Research Article

A Convergent Algorithm for Energy-Balanced Cluster-Heads Selection in Wireless Sensor Networks

Hsing-Lung Chen, Tai-An Chen, and Shu-Hua Hu

1 Department of Electronic and Computer Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan
2 Department of Computer Science and Information Engineering, Jinwen University of Science and Technology, New Taipei 231, Taiwan

Correspondence should be addressed to Hsing-Lung Chen; hlchen@mail.ntust.edu.tw

Received 4 October 2013; Accepted 8 March 2014; Published 3 April 2014

Academic Editor: Hwa-Young Jeong

Copyright © 2014 Hsing-Lung Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Due to the limited energy of sensor nodes, it is a research goal that the lifetime of sensor networks is prolonged by transmitting the sensed data to the base station in an energy-saving way. Previous algorithms aim at reducing the average energy consumption rate to extend the network lifetime. However, some nodes sometimes may be served as the cluster-head too many times to conserve their energy, resulting in reduced network lifetime. Thus, the large deviation of network lifetime makes these algorithms impractical. This paper proposes a new clustering algorithm which not only reduces the average energy consumption rate, but also converges the residual energies of all nodes on a small interval. Based on the two-region cluster-heads selection mechanism, the coordinator adaptively adjusts the far-near regions to converge the energies of all nodes on a small interval. With the exclusion-circle of cluster-heads, cluster-heads can be distributed evenly in a spatial respect for each round, resulting in reduced energy consumption. The simulation results show that the proposed algorithm not only makes cluster-heads distribute evenly in a spatial respect but also converges the residual energies of all nodes on a small interval, resulting in extending the network lifetime significantly and stably.

1. Introduction

The technological advances in microelectronics, wireless communications, and embedded systems make the development and deployment of wireless sensor networks feasible [1–3]. In 2000, the professors in UCLA proposed a wireless sensor network, called Smart Dust, for military applications. Recently, its applications have been expanded on many fields, such as industrial control, environmental monitoring, medical care, and daily living facilities [4–8].

A lot of sensor nodes are distributed in the sensing field. All the sensor nodes conduct environmental sensing periodically and send the sensed data to the sink [9]. The sink collects and analyzes all the sensed data and then transmits the results to the user terminal. It is hard to recharge the sensor nodes after they are deployed. Hence, it becomes an important research issue that the sensed data are collected in an energy-saving way to prolong the network lifetime [10].

The approaches for data collection can be categorized into a clustering scheme [11] and a multihop scheme [12]. The clustering scheme is an easier and more efficient way to save energy in wireless sensor networks. In each round, cluster-heads are elected and each cluster-head collects the sensed data from its corresponding member nodes and then transfers the collected data to the sink. The energy consumption for data transmission is proportional to the square (or quadruplicate) of the distance between the sender and receiver. It is expected that the elected cluster-heads are uniformly distributed in the sensing area such that all the transmission distances are shorter, resulting in reducing energy consumption. If some particular nodes are elected as cluster-heads too often, these nodes will use up their energy earlier and the network lifetime is reduced significantly. Hence, the energy-balanced cluster-heads selection becomes a key point in designing a wireless sensor network.

Taking the above considerations, this paper proposes the two-region cluster-heads selection mechanism. The sensing area is partitioned into the far and near regions. In the first phase, the selected coordinator picks the node with the highest residual energy in the far region as the unique cluster-head for the far region. In the second phase, all the nodes in
the near region calculate their priorities to compete for several cluster-heads by the nodes’ energy distribution window, which is estimated from the residual energies of cluster-heads of the last round. Finally, the coordinator unifies the cluster-heads by broadcasting to solve the collision problem. With the residual energies of cluster-heads of previous rounds, the coordinator adaptively adjusts the partition-state to make the system stay on two stable states back and forth after a short time period. With the exclusion-circle of cluster-heads, cluster-heads can be distributed evenly in a spatial respect for each round, resulting in reducing the energy consumption.

