Wireless Netw
DOI 10.1007/s11276-007-0035-8

An unequal cluster-based routing protocol in wireless sensor networks

Guihai Chen · Chengfa Li · Mao Ye · Jie Wu

© Springer Science + Business Media, LLC 2007

Abstract Clustering provides an effective method for prolonging the lifetime of a wireless sensor network. Current clustering algorithms usually utilize two techniques; selecting cluster heads with more residual energy, and rotating cluster heads periodically to distribute the energy consumption among nodes in each cluster and extend the network lifetime. However, they rarely consider the hot spot problem in multihop sensor networks. When cluster heads cooperate with each other to forward their data to the base station, the cluster heads closer to the base station are burdened with heavier relay traffic and tend to die much faster, leaving areas of the network uncovered and causing network partitions. To mitigate the hot spot problem, we propose an Unequal Cluster-based Routing (UCR) protocol. It groups the nodes into clusters of unequal sizes. Cluster heads closer to the base station have smaller cluster sizes than those farther from the base station, thus they can preserve some energy for the inter-cluster data forwarding. A greedy geographic and energy-aware routing protocol is designed for the inter-cluster communication, which considers the tradeoff between the energy cost of relay paths and the residual energy of relay nodes. Simulation results show that UCR mitigates the hot spot problem and achieves an obvious improvement on the network lifetime.

Keywords Wireless sensor networks · Unequal clustering · Routing · Network lifetime · Hot spot problem

G. Chen (✉) · C. Li · M. Ye
State Key Laboratory for Novel Software Technology, Nanjing University Nanjing, 210093, P. R. China
e-mail: gchen@nju.edu.cn

J. Wu
Department of Computer Science and Engineering, Florida Atlantic University Boca Raton, FL 33431, USA

1 Introduction

Rapid technological advances in micro-electro-mechanical systems (MEMS) and low-power wireless communication have enabled the deployment of large scale wireless sensor networks. The potential applications of sensor networks are highly varied, such as environmental monitoring, target tracking, and battlefield surveillance [1, 2]. Sensors in such a network are equipped with sensing, data processing and wireless communication capabilities. Distinguished from traditional wireless networks, sensor networks are characterized by severe power, computation, and memory constraints. Due to limited and non-rechargeable energy provision, the energy resource of sensor networks should be managed wisely to extend the lifetime of sensors. Although much attention has been paid to low-power hardware design and collaborative signal processing techniques, energy efficient algorithms must be supplied at various networking layers. In addition, it is very important to balance the energy consumption among all sensor nodes to prolong the network lifetime.

We consider a network of energy-constrained sensors that are deployed over a geographic area for monitoring the environment. Each sensor periodically produces information as it monitors its vicinity. The basic operation in such a network is the systematic gathering and transmission of sensed data to a base station for further processing. In order to achieve high energy efficiency and increase the network scalability, sensor nodes can be organized into clusters. Data collected from sensors are sent to the cluster head first, and then forwarded to the base station. The high density of sensor networks may lead to multiple adjacent sensors generating redundant sensed data, thus data aggregation can be used to eliminate the data redundancy and reduce the communication load [3].

For periodical data-gathering applications, a method to group sensor nodes into clusters and aggregate data within clusters has been proposed in [4].

Within a sensor node, the dominant energy consumer is the radio unit. When the network is partitioned into clusters, data transmission can be classified into two stages; intra- and inter-cluster communication. Non-cluster-head nodes first send their data to the cluster head, and then cluster heads send the data to the base station. Most of the works in the literature use single hop intra-cluster communication mode. Notice that the base station is usually located far away from the sensing area. Previous research (e.g., [5]) has shown that multihop inter-cluster communication mode is usually more energy efficient because of the characteristics of wireless channel. Thus it's better to let cluster heads cooperate with each other to forward their data to the base station. However, the many-to-one traffic pattern results in the hot spot problem when the multihop forwarding mode is adopted in inter-cluster communication. Because the cluster heads closer to the base station are burdened with heavier relay traffic, the area near the base station becomes a hot spot. Nodes in the hot spot drain their energy and die much faster than other nodes in the network, reducing sensing coverage and causing network partitions. Although many protocols proposed in the literature reduce energy consumption on forwarding paths to increase energy efficiency, they do not necessarily extend the network lifetime due to the unbalanced energy consumption.

In this paper, we propose and evaluate an Unequal Cluster-based Routing (UCR) protocol for mitigating the hot spot problem in wireless sensor networks. It is designed for long-lived, source-driven sensor network applications, such as periodical environmental information reporting. UCR does not require special node capabilities, such as location-awareness or heterogeneity. UCR consists of two parts, one is an Energy-Efficient Unequal Clustering (EEUC) algorithm for topology management, and the other is a greedy geographic and energy-aware routing protocol for inter-cluster communication. The main contribution of the paper is that we provide the first unequal cluster-based routing protocol to mitigate the hot spot problem mentioned before and thus prolong the network lifetime. EEUC is a self-organized competition-based algorithm, where cluster heads are selected based on local information (i.e., the residual energy of neighboring nodes). The node's competition range decreases as its distance to the base station decreases. The result is that clusters closer to the base station are expected to have smaller cluster sizes, thus the cluster heads will consume lower energy during the intra-cluster data processing, and can preserve some more energy for the inter-cluster relay traffic. The inter-cluster multihop routing protocol considers the tradeoff between the energy cost of relay links and the energy of relay nodes. Because the UCR protocol considers the impacts of intra- and inter-cluster traffic simultaneously, it successfully mitigates the hot spot problem and achieves a remarkable network lifetime improvement shown in simulation results.

The remainder of this paper is organized as follows. Section 2 summarizes related work. Section 3 describes the network model and elaborates the hot spot problem that we address in this work. Section 4 presents the EEUC algorithm and inter-cluster multihop routing protocol in detail. Section 5 analyzes some properties of UCR. Section 6 shows UCR's effectiveness via simulations, and compares UCR with HEED [9]. Section 7 concludes the paper with directions for future work.

2 Related work

During the last few years, there has been substantial research in the area of routing in wireless sensor networks. Proposed protocols can be classified into data-centric, hierarchical, location-based, network flow and QoS-aware routing [6]. Many energy-efficient (hierarchical) clustering algorithms have been proposed to prolong the network lifetime [4, 7–12]. We review some of the most relevant papers. Heinzelman et al. [4] first propose a clustering protocol called LEACH for periodical data-gathering applications. LEACH uses randomized rotation of cluster heads to distribute energy consumption over all nodes in the network. In the data transmission phase, each cluster head forwards an aggregated packet to the base station directly. An energy-aware variant of LEACH is proposed in [7], in which the nodes with higher energy are more likely to become cluster heads. However, the underlying routing protocol is assumed to be able to propagate node residual energy through the network. The authors analytically determine the optimum number of cluster heads. Choi et al. [8] propose a two-phase clustering (TPC) scheme. At the cluster head electing stage, each node advertises for cluster head with a random delay, and the node who overhears others' advertisement will cancel its scheduled advertisement. After forming the initial clusters, each cluster member searches for a neighbor closer than the cluster head within the cluster to set up an energy-saving data relay link. HEED [9] introduces a variable known as cluster radius which defines the transmission power to be used for intra-cluster broadcast. The initial probability for each node to become a tentative cluster head depends on its residual energy, and final heads are selected according to the intra-cluster communication cost. HEED terminates within a constant number of iterations, and achieves fairly uniform distribution of cluster heads across the network. VCA [10] is an improvement over HEED. Sensors vote for their neighbors to elect suitable cluster heads. The authors also propose two strategies to balance the intra-cluster workload among cluster heads. EECS [12] introduces a cluster head competitive algorithm without message exchange iterations. It extends LEACH and

HEED by choosing cluster heads with more residual energy. It also achieves a decent distribution of cluster heads.

