brute force method (within 10%).

7. Conclusion

In this paper, we considered clustered wireless sensor networks (WSN) where CHs either aggregate and forward data directly to BS, or via dedicated relay nodes with energy harvesting (EH) capabilities. We proposed efficient (polynomial-time) EH node placement and clustering algorithms to maximize network lifetime, where the network lifetime is the duration until α% of the nodes run out of energy. Through theoretical analysis and extensive simulations, we validated the near optimality of the proposed algorithms and demonstrated how much EH sensors can help

prolong network lifetime in different scenarios. In addition, we showed the existence of an optimal numbers of clusters for a given network configuration (defined by the node density and proximity of the network to the BS), which may not be significantly changed by the existence of EH sensors with typical energy harvesting rates.

For future work, we plan to (i) extend our study to propose distributed EH clustering mechanisms; (ii) study other configurations of introducing the EH nodes to the network; and (iii) extend our simulations to more realistic models, and implement and evaluate our algorithms in an actual WSN testbed.

Appendix A. Proof of Lemma 1

We prove by contradiction. Given $w(k + 1) \geqslant w(k)$, let's assume that $d(k + 1) < d(k)$. Let $(\bar{x}(k), \bar{y}(k)) = (x(k + 1), y(k + 1))$, and $\bar{d} = d(k + 1)$. Then we have a feasible solution for the weighted smallest circle problem in the kth iteration, $\bar{d} = d(k + 1) < d(k)$, which leads to a contradiction. $\square$

Appendix B. Proof of Theorem 1

Let $(\bar{x}(k + 1), \bar{y}(k + 1)) = (x(k), y(k))$ and $\bar{d} = \max\{w(k + 1)d_{CH,BS}(k), d(k)\}$. We have a feasible solution for the $(k + 1)$th iteration where the center of the circle is at $(x(k), y(k))$ and the radius is $\bar{d}$.

If $\bar{d} = d(k)$, then $d(k + 1) \leqslant \bar{d} = d(k)$. From Lemma 1, $d(k + 1) = d(k)$, and the algorithm terminates.

If $\bar{d} = w(k + 1)d_{CH,BS}(k)$, we have $w(k + 1)d_{CH,BS}(k + 1) \leqslant d(k + 1) \leqslant \bar{d} = w(k + 1)d_{CH,BS}(k)$, and hence $d_{CH,BS}(k + 1) \leqslant d_{CH,BS}(k)$. $\square$

Appendix C. Proof of Theorem 2

From (3), we have $w(2) > w(1)$. By Theorem 1, we have $d_{CH,BS}(2) \leqslant d_{CH,BS}(1)$. According to (3), $w(k + 1)$ increases with a decreasing value of $d_{CH,BS}(k)$. Consequently we have the conclusion that $w(k)$ is a monotonically increasing function of k, while $d_{CH,BS}(k)$ is a monotonically decreasing function of k. Since $d_{CH,BS}(k) \geqslant 0$, $d_{CH,BS}(k)$ is a lower-bounded monotonic sequence, which definitely converges. Since $w(k)$, $d(k)$ are functions of $d_{CH,BS}(k)$, they will converge as well. $\square$

References

[1] A.A. Abbasi, M. Younis, A survey on clustering algorithms for wireless sensor networks, Computer Communications 30 (2007) 2826–2841.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, Computer Networks 38 (2002) 393–422.

[3] C. Alippi, R. Camplani, C. Galperti, M. Roveri, A robust, adaptive, solar-powered wsn framework for aquatic environmental monitoring, IEEE Sensors Journal 11 (2011) 45–55.

[4] T. Arampatzis, J. Lygeros, S. Manesis, A survey of applications of wireless sensors and wireless sensor networks, in: Proceedings of the 2005 IEEE International Symposium on Intelligent Control, 2005, pp. 719–724.

