Journal of Electronic Research and Application

Research Article

Distributed Algorithms for Maximizing Lifetime in Clustered Wireless Sensor Networks Using Energy-harvesting Relay Nodes

Pengfei Zhang[1], Hwee-Pink Tan[2], Gaoxi Xiao[3]

[1]Department of Environment and Energy, Institute for Infocomm Research, Singapore, [2]School of Information Systems, Singapore Management University, Singapore, [3]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore

Abstract: Motivated by recent developments in wireless sensor networks (WSNs), we present distributed clustering algorithms for maximizing the lifetime of WSNs, that is, the duration until the first node dies. We study the joint problem of prolonging network lifetime by introducing clustering techniques and energy-harvesting (EH) nodes. First, we propose a distributed clustering algorithm for maximizing the lifetime of clustered WSN, which includes EH nodes, serving as relay nodes for cluster heads (CHs). Second, graph-based and LP-based EH-CH matching algorithms are proposed which serve as benchmark algorithms. Extensive simulation results show that the proposed algorithms can achieve optimal or suboptimal solutions efficiently.

Keyword: *Wireless sensor network; distributed algorithm; energy harvesting wireless sensor; clustering algorithm; network lifetime*

Publication date: July 2018

Publication online: 31st July 2018

Corresponding Author: Pengfei Zhang, zhangpf2005@gmail.com

0 Introduction

Nowadays, wireless sensor networks (WSNs) are widely used in many monitoring applications, such as automobile, structure health monitoring, military, and health care. Numerous research efforts are carried out by researchers worldwide to improve the performance of WSNs, e.g.[1-3].

One of the critical limitations in conventional battery powered WSNs is finite network lifetime since sensor batteries may not be conveniently replaced or recharged. Many techniques exist for maximizing WSN network lifetime, focusing on various aspects. Among these techniques, clustering techniques[4-7] make use of advanced data aggregation techniques to aggregate data from sensors and forward them to the data sink.

A clustered WSN [Figure 1] is typically composed of many clusters and a base station (BS), the latter of which acting as a data sink. Each cluster comprises a cluster head (CH) and non CHs (NCHs). NCHs collect data from the environment and send them to CHs. In addition to sensing, CHs also receive data from NCHs, aggregate the data and forward them to BS, either directly, or through relay nodes. There are typically three phases in clustering protocols for WSNs: (i) CH selection, (ii) cluster formation, and (iii) data transmission. In most network scenarios, CHs strongly affect network lifetime since CHs have to communicate with BS through a longer distance than the distance between NCHs and CHs. Cluster formation also affects the lifetime of CHs since inappropriate cluster formation may force either CH or NCH to be depleted of energy sooner.

A good survey of existing studies on clustering for WSNs can be found in Singh and Sharma[7], these include energy-efficient algorithms[8-24], MAC layer design[25-27], and many more. Most of the clustering algorithms use a rotation of CHs among sensors to reduce the burden of CHs and balance energy consumptions among all the sensors. Existing work

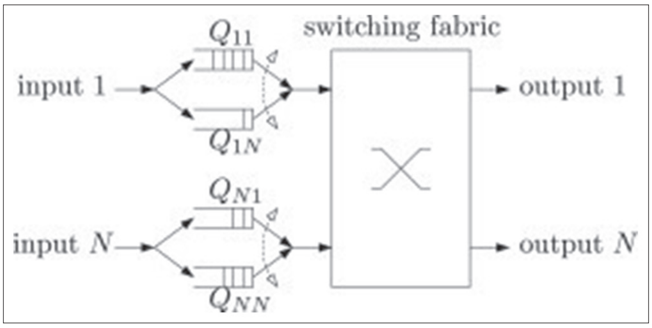

Figure 1. Abstract model of optical virtual output queueing switch[42]

on energy efficient clustering technology for WSNs is typically divided into centralized[16,20,21,23] and distributed approaches[8-14,22]. Distributed approaches make decisions based on local information exchanged between nearby sensors while centralized approaches try to solve an optimization problem based on global information, serving as a benchmark for the former.

Another way to overcome the shortage of limited battery capacity is to use energy harvesting (EH) technology[28-32] to harvest energy from the environment. EH nodes can potentially have infinite lifetime because their energy storage devices, such as super-capacitors, have a large number of recharge cycles. Since the deployments of large-scale WSNs composed solely of EH sensors remain impractical in the near future due to high costs and low achievable duty cycles, deploying EH sensors sparsely in WSNs may be a more practical approach[33,34].

Clustering methods and EH techniques can be combined together to prolong the network lifetime. Due to fluctuating energy harvesting rates[32], EH sensors may not be suitable to serve as CH nodes that need to operate continuously. In this paper, EH sensors are deployed sparsely and matched with CHs to help them relay data to BS without sensing from environment. This EH-CH matching, which is one of the main focuses of this paper, must be done optimally so that the energy consumption of CHs is reduced by communicating through a shorter distance for a certain fraction of time (e.g., when EH nodes are up and working) to relay instead of direct communication with BS.

In our previous work[23], we adopted a centralized approach to find the optimal locations for CHs and EH nodes, where we considered nodes powered by ambient EH as dedicated relay nodes for CHs, and proposed joint clustering and relay node placement algorithms for network lifetime maximization. We demonstrated the polynomial time convergence of our proposed algorithms, as well as their near optimality through extensive simulations. In a separate work[35], we considered the case where a single battery-powered relay node can communicate and coordinate with all the CHs in deciding the optimal schedule for the relay node to serve the CH nodes. Therefore, it is essentially still on a centralized scheme.

In this paper, we propose a distributed EH-CH matching algorithm with given locations for CHs and EHs. To achieve good EH-CH matching in a distributed manner, we borrow the idea adopted in virtual output queueing (VOQ) networks[36,37] to develop a simple matching scheme and provide analytical and extensive simulation results to demonstrate its fast convergence. We also benchmark its performance against algorithms based on LP and bipartite graph techniques, as well as randomized approach. Simulation results demonstrate much better performance (around 63.55% longer lifetime) compared with randomized approach.

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 present our network model and assumptions. In Section 4, we propose algorithms for CH selection, cluster formation, and CH-EH matching. In Section 5, we propose an optimized CH-EH matching algorithm as benchmarks to our proposed algorithm. In Section 6, extensive simulation results and discussions are presented for verifying the performance of the proposed algorithms. Finally, Section 7 concludes the paper and presents several directions for future research.

1 Related Work

Among the works[8-10] that maximize network lifetime, Heinzelman *et al.* proposed LEACH which uses a randomized rotation of the CH to avoid quickly draining the battery of any sensor in the network[8]. Qing *et al.* proposed and evaluated the distributed energy-efficient clustering scheme for heterogeneous WSNs (i.e., CHs and NCHs have different energies)[9]. A novel energy efficient clustering scheme was proposed in Ye *et al.*[10], which better suits periodical data gathering applications. These works consider battery powered sensor nodes where CHs directly communicate with BS. Our work extends the distributed algorithm by also utilizing EH sensors as a relay for CH nodes.

There have also been some studies on clustering in WSNs with EH nodes[38-41], typically assuming that the network is solely composed of EH sensors which have an infinite lifetime. Islam *et al.* considered a hybrid WSN which comprises both battery-powered and EH nodes[33]. However, they let EH nodes serve as CHs with a higher probability than battery-powered nodes. To the best of our knowledge, only our previous work[23] has studied on schemes maximizing network lifetime where EH nodes serve as relay nodes for CHs. However, it is for centralized system, which may only serve as a benchmark for evaluating the performance of distributed schemes.