While the clustering problem has been extensively explored, researchers have only recently started to study the strategies for balancing the workload among cluster heads while considering the inter-cluster traffic. In single hop sensor networks, cluster heads use direct communication to reach the base station, and the problem of unbalanced energy consumption among cluster heads arises. Cluster heads farther away from the base station have heavier energy burden due to the long-haul communication links. Consequently, they will die earlier. In EECS [12], a distance-based cluster formation method is proposed to produce clusters of unequal sizes. Clusters farther away from the base station have smaller sizes, thus some energy could be preserved for long-haul data transmission to the base station. On the other hand, the hot spot problem arises when multihop routing is adopted when cluster heads deliver their data to the base station. Soro and Heinzelman [13] first investigate an unequal clustering model for balancing the energy consumption of cluster heads in multihop sensor networks. The work focuses on a heterogeneous network where cluster heads (super nodes) are deterministically deployed at pre-computed locations. Thus, it's easy to control the actual sizes of clusters. Through both theoretical and experimental analyses, the authors show that unequal clustering could be beneficial, especially for heavy traffic applications. Shu et al. [14] study an optimization problem of balancing energy consumption among cluster heads which is formulated as a signomial optimization problem. The study demonstrates the significance of simultaneously considering the impacts of intra- and inter-cluster traffic.

The hot spot problem addressed in this paper is due to the many-to-one multihop data forwarding pattern on the cluster head backbones. Thus, it's similar to the common hot spot problem that appears in flat multihop sensor networks. Researchers have proposed several methods to mitigate this kind of hot spot problem. Perillo et al. [15] analyze two such strategies. Although the network lifetime can be improved by using a more intelligent transmission power control policy that balances the energy consumption of each node, they conclude that it cannot solve the hot spot problem on its own. They also investigate the effectiveness and cost efficiency of using a heterogeneous clustering hierarchy to mitigate the hot spot problem. Olariu and Stojmenovic [16] investigate the theoretical aspects of uneven energy depletion in sink-based flat sensor networks. They show that for $\alpha = 2$ (power loss rate, refer to Eq. (1) in Section 3.1) the uneven energy depletion phenomenon is intrinsic to the system and no routing strategy can avoid the creation of an energy hole around the sink. They also argue that for larger α energy-efficient sensor-to-sink routes can be designed and the energy consumption can be balanced across the network. In some works, special assumptions about the sensor network are made to facilitate solving the hot spot problem. For example, in homogenous sensor networks, additional sensor nodes can be deployed in the area near the base station as reservoirs of energy [17]. In [18], multiple sink nodes are deployed to alleviate the hot spot problem in large-scale sensor networks. It also reduces the energy dissipation at each node. Recent research begins to exploit the mobility of some nodes to facilitate the delivery of the sensed data to the base station. Gandham et al. [19] investigate the idea of employing multiple mobile sink nodes to increase the lifetime of sensor networks. They present an ILP (Integer Linear Programming) model to determine the locations of multiple sinks. In [20], a novel linear programming formulation for maximizing the network lifetime is presented to determine the movement of the sink and the times the sink visits certain network nodes. However, it is not always possible for the sink to be mobile in hostile terrains. Wang et al. [21] investigate a heterogeneous sensor network in which a mobile relay node is employed to address the hot spot problem. They show that a lifetime improvement of four times can be achieved by using a mobile relay node, and the mobile relay needs to stay only within a two hop radius of the data sink. However, these approaches inevitably increase the cost and management complexity of sensor networks, thus they are not always feasible.

Location-based geographic routing is also attractive in wireless sensor networks due to its efficiency and scalability, and it is more energy-efficient for data forwarding on the cluster head backbone compared to traditional hop-based methods. Geographic routing algorithms have been studied in the context of wireless networks [22–24]. Frey and Stojmenovic [25] provide a good review of geographic and energy-aware routing algorithms for wireless sensor networks.

In this paper, we study the hot spot problem existing in the hierarchical (cluster-based) wireless sensor networks. The advantages of UCR are as follows. UCR is the first protocol without any aid of a super node [13, 14] or mobile node [19–21]. In our study there is no need for pre-deployment [13, 14, 17, 18], which greatly simplifies system deployment. Besides that we also present a novel inter-cluster routing strategy considering the metrics of both transmitting distance and residual energy.

3 Preliminaries

3.1 System model

In this paper, we consider a sensor network consisting of N sensor nodes uniformly deployed over a vast field to continuously monitor the environment. We denote the

i-th sensor by s_i and the corresponding sensor node set $S = \{s_1, s_2, \ldots, s_N\}$, where $|S| = N$. We make some assumptions about the sensor nodes and the underlying network model:

1. There is a base station (i.e., data sink) located far from the sensing field. Sensors and the base station are all stationary after deployment.
2. Sensors are homogeneous and have the same capabilities. Each node is assigned a unique identifier (ID).
3. Sensors are capable of operating in an active mode or a low-power sleeping mode.
4. Sensors can use power control to vary the amount of transmission power according to the distance to the desired recipient.
5. Links are symmetric. A node can compute the approximate distance to another node based on the received signal strength, if the transmitting power is known.

We use a simplified model shown in [7] for the communication energy dissipation. Both the free space (d^2 power loss) and the multi-path fading (d^4 power loss) channel models are used, depending on the distance between the transmitter and receiver. The energy spent for transmission of an l-bit packet over distance d is:

$$E_{Tx}(l, d) = lE_{\text{elec}} + l\epsilon d^{\alpha} = \begin{cases} lE_{\text{elec}} + l\epsilon_{fs}d^2, & d < d_o \\ lE_{\text{elec}} + l\epsilon_{mp}d^4, & d \geq d_o. \end{cases} \tag{1}$$

The electronics energy, E_{elec}, depends on factors such as the digital coding, and modulation, whereas the amplifier energy, $\epsilon_{fs}d^2$ or $\epsilon_{mp}d^4$, depends on the transmission distance and the acceptable bit-error rate. To receive this message, the radio expends energy:

$$E_{Rx}(l) = lE_{\text{elec}}. \tag{2}$$

It is assumed that the sensed information is highly correlated, thus the cluster head can always aggregate the data gathered from its members into a single length-fixed packet. In some proposed algorithms, relay nodes can aggregate the incoming packets from other clusters together with its own packets. This assumption is impractical because the correlation degree of sensed data from different clusters is comparatively low. In this work, relay nodes don't aggregate the incoming packets. We assume that a cluster head consumes E_{DA} (nJ/bit/signal) amount of energy for data aggregation.

3.2 The problem of unbalanced energy consumption

In this work, cluster heads form a virtual backbone for inter-cluster communication. Each head node forwards the data to the base station via a multihop path through other intermediate cluster heads. The reason this is done is because multihop communication is more realistic; nodes may not be able to reach the base station due to the limited transmission range. Even if a node can use power control to send data to a farther receiver, previous research has shown that it is obviously a waste of energy. However, when multihop routing is adopted in inter-cluster communication, the many-to-one traffic pattern on the cluster head overlay leads to the hot spot problem. In a clustered sensor network, each cluster head spends its energy on intra- and inter-cluster processing. The energy consumed in intra-cluster processing varies proportionally to the number of nodes within the cluster. Proposed clustering algorithms that consider the load balance issue usually produce clusters of even sizes, thus the intra-cluster load is roughly the same for all cluster heads. On the other hand, the inter-cluster traffic load of cluster heads is highly uneven. Cluster heads closer to the base station have a higher load of relay traffic. Consequently, they will die much faster than the other cluster heads, possibly reducing sensing coverage and leading to network partitioning.

A fundamental issue in wireless sensor networks is maximizing the network lifetime subject to a given energy constraint. To achieve this goal, energy consumption must be well-balanced among nodes. In homogeneous networks, the cluster head role can be periodically rotated among nodes to balance the energy dissipation. However, the hot spot problem cannot be avoided. The main objective of the rotation is to balance the energy consumption among the sensor nodes in each cluster, and it could hardly balance the energy consumption among cluster heads in the inter-cluster multihop routing scenario. We also argue that using node's residual energy as the only criterion when selecting cluster heads is not sufficient to balance energy consumption across the network. Selecting cluster heads with more residual energy can only be helpful to balance energy consumption among nodes within a cluster radius in the long term. It is ineffective to balance the load among different cluster heads to avoid the hot spot problem if the cluster heads are uniformly distributed over the network. Because sensor nodes in the hot spot still die faster, it cannot make efficient use of all nodes' energy.