This paper is organized as follows. Section 2 reviews related clustering approaches for wireless sensor networks. Section 3 gives our proposed cluster-heads selection algorithm EnBaCH. Experimental performance comparison between EnBaCH and LEACH is given in Section 4. Section 5 concludes the paper.

2. Related Works

In LEACH [13], cluster-heads are randomly selected based on the probability by which each node will be elected as a cluster-head once in the fixed number of rounds. Probably, there is no cluster-head elected in one round and each node needs to send the sensed data to the base station by itself, resulting in wasting energy. Sometimes, cluster-heads are unevenly distributed and some member nodes employ larger energies to transmit the sensed data to their corresponding cluster-heads with the larger distance.

In TB-LEACH [14], all the nodes broadcast ADV (advertisement) packets to compete for cluster-heads only if the number of elected cluster-heads is less than the prescribed threshold. Hence, there are a fixed number of cluster-heads in each round. It only solves the problem of a few cluster-heads in one round.

LEACH-B [15] is a two-stage cluster-heads selection approach. The first stage is the same as LEACH except for broadcasting with its own residual energy. If the number of cluster-heads of the first stage is less than the prescribed threshold, in the second stage, the other nodes broadcast ADV packets to compete for cluster-heads according to time priority which is proportional to their own residual energy. It also solves the problem of a few cluster-heads in one round. Cluster-heads elected in the second stage can be with the higher residual energy while cluster-heads elected in the first stage are not.

C-LEACH [16] adjusts the number of cluster members for each cluster within the specified interval by a two-stage method. The first stage is the same as LEACH. It reforms the clusters in the second stage. However, some members may be far away from their corresponding cluster-heads, resulting in wasting energy.

N-LEACH [17] improves LEACH by giving some nodes more chances to compete for cluster-heads. For the nodes having served as a cluster-head in the fixed number of rounds, if the total number of their own member nodes is less than the prescribed number, they are allowed to compete for cluster-heads. It is assured that the number of its member nodes of each node in the fixed number of rounds is no less than the prescribed number. However, energy consumption of each node is not affected only by the number of its corresponding member nodes.

LEACH-IMP [18] partitions the sensing area into the prescribed number of regions in which there are the same number of nodes. In each round, exactly one node near the center of each region is elected as the cluster-head. The member nodes consume less energy because of the shorter transmission distance. However, the nodes near the center of each region consume more energy because of being served as cluster-heads frequently and will use up their energies earlier, resulting in reducing the network lifetime significantly.

EDL [19] enhances LEACH by considering energy and distance. The nodes within the specific distance from the elected cluster-heads are prohibited from competing for cluster-heads such that cluster-heads can be distributed uniformly in the spatial aspects. The nodes with higher residual energy are giving more chances to compete for cluster-heads. Hence, all the nodes are balanced with respect to energy, resulting in extending the network lifetime. However, it causes very high overhead to collect total energies for each round.

3. Energy-Balanced Cluster-Heads Selection Mechanism (EnBaCH)

Because all sensor nodes are distributed in the sensing area randomly, the location of all sensor nodes and distances between the base station and all sensor nodes is unknown in the beginning of the sensor network. Therefore, it is necessary to initialize the system with coordinator selection and set-up of far-near region.

After the system initialization, the system begins collecting sensed data. Each round composes two parts: set-up and data-collection, as shown in Figure 1. The set-up part consists of CH selection, coordinator broadcasting, member joining, and schedule broadcasting. The data-collection part consists of five frames, in which one sensed data from every sensor node is collected to the base station per frame. In each frame, member nodes transmit the sensed data to their corresponding cluster-heads in the scheduled slots while they are sleeping in the other slots for energy-saving. The cluster-heads collect the sensed data from their corresponding member nodes and then send all the collected data to the base station.

With transmitting the same amount of data to the base station, the farther nodes from the base station will consume more energy than the nearer nodes. The sensing area is partitioned into two regions (far and near regions) according to the distance from the base station. In the first phase, exactly one cluster head will be elected in the far region, and then others are elected in the near region, resulting in reducing energy consumption.