To pair up EH and CH nodes, a distributed matching algorithm is needed. Matching algorithms have been extensively studied with wide applications, for example, those in optical VOQ networks [Figure 2][36]. The representative algorithms include parallel iterative matching[36] and iSlip[37].

that use random and round-robin approaches, respectively, for matching between input and output ports. Both algorithms are iterative and proven to converge in O (log N) iterations on average.

We draw the following analogies between WSNs and optical VOQ networks:

- In WSNs, every node broadcasts to nodes within its range. In optical VOQ networks, input ports share the bandwidth to transmit to output ports.
- The distributed algorithm in optical VOQ networks requires fast convergence and low delay, which is also applicable in wireless sensor networks to provide low control overhead.
- The signaling scheme between input and output ports is analogous to the signaling between different nodes in WSNk, in which coordination between nodes is required.

Based on the above similarities, we propose an EH-CH matching algorithm.

2 Network Model

The network scenario we adopt in this paper is shown in Figure 1. We make the following assumptions in the model without loss of generality throughout the paper:

- We assume a large scale network with N_s sensors with the same initial energy and N_e EH nodes randomly deployed in fixed locations within a square region with lower left-hand vertex (*a, b*) and dimension *M*. The sensor nodes are partitioned into N_c clusters, each comprising one CH.

Each EH serves as the relay for one CH or multiple CHs, which can be different at different time. A typical

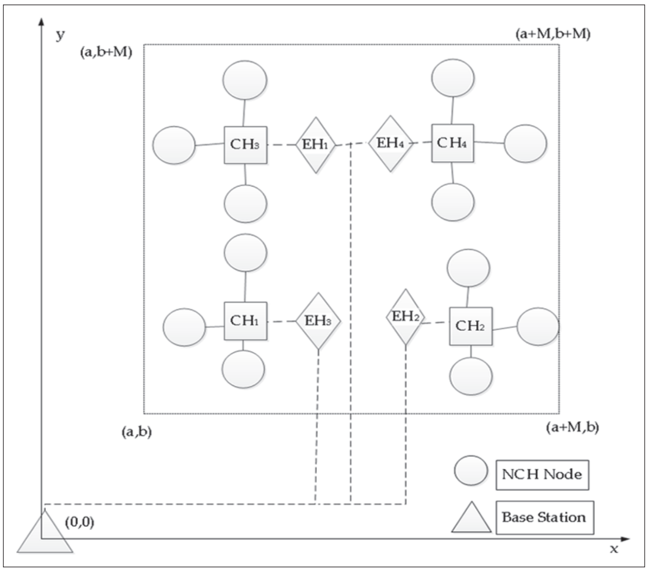

Figure 2. Abstract model of optical virtual output queueing switch[42]

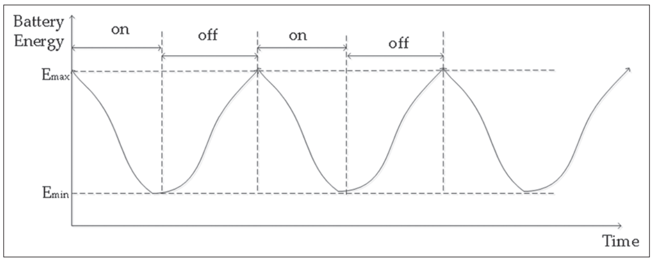

Figure 3. Typical energy profile of EH node with time

EH node's energy profile is shown in Figure 3 where it transmits data during the on phase, goes to off phase to harvest energy from environment and repeats the cycle after it has harvested enough energy. The BS is deployed at (0,0).

The distributed network operation in each round assumes time division multiple access TDMA and carrier sense multiple access (CSMA)-based communication[8] and comprises 2 phases, namely the setup phase and transmission phase, as shown in Figure 4. In the setup phase, selection of CHs, formation of clusters, and matching between EHs and CHs are performed. We assume that sufficient time is allocated in the setup phase for these operations and that each CH and T.

EH node consume overhead of a few bytes (assumed to be 4 bytes in the simulation which is enough for signaling). Each node can only make decisions based on local information such as residual energy and transmission power of nearby nodes. During the transmission phase, each NCH transmits a packet of 2000 bits to its respective CH in a TDMA time slot

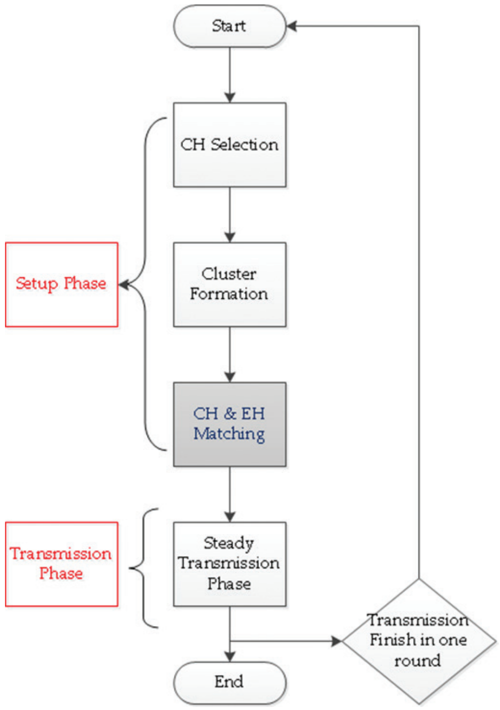

Figure 4. Overall flow chart for our design

assigned by the latter during the setup phase. Then, CHs forward data to BS using CSMA approach[1], either directly, or through EH node. The network lifetime is measured in number of TDMA rounds the network can operate until the first node dies.

We assume the Friss free-space propagation model[8], where the transmission power is proportional to the distance square between sensors. Note that the proposed algorithm can be easily extended to other propagation models, for example, multi-path fading model[43].

The notations used in the paper are shown in Table 1.

3 Distributed Clustering Algorithm for Maximizing Lifetime in WSNs with EH Sensors

Similarly, as that proposed in Heinzelman *et al.*[8], our clustering algorithm also re-selects CH periodically at the beginning of every round. We extend the work in Heinzelman *et al.*[8] by considering EH sensors as relay for CHs. Specifically, in Section 4.1, we propose CH selection algorithm; followed by the cluster formation algorithm in Section 4.2. Finally, we show our matching algorithm in Section 4.3.

1 We assume the collision of transmissions does not have a significant impact on network lifetime, similarly as assumed in[8].

3.1 CH selection algorithm

In the setup phase, sensors nominate themselves to serve as CH. Once the sensor is determined to be CH, it broadcasts an announcement message within its range. Each sensor has a probability of p_{si} to serve as CH. The expected number of CHs is $N_s p_{s_i}$. We show the optimal number of CHs, denoted as $N_c^{opt} = N_s p_{s_i}^{opt}$ through simulations in Section 6.1.

3.2 Cluster formation algorithm

Similarly, as that in Heinzelman *et al.*[8], in this step, every sensor selects the closest CH to join. The distance of a CH is indicated by the strength of signal received by NCHs. The closest CH from each NCH is the one with the strongest signal. Once NCHs determine the closest CH, they send join message to it.