[5] N. Aslam, W. Phillips, W. Robertson, Effective location management of mobile actors in wireless sensor and actor networks, Journal of Networks 4 (2009) 1–8.

[6] N. Aslam, S. Sivakumar, W. Phillips, W. Robertson, 2007 Energy efficient cluster formation using a multi-criterion optimization technique for wireless sensor networks, in: 2007 4th Annual IEEE Consumer Communications and Networking Conference, CCNC 2007, pp. 650–654.

[7] S. Banerjee, S. Khuller, A clustering scheme for hierarchical control in multi-hop wireless networks, in: Proceedings – IEEE INFOCOM 2, 2001, pp. 1028–1037.

[8] C. Bergonzini, D. Brunelli, L. Benini, Algorithms for harvested energy prediction in batteryless wireless sensor networks, in: 3rd International Workshop on Advances in Sensors and Interfaces, 2009, pp. 144–149.

[9] O. Buyanjargal, Y. Kwon, AEEC – adaptive and energy efficient clustering algorithm for content based wireless sensor networks, in: CSA '09. 2nd International Conference on Computer Science and its Applications, 2009, pp. 1–6.

[10] B. Calhoun, D. Daly, N. Verma, D. Finchelstein, D. Wentzloff, A. Wang, S.H. Cho, A. Chandrakasan, Design considerations for ultra-low energy wireless microsensor nodes, IEEE Transactions on Computers 54 (2005) 727–740.

[11] A. Chehri, H. Mouftah, An efficient clusterhead placement for hybrid sensor networks, Ad-Hoc, Mobile and Wireless Networks 6288 (2010) 123–134.

[12] Y. Chen, Q. Zhao, On the lifetime of wireless sensor networks, IEEE Communications Letters 9 (2005) 976–978.

[13] K. Dasgupta, K. Kalpakis, P. Namjoshi, An efficient clustering-based heuristic for data gathering and aggregation in sensor networks, in: Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), 2003, pp. 1948–1953.

[14] K. Dasgupta, M. Kukreja, K. Kalpakis, Topology-aware placement and role assignment for energy-efficient information gathering in sensor networks, in: IEEE ISCC, 2003, pp. 341–348.

[15] J. Deng, Y. Han, W. Heinzelman, P. Varshney, Scheduling sleeping nodes in high density cluster-based sensor networks, Mobile Networks and Applications 10 (2005) 825–835.

[16] P. Ding, J. Holliday, A. Celik, Distributed energy-efficient hierarchical clustering for wireless sensor networks, Distributed Computing in Sensor Systems 3560 (2005) 322–339.

[17] M. Gorlatova, P. Kinget, I. Kymissis, D. Rubenstein, X. Wang, G. Zussman, Challenge: ultra-low-power energy-harvesting active networked tags (enhants), in: Proceedings of the 15th Annual International Conference on Mobile Computing and Networking, 2009, pp. 253–260.

[18] H. Gou, Y. Yoo, H. Zeng, A partition-based leach algorithm for wireless sensor networks, in: Ninth IEEE International Conference on Computer and Information Technology, vol. 2, 2009, pp. 40–45.

[19] D. Hasenfratz, A. Meier, C. Moser, J.J. Chen, L. Thiele, Analysis, comparison, and optimization of routing protocols for energy harvesting wireless sensor networks, in: IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), 2010, pp. 19–26.

[20] D.W. Hearn, J. Vijay, Efficient algorithms for the (weighted) minimum circle problem, Operations Research 30 (1982) 777–795.

[21] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in: Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, vol. 2, 2000, p. 223.

[22] T.C. Hou, T.J. Tsai, A access-based clustering protocol for multihop wireless ad hoc networks, IEEE Journal on Selected Areas in Communications 19 (2001) 1201–1210.

[23] Y.F. Huang, W.H. Luo, J. Sum, C.W. Chang, L.H. Chang, R.C. Chen, Lifetime performance of an energy efficient clustering algorithm for cluster-based wireless sensor networks, in: Frontiers of High Performance Computing and Networking ISPA 2007 Workshops, vol. 4743, 2007, pp. 455–464.