For competing for cluster-heads with broadcasting, it is possible to collide with each other. The nodes with collision may perceive themselves as cluster-heads and send their data to the base station solely, resulting in wasting energy. By
choosing a suitable node as the coordinator, it broadcasts the received cluster-heads information to unify the cluster-heads information of all nodes, resulting in solving the collision problem.

In addition to unifying the cluster-heads information, the coordinator must adjust far-near region to achieve the balance of energy consumptions. With the energy information of cluster-heads of previous rounds, the coordinator adaptively adjusts the far-near regions to converge the residual energies of all the nodes to a small interval.

3.1. System Initialization. Initially, the base station broadcasts the request to all sensor nodes and all sensor nodes send their own distances from the base station back to the base station, which are estimated by the received signal strength indication (RSSI). In order to partition the sensing area into far and near regions by the estimated distances, it is necessary that each sensor node has a unique \((\text{distance}, \text{node-id})\) tuple. The operation \(<\) on \((\text{distance}, \text{node-id})\) tuples is defined in the following, which possesses the property of total ordering.

\[
\text{Definition 1.} \quad (\text{dis}_1, \text{id}_1) < (\text{dis}_2, \text{id}_2) \quad \text{if and only if} \quad \text{dis}_1 < \text{dis}_2 \quad \text{or} \quad (\text{dis}_1 = \text{dis}_2 \quad \text{and} \quad \text{id}_1 < \text{id}_2).
\]

With the property of total ordering, the base station sorts all the \((\text{distance}, \text{node-id})\) tuples in decreasing order into Distance-Id List (DIL) and then broadcasts DIL to all sensor nodes. The partition-state of the sensing area can be defined with DIL.

\[
\text{Definition 2.} \quad \text{Let DIL}_k \quad \text{be the kth tuple in DIL. The partition-state} \quad \text{S}_k \quad \text{is defined as that in which the sensor nodes with tuples DIL}_1, \ldots, \text{DIL}_k \quad \text{are in the far region and the other nodes are in the near region.}
\]

From Definition 2, each sensor node can determine its belonging region with DIL and \(S_k\). The adaptive adjustment of partition-state will be described in the following subsection.

The coordinator is responsible for collecting the residual energies of all the nodes in the far region and selecting the node with the highest residual energy as the cluster-head of the far region. The nodes nearby the center of the sensing area are suitable for the candidates of the coordinator. A feasible candidate can reduce energy consumption due to short distances.

Initially, all the sensor nodes calculate their own broadcast waiting time which is in inverse proportion to their own distances from the base station. When each node has run out of its waiting time, it determines whether it broadcasts a Co_ADV packet or not. If a node is within the exclusion-circle of some coordinator candidates or there are three or more coordinator candidates, it will cancel broadcasting; otherwise, it will broadcast a Co_ADV packet, which includes other candidates’ information to claim itself as a coordinator candidate.

Each candidate has three distances among these three candidates through broadcasting messages. Each candidate has two related distances. The candidate with the minimum sum of its related distance should be in the middle of three. In order to let the coordinator be nearby the center of the sensing area, only the middle candidate with less distance (< threshold \(\alpha\)) from the base station can be elected as the coordinator. If the distance between the middle candidate and the base station is greater than threshold \(\alpha\), the middle candidate broadcasts a Co_NAK packet to cancel its candidacy. When other nodes receive the Co_NAK packet, they will remove the middle candidate from the candidate list and then compete for being the coordinator candidate. If the distance between the middle candidate and the base station is not greater than threshold \(\alpha\), the middle candidate broadcasts a Co_ACK packet to assume its coordinator-ship. When other nodes receive Co_ACK message, they will record the middle candidate as the coordinator.

3.2. The Cluster-Head Selection in the Far Region. Because the nodes in the far region consume more energy than the nodes in the near region, only one cluster-head is elected in the far region and others are elected in the near region. In the first phase, the coordinator is expected to select the node in the far region with the highest residual energy as the cluster-head, resulting in the residual energies of all the nodes in the far region being converged on a small interval.