3.3 Matching algorithm

After CH selection and cluster formation, our next step is to design the matching algorithm between CHs and EHs to maximize the time until the first CH dies. We consider the scenario when each EH node can serve as relay for only one CH in each round. We extend the consideration to the case when each EH node can serve as relay for multiple CHs in around in Section 5.2.

We assume that every sensor including CH or EH makes decision based on its local information collected (defined in Section 3). The main idea of our algorithm is that through several iterations, the algorithm can find a matching between CHs and EHs such that the CH node that EH node needs to serve is the sensor with lowest residual energy that has not been served by other EHs. By serving the sensor with lowest residual energy every round, the number of rounds until the first CH dies can be increased. For the CH, the best EH node that it could find is the closest EH node in its neighborhood followed by the second closest and so on.

To calculate the residual energy of CHs, we design the following approaches. We use $E_{CH_i}^{off}$ to denote the residual energy of CHi if without EH nodes serving as relay for CHs and each CH communicates directly with BS. E_{CH_i} denotes the residual energy of CH_i at the beginning of a round.

Let $E_{CH_i}^{on}$ denote the residual energy in CH_i after each round with EH node serving as relay for CHs, and denote the residual energy for CH_i after a round if it directly transmits to EH_j for a whole round as E_{CH_i,EH_i}^{on}.

Table 1: Notations used throughout this paper

Notation	Description	Value
(a, b)	Lower left-hand vertex of deployed region	N.A.
M	Dimension of square region	N.A.
N_s	Number of sensors in the network	{100, 125, 150, 175, 200}
N_e	Number of EH sensors in the network	N.A.
N_c	Number of clusters in the network	N.A.
i	Cluster index	$\{i=1,2,\ldots, N_c\}$
j	EH index	$\{j=1,2, \ldots, N_e\}$
T	Time duration for each round	N.A.
tr	Length of re-clustering period	N.A.
E_s	Energy stored in battery for each sensor	0.5J
CH_i	The CH node for cluster i	N.A.
E_{CH_i}	Residual energy in CH_i	N.A.
E_{th}	Energy threshold for node to wake up	N.A.
$P_{EH, h}$	EH rate for EH node	N.A.
$P_{CH_i, BS}$	Transmission power between CH_i and BS	N.A.
P_{CH_i, EH_j}	Transmission power between CH_i and EH_j	N.A.

After a round (duration T), $E_{CH_i}^{off}$ is shown in the following equation:

$$E_{CH_i}^{off} = E_{CH_i} - P_{CH_i,BS}T \tag{1}$$

Let t_{ij} represent the time EH_j serves as relay for CH_i in each round. In our scheme, EH node serves as relay for CH for a whole round, hence if $t_{i^*j^*} = T$ for CH_i^* and EH_j^*, then $t_{ij} = 0$ for $i = i^*, j \neq j^*$ and $i \neq i^*, j = j^*$. For simplicity of the scheme design, we assume EH node can serve as relay for CHs only if its residual energy can sustain transmission of at least one round with duration T. Besides, we assume that, during the transmission, EH nodes that wake up do not broadcast awake messages until the beginning of the next round[2]. We have:

$$E_{CH_i} - E_{CH_i}^{on} = (T - \sum_{j=1}^{Ne} t_{ij})P_{CH_i,BS} + \sum_{j=1}^{Ne} t_{ij}P_{CH_i,EH_j} \tag{2}$$

$$E_{CH_i}^{on} = E_{CH_i} - P_{CH_i,BS}T + \sum_{j=1}^{Ne} t_{ij}(P_{CH_i,BS} - P_{CH_i,EH_j}) \tag{3}$$

Let

$$\alpha_{ij} = P_{CH_i,BS} - P_{CH_i,EH_j} \tag{4}$$

2 Since EHs may wake up during the transmission, they may broadcast *awake* message to their surrounding CHs. If they do so, however, CHs may consume quite a lot of energy to re-match the new EHs. For simplicity, when EHs wake up during the transmission, they will wait until the beginning of the next round to broadcast *awake* message.

From (1), (2), (3), (4), we have:

$$E_{CH_i}^{on} = E_{CH_i}^{off} + \sum_{j=1}^{Ne} t_{ij}\alpha_{ij} \tag{5}$$

Specifically, E_{CH_i,EH_j}^{on} is expressed as follows:

$$E_{CH_i,EH_j}^{on} = E_{CH_i}^{on} + T\alpha_{ij}$$

At the setup phase, we use Algorithm 1 to match CHs and EHs. It is important to analyze the number of iterations to converge and the performance of the algorithm. A near optimal matching between CHs and EHs can be achieved through many iterations of signaling between CHs and EHs. This can cause higher energy consuming that in the overhead. Hence, there is a tradeoff between the performance of the algorithm and number of iterations/signals that each EH/CH may spend for the matching.

Since in every iteration, at least one CH is matched to either EH or BS, Algorithm 1 requires at most N iterations to converge, where $N = \min(N_c, N_e)$.

Figure 5 shows the average number of iterations required for convergence while 100 simulations are performed for each case. The same number of CHs and EH nodes is randomly deployed in 100–200 m^2 region. We change the number of CHs and EHs from 6 to 55. The simulation result shows that the process will converge in approximately O ($\log_2[N]$) iterations on average.

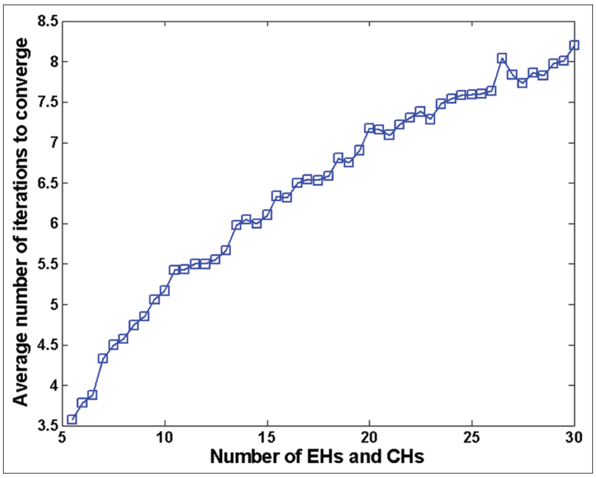

Figure 5. The average numbers of iterations required to converge while using the Algorithm 1 in WSNs with different numbers of EHs/CHs

We use the timing diagram in Figure 6 to show the overhead messages exchanged during the setup phase.

4 Benchmark Matching Algorithms for Maximizing Lifetime of WSNs with EH Sensors

Algorithm 1 in Section 4.3 provides distributed matching between CHs and EHs to maximize lifetime. To show the performance of our algorithm, we propose centralized benchmark algorithms using information about all CHs and EHs in this section. Specifically, in Section 5.1, we propose graph-based benchmark which serves as upper bound for Algorithm 1. In Section 5.2, we propose an LP-based benchmark.

