Energy-balanced Clustering Routing Protocol Based on Task Separation in Wireless Sensor Networks

Wei Sun[1,a], Xiaoying Song[2], Fasheng Wang[1]

1. Liaoning Key Lab of Network Security and Computing Technology
Dalian Neusoft University of Information
Dalian 116023 China
2. Software Center
Northeastern University
Shenyang, 110003 China.
{sunwei,songxiaoying,wangfasheng}@neusoft.edu.cn

Abstract: Clustering protocol for data gathering is one of the effective ways to solve energy hole problem in wireless sensor networks. However, most of the existing clustering protocols focus on the network model with uniform node distribution. They do not effectively apply to the real network where the sensor nodes are randomly non-uniformly deployed. In this paper, we propose an energy-balanced clustering routing protocol (EBCRP) based on task separation. In this scheme, the network is firstly divided into clusters by using global information. And each of them has the same number of sensor nodes in order to balance the energy consumption of intra-cluster. In succession, task separation, the tasks of traditional single cluster head are separated and achieved by two cluster heads respectively, is proposed to reduce the traffic burden for single cluster head. Then, we explore an energy-efficient and reliable inter-cluster routing algorithm, which considers comprehensively three factors: residual energy, distance and available buffer space of nodes. Simulation results and performance evaluation of EBCRP show significant improvement in network lifetime and energy balance.

Key-words: Wireless Sensor Networks; Balancing Energy Consumption; Task Separation; Clustering Routing Protocol; Random Deployment

1 Introduction

Wireless sensor networks have become a hot research topic in recent years due to their potential wide use in applications such as target tracking, biomedical health monitoring [1]. Sensor nodes are responsible for obtaining environmental information and sending it towards Sink through the relays of neighbor nodes. When the data traffic follows a many-to-one communication pattern, the nodes near the hotspot are burdened with heavier relay traffic and trend to die early. This phenomenon is called “energy hole” problem [2][3][4].

The clustering scheme for data gathering protocol is one of the effective ways to solve energy hole problem [5][6]. In the process of clustering for data gathering protocol, the determination of cluster heads plays an important role for the data traffic burdened and energy consumption [7][8]. Clustering schemes proposed can be grouped in two categories concerning the sequence of determining cluster head. One is determining the cluster head first before other nodes select the nearest cluster head to form a unit of clustering network [9][10][11]. The other is clustering the network first and then selecting cluster head for each cluster [12][13].

Concerning the first data gathering protocol, the member nodes join in the nearest cluster head after determining cluster head. LEACH [9] has been firstly proposed with clustering to gather data in the whole network. It selects the cluster head based on the rotation mode according to the same probability as which a cluster head is pre-determined. After gathering data from member nodes, the cluster heads transmit data to Sink directly. Heinzelman et al. in [10] propose a centralized clustering protocol called LEACH-C, which reduces the nodes energy consumption due to the reduction on control information. Chen et al. in [6] explore an Unequal Cluster-based Routing (UCR) protocol that groups the nodes into clusters of unequal sizes. Clusters near the region of Sink have smaller sizes than those far from Sink. Wang et al. in [11] present LEACH-SWDN, which sets up a sliding window to adjust the electing probability of cluster heads and keeps stable the expected number of the cluster heads. However, Chen et al. point out in [14] that the determining cluster heads first before clustering network cannot easily balance energy consumption of the network.

For the other clustering protocols, which clustering is firstly carried out and then cluster heads are selected, have been proposed in recent years. The protocols [12][13][15][16][17] usually adopt global information to fix a cluster shape. Khalil and Attea [15] prove that clustering according to global information is more reasonable. Lai et al. in [12] presents a cluster-based routing protocol called arranging cluster sizes and transmission ranges for wireless sensor networks (ACT). The protocol uses global information to calculate each cluster radius based on the relaying load of CH by Sink and cross-level transmissions to prolong network lifetime. Fu et al. in [13] propose an energy-balanced separating algorithm for cluster-based data aggregation (SCA) in wireless sensor networks. SCA firstly arranges cluster sizes based on the equal inter-cluster energy consumption. Then, it designs the intra-cluster communication algorithm from the task separation perspective. However, ACT and SCA protocols present better performance than other protocols of determining cluster head first, the assumption of two protocols fits for the situation of nodes uniform distribution so that they cannot be suitable for the real situation with nodes non-uniform distribution. Xu et al. in [16] explore Geographic the Adaptive Fidelity (GAF) protocol, the sensing area is divided into several fixed regions by virtual square grid. Salzmann et al. in [17] compare network performance among cluster shapes and find that virtual regular hexagon grid is the best one on energy consumption and network connectivity. Although the assumption of nodes non-uniform distribution is considered by these two protocols, the optimal number of clusters cannot be obtained. Accordingly, the protocols for firstly clustering have been received increasing attention.