At first, the coordinator collects the quantities of the residual energies of all the nodes in the far region. All the nodes in the far region will broadcast their own quantity of the residual energy to the coordinator at a time slot randomly. The coordinator then selects the node with the highest residual energy and notifies it. The notified node will broadcast a CH_ADV packet to assume its cluster-headship.
Due to collision, it is possible that the notified node is not the node with the highest residual energy.

3.3. The Cluster-Heads Selection in the Near Region. In the second phase, it is expected that the nodes in the near region with the highest residual energy are elected as the cluster-heads, resulting in the residual energies of all the nodes in the near region being converged on a small interval. As described before, in order to achieve the spatial balance, the nodes within the exclusion-circle of cluster-heads are prohibited from being cluster-heads. It is useless that the coordinator collects the quantities of residual energies of all the nodes in the near region. Based on the estimated energy distribution, it allows the nodes with the higher residual energy to compete for being cluster-heads with less collision.

Reference Energy (RE) is defined as the estimation of the highest residual energy in the near region for the current round. Let RE’ be the RE of the last round. Assume that Max E is the highest residual energy of cluster-heads in the near region in the current round. Let δ be an allowable energy deduction. RE can be estimated at the end of cluster-heads selection as follows:

\[
RE = \begin{cases} 
RE' & \text{if } \text{Max } E \geq RE' \\
\text{Max } E & \text{if } RE' > \text{Max } E \geq RE' - \delta \\
RE' - \delta & \text{if } \text{Max } E < RE' - \delta. 
\end{cases}
\]  

Figure 2: The contention window.

From the definition, it is obvious that RE should be equal to Max E. The node with the highest residual energy may not be elected as one of the cluster-heads, because it may collide with other nodes or because it is in the exclusion-circle of the cluster-head in the far region. When Max E ≥ RE’, it means that the node with the highest residual energy in the last round is not elected as one of the cluster-heads or the cluster-head in the current round with the highest residual energy was in the far region in the last round. Then RE should stay the same as RE’; otherwise, the cluster-heads selection may not work well in the next round because the contention window cannot cover most of the nodes with higher residual energy. When Max E is less than RE’ − δ, RE is set to RE’ − δ in order to keep the estimation error small. At the end of cluster-heads selection, the coordinator will broadcast RE to all sensor nodes for cluster-heads selection of the next round.

Recall that the nodes with higher residual energy are expected to be elected as cluster-heads. Let us introduce the concept of contention window. It is expected that the contention window covers most of the nodes with higher residual energy. Hence, let the beginning of contention window (WB) be RE + 0.1 and the width of the contention window be 0.7 Joules. Because the broadcasting interval is limited, the nodes with a residual energy less than RE + 0.6 do not join the competition for cluster-heads. Due to estimation error, maybe there exist some nodes with a residual energy greater than WB, namely, priority group, and it is necessary to give these nodes higher priority, as shown in Figure 2. ADVtime1 time slots are allocated for priority group. The nodes within the contention window are partitioned into usual group and backup group according to the energy threshold WB − 0.3. ADVtime2 and ADVtime3 time slots are allocated for usual group and backup group, respectively. Because usual group must have higher priority, ADVtime2 is set to be two times that of ADVtime3. Thus, the broadcast waiting time of priority group (Wp), usual group (Wu), and backup group (Wb) can be derived as follows, where ER is the current residual energy of the node:

\[
W_p = \text{random}(0, \text{ADVtime1}),
\]
\[
W_u = \frac{(\text{WB} - \text{ER})}{0.3} \ast \text{ADVtime2} + \text{ADVtime1},
\]
\[
W_b = \frac{((\text{WB} - 0.3) - \text{ER})}{0.4} \ast \text{ADVtime3}
\]
\[\ast \text{ADVtime2} + \text{ADVtime1}.
\]

In order to reduce collision, Wu and Wb are increased by a random number (e.g., random(0, 3)).