Algorithm 1 Distributed matching algorithm at setup phase

Require: N_c CHs and N_e EHs
Ensure: Matching between CHs and EHs

At beginning, EH node broadcasts *awake* message to CHs in their broadcast range. Each CH estimates P_{CH_i}, EH_j according to the signal strength and calculates $E^{on}_{CH_i,EH_j}$. It also records the smallest $E_i = min\, E^{on}_{CH_i,EH_j}$.

while there are unmatched CHs **do**
For each unmatched CH:
if there are unmatched EH nodes **then**
It selects the *closest* unmatched EH node and send *request* message including E_i to this EH node.
else[CH node does not receive any *awake* message]
It directly communicates with *BS*.
end if
For each unmatched EH:
EH node receives several *request* messages from CH nodes in its range.
Denote the number of *requests* as N_r, N_r may be equal to 0
if $N_r > 0$ **then**
EH node sends *grant* to the CH node with smallest E_i.
else[$N_r = 0$]
EH node waits for the next iteration.
end if
For each unmatched CH:
CH node may or may not receive *grant* messages from EH nodes in its range, let N_g be the number of *grant* messages it receives.
if $N_g = 1$ **then**
It selects the closest EH node and sends *accept* message to it.
else $N_g = 0$
Return to while loop.
end if
end while

At the beginning of setup phase, after some information exchanges between CHs and EHs (i.e., the overhead shown in Figure 6 that we take into consideration), we assume that:

- N_e EH nodes know the residual energy and energy consumption rate of each CH.
- The energy consumed in the overhead occurs once at the beginning of each round.

4.1 Graph-based matching algorithm

The objective for the algorithm is to find a matching between CHs and EHs or BS to maximize the minimal $E^{on}_{CH_i,EH_j}$ or $E^{off}_{CH_i}$ (defined in Section 4.3). In other words, the problem is similar to a maximum-minimum matching problem in the bipartite graph. CH represents one set in a bipartite graph while EH and BS represent the other set. The problem is to look for the "largest set of edges" that cannot be removed. A matching exists if all CHs have been matched to either EH node or BS. Each EH node can only match one CH node while BS can match multiple CH nodes. Each CH node can only be matched to one EH node or BS.

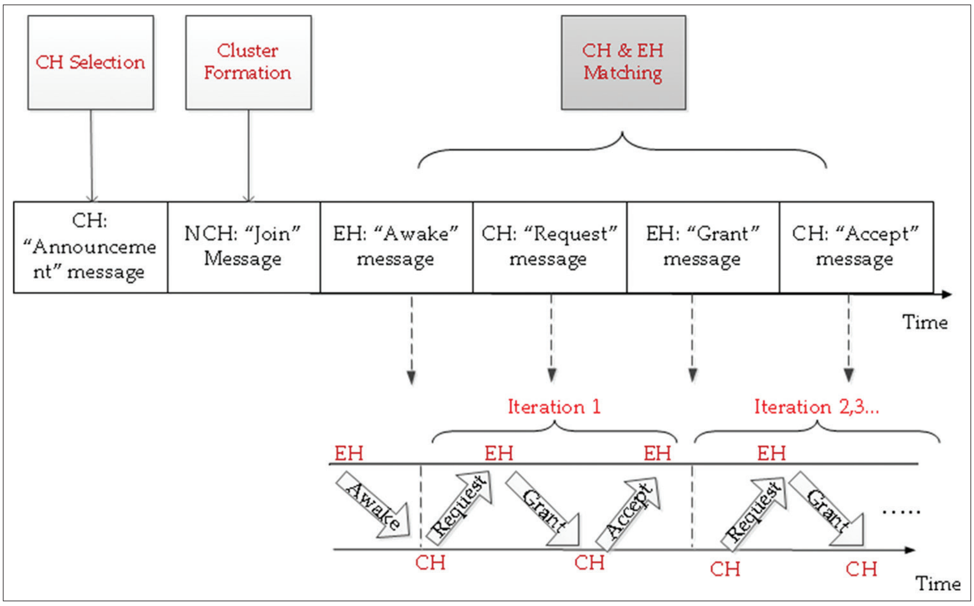

Figure 6. Timing diagram for message exchanges in overhead

Our idea is to assign proper weights to edges between two sets taking into consideration that BS can be matched to multiple CHs. Then, we sort these edges in ascending order and use binary search and Hopcroft–Karp Algorithm[44] to find the set of edges that cannot be removed.

Let $W(i, j)$ be the weight assigned to the edge between CHi and EH_j or BS. First, we derive an $N_c(N_e + N_c)$ matrix W with row index $1 \leq i \leq N_c$ and column index $1 \leq j \leq N_e + N_c$. $W(i, j)$ in row i and column j where $1 \leq j \leq N_e$ represents $E^{on}_{CH_i,EH_j}$ according to equation (5). $W(i, j)$ in row i and column j where $N_e + 1 \leq j \leq N_e + N_c$ represents E_{CH_i} according to equation (1).

We sort $W(i, j)$ in an ascending order and then remove the smallest $W(i, j)$ one by one until we cannot find a matching. In our case, a matching exists if the maximal number of matches is larger than or equal to N_c. By adopting binary search to cut the "largest" edge and check if there is a matching, that will reduce the time complexity from $O(n^4\sqrt{n})$ to $O(n^2\sqrt{n}log(n))$. The algorithm is shown in Algorithm 2[3].

The optimality of the algorithm is defined as whether it can generate a matching such that within the matching, min $(W[i, j])$ can be maximized in each round. We prove Algorithm 2 can maximize min $(W(i, j))$ in a *matching* for each round in Theorem 1.

Algorithm 2 Optimized Graph-based Algorithm for multiple EH nodes to re-select CH as relay at setup phase

- **Step 1**
  Sort $W(i, j)$ in ascending order, where $1 \leq i \leq N_c$ and $1 \leq j \leq N_e + N_c$. We record the sorted array as L_i, where $1 \leq i \leq N_c(N_e + N_c)$. We have $L_1 \leq L_2$. $\leq L_{N_C}(N_e+N_c)$. Let iteration $k = 1$, define $lb = 1$, $ub = N_c(N_e + N_c)$.
- **Step 2**
  Let $b_k = [\frac{lb+ub}{2}]$. Use the Hopcroft–Karp Algorithm[44] to check if there still exists a matching if we remove all L_i for $i \leq b_k$. If yes, remove all the L_i for $i \leq b_k$ and let $l_b = b_k$; otherwise, do not remove any element but let $ub = b_k$.
- **Step 3**
  Let $k = k + 1$, repeat Step 2 until $lb = ub$. We generate the matches between corresponding CH_i and EH_j and record L_{lb}.

Theorem 1: *Algorithm 2 can maximize* min$(W(i, j))$ *in a matching for each round*

Proof. Proof by contradiction: Suppose there is an optimal *matching* denoted as W^* where CHs have been matched to either EHs or BS. Let $L^* = \min(W^*(i, j))$, $1 \leq i \leq N_c$, $1 \leq j \leq N_e + N_c$. We denote the *matching* from Algorithm 2 as $\overline{W}$ (i, j), 1 i N_c, 1 j $N_e + N_c$. Let $\overline{L}$ = min $(W(i, j)) = L_{lb} = L_{ub}$

3 There are many possible solutions to maximize the minimal $E^{on}_{CH_i,EH_j}$ or $E^{off}_{CH_i}$ scheme can finds one solution among them. The algorithm can be extended to maximize the second minimal $E^{on}_{CH_i,EH_j}$ or $E^{off}_{CH_i}$, the third minimal $E^{on}_{CH_i,EH_j}$ or $E^{off}_{CH_i}$ and so on, which is out of scope of this paper.