After gathering data for their clusters, cluster heads relays data from their cluster members to Sink. As each cluster head needs to transfer large amount of data, we should aware about the data delivery from cluster heads to Sink. The cluster heads of LEACH [9] transmits data to Sink directly. However, the direct transmission from cluster head to Sink consumes higher energy since energy consumption of sensor nodes is exponentially related to the distance. Multi-hops transmission in [18][19] has longer network lifetime than direct communication [20]. In [19], the author defines hotspot and proposes a solution to address this issue through a hybrid approach that combines two routing strategies. In the hotspot, flat multi-hop routing aims to minimize the total power consumption in the network. However, the authors do not suggest any solution for performing energy-efficient and reliable data delivery in the hotspot area. Anisi et al. in [18] redefines the hotspot that is the whole area from the farthest cluster head to Sink. Furthermore, they explore an energy- efficient and reliable routing approach in this area. However, the proposed network is a heterogeneous network model and they still adopt one cluster head to gathering data in the cluster.

In the case of nodes non-uniform distribution, we make an analysis of energy balancing problem, which includes two sub-problems, both intra-cluster and inter-cluster. For the former sub-problem, we propose a task separation algorithm of cluster head based on broadcasting time. For the latter sub-problem, we try to make every cluster to approximately consume the same amount of energy by adjusting the number of nodes for each cluster. Moreover, we provide an energy-efficient and reliable routing algorithm to make energy consumption balance further. Although many literatures about dividing the network into clusters deal with the problem of unbalanced energy consumption in WSNs, none of the existing algorithms consider separating the tasks of cluster heads to two nodes for intra-cluster energy balance in nodes non-uniform distribution. The main contributions of this paper are summarized.

The rest of the paper is organized as follows. After introducing network model in Section 2, we propose the intra-cluster balance of EBCRP in Section 3. In this section, we introduce how to calculate the optimal number of cluster in detail and propose an adjusting algorithm that makes every cluster with the same number of cluster node. Section 4 explores the inter-cluster balance of EBCRP, an energy-efficient and reliable routing algorithm between clusters. We analyze the performance of EBCRP in Section 5. Section 6 presents the four performances of EBCRP and we conclude the paper in Section 7.

2 Network model

In this paper, we consider N sensor nodes randomly deployed in $R \times R$ square network. The sensor nodes have the same initial energy and processing capacity. There are three kinds of sensor nodes which are master cluster heads, slave cluster heads and common nodes. The responsibility of master cluster heads is receiving and fusing data, and that of slave cluster heads is transmitting data. Common nodes transmit data to master cluster head by single hop. Slave cluster heads relay data to other master cluster head by multi-hops.

In addition, we make some assumptions about the sensor nodes and the underlying network model:

(1) After deploying sensor nodes and Sink, the positions of them are fixed.
(2) There is one Sink that locates in the center of the circular sensing field.
(3) Communication power of all sensor nodes is adjustable, i.e., sensor nodes can select adaptive transmission power to send data in terms of the distance between two nodes, and the biggest transmission power of nodes can send data to Sink.
(4) Sensor nodes can recognize their geographical position and Sink's position by exchanging information.
(5) Every sensor node is able to compute its residual energy and its available buffer size.
(6) The background is period data gathering, i.e., sensor nodes gather data and send to Sink periodically.

3 Intra-cluster energy balancing

Similar to [6], a certain period of time is defined as a round. Every round has two phases: set-up phase and steady-state phase. In set-up phase, there are two sub-phases: clustering and master, slave cluster head selection. Data are transmitted from nodes to Sink in steady-state phase.

3.1 Clustering

In clustering phase, every node sends a message *N_Msg* (*ID*, Energy, $L(x, y)$) to Sink. This message includes the node ID, energy message and the position message of the node. Firstly, Sink calculates the optimal number of clusters, according to the message of every node in this network model, so that the number of nodes can be obtained in every cluster. Furthermore, the whole $R\times R$ network will be divided many clusters by the virtual circle with same radius. At last, we need design an adjustable algorithm keeping the optimal number of cluster and that of nodes in every cluster, for non-uniform node deployment makes energy imbalanced in every cluster.

3.1.1 Optimal number of clusters

We adopt the energy model in [6].We assume that the distance of intra-cluster communication does not exceed the distance threshold of energy model so that free space model will be used by calculating energy consumption. The energy consumption of node *i* in one cluster is

$$E_{i-node} = l(E_{elec} + \varepsilon_{fs} d_{itoch}^2) \tag{1}$$

In the process of clustering, the main energy consumption of Master cluster head is receiving data from this and other clusters, meanwhile, fusing the receving data. And the power consumption of Slave cluster head is transmitting the fused data. For simplicity, suppose that there is one Master cluster head, one Slave cluster head and (m-2) common nodes in one cluster, the energy consumption of Master cluster head E_{M_ch} is

$$E_{M\_ch} = E_r(l) + E_f(m,l) = mlE_{elc} + mlE_{fusion} \tag{2-1}$$

$$E_{S\_ch} = E_r(l) + E_t(l) = 2mlE_{elc} + l\varepsilon_{fs} d_{toNch}^2 \tag{2-2}$$

where, d^2_{toNch} is the square of distance between cluster head to next cluster head. According to formula (1) and (2), the total energy consumption of one cluster $E_{cluster}$ is

$$\begin{aligned} E_{cluster} &= (m-2)E_{i-node} + E_{M\_ch} + E_{S\_ch} \\ &= 2l(2m-1)E_{elec} + (m-2)l\varepsilon_{fs} \sum_{i=1}^{m-2} d_{itoch}^2 \\ &\quad + mlE_{fusion} + l\varepsilon_{fs} d_{toNch}^2 \end{aligned} \tag{3}$$

As shown in formula (3), the energy consumption of one cluster is relative to the number of common nodes of every cluster. Accordingly,the number of nodes of every cluster must be same, which balances the energy consumption of every cluster.