To mitigate the hot spot problem, we introduce a novel unequal clustering protocol for hierarchical routing, called UCR. Both the rotation of cluster heads and choosing cluster heads with more residual energy are adopted into the clustering algorithm EEUC. It organizes the network into clusters of unequal sizes. By decreasing the number of nodes in clusters with higher relay load near the base station, we can maintain more uniform energy consumption among cluster heads in the long run.

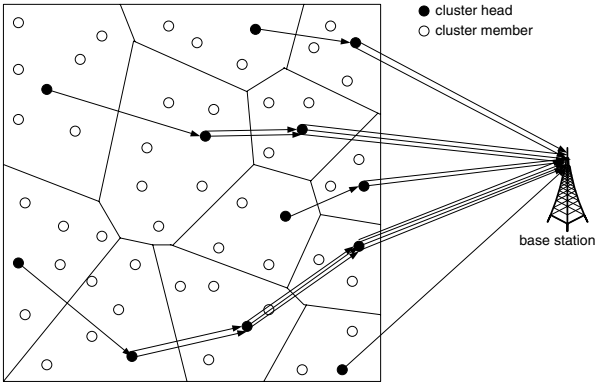

Fig. 1 An overview of the UCR protocol

4 The unequal cluster-based routing protocol

The UCR protocol consists of two parts: an energy-efficient unequal clustering algorithm called EEUC and an inter-cluster greedy geographic and energy-aware routing protocol. At the network deployment stage, the base station broadcasts a beacon signal to all sensors at a fixed power level. Therefore each sensor node can compute the approximate distance to the base station based on the received signal strength. It not only helps nodes to select the proper power level to communicate with the base station, but also helps us to produce clusters of unequal sizes. Detailed descriptions of the unequal clustering algorithm and intra-cluster multihop routing protocol are in the following two subsections. Figure 1 gives an overview of the UCR protocol, where the unequal Voronoi cells represent the unequal clusters formed by EEUC and the traffic among cluster heads illustrates our multihop forwarding method.

4.1 Unequal clustering algorithm

Clustering a wireless sensor network means partitioning its nodes into clusters, each one with a cluster head and some ordinary nodes as its members. Similar to LEACH, the operation of UCR is divided into rounds. The task of being a cluster head is rotated among sensors in each round to distribute the energy consumption across the network. EEUC is a distributed cluster head competitive algorithm, where the cluster head selection is primarily based on the residual energy of tentative cluster heads. Furthermore, EEUC produces clusters of unequal sizes to mitigate the hot spot problem. Clusters closer to the base station have smaller cluster sizes, thus they will consume less energy during the intra-cluster data processing, and can conserve some more energy for the inter-cluster relay traffic. The pseudocode for each sensor node at the cluster head selecting stage is given in Fig. 2.

Here we explain the clustering algorithm in detail. First, several tentative cluster heads are randomly selected to compete for final cluster heads. Ordinary nodes become tentative cluster heads with the same probability T which is a predefined threshold. Nodes that fail to be tentative heads keep sleeping until the cluster head selection stage ends.

Each tentative cluster head s_i has a competition range R_i. Different competition ranges are used to produce clusters of unequal sizes. Only one final cluster head is allowed in each competition range. If s_i becomes a cluster head at the end of the competition, there will not be another cluster head s_j in s_i's competition range. Figure 3 illustrates a topology of tentative cluster heads, where the circles represent different competition ranges of tentative cluster heads. In Fig. 3 both s_1 and s_2 can be final cluster heads, but s_3 and s_4 cannot. Therefore, the distribution of cluster heads can be controlled over the network. Cluster heads closer to the base station should support smaller cluster sizes, thus more clusters need to be produced closer to the base station. That is to say, the tentative cluster head's competition range should decrease as its distance to the base station decreases. We need to select a proper scope of competition ranges in the network. Suppose R_0 is the maximum competition range which is predefined, and the minimum competition range is set to $(1-c)R_0$ correspondingly, where c is a constant coefficient between 0 and 1. Thus the tentative cluster head s_i's competition range R_i can be expressed as a linear function of its distance to the base station:

$$R_i = \left(1 - c\frac{d_{\max} - d(s_i, BS)}{d_{\max} - d_{\min}}\right)R_0 \tag{3}$$

where $d_{\max}$ and $d_{\min}$ denote the maximum and minimum distance between sensor nodes in the network and the base station, and $d(s_i, BS)$ denotes the distance between s_i and the base station. According to Eq. (3), if c is set to $\frac{1}{3}$, R_i varies from $\frac{2}{3}R_0$ to R_0 according to the distance between s_i and the base station.

Each tentative cluster head maintains a set S_{CH} of its "adjacent" tentative cluster heads. In lines 10–13 of Fig. 2, each tentative head constructs its S_{CH}. Tentative head s_j is an "adjacent" node of s_i if s_j is in s_i's competition diameter or s_i is in s_j's competition diameter. Whether a tentative cluster head s_i will become a final cluster head depends on the nodes in $s_i.S_{CH}$ only, i.e., the algorithm is localized.

In the cluster head selecting algorithm, the broadcast radius of every control message is R_0, thus s_i can hear all messages from nodes in its S_{CH}. In line 6 of Fig. 2, each tentative cluster head broadcasts a `COMPETE_HEAD_MSG` which contains its competition radius and residual energy (RE). After the construction of S_{CH} has finished in lines 10–13, each tentative cluster head checks its S_{CH} and makes a decision as to whether it can act as a cluster head in lines 14–26. Before deciding what its role is going to be, s_i needs to know what

Fig. 2 Cluster head competitive algorithm

Cluster head competitive algorithm for node s_i

1: $\mu \leftarrow RAND(0,1)$
2: **if** $\mu < T$ **then**
3: $\quad beTentativeHead \leftarrow TRUE$
4: **end if**
5: **if** $beTentativeHead = TRUE$ **then**
6: $\quad$ broadcast a COMPETE_HEAD_MSG$(s_i.ID, R_i, s_i.RE)$
7: **else**
8: $\quad$ EXIT
9: **end if**
10: on receiving a COMPETE_HEAD_MSG from node s_j
11: **if** $d(s_i, s_j) < max(R_i, R_j)$ **then**
12: $\quad$ add s_j to $s_i.S_{CH}$
13: **end if**
14: **while** the time slot for cluster head competing has not expired **do**
15: $\quad$ **if** $s_i.RE > s_j.RE, \forall s_j \in s_i.S_{CH}$ **then**
16: $\quad\quad$ broadcast a FINAL_HEAD_MSG$(s_i.ID)$ and then EXIT
17: $\quad$ **end if**
18: $\quad$ on receiving a FINAL_HEAD_MSG from node s_j
19: $\quad$ **if** $s_j \in s_i.S_{CH}$ **then**
20: $\quad\quad$ broadcast a QUIT_ELECTION_MSG$(s_i.ID)$ and then EXIT
21: $\quad$ **end if**
22: $\quad$ on receiving a QUIT_ELECTION_MSG from node s_j
23: $\quad$ **if** $s_j \in s_i.S_{CH}$ **then**
24: $\quad\quad$ remove s_j from $s_i.S_{CH}$
25: $\quad$ **end if**
26: **end while**

each node x in its S_{CH} such that $x.RE > s_i.RE$ has decided for itself. In case of a tie, the smaller node ID is chosen. In lines 15–17, once s_i finds that its residual energy is more than all the nodes in its S_{CH}, it will win the competition and broadcast a FINAL_HEAD_MSG to inform its adjacent tentative cluster heads. In lines 18–21, if s_j belongs to $s_i.S_{CH}$ and s_i receives a FINAL_HEAD_MSG from s_j, s_i will give up the competition immediately, and inform all nodes in its S_{CH} by broadcasting a QUIT_ELECTION_MSG. In lines 22–25, if s_i receives a QUIT_ELECTION_MSG form s_j and s_j belongs to $s_i.S_{CH}$, s_i will remove s_j from its S_{CH}.

After cluster heads have been selected, sleeping nodes now wake up and each cluster head broadcasts a CH_ADV_MSG across the network field. Each ordinary node chooses its closest cluster head with the largest received signal strength and then informs the cluster head by sending a JOIN_CLUSTER_MSG. A Voronoi diagram of sensor nodes is then constructed. The cluster head sets up a TDMA schedule and transmits it to the nodes in the cluster. After the TDMA schedule is known by all nodes in the cluster, the set-up phase is completed and the steady-state operation (data transmission) can begin.