We have $L^* > \overline{L}$ as the assumption of contradiction, which means matching W^* exists for L_i when $i > lb$. However, according to Algorithm 2, there is no *matching* for L_i when $i > lb$, which causes contradiction.

4.2 LP-based matching algorithm

In our previous analysis, we considered a scenario when each EH serves as relay for one CH in around, which was denoted as Scenario A. In this section, we consider a new scenario, denoted as scenario B, where each EH node can serve as relay for multiple CHs in a round. Intuitively, Scenario B can have better lifetime compared with Scenario A if overhead is small. Scenario A is simple for a large scale network and has lower control overhead compared with Scenario B[4].

LP approach is a benchmark algorithm for Scenario B. It provides longer lifetime compared with Algorithm 2 for Scenario A when the control overhead is ignored. We compare these two different scenarios and see the improvements Scenario B can have over Scenario A.

The problem can be formulated as follows: We would like to find t_{ij} for $I = 1,2.,N_c$ and $j = 1,2.,N_e$ such that the minimum residual energy left in CH_i is maximized after each round. Quantitatively, this can be expressed as the following maximum-minimum problem:

MMWSN: To maximize z = min

$$\left(E^{on}_{CH_i} = E^{off}_{CH_i} + \sum_{j=1}^{Ne} t_{ij}\alpha_{ij}\right)\forall i$$

1. $$\sum_{j=1}^{Ne} t_{ij} \le T \forall i$$

2. $$\sum_{j=1}^{Ne} t_{ij} P_{CH_i, EH_j} \le E_{CH_i}$$

3. $t_{ij} \geq 0$, □$_i$, □$_j$

Note that $E^{on}_{CH_i} = E^{off}_{CH_i}$ if $t_{ij} = 0$ for all j. $\sum_{j=1}^{Ne} t_{ij}$ in constraint (1) represents the total time CH_i has been served by all EHs in a round. Since the duration of each round is T, constraint (1) ensures that the total time all EH_j serve as relay for each CH cannot exceed T. Constraint (2) ensures that the energy consumption by CH_i does not exceed its residual energy. Constraint (3) makes sure that t_{ij} is not negative. This maximum-minimum optimization problem is similar to the problem[45], denoted by MM, of allocating resources $j \in 1,\ldots,J$ to activities $k \in 1,\ldots,K$, as shown below:

MM: To maximize $z = \min_i \left[r_i \left(\sum_{j=1}^{J} \beta_{ij} x_{ij} \right) \right]$

s.t.:

1. $$\sum_{i=1}^{K} x_{ij} = j = 1,2,\ldots,J$$

2. $x_{ij} \geq 0$

x_{ij} is the quantity of resource j allocated to activity i; h_j is the available quantity of resource j; β_{ij} is the effectiveness of resource j when allocated to activity i; and r_i is a return function for activity i. Specifically,

$$r_i(x_{ij}) = a_i + b_i \left(\sum_{j=1}^{J} \beta_{ij} x_{ij} \right)$$

By letting $r_i = E^{off}_{CH_i} + \sum_{j=1}^{Ne} t_{ij}\alpha_{ij}, x_{ij} = t_{ij}, h_{ij} = T$, **MMWSN** is similar to **MM** for both objective and constraints.

According to Mjelde[45], an optimal solution of this problem can be obtained by solving the following linear programming problem:

MMWSN-LP: To maximize z

s.t.:

1. $$E^{off}_{CH_i} + \sum_{j=1}^{Ne} t_{ij}\alpha_{ij} = E^{on}_{CH_i} \ge z \forall i$$

2. $$\sum_{j=1}^{Ne} t_{ij} \le T \forall i$$

3. $$\sum_{j=1}^{Ne} t_{ij} P_{CH_i, EH_j} \le E_{CH_i}$$

4. $t_{ij} \geq 0$, □$_i$, □$_j$

Since the constraints are linear, this **MMWSN-LP** can be optimally solved according to Mjelde[45]. We describe the algorithm in Algorithm 3.

Algorithm 3 LP Algorithm for multiple EH nodes to re-select CH as relay at setup phase

---

- **Step 1**
  Calculate $E^{off}_{CH_i}$ and α_{ij} for CH_i and EH_j according to equations (1) and (4).
- **Step 2**
  Formulate the problem as maximum-minimum-sum problem as shown in **MMWSN-LP**. After solving this problem, we use t_{ij} to represent the optimal time EH_j serves as relay for CH_i.
- **Step 3**
  Repeat Steps 1-2 at the beginning of each round until one of the CHs has $E_{CH_i} < 0$. Finally, we record the network lifetime in terms of number of rounds.

---

4 Distributed algorithm for scenario A can be extended to suit scenario B with more overhead and higher complexity. We show in Section 6 that, under our deployment cases, the optimal lifetime for Scenario A and Scenario B has slight difference. Thus we only derive the centralized algorithm for Scenario B in this section.

5 Simulation Results

We demonstrate the performance of our proposed algorithms through simulations using MatLab for a 2D network with a BS deployed at (0,0) and N_s={100,125,150,175,200} 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: **Case I** a = 100, b = 100, M = 100; **Case II** a = 200, b = 200, M = 100; **Case III** a = 0, b = 0, M = 200. Although Cases I and II have the same area, nodes are located closer to the BS in Case I than Case II. Cases I and III differ in terms of node density. Using these three cases, we can determine the influence of proximity to the BS and node density on the performance of our proposed algorithms.

The performance metric used in this paper is the number of rounds (defined in Section 3) the network could operate until the first node dies. We first show the effects of parameters N_c, t_r, N_e, $P_{eh,h}$ on the network lifetime in Section 6.1. Due to the page constraint, we show the effects of parameters in Case I only, the same conclusions also hold for Case II and Case III. $P_{eh,h}$ is assumed according to the previous literature[28,46], typically tens of milliwatts. Then we verify the sub-optimality of our algorithm in Section 6.2. We benchmark our distributed algorithm with algorithms proposed in Section 5.1 and Section 5.2. We also compare our results with PIM shown in Section 2.

5.1 Performance characteristics of distributed algorithm [Case I]

We first examine the effects of number of clusters on network lifetime when N_e = 4, $P_{eh,h}$ = 0.04 W, t_r = T and E_{th} = E_s. We plot the results from our simulation in Figure 7 when 1 N_c 10 for Case I. The results show that the network lifetime is longest when $N_c^{opt} = 4$.

The effects of t_r on the algorithm performance in Case I are shown in Figure 8 when p_{si} = 0.04, N_e = 4, $P_{eh,h}$ = 0.04 W, and E_{th} = E_s. The results, obtained by averaging over 20 runs, show that the length of reclustering period has effects on the lifetime. Specifically, in Case I, when p_{si} is varied between 0.02, 0.025, and 0.03, the optimal length of re-clustering period also varies. Specifically, the optimal length of re-clustering period is 2T, 3T, and 2T for these three different p_{si}, respectively. When p_{si} = 0.025, the lifetime when t_r = 3T is 3.7% longer than the lifetime when t_r = T. This is because the more frequently CHs are re-selected, more energies are consumed. It is interesting, though not a big surprise, to see that when the length of re-clustering period increases, network lifetime increases first and then decreases. When re-clustering period increases, those CHs which were not *bottleneck* CHs previously may become *bottleneck* CHs since, until the next re-clustering, they might have to consume high transmission power to transmit to farther EHs. Therefore, the network lifetime decreases due to the high power consumption over long time by these new *bottleneck* CHs.