Assuming that there are k clusters in network, the numer of common nodes in one cluster is $m=N/k$ and the average area of one cluster is $S=R^2/k$. As we know, the radius of one cluster is $R/\sqrt{\pi k}$, the average square of the distance from a common node to its cluster head can be obtained

$$E[d_{itoch}^2] = \int_0^{2\pi} \int_0^{\frac{R}{\sqrt{\pi k}}} \rho \cdot r \cdot r^2 dr d\theta = \frac{\rho R^4}{2\pi k^2} \tag{4}$$

where, ρ is the probability of nodes distribution. According to formula (3), the total energy consumption of the whole network E_{sum} is

$$\begin{aligned} E_{sum} &= kE_{cluster} \\ &= 2NlE_{elec} + lNE_{fusion} + l\varepsilon_{fs}[(N-k)\frac{\rho R^4}{2\pi k^2} + kd_{toNch}^2] \end{aligned} \tag{5}$$

For minimizing the total energy consumption of the network, the extreme value of k should be calculated so that the optimal number of cluster can be obtained. Then, the number of nodes of one cluster is calculated according to the formula $m=N/k_{opt}$. It is worthy to note that we obtain a cubic equation of one unknown about variable k, and its solution can be obtained according to the formula about cubic equation of one unknown.

3.1.2 Adjusting for clustering algorithm

Firstly, Sink divides the whole network via the virtual circle with the same radius. From above subsection, the radius of same virtual circle is $r = R/\sqrt{\pi k_{opt}}$. Due to non-uniform node deployment, there are more or less nodes in one cluster so that the energy consumption of whole network does not balance. Moreover, as known from above subsection, the number of clusters k divided by virtual circle is bigger than that obtained k_{opt} by mathematical calculation. What's more, the number of dead nodes increases with more and more working rounds. For keeping the optimal number of clusters, we need design an adjusting clustering algorithm to resolve this problem.

Suppose the number of alive nodes is M_{alive}, meanwhile, $M_{max} = \lceil M_{alive}/k_{opt} \rceil$ and $M_{min} = \lfloor M_{alive}/k_{opt} \rfloor$, and every initial cluster has an ID. The algorithm has three phases: initialization phase, determining the optimal number of clusters phase and determining the member of every cluster phase. Where, there are three key parameters of every cluster: *Ifdelete*, *Iftogethe*, *Ifstayed*. *Ifdelete* demonstrates whether the initial cluster exists. *Iftogether* denotes whether there are new members participating in this cluster. *Ifstayed* represents whether all members of this cluster are fixed.

After the process of clustering, every cluster will select master and slave cluster head. Firstly, every node broadcasts INI_NEI_MSG message, with fixed transmission radius r_0, which includes the node ID, the current residual energy and the position coordinateof node. The node receiving this message is the neighbor node of this node in the transmission range r_0 and updates the new table of neighbor node message. Furthermore, the average residual energy of neighbor nodes $\overline{E}_{Ni}$ can be obtained by every candidate node with formula (6).

$$\overline{E}_{Ni} = \sum_{i=1}^{m} \frac{E_{rem}}{m} \tag{6}$$

Moreover, every candidate node broadcasts a competition massage COM_HEAD_MSG, with the transmission radius r_{ID} of control message, which includes every candidate node ID, the residual energy and competition range r_{ID}. After receiving this message, the candidate node with the same ID updates the table of candidate neighbor node. The broadcasting radius of control message r_{ID} ensures that candidate node can receive the message from neighbor candidate node. Common node will not sleep, during the process of the completion of two cluster heads, until ending the algorithm of two cluster heads selection, which can save the much energy.