The organization of intra-cluster data transmission is similar to LEACH after clusters have been set up, so we omit it in this section.

4.2 Inter-cluster multihop routing

When cluster heads deliver their data to the base station, each cluster head first aggregates the data from its cluster members, and then sends the packet to the base station via a multihop path through other intermediate cluster heads. The routing problem here differs substantially from that of traditional ad-hoc wireless networks because of the many-to-one traffic pattern. On the other hand, both query-driven and event-driven routing protocols for wireless sensor networks are not suitable for the cluster head virtual backbone. In [9], Younis et al. prove that HEED can produce a connected multihop cluster head backbone using a fixed inter-cluster transmission range. Using the fixed transmission power

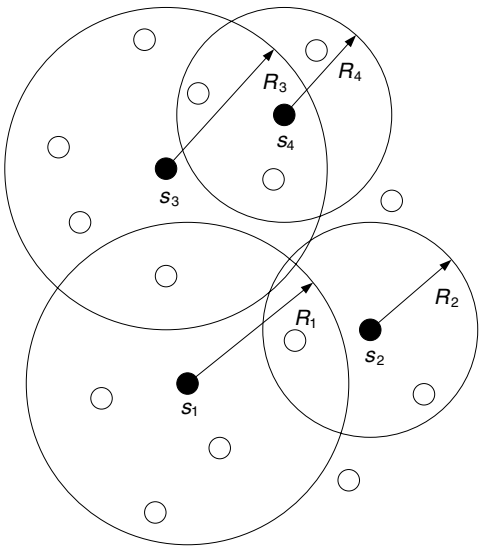

Fig. 3 The competition among tentative cluster heads

facilitates its implementation on the TinyOS platform, where the multihop routing uses a shortest-path-first algorithm [27]. By using adjustable transmission range and weak location information, we design a greedy geographic and energy-aware multihop routing protocol to extend the network lifetime.

Before selecting the next hop node, each cluster head broadcasts a short beacon message across the network at a fixed power which consists of its node ID, residual energy, and distance to the base station. Distance between each pair of cluster heads can be calculated approximately according to the received signal strength. We introduce a threshold TD_MAX in the multihop routing protocol. If a node's distance to the base station is smaller than TD_MAX, it transmits its data to the base station directly; otherwise it's better to find a relay node which can forward its data to the base station. In the previous work [28], Ye et al. defines the region within TD_MAX as the hot spot, and investigates the optimal value of TD_MAX to maximize the network lifetime in an equal clustered network.

It is worth explaining that the value of TD_MAX is always smaller than the actual maximum transmission range of a sensor node as we try to avoid the long-distance direct communication of heavy traffic. A node could still use a transmission range larger than TD_MAX if necessary.

To reduce wireless channel interference, it is better to choose an adjacent node as the relay node [29]. Because the transmission power of cluster heads is adjustable, hop count is improper to be used is a poor method of defining neighboring relations. In this paper, the multihop forwarding algorithm considers nodes on the cluster head backbone in the forward direction (i.e., closer to the base station) only. The neighboring node set R_{CH} of cluster head s_i is defined as

$$s_i.R_{CH} = \{s_j \mid d(s_i, s_j) \le xR_i, d(s_j, BS) < d(s_i, BS)\}. \quad (4)$$

x is the minimum integer that lets $s_i.R_{CH}$ contain at least one item (if there doesn't exist such an x, define $s_i.R_{CH}$ as a null set, and s_i will send its own data together with forwarding data directly to the base station).

Choosing the relay node with more residual energy could balance the energy consumption to extend the network lifetime. On the other hand, decreasing the energy cost per packet also contributes to the network lifetime. Here we propose a greedy geographic forwarding algorithm that aims to minimize the energy cost per packet. Suppose s_i chooses s_j as its relay node. For simplicity, we assume a free space propagation channel model. Because a localized algorithm is desirable, we assume there is a virtual hop between s_j and the base station. To deliver an l-length packet to the base station, the total energy consumed by s_i and s_j is

$$\begin{aligned} &E_{2-hop}(s_i, s_j) \\ &\quad = E_{Tx}(l, d(s_i, s_j)) + E_{Rx}(l) + E_{Tx}(l, d(s_j, BS)) \\ &\quad = l(E_{\text{elec}} + \epsilon_{fs} d^2(s_i, s_j)) + lE_{\text{elec}} \\ &\qquad + l(E_{\text{elec}} + \epsilon_{fs} d^2(s_j, BS)) \\ &\quad = 3lE_{\text{elec}} + l\epsilon_{fs}(d^2(s_i, s_j) + d^2(s_j, BS)) \end{aligned} \quad (5)$$

according to Eqs. (1) and (2). Thus we define

$$E_{\text{relay}}(s_i, s_j) = d^2(s_i, s_j) + d^2(s_j, BS) \quad (6)$$

as the energy cost of the path $s_i \to s_j \to BS$. We use the distance between nodes rather than precise location information of s_j to define the energy cost of the relay path. The bigger the E_{relay} is, the more energy will be consumed for transmitting packets on the path.

In the localized routing algorithm, s_i first chooses k eligible neighbor nodes from $s_i.R_{CH}$, denoted as the set S_{eligible}:

$$\begin{aligned} &s_i.S_{\text{eligible}} \\ &\quad = \{s_j \mid s_j \in s_i.R_{CH}, E_{\text{relay}}(s_i, s_j) \textit{ is the k smallest}\}. \end{aligned} \quad (7)$$

The pseudocode for constructing $s_i.S_{\text{relay}}$ is given in Fig. 4.

To reduce inefficiencies of energy consumption, a tradeoff should be made between the two criteria of residual energy and link cost E_{relay}. In our mechanism, s_i chooses as its relay node the neighbor in $s_i.S_{\text{eligible}}$ that has the biggest residual energy.

Besides the tradeoff between the two different goals, we propose another goal to balance the energy consumption. The nodes near the base station send the forwarding data directly to the base station, thus they may deplete their energy quickly if the base station is located far from the network field. In our solution, if cluster head s_j's distance to the base

Fig. 4 Eligible neighbor nodes choosing algorithm

```
while ∀s_j ∈ R_CH is not null do
    compute E_relay(s_i, s_j)
    if |S_relay| < k then
        put s_j into S_eligible
    else
        find out s_m that satisfies
        E_relay(s_i, s_m) = max{E_relay(s_i, s_n)}, ∀s_n ∈ S_eligible
        if E_relay(s_i, s_j) < E_relay(s_i, s_m) then
            replace s_m with s_j in S_eligible
        end if
    end if
    remove s_j from R_CH
end while
```

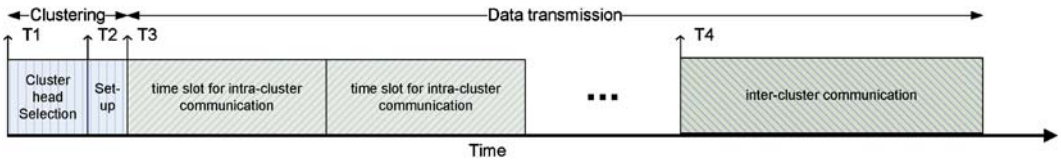

Fig. 5 Time line showing UCR operation

station is smaller than TD_MAX, and cluster head s_i selects s_j as its relay node according to the approach described before, and if the residual energy of s_j is smaller than that of s_i, we let s_i communicate with the base station directly rather than aggravating the load of s_j. In this way, energy of s_j can be saved and the network lifetime is extended further.

After each cluster head has chosen a relay node or decided to transmit its data to the the base station directly, a tree rooted at the base station is constructed. A cluster head receives data packets from tree descendants and sends them with the cluster's own packets up to the root.

Figure 5 illustrates the operation of UCR by the time line for one data gathering round. It begins with a clustering phase when cluster heads are selected and the intra-cluster TDMA schedule are set-up, followed by a data transmission phase where data are transferred from the nodes to the cluster head and on to the base station via a multihop path.

5 Performance analysis and discussion

5.1 Complexity and correctness analysis

This section presents the analysis of the UCR protocol. According to Algorithm 1, the cluster head selection process is message driven, thus we first discuss its message complexity.