3.4. Adaptive Adjustment of Far-Near Regions. For transmitting the same amount of data to the base station, the farther nodes from the base station will consume more energy than the nearer nodes. The sensing area is partitioned into two regions (far and near regions) according to the distance from the base station. In the first phase, exactly one cluster head will be elected in the far region, and then others are elected in the near region, resulting in reducing energy consumption. If the partition-state is not in energy-balanced state and one of far-near regions will run out of energy earlier, the network lifetime is reduced significantly. However, the balanced state is affected by many factors and it is very difficult to derive the balanced state in advance. In this paper, we propose a partition-state adjustment scheme which adaptively adjusts partition-state by the energy information of cluster-heads such that average energy consumption per node is almost equal. Thus, it can extend the network lifetime significantly and stably.

Recall that there are k nodes in the far region for partition-state Sf. It is impossible that average energy consumption per node in the far region is equal to that in the near region for any partition-state. Hence, we introduce the
The concept of virtual partition-state. Virtual partition-state $S_v$ is defined as that which consists of $(v - u + 1) \times 100\%$ of $S_u$ and $(u - v) \times 100\%$ of $S_{u-1}$, where $u = \lceil v \rceil$. The balance point is defined as a virtual partition-state $S_v$ in which all energy consumptions per node for all the nodes are almost equal. Two partition-states near the balance point are called stable states. It is expected that after a time period, the system could stay on two stable states back and forth. Thus, the residual energies of all the nodes are converged on a small interval.

Recall that the coordinator selects the node with the highest residual energy in the far region as the cluster-head. One turn is defined as a number of rounds in which every node in the far region has been the cluster-head once. That is, one turn consists of $k$ rounds for partition-state $S_k$. The coordinator determines if the partition-state is adjusted at the end of the current turn according to the energy information of cluster-heads. Assume that $\text{AVG}_n$ and $\text{AVG}_f$ are the average residual energies of cluster-heads of the current turn for the near region and the far region, respectively. Let $\Delta_i = \text{AVG}_n - \text{AVG}_f$. $\Delta_i$ is employed as the estimation of difference of the average residual energies of all the nodes between the far region and the near region. Let $\Delta_r$ be $\Delta_i$ of last turn with different partition-states from the current turn.

The partition-state adjusting scheme consists of two stages: initialization stage and adjusting stage. For the case of the current partition-state being at the left-hand side of the balance point, $\Delta_i$ is increased by a small amount per turn because there are a number of nodes in the far region, compared with the number of nodes at the far region in the balance point, as shown in Figure 3. There is the similarity for the case of the current partition-state being at the right-hand side of the balance point. Initially, all the nodes possess the same energy. Ideally, at the end of the first turn and for the case of the initial partition-state being at the left-hand (or right-hand) side of the balance point, $\Delta_i$ is greater (or less) than zero. Due to estimation error, there may not exist the corresponding $\Delta_i$ value in the first three turns and then the partition-state adjustments of the first three turns are skipped. At the end of the fourth turn, the partition-state is changed rightward (or leftward) if $\Delta_i > 0$ (or $\Delta_i < 0$). Then, the system enters the adjusting stage.

In the adjusting stage, it is not assured that the current partition-state is in the left-hand (or right-hand) side of the balance point if $\Delta_i > 0$ (or $\Delta_i < 0$). The other measurement is needed. Recall that $\Delta_r$ is $\Delta_i$ of last turn with different partition-state from the current turn. If the current partition-state is at the left-hand (or right-hand) side of the balance point, $\Delta_i$ is increasing (or decreasing). Hence, it is assured that the current partition-state is in the left-hand (or right-hand) side of the balance point if $(\Delta_i - \Delta_r) > 0$ (or $(\Delta_i - \Delta_r) < 0$), as shown in Figure 3. Cases 1 and 2 (or Cases 3 and 4) are in the left-hand (or right-hand) side of the balance point. In Case 1, the partition-state must be adjusted rightward to approach the balance point. Similarly, in Case 4, the partition-state must be adjusted leftward to approaching the balance point. For Cases 2 and 3, the partition-state remains unchanged such that $\Delta_i$ approaches zero. In the following, we consider the case of the initial partition-state being in the left-hand side of the balance point. Initially, the system is in Case 1. After a couple of state changings, the system will enter Case 3 and stay on one of stable states to reduce $\Delta_i$. When $\Delta_i$ becomes negative, the system is in Case 4 and the partition-state will be adjusted to one of the other stable states (i.e., Case 2 in the next turn). The system with Case 2 will stay on one of the stable states to increase $\Delta_i$. When $\Delta_i$ becomes positive, the system is in Case 1 and the partition-state will be adjusted to one of the other stable states (i.e., Case 3 in the next turn). From now on, the system stays on two stable states back and forth. There is the similarity for the case of the initial partition-state being in the right-hand side of the balance point.