Algorithm 1 Adjusting Algorithm

Begin
//Initialization phase
According to the number of initial cluster member, rank all clusters in the decreasing order of nodes number, and save the first k_{opt} initial cluster, set *Ifdelete*=0, *Ifdelete*=1 in other initial cluster. Meanwhile, set *Ifstayed*=0 for every cluster. If $m>M_{max}$, set *Iftogether*=0, else *Iftogether*=1.
//Determining the optimal number of clusters phase
if (*Iftogether*=1)
Set the parameter m of this cluster = 0 and *Iftogether*= 0. Every node of this cluster joins into the nearest cluster that *Ifdelete* of the cluster is 0.
//Determining the member of every cluster phase
for (the every cluster with*Ifstayed*=0)
if ($m<M_{min}$)
In the cluster j which has the nearest distance to cluster i with *Ifstayed*=0, find a node n_j which has the nearest distance to the centre of cluster i, joins the node n_j into this cluster i. If *Ifstayed* of neighbor clusters is not 0, find the node n_j which has the nearest distance to the centre of neighbor cluster i in the neighbor cluster j with *Ifstayed* = 1 and $m = M_{max}$. n_j joins into cluster i until $m = M_{min}$ and set *Ifstayed* = 1. Update the two parameters of m and *Iftogether*.
else if ($M_{min} \le m \le M_{max}$)
set *Ifstayed* of this cluster = 1
else
For every node n_i in the cluster, records d_{ij} between n_i to the centre of cluster j with *Iftogether*=1, the node with smallest d_{ij} joins to cluster j. If there is not one cluster with *Iftogether*=1, the node with the smallest d_{ij} joins to cluster j in the neighbor cluster with *Ifstayed*, until $m = M_{max}$ and set *Ifstayed* = 1. Update the number of nodes m and parameter *Iftogether* of the two clusters.
end

Finally, after calucating the final broadcasting time CM and CS of two cluster head according to formula (7) and (8), candidate node i broadcasts message FIN_MASHEAD_MSG or FIN_SLAHEAD_MSG with transmission radius r_{ID} inorder to notice who is master or slave cluster head. Other candidate node receiving this message will cancel the competition of master and slave cluster head. In [12], the competition of candidate node adopts message negotiation mechanism, i.e., the candidate node with the most residual energy broadcasts message to notice all neighbor candidate node. The neighbor candidate node cancels the competition of cluster head and notices other nodes, after it receives the message of final cluster head.

$$CM_i = T_C \times \frac{\overline{E}_{Ni}}{RE_i} \times \frac{\sum_{i=1}^{m} d_{ij}^2}{m r_{ID}^2} \tag{7}$$

$$CS_i = T_C \times \frac{\overline{E}_{Ni}}{RE_i} \times \frac{d_{(i,Sink)}}{d_{(i,Sink)} + r_{ID}} \tag{8}$$

where, T_c is the predefined time of forming clusters. RE_i is the residual energy of node *i*. $\sum_{i=1}^{m} d_{ij}$ denotes the sum of distance square between candidate node*i* and other member nodes. $d_{(i, Sink)}$ represents the distance between candidate node *i* and Sink.

According to formula (7) and (8), if the residual energy of node *i* is less than the average residual energy of neighbor nodes, candidate node *i* gives up the competition of master and slave cluster head. The broadcasting time of master cluster head is relative to the residual energy of node *i* and the sum of the distance square between node *i* and other common nodes.

4 Inter-cluster energy balancing

EBCRP protocol adopts single hop in intra-cluster and multi-hops in inter-cluster. In steady-state phase, cluster member node firstly transmits data to master cluster head of this cluster, master cluster head fuses data from itself and all member nodes except for slave cluster head. Then, transmitting data fused to slave cluster head of this cluster. After fusing data from itself and master cluster head, slave cluster head transmits data fused to the master cluster head of other clusters. Finally, data can be transmitted to Sink by multi-hops relays.

4.1 Multi-hops routing construction of inter-cluster

Multi-hops routing of EBCRP protocol is distributed. The network model, which is one Sink and many sensor nodes, appears energy hole near Sink when cluster heads transmit data by multi-hops relays. Accordingly, the design goal of multi-hop routing is to establish an energy efficient and reliable routing path, which can reduce energy consumption of transmitting data from slave and master cluster head. In addition, there can be enough data buffering space to avoid data loss when two cluster heads receive data. The routing algorithm can efficiently alleviate energy hole and prolong the network lifetime.

There are two parts during multi-hops routing construction of EBCRP**:** slave cluster head transmits data to Sink directly or relays data through multi-hops to Sink. Transmission distance is the most important factor of energy consumption and Sink has enough buffering space. Therefore, slave cluster head compares the distance between itself and Sink to that of master cluster head, which is the next step of this slave cluster head, to Sink, which chooses the shorter distance to relay data.

The specific implement process is shown in following. Every slave cluster head C_{Si} (i=1…K, K is the number of slave cluster head) broadcasts a message CLUSTER_HEAD_MSG, with the radius that covers ξ times master cluster head, which includes slave cluster head ID, residual energy, available buffering space and the distance to Sink. The master cluster head receiving this message C_{Mj} judges whether the distance to Sink exceeds the thresholdd_0.If $d_{(Mj,BS)} > d_0$, the master cluster head must adopt multi-hops to transmit data. Slave cluster head C_{Si} records the distance to master cluster head and establishes the table of neighbor cluster head. The routing cost from salve C_{Si} to master C_{Mj} cluster head can be calculated by this table, which includes five factors: ID of master cluster head C_{Mj}, the residual energy of C_{Mj}, the available buffering space of C_{Mj}, the distance between C_{Mj} and C_{Si}, the distance between C_{Mj} and Sink.