[24] A. Iranli, M. Maleki, M. Pedram, Energy efficient strategies for deployment of a two-level wireless sensor network, in: Proceedings of the 2005 International Symposium on Low Power Electronics and Design, 2005, pp. 233–238.

[25] M. Islam, M. Islam, M. Islam, A-sleach: an advanced solar aware leach protocol for energy efficient routing in wireless sensor networks, in: Sixth International Conference on Networking, vol. 4, 2007.

[26] H. Karl, A. Willig, Protocols and Architectures for Wireless Sensor Networks, Wiley, Hoboken, NJ, 2005.

[27] K. Kinoshita, T. Okazaki, H. Tode, K. Murakami, A data gathering scheme for environmental energy-based wireless sensor networks, in: 5th IEEE Consumer Communications and Networking Conference, 2008, pp. 719–723.

[28] D. Kumar, T.C. Aseri, R. Patel, Eehc: energy efficient heterogeneous clustered scheme for wireless sensor networks, Computer Communications 32 (2009) 662–667.

[29] I.H. Li, I.E. Liao, F.N. Wu, An uncapacitated facility location based cluster hierarchy scheme on wireless sensor networks, WSEAS Transactions on Communications 9 (2010) 740–757.

[30] N.H. Mak, W. Seah, How long is the lifetime of a wireless sensor network?, International Conference on Advanced Information Networking and Applications (2009) 763–770

[31] A. Manjeshwar, D. Agrawal, Teen: a routing protocol for enhanced efficiency in wireless sensor networks, in: Proceedings 15th International Parallel and Distributed Processing Symposium, 2001, pp. 2009–2015.

[32] B. Medepally, N. Mehta, Voluntary energy harvesting relays and selection in cooperative wireless networks, IEEE Transactions on Wireless Communications 9 (2010) 3543–3553.

[33] X. Ning, C.G. Cassandras, Optimal cluster-head deployment in wireless sensor networks with redundant link requirements, in: Proceedings of the 2nd International Conference on Performance Evaluation Methodologies and Tools, 2007, pp. 241–249.

[34] E. Oyman, C. Ersoy, Multiple sink network design problem in large scale wireless sensor networks, IEEE International Conference on Communications 6 (2004) 3663–3667.

[35] J. Peng, L. Li, S. Xu, A novel energy efficient and reliable clustering algorithm in wireless sensor networks, in: IET Conference Publications, 2007, pp. 596–599.

[36] T. Prabhakar, S. Devasenapathy, H. Jamadagni, R. Prasad, Smart applications for energy harvested wsns, in: Second International Conference on Communication Systems and Networks (COMSNETS), 2010, pp. 1–7.

[37] L. Qing, Q. Zhu, M. Wang, Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks, Computer Communications 29 (2006) 2230–2237.

[38] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, M. Srivastava, Design considerations for solar energy harvesting wireless embedded systems, in: Fourth International Symposium on Information Processing in Sensor Networks, 2005, pp. 457–462.

[39] T.S. Rappaport, Wireless Communications: Principles and Practice, Prentice Hall PTR, New Jersey, 2002.

[40] G. Smaragdakis, I. Matta, A. Bestavros, SEP: a stable election protocol for clustered heterogeneous wireless sensor networks, in: Proc. of the Int'l Workshop on SANPA, 2004.

[41] S. Sudevalayam, P. Kulkarni, Energy harvesting sensor nodes: survey and implications, IEEE Communications Surveys Tutorials (2010) 1–19.

[42] T. Van Dam, K. Langendoen, An adaptive energy-efficient mac protocol for wireless sensor networks, in: SenSys'03: Proceedings of the First International Conference on Embedded Networked Sensor Systems, 2003, pp. 171–180.