Lemma 1. *The message complexity of the cluster formation algorithm is $O(N)$ in the network.*

Proof: At the beginning of the cluster head selection phase, $N \times T$ tentative cluster heads are produced and each of them broadcasts a `COMPETE_HEAD_MSG`. Then each of them makes a decision by broadcasting a `FINAL_HEAD_MSG` to act as a final cluster head, or a `QUIT_ELECTION_MSG` to act as an ordinary node. Suppose k cluster heads are selected. They send out k `CH_ADV_MSG`s, and then $(N-k)$ ordinary nodes transmit $(N-k)$ `JOIN_CLUSTER_MSG`s. Thus the messages add up to $2N \times T + k + N - k = (2T+1)N$ at the cluster formation stage per round, i.e., $O(N)$. □

Lemma 2. *There is no chance that two nodes are both cluster heads if one is in the other's competition range.*

Proof: Suppose s_j and s_k are both tentative cluster heads, and s_k is located within the circle of s_j's competition range. According to Algorithm 1, each node belongs to the other node's S_{CH}. If s_j first becomes a head node, then it will notice s_k its state, so s_k quits the competition and becomes an ordinary node; vice versa. □

Lemma 1 shows that the message overhead of EEUC is small. In HEED, the clustering algorithm terminates in N_{iter} iterations which can be bounded by a constant, and each tentative cluster head generates at most N_{iter} messages in the process. Because we have avoided the message iteration in the cluster head selection algorithm, the message exchange overhead in EEUC is much lower than that in HEED.

As described before, the threshold T determines the number of tentative cluster heads. Enough tentative cluster heads

guarantee good head choosing in terms of residual energy. On the other hand, too many tentative cluster heads cause a considerable message overhead. Thus a proper value of T should be chosen in order to guarantee the quality of head selection and reduce the message overhead.

Then we simply analyze the impact of R_0 and c on the network lifetime. According to Eq. (3), c dominates the unequal extent of cluster sizes. The bigger c is, the bigger the scope of competition range is, and the greater difference the cluster sizes exhibit. When c is set to 0, EEUC just performs as an equal clustering algorithm and cannot balance the energy consumption among cluster heads very well. The number of clusters constructed in each round is determined by both R_0 and c. Intuitively, it decreases with the increase of R_0 when c is fixed, and it increases with the increase of c when R_0 is fixed. In order to balance the energy consumption well, R_0 and c should be properly set. Formulating the parameters to maximize the network lifetime is left for future work.

Finally, we give an explanation of the guideline for UCR to balance energy consumption of sensor nodes across the network. Due to smaller sizes of clusters in the hot spot, nodes are selected as cluster heads more frequently than these not in the hot spot. Energy holes may still appear though the intra-cluster load of cluster heads in hot spot could be reduced via unequal clustering. Our solution is that in each round, let heads in the hot spot consume less energy (intra- and inter-cluster cost in total) than these not in the hot spot, rather than consuming the same energy. This can be accomplished through decreasing the cluster size and inter-cluster transmission range of cluster heads in the hot spot simultaneously. As a rough example, suppose s_1 is a node in the hot spot, and it is selected as a head every 10 rounds; s_2 not in hot spot, as a head every 15 rounds. Suppose the energy consumption of s_1 in a round is 0.1J if s_1 is a head, and for s_2 0.15J. We ignore the energy consumption in rounds when it is an ordinary node. In every 30 rounds, s_1 and s_2 consume the same amount of energy: for s_1, it is 0.1J*3, and for s_2 it is 0.15J*2. So the energy hole is avoided. Although the explanation is not formalized, it serves as a reasonable guideline to balancing the load across the network.

5.2 Discussion

In this section, we discuss the design details for practical deployment of UCR. In Fig. 6, we see that UCR includes four time triggers: (1) cluster head selection trigger (T1), (2) cluster set-up trigger (T2), (3) intra-cluster communication trigger (T3), and (4) inter-cluster communication trigger (T4). These triggers are used to switch MAC protocols and schedule duty cycles.

First we describe the MAC mechanism together with the duty cycles schedule in various phases of the UCR protocol. T1 triggers the cluster heads competition phase. Non-tentative cluster heads turn off their radio to save energy at the moment. Tentative cluster heads exchange the control messages using the carrier-sense multiple access (CSMA) MAC protocol. T2 triggers waking up of non-cluster head nodes. The `CH_ADV_MSG`, the `JOIN_CLUSTER_MSG`, and the TDMA schedule message are transmitted using CSMA too. Note that the number of time slots in each cluster depends on the number of member nodes in the cluster. T3 triggers intra-cluster data transmission. Member nodes turn off their radio at all times except during their transmit time. To reduce inter-cluster interference, nodes in each cluster communicate using direct-sequence spread spectrum (DSSS). Each cluster uses a unique spreading code; all the nodes in the cluster transmit their data to the cluster head using this spreading code. Readers can refer to [31] for details about code assignments for each cluster. A cluster head turns off its radio once the cluster's TDMA time slots run out. T4 triggers waking up of all cluster heads. They transmit control messages and data packets using CSMA.

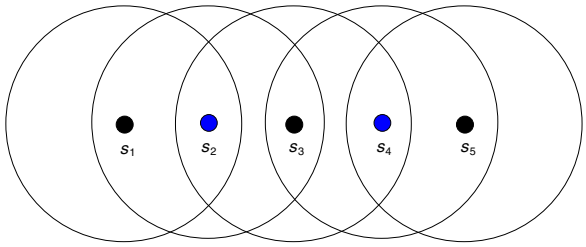

Fig. 6 A monotonic energy chain of five nodes

In each phase, an appropriate time interval should be chosen to let UCR run correctly. The time interval depends on the network scale and wireless channel quality.

In the waiting time between T1 and T2, the cluster head selection algorithm needs several message propagation steps to finish. In order to decide whether it is going to be a cluster head or an ordinary node, the tentative head s_i waits for the decision of each node x in its S_{CH} such that $x.RE > s_i.RE$. When T2 starts, all nodes who have not made their decisions exit the competing process immediately, and then final cluster heads broadcast their wills. Let's refer to Fig. 6 to gain an insight into the problem of waiting time. Suppose $s_1.RE < s_2.RE < s_3.RE < s_4.RE < s_5.RE$, i.e., they form a monotone incremental energy chain. The following events will happen one after another: first s_5 claims that it is a final cluster head, so s_4 quits the competition, then s_3 announces that it wins the competition too, so s_2 decides to be an ordinary node, and at last s_1 becomes a cluster head. It takes four message steps for s_1 to make its decision in such a chain of five nodes. The example shows that the waiting time depends on the length of the longest monotone energy chain. However, because the residual energy of tentative cluster

heads is distributed randomly, the longer a monotone energy chain is, the smaller the probability is. In [30], Basagni analyzes a similar problem and points out that the waiting time depends on the energy topology of the network rather than on the number of nodes in the network.

The length of time interval between T2 and T3 should allow the CH_ADV_MSG, JOIN_CLUSTER_MSG, and TDMA schedule messages to successfully propagate. T3 equals to the beginning of the first time slot for intra-cluster communication.

The length of the time interval between T3 and T4 depends on the maximum size of clusters in the network because each cluster member holds a TDMA time slot. The maximum size can be estimated with R_0 and the node density of the network.

Synchronization is important for the operation of UCR. We assume that all sensor nodes are synchronized and start the clustering phase at the same time. This could be achieved, for example, by having the base station periodically broadcast synchronization pulses. Readers can refer to [26] for further study about the time synchronization issue in clustered wireless sensor networks.

There is a cost in terms of energy and time to set up the clusters for UCR. If the clustering overhead is incomparable to the application packet load, clustering can be triggered every data-gathering round. For applications where all nodes are continuously sending reports, however, frequent cluster rotation will cause the system to always be in an unstable state which might lead to data losses and delayed response. Therefore, there is a trade-off in determining how long to make the steady-state phase, and it is application specific.