The partition-state is changed only if the system is in Cases 1 or 4. Due to the estimation error, it is possible that an incorrect state changing is made. To prevent the incorrect state changing, two small amounts of thresholds $\text{TH}_1$ and $\text{TH}_2$ are introduced. The conditions of state changing for Cases 1 and 4 become $\Delta_i > \text{TH}_1$ and $(\Delta_i - \Delta_r) > \text{TH}_2$ and $\Delta_i < -\text{TH}_1$ and $(\Delta_i - \Delta_r) < -\text{TH}_2$, respectively.

For Cases 2 and 3, the system stays on one of two stable states to let $\Delta_i$ approach zero. However, the difference of $\Delta_i$’s between two consecutive turns for stable states is very small. The time for $\Delta_i$ reaching around zero may be very long. Hence, we introduce the counter threshold. Initially, the counter threshold is set to 3. When the system has stayed on the same state for more turns than the counter threshold, its partition-state must be adjusted according to $\Delta_i$. For Case 3 (or Case 2), the partition-state must be adjusted rightward (or leftward) to decrease (or increase) $\Delta_i$ with a larger amount. As a consequence, the time for $\Delta_i$ reaching around zero could be reduced significantly. If the direction of adjustment of the partition-state has been changed twice, $\Delta_i$ will reach around zero for partition-states nearby the balance point and the counter threshold can be set to 9 to let the system stay on two stable states back and forth.

### 4. Simulations and Results

The simulations are conducted with NS2 [20]. There are 100 sensor nodes randomly distributed in the 100 m $\times$ 100 m area. The base station is located at the top of the sensing area with the distance of 75 m, as shown in Figure 4. The base station and all the sensor nodes are assumed to be fixed. All the sensor nodes are not equipped with the devices for detecting their own locations and their own location may be estimated with some localization algorithms. All the sensor nodes are homogeneous with limited energy.
The simple first-order radio model is adopted to calculate the energy consumption for transmitting and receiving data packets. Let $E_{\text{elec}}$, $\varepsilon_{fs}$, and $\varepsilon_{mp}$ denote the energy consumption rates for operating the electronics in a radio transceiver and transmitter amplifier of near-far ranges, respectively. For receiving a $k$-bit packet, a sensor node consumes $E_{\text{Rx}}(k)$ Joule of energy or $E_{\text{Rx}}(k) = kE_{\text{elec}}$. For transmitting a $k$-bit packet to another node over a distance of $d$ meters, the energy consumption is as follows

$$E_{\text{Tx}}(k, d) = \begin{cases} kE_{\text{elec}} + k\varepsilon_{fs}d^2, & d < d_0 \\ kE_{\text{elec}} + k\varepsilon_{mp}d^4, & d \geq d_0. \end{cases}$$ (3)

In the simulations of this paper, let $E_{\text{elec}} = 50$ nJ/bit, $\varepsilon_{fs} = 10$ pJ/bit/m$^2$, $\varepsilon_{mp} = 0.0013$ pJ/bit/m$^4$, and $d_0 = 250$ m.

The simulations with initial partition-states from $S_{25}$ to $S_{40}$ are conducted in the 100 m × 100 m sensing area. Due to the limitation of paper size, only the state transition diagrams for $S_{25}$, $S_{30}$, $S_{35}$, and $S_{40}$ are shown in Figures 5, 6, 7, and 8. In this scenario, two stable states are $S_{32}$ and $S_{33}$, indicated by red solid lines.