[43] T. Voigt, A. Dunkels, J. Alonso, H. Ritter, J. Schiller, Solar-aware clustering in wireless sensor networks, in: Ninth International Symposium on Computers and Communications, vol. 1, 2004, pp. 238–243.

[44] R. Vullers, R. Schaijk, H. Visser, J. Penders, C. Hoof, Energy harvesting for autonomous wireless sensor networks, IEEE Solid-State Circuits Magazine 2 (2010) 29–38.

[45] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, C. Gill, Integrated coverage and connectivity configuration in wireless sensor networks, in: SenSys'03: Proceedings of the First International Conference on Embedded Networked Sensor Systems, 2003, pp. 28–39.

[46] K. Xu, M. Gerla, A heterogeneous routing protocol based on a new stable clustering scheme, Proceedings – IEEE Military Communications Conference MILCOM, vol. 2, 2002, pp. 838–843.

[47] M. Ye, C. Li, G. Chen, J. Wu, Eecs: an energy efficient clustering scheme in wireless sensor networks, in: 24th IEEE International Performance, Computing, and Communications Conference, 2005, pp. 535–540.

[48] W. Ye, J. Heidemann, D. Estrin, An energy-efficient mac protocol for wireless sensor networks, in: IEEE INFOCOM 3, vol. 3, 2002, pp. 1567–1576.

[49] M. Younis, M. Youssef, K. Arisha, Energy-aware management for cluster-based sensor networks, Computer Networks 43 (2003) 649–668.

[50] O. Younis, S. Fahmy, Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks, IEEE Transactions on Mobile Computing 3 (2004) 366–379.

[51] M. Zhang, C. Gong, Y. Lu, An novel dynamic clustering algorithm based on geographical location for wireless sensor networks, International Symposium on Information Science and Engineering 1 (2008) 565–568.

Pengfei Zhang received his Bachelor of Engineering degree in Electrical & Electronic Engineering from Nanyang Technological University in 2010. He is now pursuing his Ph.D. at Infinitus Lab in NTU, in conjunction with the Institute for Infocomm Research, supported by A*STAR. He is jointly supervised by Dr. Hwee-Pink Tan and Prof. Gaoxi Xiao. His research interests include energy efficient clustering algorithms, energy harvesting wireless sensor networks (EH-WSN), and multi-cluster distributed algorithms, etc.

Gaoxi Xiao received the B.S. and M.S. degrees in applied mathematics from Xidian University, Xi'an, China, in 1991 and 1994 respectively. In 1998, he received the Ph.D. degree in computing from the Hong Kong Polytechnic University. He was a Postdoctoral Research Fellow in Polytechnic University, Brooklyn, New York in 1999; and a Visiting Scientist in the University of Texas at Dallas in 1999–2001. He joined the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, in 2001, where he is now an Associate Professor. His main research interests include complex systems and networks, optical and wireless networking, and Internet technologies.

Hwee-Pink Tan is currently a Senior Scientist at the Institute of Infocomm Research (I2R), A*STAR and is also the SERC Programme Manager for the A*STAR Sense and Sense-abilities Program, where he leads a team of 20 full-time research scientists and engineers. He graduated from the Technion, Israel Institute of Technology, Israel in August 2004 with a Ph.D. Prior to joining I2R, he was awarded the A*STAR International Fellowship to conduct postdoctoral research at EURANDOM, Eindhoven University of Technology (December 2004–June 2006), and the Telecommunications Research Centre (CTVR), Trinity College Dublin (July 2006–March 2008). His research has focused on the design, modeling and performance evaluation of networking protocols for wireless networks, and his current research interests include underwater acoustic sensor networks and wireless sensor networks powered by ambient energy harvesting. He has been a Principal Investigator for several industry-projects in the above research areas. He has published more than 70 papers and has served on the TPC of numerous conferences and reviewer of papers for many key journals and conferences in the area of wireless networks.