If satisfying $\{C_{Si}|\ d(C_{Si},\text{Sink}) < d_0 \&\& d^2(C_{Si}, \text{Sink}) < d^2(C_{Mj}, \text{Sink})\}$, slave cluster head C_{Si} transmits data to Sink directly. If satisfying $\{C_{Si}|\ d^2(C_{Si},C_{Mj}) + d^2(C_{Mj}, \text{Sink}) < d^2(C_{Si},\text{Sink})\}$, slave cluster head C_{Si} relays data to Sink by multi-hops. Because slave cluster head C_{Si} has not only one neighbor cluster head, slave cluster head C_{Si} selects the master cluster head with smallest routing cost R_j as the next step node in the set of neighbor nodes φ_{si} when data is relayed by multi-hops, i.e., $R_j = \underset{j \in \varphi_{si}}{argmin}(cost(i, j))$. The routing cost function of inter-cluster is shown as follows.

$$cost(i,j) = \alpha(1 - \frac{E_{rem}(j)}{E_{ini}(j)}) + \beta(1 - \frac{B_{spa}(j)}{B_{total}(j)}) + \gamma(\frac{d_{(i,j)}^2 + d_{(j,BS)}^2}{d_{(i,BS)}^2}) \tag{9}$$

where $E_{rem}(j)$ is the residual energy of master cluster head j. $E_{ini}(j)$ is the initial energy of $E_{ini}(j)$. $B_{spa}(j)$ denotes the available buffering space of master cluster head j.$B_{total}(j)$ represents the total buffering space of master cluster head j. $d_{(i,j)}$ is the distance between slave and master cluster head. $d_{(j,BS)}$is the distance between master cluster head j and Sink. $d_{(i,BS)}$ denotes the distance between slave cluster head i and Sink. α, β and γ are the weighting factors and the sum of them is one.

Slave cluster head transmits data to Sink directly, when the next relay node is itself. However, when the next relay node is not itself, the first item of cost function means to select the node with more residual energy as the next step node. The more residual energy is, the less the value of cost function is at this time. The energy of nodes in WSNs is limited, regardless of transmitting data, receiving data or cluster head selecting, which consumes much more energy. Hence, selecting more residual energy as the next step node is one of the more important factors.

The second item of cost function selects the node with the more available space as the next step relay node. When the receiving data exceeds its transmitting data, the redundant data will be saved. However, when the limited buffering space is full, data will be blocked so that the data that will be received has to be lost. Therefore, the routing cost must consider the current available buffering space of neighbor cluster head in order to provide reliable data transmission. The greater the available buffer space is, the more approximate the value of the second item is to zero. At this time, the value of cost function is less.

The third item of cost function selects the node with the best position as the next step relay node. The energy consumption is relative to the distance of nodes. Selecting the node with the best position can save much energy in order to prolong the network lifetime.

EBCRP protocol includes four parts: clustering, selecting master and slave cluster head, the routing path construction of inter-cluster and steady-state transmission.

4.2 Clusters maintenance

Regardless of master or slave cluster head may be exhausted quickly due to the much larger loads imposed on them. It is significant that master and slave cluster head have to re-select so that the energy consumption will balance for intra-cluster. In EBCRP, the cluster maintenance phase consists of three parts: update clustering, cluster head rotations and slave cluster head transmission to Sink directly.

(1) Update clustering. In [6], Sink notices the nodes to update clustering at a fixed time interval so that the nodes will be clustered again periodically. However, this update clustering will improve message cost and consume the limited energy. We adopt a threshold of alive nodes to judge whether re-clustering the network or not. If the number of alive nodes is less than $N/m_{opt}\times 15\%$, the network will be re-clustered. The reason is that the number of alive nodes in every cluster is not equal with the change of the number of alive node in the network. The network need be re-clustered to have same alive node in every cluster for the energy consumption balancing of intra-cluster.

(2) Cluster head rotations. Regardless of master or slave cluster head, they need re-select them when their residual energy is less than 20% of their initial energy. The new cluster head is the one with the nearest distance and broadcasts message CHANGE_HEAD_MSG with transmission radius r_{ID} to notice other member node.

(3) Slave cluster head transmission to Sink directly. The cluster head near Sink relays more data from other cluster head. Hence, the energy in this area exhausts so quickly. There is not suitable relay node in the table of neighbor cluster head, when the node far away from Sink transmits data to Sink by multi-hop. At this time, this cluster head can send data to Sink directly in order to prolong the network lifetime.

5 Theory analysis

Lemma 1. In clustering phase, the message complexity of EBCRP protocol is $O(N)$, where, N is the number of nodes in the network.

Proof: After Sink sending the initial message to every node, N nodes firstly transmit the message to calculate the optimal number of clusters and determine the member nodes in every cluster. Furthermore, N nodes send the message INI_NEI_MSG to establish the table of neighbor node informiation. Moreover, assuming the probability of becoming candidate cluster head is K, there are KN candidate cluster head send COM_HEAD_MSG to establish the table of neighbor node information. Due to the optimal number of clusters k_{opt} in the network, k candidate cluster heads send the message FIN_MASHEAD_MSG to become the final mastre cluster head. Otherwise, k candidate cluster heads send the message FIN_SLAHEAD_MSG to become the final slave cluster head. At last, master cluster head send k messages MASTER_ADV_MSG to change the state of nodes from sleeping to working. Based on the above analysis, the total message complexity is $N + N + KN + k + k + k = (2+K)N + 3k$. Accordingly, the message complexity of EBCRP protocol is $O(N)$.