For the cases of initial partition-state being far away from stable states, the system stays on two stable states back and forth after turns 45 and 60 for initial states $S_{25}$ and $S_{40}$, respectively. For the cases of initial partition-state being nearby stable states, the system stays on two stable states back and forth after turns 22 and 24 for initial states $S_{30}$.
Table 1: The network lifetime for LEACH and EnBaCH.

|                  | LEACH   | EnBaCH   |
|------------------|---------|----------|
| Average network lifetime (s) | 53538   | 79986    |
| Average energy consumption per round (J) | 2.77985869 | 1.9937212 |
| Average residual energy (J) | 563.291104 | 26.45153797 |
| Network lifetime for the worst case | 51380 | 79820 |

![Figure 9: Residual energy distribution for all the nodes at time 50000 second.](image)

Consider the average case in the following. From Table 1, because the energy utility of EnBaCH is 99.67% and the energy utility of LEACH is 92.96%, EnBaCH gets 7.22% improvements in the energy utility over LEACH. For average energy consumption per round, EnBaCH obtains 39.43% improvements over LEACH. Totally, EnBaCH obtains 49.40% improvements in the network lifetime over LEACH. In the worst case, EnBaCH obtains 55.35% improvements in the network lifetime over LEACH. Table 3 shows that EnBaCH obtains much more improvements in the network lifetime than other cluster-head selection algorithms.

5. Conclusions

This paper proposes a two-region cluster-heads selection mechanism. The sensing area is partitioned into the far and near regions. In the first phase, the selected coordinator picks the node with the highest energy in the far region as the unique cluster-head of the far region. In the second phase, all the nodes in the near region calculate their priorities to compete for several cluster-heads by the estimation band of nodes’ energies. Finally, the coordinator unifies the cluster-heads by broadcasting to solve the collision problem. With the energy information of cluster-heads of previous rounds, the coordinator adaptively adjusts the partition-state to make the system stay on two stable states back and forth after a short time period. With the exclusion-circle of cluster-heads, cluster-heads can be distributed evenly in spatial respect for each round, resulting in reducing the energy

Table 2: The standard deviation for LEACH and EnBaCH.

|                  | LEACH   | EnBaCH   |
|------------------|---------|----------|
| SD for network lifetime | 1434.746 | 89.14408 |
| SD for average energy consumption per round | 0.082289 | 0.002001 |
| SD for residual energy | 60.62555 | 2.330875 |

Table 3: The performance comparisons for cluster-head selection algorithms.

| Algorithms       | The improvements over LEACH |
|------------------|-----------------------------|
| N-LEACH          | 12%                         |
| TB-LEACH         | 15%                         |
| LEACH-IMP        | 21%                         |
| LEACH-B          | 25%                         |
| Low-energy [21]  | 30%                         |
| EnBaCH           | 49.40%                      |
consumption. The simulation results show that EnBaCH not only distributes cluster-heads evenly in spatial respect but also converges the energies of all the nodes on a small interval, resulting in extending the network lifetime significantly and stably.