Note that we have four EH nodes in the previous few figures. Now we show the effect of number of EH nodes when p_{si} = 0.04, $P_{eh,h}$ = 0.04 W, t_r = T and E_{th} = E_s in Figure 9. As expected, when the number of EH nodes

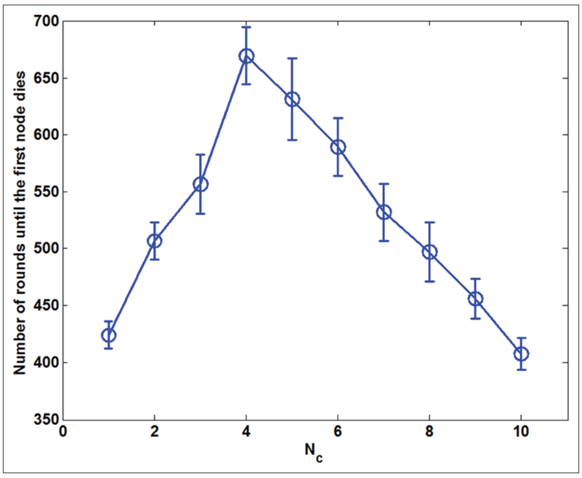

Figure 7. Effects of number of clusters on network lifetime in Case I (N_e = 4, $P_{eh,h}$ = 0.04 W, t_r = T and E_{th} = E_s

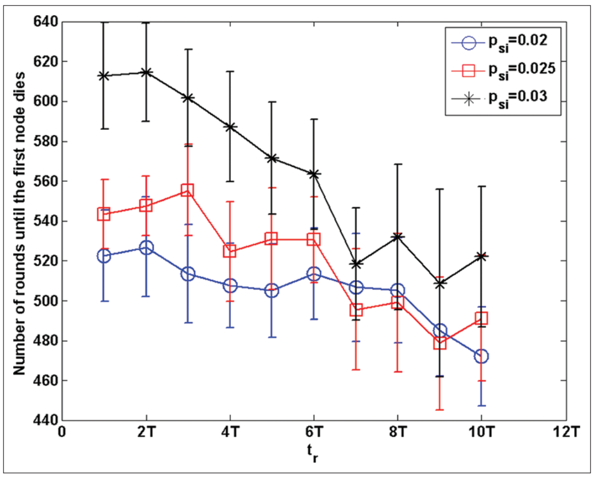

Figure 8. Effects of *tr* on network lifetime in Case I (p_{si} = 0.04, N_e = 4, $P_{eh,h}$ = 0.04 W, E_{th} = E_s)

increases, network lifetime increases. Interestingly, when N_e is increased further, that is, $N_e > 10$, the increase in the network lifetime becomes less and less significant. With further increase of N_e, the network lifetime will not increase. Theoretically, when Ne approaches infinity, CHs will not become bottleneck nodes. Instead, NCHs which are farthest from CHs will become bottleneck nodes, thus affecting the network lifetime.

We also show the effects of EH rate on network lifetime in Figure 10 when $N_e = 4$, $p_{si} = 0.04$, $t_r = T$ and $E_{th} = E_s$. The EH rate is increased from 0.01 W to 0.1 W in steps of 0.01 W. When EH rate is 0, the case simplifies to battery powered network, already shown in Zhang *et al.*[23] As expected, when harvesting rate increases, network lifetime increases as well. Specifically, when harvesting rate increases from 0.01 W to 0.04 W for 100 sensors, network lifetime increases by 44.4% from 432.55 to 624.65. Interestingly, we also notice when harvesting rate continues to increase, the increase in lifetime becomes less and less significant; with further increase of harvesting rate, network lifetime will approach constant.

Theoretically when harvesting rate approaches infinity, CHs will not become bottleneck nodes. Instead, NCHs which are farthest from CHs will become bottleneck nodes, thus affecting the network lifetime.

Then, we illustrate the performance of the proposed algorithms for different E_{th} ranging from {$0.1E_s$, $0.2E_s$., E_s} for case I when $N_e = 4$, $p_{si} = 0.04$, $t_r = T$ and $P_{eh,h} = 0.04W$. We compare the performance of our algorithm when number of sensors is 100 and 150 for Case I, and plot the results in Figure 11. As seen from this Figure 11, the network lifetime is not a monotonic function of E_{th}. For example, the optimal lifetime is achieved when remaining energy is $0.7E_s$ when number of sensors is 100 and is $0.6E_s$ when number of sensors is 150. As shown in Section 4.3, each EH node broadcasts to awake message when they can transmit for the whole round. From the Figure 11, we can see that E_{th} does not significantly affect the network lifetime. For example, the increase in the lifetime when E_{th} is $0.7E_s$ is only 3.16% more than the case when E_{th} is E_s when $N_s = 100$. When $E_{th} = E_s$, the time EH node to recharge is longer compared with the time for EH node to recharge when $E_{th} = 0.7E_s$. On the other hand, the time EH node could serve relay for CHs when $E_{th} = E_s$ is also longer compared with the time when $E_{th} = 0.7E_s$.

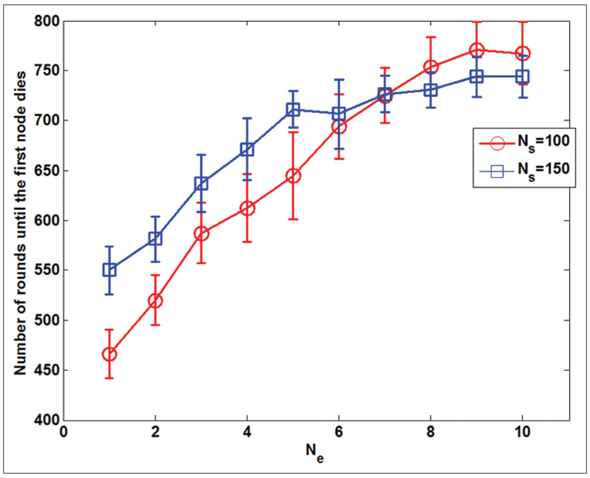

Figure 9. Effects of number of EH nodes on the network lifetime in Case I ($p_{si} = 0.04$, $P_{eh,h} = 0.04\ W$, $t_r = T$, $E_{th} = E_s$).

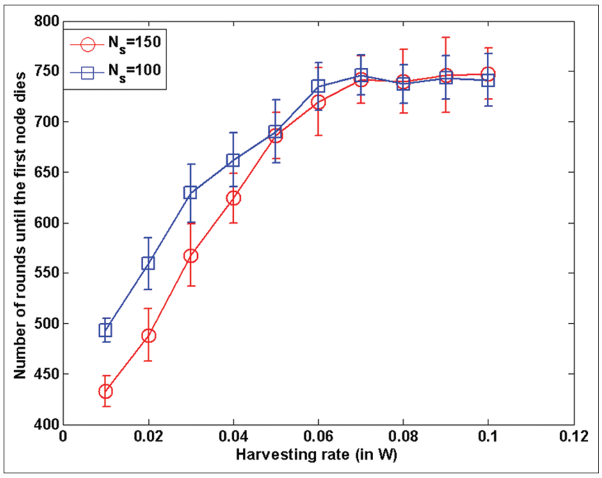

Figure 10. Effects of EH node harvesting rate on the network lifetime in Case I ($N_e = 4$, $p_{si} = 0.04$, $t_r = T$, $E_{th} = E_s$).