□

Lemma 2. Supposing the residual energy of slave cluster head C_{si} is $E_{rem}(i)$, the distance between C_{si} and Sink is $d(i, \text{Sink})$, there are k candidate cluster head in the table of neighbor cluster head, the residual energy of any master cluster head C_{mj} is $E_{rem}(j)$. Meanwhile, the available buffering space of master cluster head C_{mj} is $B_{spa}(j)$, the total buffering space is $B_{total}(j)$, the factors are known, if the relaying master cluster head satisfies

$$\frac{E_{rem}(i)+E_{rem}(j)}{E_{i\_j}}\cdot\frac{B_{spa}(j)}{B_{total}(j)}>\frac{E_{rem}(i)}{E_{T(i\_Sink)}} \quad (10)$$

Here, the distance between C_{si} and C_{mj} is $d(i, j)$ and the distance between C_{mj} and Sink, it can save much energy that multi-hops communication between C_{si} and C_{mj} substitutes for direct communication.

Proof: Assuming there are k candidate neighbor cluster heads in the table of slave cluster head C_{si}. If slave cluster head transmits data to Sink by multi-hops communication, the network energy consumption consists of four parts: C_{si} transmitting data, C_{mj} receiving data, C_{mj} fusing data and C_{mj} transmitting data. According to formula (1), (2) and (3), we can obtain the energy consumption of respective part.

(1) The energy consumption of transmitting data from C_{si} to Sink is

$$E_{T(i\_j)}=\begin{cases} l(E_{elec}+\varepsilon_{fs}d^2(i,j)) & d(i,j)<d_0 \\ l(E_{elec}+\varepsilon_{amp}d^4(i,j)) & d(i,j)\geq d_0 \end{cases} \quad (11)$$

(2) The energy consumption of transmitting data from Csi to Cmj is

$$E_{T(i\_j)}=\begin{cases} l(E_{elec}+\varepsilon_{fs}d^2(i,j)) & d(i,j)<d_0 \\ l(E_{elec}+\varepsilon_{amp}d^4(i,j)) & d(i,j)\geq d_0 \end{cases} \quad (12)$$

(3) The energy consumption of receiving data of Cmj is

$$E_{R(j)}=lE_{elec} \quad (13)$$

(4) The energy consumption of fusing data of Cmj is

$$E_f(m,l)=mlE_{fusion} \quad (14)$$

Here, m is the number of fusing data package. l bits are the size of every package.

(5) The energy consumption of transmitting data from Cmj to Sink is

$$E_{T(j\_Sink)}=\begin{cases} l(E_{elec}+\varepsilon_{fs}d^2(j,Sink)) & d(j,Sink)<d_0 \\ l(E_{elec}+\varepsilon_{amp}d^4(j,Sink)) & d(j,Sink)\geq d_0 \end{cases} \quad (15)$$

If only satisfying the formula (16), i.e., the product of two ratios is more than the ratio of $E_{T(i\_Sink)}$ and the residual energy of slave cluster head C_{si}, it can save much energy that multi-hops communication between C_{si} and C_{mj} substitutes for direct communication.

$$\frac{E_{rem}(i)+E_{rem}(j)}{E_{T(i\_j)}+E_{R(j)}+E_f(m,l)+E_{T(j\_Sink)}}\cdot\frac{B_{spa}(j)}{B_{total}(j)} > \frac{E_{rem}(i)}{E_{T(i\_Sink)}}\cdot 1 \quad (16)$$

□

6 Simulations

6.1 Simulation environment and performance metrics

The simulation environment is on Intel Pentium with double cores (2.2 GHz), 2GRAM. We implement EBCRP and other four data gathering protocols by OMNet++ in the same experiment condition.

Table 1 Network parameters

parameters	value
Network area R	120×120m
Number of nodes	100
Node distribution	Randomly non-uniform
The initial energy of node	2J
The size of data packets	4000 bits
α, β, γ	0.3, 0.3, 0.4
r_0	30m
d_0	87m

The four data gathering protocols are LEACH, UCR, ACT and SCA. The network area is square with the length of side 120m, the position of Sink locates on (100,250) out of the network area. Other experiment parameters are shown in table 1. A typical energy consumption model is adopted, and the specific details of this energy consumption model can be found in [4].

6.2 Simulation results

6.2.1 Total energy consumption

Figure 1 shows the total energy consumption of five data gathering protocols. We can observe that the total energy consumption of five data gathering protocol increases with the network working time increases. Originally, the energy consumption of EBCRP is more than SCA protocol. After about 400($\times 10^3$)s, the energy consumption of EBCRP is less than SCA protocol. This can be explained by the fact that the clustering of EBCRP depends on the centralization of Sink to which every node needs sends its node information. However, SCA protocol is a distribution data gathering protocol that every node does not send node information to Sink. After about 400($\times 10^3$) s, the energy consumption of SCA is more than EBCRP, for SCA protocol selects two cluster heads with message negotiation mechanism. Therefore, the energy consumption of EBCRP at this time is less than SCA protocol.