6 Simulation results

In this section, we evaluate the performance of UCR via simulations. First we study the cluster head characteristics of the unequal clustering algorithm, and then we investigate the parameter settings and the energy efficiency of UCR in terms of the network lifetime. Because this paper focuses on energy efficient routing in the network layer, an ideal MAC layer and error-free communication links are assumed for simplicity. For these simulations, energy is consumed whenever a sensor transmits or receives data or performs data aggregation. Because HEED is the most similar self-organized clustering protocol, we use it for comparison. For HEED, we use the average minimum reachability power (AMRP) [9] as the intra-cluster communication cost function. The simulation parameters are given in Table 1, in which the parameters of radio model are the same as those in LEACH [7]. It is worth mentioning that we assume a rectangle network field other than a square field. Under this model longer multihop paths to the base station are produced, which helps evaluate the effects of the UCR protocol to mitigate the severe hot spot problem. Unless otherwise specified, we set T to 0.2, R_0 to 80 m, c to 0.3 in Eq. (3), TD_MAX to 200 m, and k to 2. Actually, these values are found through simulations described below.

Table 1 Simulation parameters

Parameter	Value
Network field	(0,0)–(400,200) m
Base station location	(500,100) m
N	600
Initial energy	1 J
E_{elec}	50 nJ/bit
ϵ_{fs}	10 pJ/bit/m^2
ϵ_{mp}	0.0013 pJ/bit/m^4
d_o	87 m
E_{DA}	5 nJ/bit/signal
Data packet size	4000 bits

6.1 Cluster head characteristics

As we have explained in the previous section, the number of selected cluster heads varies according to the specified R_0 and c. Figure 7 shows the average number of cluster heads generated by UCR. For a fixed value of c, the number of clusters decreases as the value of R_0 increases. Notice that when R_0 is fixed and c increases, the competition radius decreases accordingly, thus UCR generates more clusters as shown in the figure. It testifies our analysis, i.e., the smaller the competition radius, the larger the required number of clusters to cover the network. Since each cluster head is responsible for aggregating the data from its cluster members into a single length-fixed packet, only one data packet needs to be delivered to the base station out of a cluster. Thus the more clusters are present, the more messages need to

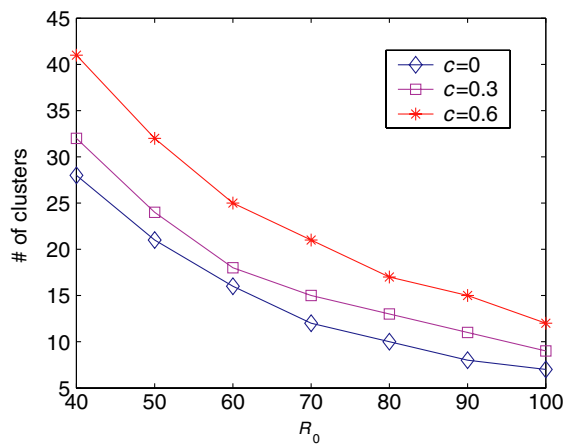

Fig. 7 The number of clusters

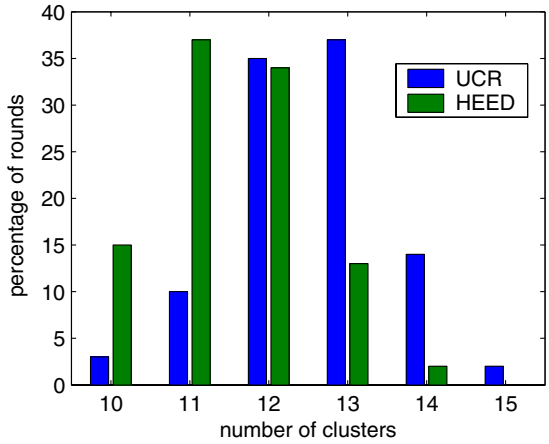

Fig. 8 Distribution of the number of clusters

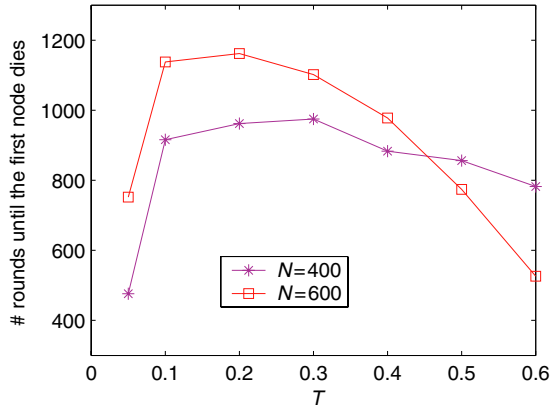

Fig. 9 The network lifetime with different T

be delivered to the base station, resulting in overall energy consumption increases. In LEACH [7], the authors give an estimation of the optimum number of clusters in single hop clustered networks. However, it cannot be applied to the unequal cluster-based routing protocol proposed in this work. We give simulation results of the effect of R_0 and c on the network lifetime later on.

Next we examine the stability of our clustering algorithm. Figure 8 shows the distribution of the number of clusters in UCR, HEED. The percentage is calculated from 100 randomly selected rounds of the simulation. In this scenario the cluster radius of HEED is also set to R_0 too. UCR generates 13 or 14 cluster heads in 70% rounds, and HEED also generates 12 or 13 cluster heads in 70% rounds. Thus the number of clusters in both UCR and HEED is steady. For UCR, a certain proportion of nodes voluntarily join the cluster head competition, thus the number of selected cluster heads won't be too small. On the other hand, according to Lemma 2 the number of selected heads won't be too large. As a matter of fact, the number of clusters in UCR depends on the competition range of all tentative cluster heads. Thus UCR achieves a steady number of clusters. For HEED, the probability that two nodes within each other's cluster range are both cluster heads is small. Therefore the clustering approach also generates a steady number of clusters. It is worth mentioning that UCR generates more clusters than HEED does because the unequal clustering method produces more cluster heads in the area closer to the base station to afford the forwarding traffic.

6.2 Parameter setting

There are several parameters in UCR, namely T, R_0, c, TD_MAX, and k. In this part we study the parameter setting with regard to the network lifetime. We measure the lifetime in terms of the data gathering rounds when the first node dies, because a certain area cannot be monitored any more once a sensor node exhausts its energy.

First, we examine the effect of T on the network lifetime. We add a sparse network scenario in which the number of sensor nodes N is decreased to 400. As T varies from 0.05 to 0.6, Fig. 9 shows the relation between T and the network lifetime in the two scenarios. There is an optimal range for the value of T, i.e., 0.1–0.3. According to the explanation about T in Section 5, the value of T should be properly chosen to guarantee the cluster heads quality and reduce the message overhead. Another point to be mentioned is that for the dense network a smaller T is preferred. When T increases from 0.3 to 0.6, the network lifetime of the dense network ($N = 600$) decreases more dramatically than that of the sparse network ($N = 400$). The reason is that it suffers serious message overhead according to Lemma 1.

Second, we investigate the impact of the parameters R_0 and c on the network lifetime. We observe the network lifetime for different settings of R_0 and c, and the result is shown in Fig. 10. It suggests that there is a tradeoff among R_0, c, and the network lifetime. On one hand, R_0 is the dominant factor that impacts the network lifetime. The reason is that

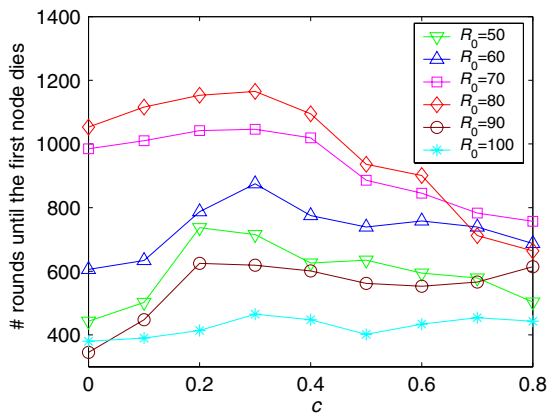

Fig. 10 The network lifetime with different R_0 and c

the number of clusters in a given network scale is mainly determined by R_0. When R_0 is set to 80 m, the network lifetime is prolonged furthest. On the other hand, it shows that the unequal clustering method can extend the network lifetime. As we have explained previously, c determines the difference of cluster sizes. Under each setting of R_0, we can see the network lifetime varies as c varies. When c is set to 0, EEUC performs as an equal clustering approach. When c increases from 0, the energy consumption becomes gradually balanced among cluster heads, therefore the network lifetime increases. However, the lifetime decreases when c is too large. This is because too many clusters will be produced closer to the base station, and each of them will deliver a data packet to the base station, which causes a waste of energy. Therefore, there exists an optimal value of c for a given R_0 that could best extend the network lifetime. Under the network scale of this simulation, 0.3 is approximately the optimal value of c for R_0 among 60 m and 80 m.