5.2 Comparison of distributed algorithm with benchmark algorithms

Next, we compare the performance of our distributed algorithm (DA in short) with respect to graph-based benchmark (GB in short) (shown in Section 5.1), LP-based benchmark (LP in short) (shown in Section 5.2), and PIM (random matching shown in Section 2). We adopt the same parameters in these approaches, that is $p_{si} = 0.04$, $N_e = 4$, $t_r = T$, $P_{eh,h} = 0.04\ W$, and $E_{th} = E_s$, which are selected according to Section 6.1.

For Case I, network lifetime increases when N_s increases as shown in Figure 12. The reason is that when number of sensors increases, more sensors will be able to serve as CHs, which reduces the chances CHs, become bottleneck nodes. The result shows that our algorithm outperforms PIM and is quite close to the benchmarks. Specifically, for Case I with 100 and

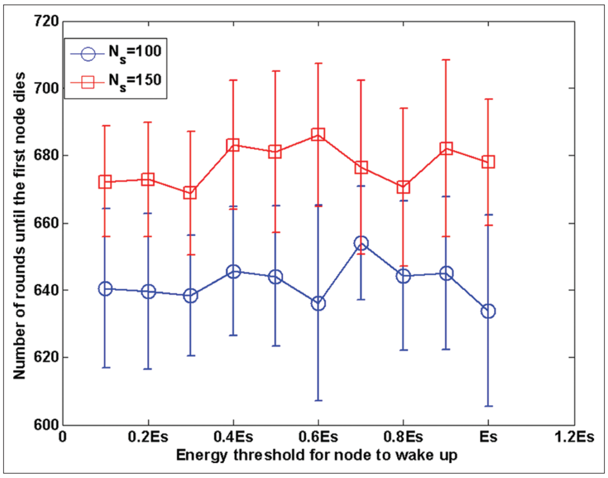

Figure 11. Effects of E_{th} on the network lifetime in Case I ($N_e = 4$, $p_{si} = 0.04$, $t_r = T$, $P_{eh,h} = 0.04$ W).

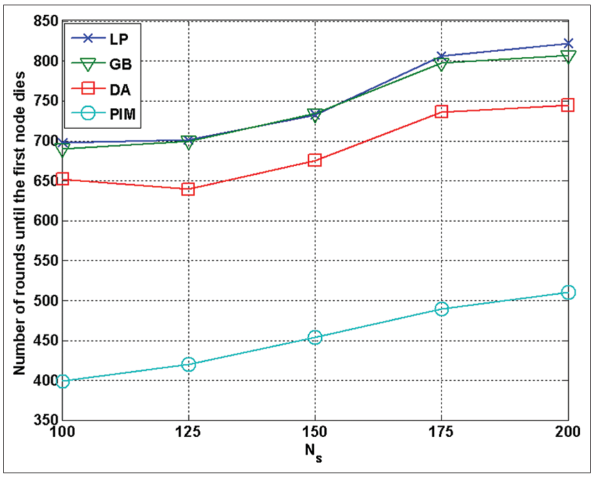

Figure 12. Comparison between our algorithm with benchmark algorithms in Case I ($p_{si} = 0.04$, $N_e = 4$, $t_r = T$, $P_{eh,h} = 0.04$ W, $E_{th} = E_s$)

150 sensors, the network lifetime from DA is 63.5% and 48.8% longer compared with PIM. This is because our algorithm selects the EH node to serve the best CHs it could find; while PIM considers using EH nodes to randomly serve a certain CH. Thus, our algorithm maximizes the lifetime until the first node dies while PIM is trying to find a matching within fewer iterations. When we compare DA with the benchmark algorithms, we found the network lifetime in Case I for DA with 100 and 150 sensors can achieve 93.4%, 94.4% and 92.1%, 92.0% optimality compared with LP and GB. Note that[5]:

i. We are optimizing the performance within a few iterations while LP and GB adopt more complicated calculations.
ii. Our algorithm is based on the local information that each EH or CH node receives while the benchmark algorithms uses global information to make decisions.

Hence, we may claim that the proposed algorithm achieve satisfactory performance and good scalability.

As mentioned in Section 5.2, we compare the Scenario A and Scenario B in this section. GB and DA are for Scenario A while LP is for Scenario B. The result shows that LP only provides slightly longer lifetime compared with GB. For example, with 100 and 150 sensors, the differences between LP and GB results are 1% and 0.1%, respectively. The differences between Scenario A and Scenario B is small in Case I, however, for some special deployment scenarios and parameters, the difference may be large, which is not covered in this work.

6 Conclusion

In this paper, we considered clustered WSNs where CHs either aggregate and forward data directly to BS, or through dedicated relay nodes with EH capabilities. We proposed efficient distributed matching algorithm between CHs and EHs to maximize network lifetime, where the network lifetime is the duration until the first node runs out of energy. Through theoretical analysis and extensive simulations, we validated the performance of the proposed algorithms. Simulation results demonstrate much better performance (around 63.55% longer lifetime) compared with randomized approach. Specifically, with low complexity ($\log_2(n)$), our algorithm achieves sub-optimality compared with benchmarks for matching between CHs and EHs in our deployment scenario. In addition, we showed the existence of an optimal length of re-clustering period for a given network configuration as well as the effect of energy threshold for EH node to wake up.

For future work, we plan to (i) study the effects of EH rates obtained from real measurements on lifetime; (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.

7 Acknowledgment

This work was carried out while Zhang was a postgraduate student at Nanyang Technological

5 Strictly speaking, these benchmarks only provide optimized results for each round rather than optimizing the overall network lifetime. Optimizing overall network lifetime requests much more complicated approaches, which is out of scope of the paper.

University sponsored by A*STAR Graduate Scholarship (AGS) from September 2010.

References

[1] Rawat P, Singh KD, Chaouchi H, Bonnin JM. Wireless sensor networks: A survey on recent developments and potential synergies. J Supercomput 2014;68:1-48.

[2] Vullers R, Schaijk R, Visser H, Penders J, Hoof C. Energy harvesting for autonomous wireless sensor networks. IEEE Solid State Circuits Mag 2010;2:29-38.

[3] Khan I, Belqasmi F, Glitho R, Crespi N, Morrow M, Polakos P. Wireless sensor network virtualization: A survey. IEEE Commun Surv Tutor 2015;18:553-76.

[4] Singh J, Singh BP, Shaw SB. A survey on leach-based hierarchical routing protocols in wireless sensor network. Int J Eng Res Technol 2014;3:98-102.

[5] Liu X. A survey on clustering routing protocols in wireless sensor networks. Sensors 2012;12:11113-53.

[6] Tyagi S, Kumar N. A systematic review on clustering and routing techniques based upon leach protocol for wireless sensor networks. J Netw Comput Appl 2013;36:623-45.

[7] Singh SP, Sharma S. A survey on cluster based routing protocols in wireless sensor networks. Procedia Comput Sci 2015;45:687-95.

[8] Heinzelman W, Chandrakasan A, Balakrishnan H. Energy Efficient Communication Protocol for Wireless Microsensor Networks. Proceedings of the 3rd Annual Hawaii International Conference on System Sciences. Vol. 2. Maui, Hawaii: 2000. p. 223.