□

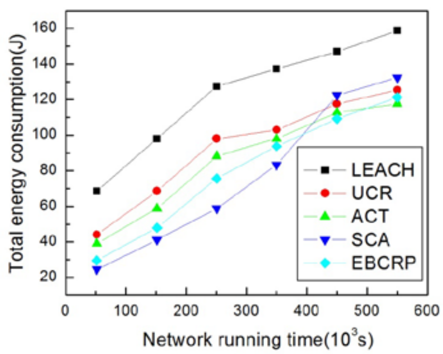
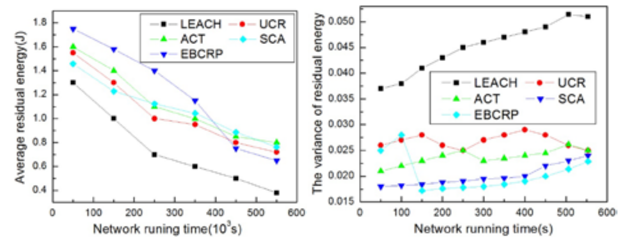


Fig.1. Total energy consumption

The energy consumption of LEACH is the most in the five data gathering protocols. This can be explained by the fact that cluster heads of LEACH sends data to Sink directly after gathering data from intra-cluster so that the energy consumption of LEACH is the most. UCR protocol is unequal clustering protocol, which the size is small near Sink so that there is enough energy to relay data from other clusters. Hence, the total energy consumption of UCR is less than LEACH. Although ACT protocol is also unequal cluster protocol, the size of clustering is the global optimization; however, the clustering size of UCR protocol is not the global optimization theoretically. Therefore, the total energy consumption of ACT protocol is less than that of UCR protocol.

6.2.2 Energy balance Factor

Figure 2 shows the average and the variance of residual energy of five data gathering protocols. As shown in figure 2(a), the average residual energy of EBCRP is more and the variance is less. However, we can observe that the change of the variance of residual energy is huge in figure 2(b) when the running time is about 100($\times 10^3$)s. This can be explained by the fact that every node sends its node information to Sink when the network starts. Due to the different distance from the nodes to Sink, the energy consumption of nodes transmission is different. Hence, the variance of residual energy at the running time is more. Synthetically, the variance of residual energy of EBCRP is the least and more stable than other four data gathering protocols. Accordingly, energy balance Factor of EBCRP is the best in the five data gathering protocols.



(a) (b)

Fig.2. Average residual energy and Residual energy variance

6.2.3 Number of alive nodes

Figure 3 shows the number of alive nodes in five data gathering protocols. As shown in figure 3, the stability of EBCRP is better and the number of alive nodes at the same running time is obviously more than other four data gathering protocols. Although EBCRP appears the first node dead, the number of alive nodes linearly decreases rather than the network separation that causes the function of data gathering lost. This can be explained by the fact that the network is clustering again in order that the energy consumption of alive nodes is balanced again when the number of alive nodes is less than $N/m_{\text{opt}} \times 15\%$ in the clustering maintain phase of EBCRP.

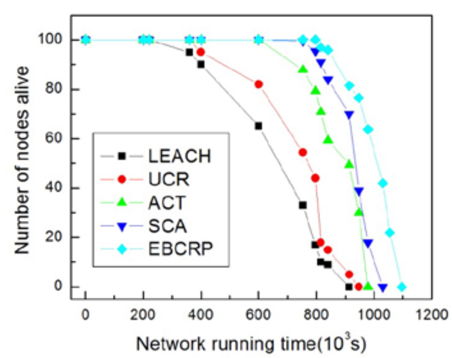


Fig.3. Number of alive nodes

In addition, if there is no relay node to Sink during inter-cluster data transmission, nodes can send data to Sink directly so that the network lifetime can be prolonged. However, UCR, ACT and SCA fit for the nodes with uniform distribution. When the three protocols move to the nodes with non-uniform distribution, the energy consumption balance is seriously influenced. Synthetically, the stability of EBCRP is better than other four data gathering protocols in the case of the nodes with non-uniform distribution.

7 Conclusion

In this paper, we propose an energy-balanced clustering routing protocol (EBCRP) based on task separation in wireless sensor networks. In EBCRP, the network is firstly divided into clusters by using global information. And each of them has the same number of sensor nodes so as to balance the energy consumption of intra-cluster. Furthermore, task separation, the tasks of traditional single cluster head are separated and achieved by two cluster heads respectively, is proposed to reduce the traffic burden for single head. Moreover, we explore an energy-efficient and reliable inter-cluster routing algorithm, which considers comprehensively three factors: residual energy, distance and available buffer space of nodes.