Third, we study the impact of R_0 and TD_MAX on the network lifetime. TD_MAX determines the area where nodes should communicate with the base station directly. Since the cost of forwarding packets from other clusters to the far away base station is considerably high, the size of this area should be properly set to save and balance the energy consumption. We observe the network lifetime for different settings of R_0 and TD_MAX, and the result is shown in Fig. 11. Similar to Fig. 10, it also demonstrates that R_0 plays a critical role in the network lifetime. We also see that the network lifetime varies as TD_MAX varies, and there exists an optimal value of TD_MAX for a given R_0. Under the network scale of this simulation, 200 m is approximately the optimal value of TD_MAX for R_0 among 60 m and 80 m. If TD_MAX becomes larger, too many cluster heads directly send their data to the base station, resulting in a waste of energy. On the other hand, if TD_MAX becomes smaller, the average load of cluster heads in the direct communication area is too high, resulting in premature creation of energy holes in that area.

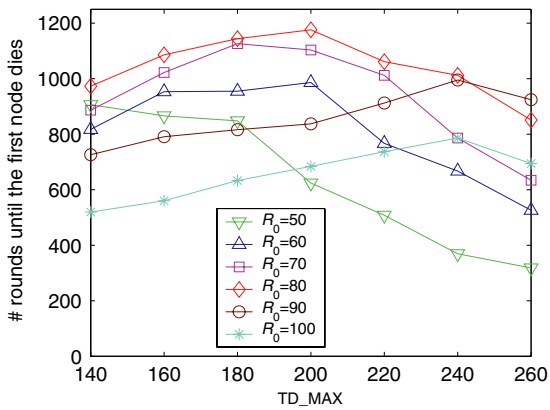

Fig. 11 The network lifetime with different TD_MAX

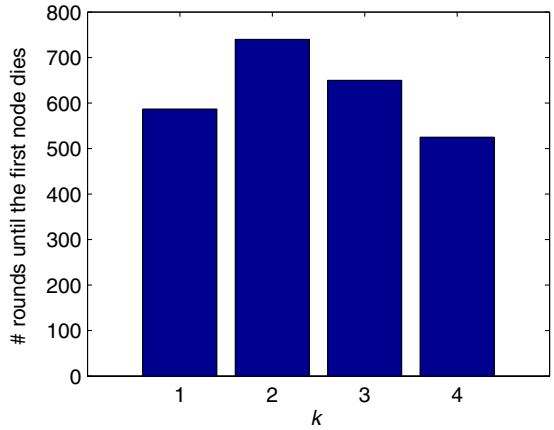

Fig. 12 The network lifetime with different k

Finally, we study the effect of k in the inter-cluster routing protocol on the network lifetime. In the routing algorithm, node s_i chooses as its relay node the neighbor that has the biggest residual energy among the k smallest E_{relay} neighbors. To produce a dense cluster head backbone for the validation of multihop routing algorithm, R_0 is set to 50 m in this simulation to generate more clusters. Figure 12 shows the relation between k and the network lifetime. When k is set to 1, the routing algorithm is just a greedy geographic forwarding approach that doesn't consider the relay node's residual energy. When k becomes larger, the routing algorithm considers the relay node's residual energy more for load balance. Figure 12 illustrates the improvement of the network lifetime. When k is larger than 3, however, the network lifetime is lower than that of the circumstance when $k = 1$. It suggests the importance of decreasing the total energy cost per packet. In conclusion, the tradeoff in the routing protocol is indeed effective in extending the network lifetime.

6.3 Energy efficiency

In this part, we evaluate the energy efficiency of UCR. First, we compare the network lifetime of UCR and HEED. A similar inter-cluster multihop routing protocol is implemented for HEED as described in [27], in which the link estimation module is omitted because we assume an ideal wireless channel. We run extensive experiments to determine the optimal cluster radius for HEED. Figure 13 shows the number of sensor nodes still alive over the simulation time. UCR clearly improves the network lifetime (both the time until the first node dies and the time until the last node dies) over HEED. In HEED, tentative cluster heads are randomly selected based on their residual energy. Therefore, sensors with low residual energy can still become cluster heads because it uses the intra-cluster communication cost to select final cluster heads. Furthermore, the energy consumption of cluster

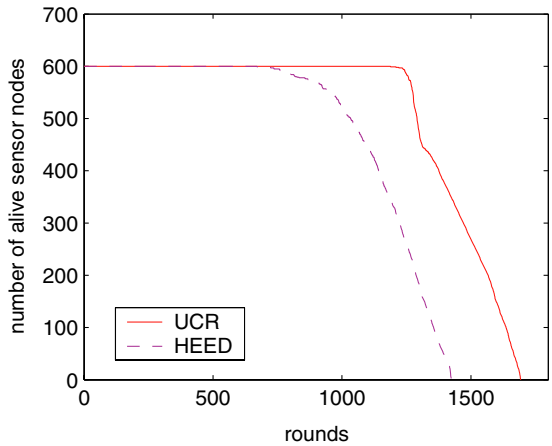

Fig. 13 The number of alive sensor nodes over time

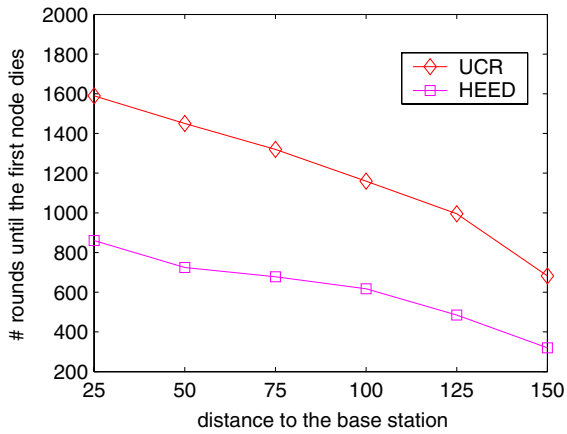

Fig. 14 The network lifetime as the base station travels farther

heads is not well-balanced. Thus nodes in the hot spot die much faster in HEED. This is avoided in UCR because the unequal cluster-based routing protocol aims to mitigate the hot spot problem, thus energy consumption is well-balanced among nodes. The small interval between the time until the first node dies and the time until the last node dies implies that UCR has successfully mitigated the hot spot problem.

Second, we study the impact of the distance to the base station on the network lifetime. We fix the y-coordinate of the base station and adjust its x-coordinate. The distance is computed from the base station to the closest point of the network field. Under each setting of the distance, an optimum TD_MAX is chosen to extend the network lifetime. We measure the number of rounds until the first node dies in UCR and HEED, and the result is shown in Fig. 14. Although the network lifetime severely deteriorates as the distance increases, UCR achieves about 2 $\times$ network lifetime as that in HEED for all base station locations we simulated. It suggests that UCR is more energy-efficient than HEED though they both use multihop routing in inter-cluster communication.

7 Conclusions and future work

In this paper, we have introduced a novel unequal cluster-based routing protocol for wireless sensor networks. The hot spot problem arises when employing the multihop routing in a clustered sensor network. We argue that both the rotation of cluster heads and the metric of residual energy are not sufficient to balance the energy consumption across the network. To address the problem, we first introduce an unequal clustering algorithm. Cluster heads closer to the base station have smaller cluster sizes than those farther from the base station, thus they can preserve some energy for the purpose of inter-cluster data forwarding. What is more, we propose an energy-efficient multihop routing protocol for the inter-cluster communication. Simulation results show that UCR clearly improves the network lifetime over HEED.

We assume that a sensor node can compute the approximate distance to another node based on the received beacon signal strength, if the transmitting power is known. However, error will arise due to the noise in the real network environments. As a future work, we plan to extend the method to increase its robustness.