The simulation results show that EnBaCH obtains 39.43% improvements in the average energy consumption per round over LEACH, because of cluster-heads unification by the coordinator and even distribution of cluster-heads in space. EnBaCH also gets 72.22% improvements in the energy utility over LEACH by adaptive partition-state adjustment. Totally, EnBaCH obtains 49.40% improvements in the network lifetime over LEACH.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was supported in part by the National Science Council, Republic of China, under Grant no. NSC 102-2221-E-011-067.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” Computer Networks, vol. 38, no. 4, pp. 393–422, 2002.
[2] H.-R. Lee, K.-Y. Chung, and K.-S. Jhang, “A study of wireless sensor network routing protocols for maintenance access hatch condition surveillance,” Journal of Information Processing Systems, vol. 9, pp. 237–246, 2013.
[3] R. Sumathi and M. G. Srinivas, “A survey of QoS based routing protocols for wireless sensor networks,” Journal of Information Processing Systems, vol. 8, pp. 589–602, 2012.
[4] H. Viswanathan, B. Chen, and D. Pompili, “Research challenges in computation, communication, and context awareness for ubiquitous healthcare,” IEEE Communications Magazine, vol. 50, pp. 92–99, 2012.
[5] S. Dhurandher, M. Obaidat, and M. Gupta, “An acoustic communication based AQUA-GLOMO simulator for under-water networks,” Human-Centric Computing and Information Sciences, vol. 2, article 3, 2012.
[6] Q.-L. Liu and D.-H. Oh, “Performance evaluation of multi-hop communication based on a mobile multirobot system in a subterranean laneway,” Journal of Information Processing Systems, vol. 8, pp. 471–482, 2012.
[7] J. K. Y. Ng, “Ubiquitous healthcare: healthcare systems and applications enabled by mobile and wireless technologies,” Journal of Convergence, vol. 3, pp. 15–20, 2012.
[8] T. Teraoka, “Organization and exploration of heterogeneous personal data collected in daily life,” Human-Centric Computing and Information Sciences, vol. 2, article 1, 2012.
[9] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, “Wireless sensor networks for habitat monitoring,” in Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications, pp. 88–97, Atlanta, Ga, USA, September 2002.
[10] G. H. S. Carvalho, I. Younang, A. Anpalagan, and S. K. Dhurandher, “Energy-efficient radio resource management scheme for heterogeneous wireless networks: a queueing theory perspective,” Journal of Convergence, vol. 3, pp. 15–22, 2012.
[11] B. Singh and D. Lobiyal, “A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks,” Human-centric Computing and Information Sciences, vol. 2, article 13, 2012.
[12] X. Li, N. Mitton, A. Nayak, and I. Stoimenovic, “Achieving load awareness in position-based wireless Ad Hoc routing,” Journal of Convergence, vol. 3, pp. 17–22, 2012.
[13] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” in Proceedings of the 33rd Annual Hawaii International Conference on System Sciences (HICSS ’00), pp. 1–10, Maui, Hawaii, USA, January 2000.
[14] H. Junping, J. Yuhui, and D. Liang, “A time-based cluster-head selection algorithm for LEACH,” in Proceedings of the 13th IEEE Symposium on Computers and Communications (ISCC ’08), pp. 1172–1176, Marrakech, Morocco, July 2008.
[15] T. Mu and M. Tang, ”LEACH-B: an improved LEACH protocol for wireless sensor network,” in Proceedings of the 6th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM ’10), pp. 1–4, Chengdu, China, September 2010.
[16] R. Mehta, A. Pandey, and P. Kapadia, “Reforming clusters using C-LEACH in wireless sensor networks,” in Proceedings of the International Conference on Computer Communication and Informatics (ICCCI ’12), pp. 1–4, Coimbatore, India, January 2012.
[17] R. K. Tripathi, Y. N. Singh, and N. K. Verma, “NLEACH, a balanced cost cluster-heads selection algorithm for wireless sensor network,” in Proceedings of the 18th National Conference on Communications (NCC ’12), pp. 1-5, Kharagpur, India, 2012.
[18] X. Hu, J. Luo, Z. Xia, and M. Hu, “Adaptive algorithm of cluster head in wireless sensor network based on LEACH,” in Proceedings of the IEEE 3rd International Conference on Communication Software and Networks (ICCSN ’11), pp. 14–18, Xi’an, China, May 2011.
[19] R. Hou, W. Ren, and Y. Zhang, “A wireless sensor network clustering algorithm based on energy and distance,” in Proceedings of the 2nd International Workshop on Computer Science and Engineering (WCSE ’09), pp. 439–442, Qingdao, China, October 2009.
[20] K. Fall and K. Vardhan, “The Network Simulator (ns-2),” http://www.isi.edu/nsnam/ns.
[21] M. J. Handy, M. Haase, and D. Timmermann, “Low energy adaptive clustering hierarchy with deterministic cluster-head selection,” in Proceedings of the 4th International Workshop on Mobile and Wireless Communications Network, pp. 368–372, Stockholm, Sweden, 2002.