Although EBCRP can implement better performance in the assuming two dimension network, the nodes are really deployed in three dimension network. To meet this requirement of the real network better, the future work will adopt other network model and mechanism to resolve the problem of energy hole. For example, node deployment, mobile Sink etc. in three dimension networks.

Acknowledgments

This research is supported by the National Nature Science Foundation of China (61170169, 61170168), the Science foundation of the Liaoning Education Ministry (L2013517), the Foundation of Dalian Scientific and Technical Planning Project (2013A16GX115), the Program for Liaoning excellent talents in University (LJQ2015006) and Liaoning Natural Science Foundation (2015020015).

References

[1] Akyildiz I, Weilian S, Sankarasubramaniam Y, and Cayirci C, A survey on sensor networks, *IEEE Communications Magazine*, Vol.40, No.8, 2012, pp. 102-114.

[2] Asharioun H, Asadollahi H, Wan T, *et al*. A Survey on Analytical Modeling and Mitigation Techniques for the Energy Hole Problem in Corona-Based Wireless Sensor Network, *Wireless Personal Communications*, Vol.81, 2015, pp. 161–187.

[3] Jia Jie, Chen Jian, Wang Xingwei, *et al*. Energy-balanced density control to avoid energy hole for wireless sensor networks, *International Journal of Distributed Sensor Networks*, 2012, Article No. 812013, pp. 1-10.

[4] Liu Anfeng, Jin Xin, Cui Guohua, *et al*. Deployment guidelines for achieving maximum lifetime and avoiding energy holes in sensor networks, *Information Sciences*, Vol. 230, 2013, pp. 197-226.

[5] Liu T. Avoiding energy holes to maximize network lifetime in gradient sinking sensor networks, *Wireless personal communications*, Vol. 70, No. 2, 2013, pp. 581-600.

[6] Chen Guihai, Li Chengfa, Ye Mao, *et al*. An unequal cluster-based routing protocol in wireless sensor networks, *Wireless Networks*, Vol. 15, Issue 2, 2009, pp. 193-207.

[7] Kacimi R, Dhaou R, Beylot A L, Load balancing techniques for lifetime maximizing in wireless sensor networks, *Ad Hoc Networks*, Vol. 11, No. 8, 2013, pp. 2172-2186.

[8] Abdulla A, Nishiyama H, Kato N, Extending the lifetime of wireless sensor networks: A hybrid routing algorithm, *Computer Communications*, Vol. 35, No. 9, 2012, pp. 1056-1063.

[9] Bandyopadhyay S, Coyle E J, An energy efficient hierarchical clustering algorithm for wireless sensor networks, INFOCOM 2003, IEEE, Vol. 3, pp. 1713-1723.

[10] Heinzelman WB, Chandrakasan AP and , An application-specific protocol architecture for wireless microsensor networks, *IEEE Transactions on Wireless Communications*, Vol. 1, No. 4, 2002, pp. 660-670.

[11] Wang A, Yang D, Sun D, A clustering algorithm based on energy information and cluster heads expectation for wireless sensor networks, *Computers & Electrical Engineering*, Vol. 38, No. 3, 2012, pp. 662-671.

[12] Lai W K, Fan C S, Lin L Y. Arranging cluster sizes and transmission ranges for wireless sensor networks, *Information Sciences*, Vol. 183, No. 1, 2012, pp. 117-131.

[13] Fu S, Ma J, Li H, *et al*. Energy-Balanced Separating Algorithm for Cluster-Based Data Aggregation in Wireless Sensor Networks, *International Journal of Distributed Sensor Networks*, 2013, ID:570805, pp. 1-15.

[14] Chen J, Li Z, Kuo Y H. A Centralized Balance Clustering Routing Protocol for Wireless Sensor Network, *Wireless Personal Communications*,

Vol. 72, No. 1, 2013, pp. 623-634.

[15] Khalil, E. A. and Attea, B. A. Energy-aware evolutionary routing protocol for dynamic clustering of wireless sensor networks, *Swarm and Evolutionary Computation*, Vol. 1, No. 4, 2011, pp. 195-203.

[16] Xu Y, Heidemann J and Estrin D. Geography-informed energy conservation for Ad hoc routing, Proceedings of the 7th ACM/IEEE international conference on mobile computing and networking, ACM, New York, 2001, pp. 70-84.

[17] Salzmann J, Behnke R and Timmermann D, Tessellating cell shapes for geographical clustering, Proceedings of IEEE international conference on computer and information technology, IEEE, Bradford, 2010, pp. 2891-2896.

[18] Anisi M H, Abdullah A H, Razak S A, Energy-efficient and reliable data delivery in wireless sensor networks, *Wireless networks*, Vol. 19, No. 4, 2013, pp. 495-505.

[19] Mhatre V, Rosenberg C, Design guidelines for wireless sensor networks: communication, clustering and aggregation, *Ad Hoc Networks*, Vol. 2, Issue 1, 2004, pp. 45-63.

[20] Ahmed E.A.A.A, Hiroki N, Nei K, Extending the lifetime of wireless sensor networks: a hybrid routing algorithm, *Computer Communications*, Vol. 35, Issue 9, 2012, pp. 1056-1063.