[9] Qing L, Zhu Q, Wang M. Design of a distributed energy efficient clustering algorithm for heterogeneous wireless sensor networks. Comput Commun 2006;29:2230-7.

[10] Ye M, Li C, Chen G, Wu J. EECS: An Energy Efficient Clustering Scheme in Wireless Sensor Networks 24th IEEE International Performance, Computing, and Communications Conference; 2005. p. 535-40.

[11] Younis O, Fahmy S. Heed: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. IEEE Trans Mobile Comput 2004;3:366-79.

[12] Zhang M, Gong C, Lu Y. An novel dynamic clustering algorithm based on geographical location for wireless sensor networks. Int Symp Inf Sci Eng 2008;1:565-8.

[13] Huang YF, Luo WH, Sum J, Chang LH, Chang CW, Chen RC. Lifetime Performance of an Energy Efficient Clustering Algorithm for Cluster-Based Wireless Sensor Networks. Vol. 4743. Frontiers of High Performance Computing and Networking ISPA 2007 Workshops; 2002. p. 455-64.

[14] Peng J, Li L, Xu S. A Novel Energy Efficient and Reliable Clustering Algorithm in Wireless Sensor Networks. IET Conference Publications; 2007. p. 596-9.

[15] Wang SS, Chen ZP. LCM: A link-aware clustering mechanism for energy-efficient routing in wireless sensor networks. IEEE Sens J 2013;13:728-36.

[16] Chehri A, Mouftah H. An efficient cluster head placement for hybrid sensor networks. Ad-Hoc Mobile Wirel Netw 2010;6288:123-34.

[17] Peiravi A, Mashhadi HR, Javadi SH. An optimal energy efficient clustering method in wireless sensor networks using multi objective genetic algorithm. Int J Commun Syst 2013;26:114-26.

[18] Bagci H, Yazici A. An energy aware fuzzy approach to unequal clustering in wireless sensor networks. Appl Soft Comput 2013;13:1741-9.

[19] Kumar D. Performance analysis of energy efficient clustering protocols for maximizing lifetime of wireless sensor networks. IET Wirel Sens Syst 2014;4:9-16.

[20] Hoang DC, Yadav P, Kumar R, Panda SK. Real time implementation of a harmony search algorithm-based clustering protocol for energy efficient wireless sensor networks. IEEE Trans Ind Inf 2014;10:774-83.

[21] Kuila P, Jana PK. Energy efficient clustering and routing algorithms for wireless sensor networks: Particle swarm optimization approach. Eng Appl Artif Intell 2014;33:127-40.

[22] Javaid N, Qureshi T, Khan A, Iqbal A, Akhtar E, Ishfaq M. EDDEEC: Enhanced developed distributed energy-efficient clustering for heterogeneous wireless sensor networks. Procedia Comput Sci 2013;19:914-9.

[23] Zhang P, Xiao G, Tan HP. Clustering algorithms for maximizing the lifetime of wireless sensor networks with energy-harvesting sensors. Comput Netw 2013;57:2689-704.

[24] Pantazis NA, Nikolidakis SA, Vergados DD. Energy efficient routing protocols in wireless sensor networks: A survey. IEEE Commun Surv Tutor 2013;15:551-91.

[25] Huang P, Xiao L, Soltani S, Mutka MW, Xi N. The evolution of mac protocols in wireless sensor networks: A survey. IEEE Commun Surv Tutor 2013;15:101-20.

[26] Dong Q, Dargie W. A survey on mobility and mobility aware mac protocols in wireless sensor networks. IEEE Commun Surv Tutor 2013;15:88-100.

[27] Pawar PM, Nielsen RH, Prasad NR, Prasad R. Mobility impact on cluster based mac layer protocols in wireless sensor networks. Wirel Pers Commun 2014;74:1213-29.

[28] Sudevalayam S, Kulkarni P. Energy harvesting sensor nodes: Survey and implications. IEEE Commun Surv Tutor 2010;99:1-19.

[29] Gorlatova M, Kinget P, Kymissis I, Rubenstein D, Wang X, Zussman G. Challenge: Ultra Low Power Energy Harvesting Active Networked Tags (EnHants). Proceedings of the 15th Annual International Conference on Mobile Computing and Networking; 2009. p. 253-60.

[30] Yoo H, Shim M, Kim D. Dynamic duty-cycle scheduling schemes for energy-harvesting wireless sensor networks. IEEE Commun Lett 2012;16:202-4.

[31] Hasenfratz D, Meier A, Moser C, Chen JJ, Thiele L. Analysis, Comparison, and Optimization of Routing Protocols for Energy Harvesting Wireless Sensor Networks. 2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC); 210. p. 19-26.

[32] Bergonzini C, Brunelli D, Benini L. Algorithms for Harvested Energy Prediction in Battery less Wireless Sensor Networks. 3rd International Workshop on Advances in sensors and Interfaces; 2009. p. 144-9.

[33] Islam M, Islam M, Islam M. A-sLEACH: An Advanced

Solar Aware Leach Protocol for Energy Efficient Routing In Wireless Sensor Networks. 6th International Conference on Networking; 2007. p. 4.

[34] Medepally B, Mehta N. Voluntary energy harvesting relays and selection in cooperative wireless networks. IEEE Trans Wirel Commun 2010;9:3543-53.

[35] Zhang P, Xiao G, Tan HP. Distributed relay scheduling for maximizing lifetime in clustered wireless sensor networks. 2012 IEEE International Conference on Communication Systems (ICCS); 2012. p. 11-5.

[36] Anderson TE, Owicki SS, Saxe JB, Thacker CP. Highspeed switch scheduling for local-area networks. ACM Trans Comput Syst 1993;11:319-52.

[37] McKeown N. The islip scheduling algorithm for input-queued switches. IEEE ACM Trans Netw 1999;7:188-201.

[38] Voigt T, Dunkels A, Alonso J, Ritter H, Schiller J. Solar aware clustering in wireless sensor networks. Vol. 1. Ninth International Symposium on Computers and Communications; 2004. p. 238-43.

[39] Kinoshita K, Okazaki T, Tode H, Murakami K. A Data Gathering Scheme for Environmental Energy-Based Wireless Sensor Networks. 5th IEEE Consumer Communications and Networking Conference; 2008. p. 719-23.

[40] Alippi C, Camplani R, Galperti C, Roveri M. A robust, adaptive, solar-powered WSN framework for aquatic environmental monitoring. IEEE Sens J 2011;11:45-55.

[41] Gou H, Yoo Y, Zeng H. A partition-based leach algorithm for wireless sensor networks. Vol. 2. 9th IEEE International Conference on Computer and Information Technology; 2009. p. 40-5.

[42] Momčilović P. A distributed switch scheduling algorithm. Perf Eval 2007;64:1053-61.

[43] Rappaport TS. Wireless Communications: Principles and Practice. New Jersey: Prentice Hall PTR; 2002.

[44] Hopcroft JE, Karp RM. A n5/2 Algorithm for Maximum Matching in Bipartite. 12th Annual Symposium on Switching and Automata Theory; 1971. p. 122-5.

[45] Mjelde K. Max-min resource allocation. BIT 1983;23:529-37.

[46] Calhoun B, Daly D, Verma N, Finchelstein D, Wentzloff D, Wang A, *et al*. Design considerations for ultra-low energy wireless microsensor nodes. IEEE Trans Comput 2005;54:727-40.