Acknowledgments The preliminary version of this paper has been presented at IEEE MASS'05. We thank the anonymous reviewers for their valuable comments. The work is supported by China 973 project (2006CB303004), National NSF (60573131, 60673154), China Jiangsu Provincial NSF grant (BK2005208) and TRAPOYT award of China Ministry of Education, and US NSF grants CCR 0329741, CNS 0422762, and CNS 0434533.

References

1. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, A survey on sensor networks, IEEE Communications Magazine 40 (August 2002) 102–114.
2. A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler and J. Anderson, Wireless sensor networks for habitat monitoring, in: *Proceedings of ACM Workshop on Wireless Sensor Networks and Applications* (September 2002) pp. 88–97.
3. B. Krishnamachari, D. Estrin and S. Wicker, The impact of data aggregation in wireless sensor networks, in: *Proceedings of IEEE International Conference on Distributed Computing Systems Workshops* (July 2002) pp. 575–578.
4. W. Heinzelman, A. Chandrakasan and H. Balakrishnan, Energy-efficient communication protocols for wireless microsensor networks, in: *Proceedings of the 33rd Hawaiian International Conference on Systems Science* (January 2000).
5. V. Mhatre and C. Rosenberg, Design guidelines for wireless sensor networks: communication, clustering and aggregation, Ad Hoc Networks 2(1) (2004) 45–63.
6. K. Akkaya and M. Younis, A survey on routing protocols for wireless sensor networks, Ad Hoc Networks 3(3) (2005) 325–349.
7. W. Heinzelman, A. Chandrakasan and H. Balakrishnan, An application-specific protocol architecture for wireless microsensor networks, IEEE Transactions on Wireless Communications 1(4) (2005) 660–670.

8. W. Choi, P. Shah and S.K. Das, A framework for energy-saving data gathering using two-phase clustering in wireless sensor networks, in: *Proceedings of International Conference on Mobile and Ubiquitous Systems* (August 2004) pp. 203–212.
9. O. Younis and S. Fahmy, HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks, IEEE Transactions on Mobile Computing 3(4) (2004) 660–669.
10. M. Qin and R. Zimmermann, An energy-efficient voting based clustering algorithm for sensor networks, in: *Proceedings of 1st ACIS Workshop on Self-Assembling Wireless Networks* (May 2005).
11. J.S. Liu and C.H. Lin, Energy-efficiency clustering protocol in wireless sensor networks, Ad Hoc Networks 3(3) (2005) 371–388.
12. M. Ye, C. Li, G. Chen and J. Wu, An energy efficient clustering scheme in wireless sensor networks, Ad Hoc & Sensor Wireless Networks to appear.
13. S. Soro and W. Heinzelman, Prolonging the lifetime of wireless sensor networks via unequal clustering, in: *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (April 2005).
14. T. Shu, M. Krunz and S. Vrudhula, Power balanced coverage-time optimization for clustered wireless sensor networks, in: *Proceedings of the Sixth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)* (May 2005).
15. M. Perillo, Z. Cheng and W. Heinzleman, An analysis of strategies for mitigating the sensor network hot spot problem, in: *Proceedings of the Second International Conference on Mobile and Ubiquitous Systems* (July 2005).
16. S. Olariu and I. Stojmenovic, Design guidelines for maximizing lifetime and avoiding energy holes in sensor networks with uniform distribution and uniform reporting, in: *Proceedings of IEEE INFOCOM* (April 2006).
17. J. Lian, K. Naik and G. Agnew, Data capacity improvement of wireless sensor networks using non-uniform sensor distribution, International Journal of Distributed Sensor Networks to appear.
18. E.I. Oyman and C. Ersoy, Multiple sink network design problem in large scale wireless sensor networks, in: *Proceedings of the International Conference on Communications* (June 2004).
19. S.R. Gandham, M. Dawande, R. Prakash and S. Venkatesan, Energy efficient schemes for wireless sensor networks with multiple mobile base stations, in: *Proceedings of IEEE IEEE Global Telecommunications Conference* (December 2003) pp. 377–381.
20. Z.M. Wang, S. Basagni, E. Melachrinoudis and C. Petrioli, Exploiting sink mobility for maximizing sensor networks lifetime, in: *Proceedings of the 38th Hawaii International Conference on System Sciences* (January 2005).
21. W. Wang, V. Srinivasan and K.C. Chua, Using mobile relays to prolong the lifetime of wireless sensor networks, in: *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking (MobiCom)* (August 2005) pp. 270–283.
22. G.G. Finn, Routing and addressing problems in large metropolitan-scale internetworks, *Technical Report ISI/RR-87-180* Information Sciences Institute (ISI) (1987).
23. E. Kranakis, H. Singh and J. Urrutia, Compass routing on geometric networks, in: *Proceedings of the 11th Canadian Conference on Computational Geometry* (August 1999) pp. 51–54.
24. B. Karp and H.T. Kung, GPSR: greedy perimeter stateless routing for wireless networks, in: *Proceedings of the 6th ACM/IEEE Annual International Conference on Mobile Computing and Networking (MobiCom)* (August 2000) pp. 243–254.
25. H. Frey and I. Stojmenović, Geographic and energy-aware routing in sensor networks, in: *Handbook of Sensor Networks: Algorithms and Architectures* (Wiley, 2005).
26. O. Younis and S. Fahmy, A scalable framework for distributed time synchronization in multi-hop sensor networks, in: *Proceedings of the Second IEEE International Conference on Sensor and Ad Hoc Communications and Networks* (September 2005) pp. 13–23.
27. O. Younis and S. Fahmy, An experimental study of routing and data aggregation in sensor networks, in: *Proceedings of the International Workshop on Localized Communication and Topology Protocols for Ad hoc Networks* (November 2005) pp. 50–57.
28. M. Ye, E. Chan and G. Chen, On mitigating hot spots for clustering mechanisms in wireless sensor networks, in: *Proceedings of the Third IEEE International Conference on Mobile Ad-hoc and Sensor Systems* (October 2006).
29. V. Kawadia and P.R. Kumar, Power control and clustering in ad hoc networks, in: *Proceedings of IEEE INFOCOM* (April 2003) pp. 459–469.
30. S. Basagni, Distributed clustering for ad hoc networks, in: *Proceedings of the 4th International Symposium on Parallel Architectures, Algorithms, and Networks* (June 1999) pp. 310–315.
31. L. Hu, Distributed code assignments for CDMA packet radio networks, IEEE/ACM Transactions on Networking 1(6) (1993) 668–677.

Guihai Chen obtained his B.S. degree from Nanjing University, M. Engineering from Southeast University, and PhD from University of Hong Kong. He visited Kyushu Institute of Technology, Japan in 1998 as a research fellow, and University of Queensland, Australia in 2000 as a visiting professor. During September 2001 to August 2003, he was a visiting professor at Wayne State University. He is now a full professor and deputy chair of Department of Computer Science, Nanjing University. Prof. Chen has published more than 100 papers in peer-reviewed journals and refereed conference proceedings in the areas of wireless sensor networks, high-performance computer architecture, peer-to-peer computing and performance evaluation. He has also served on technical program committees of numerous international conferences. He is a member of the IEEE Computer Society.

Chengfa Li was born 1981 and obtained his Bachelor's Degree in mathematics in 2003 and his Masters Degree in computer science in 2006, both from Nanjing University, China. He is now a system programmer at Lucent Technologies Nanjing Telecommunication

Corporation. His research interests include wireless ad hoc and sensor networks.

Mao Ye was born in 1981 and obtained his Bachelor's Degree in computer science from Nanjing University, China, in 2004. He served as a research assistant at City University of Hong Kong from September 2005 to August 2006. He is now a PhD candidate with research interests in wireless networks, mobile computing, and distributed systems.

Jie Wu is a professor in the Department of Computer Science and Engineering at Florida Atlantic University. He has published more than 300 papers in various journal and conference proceedings. His research interests are in the areas of mobile computing, routing protocols, fault-tolerant computing, and interconnection networks. Dr. Wu serves as an associate editor for the IEEE Transactions on Parallel and Distributed Systems and several other international journals. He served as an IEEE Computer Society Distinguished Visitor and is currently the chair of the IEEE Technical Committee on Distributed Processing (TCDP). He is a member of the ACM, a senior member of the IEEE, and a member of the IEEE Computer Society.