Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computer Communications 28 (2005) 1293–1302

computer communications

www.elsevier.com/locate/comcom

Cell based energy density aware routing: a new protocol for improving the lifetime of wireless sensor networks

Jae Young Choi*, Hyung Seok Kim[1], Iljoo Baek, Wook Hyun Kwon

School of Electrical Engineering and Computer Science, Seoul National University, San 56-1, Shilimdong, Kwanakgu, Seoul 151-744, South Korea

Received 2 May 2003; revised 8 November 2004; accepted 24 November 2004
Available online 8 December 2004

Abstract

Wireless sensor networks have unique features not shared by mobile ad-hoc networks. Taking these features into consideration, we propose a new routing protocol specifically designed for wireless sensor networks. This protocol, referred to as cell based energy density aware routing (CEDA), divides a sensor field into uniform cells, thereby reducing energy consumption caused by sensor data flooding. Energy density, a novel routing metric, is used to avoid forwarding packets to subareas whose nodes have lower residual energies. By ensuring fair energy consumption of sensor nodes, CEDA makes it possible for monitoring stations to monitor all subareas for longer periods of time. Simulations were carried out to compare the performance of CEDA with those of several existing protocols for wireless networks or sensor networks. The time required for a subarea to run out of energy, called the lifetime of that area, is measured in the simulations. The simulation results show that CEDA gives a longer lifetime than the existing routing protocols. In addition, it is proved that CEDA guarantees the maximum hop count regardless of the node density, and hence does not suffer from unpredictable delays.

Keywords: Wireless sensor networks; Lifetime; Energy density; Routing protocol; Cell; Hop count

1. Introduction

Recent advances in wireless communication and micro-sensor technologies have made it possible to develop wireless networks of low-cost, small-sized sensor nodes [1]. These sensor nodes can be deployed in physical environments to collect information such as acoustic, light, and seismic data, in an autonomous manner [2]. This technology has potential applications in diverse areas, including surveillance, rescue efforts in disaster areas, and military intelligence collection.

An important feature of wireless sensor networks is that the sensor nodes are equipped with batteries that *cannot* be recharged. The sensor nodes have this feature because they are intrinsically immobile; typically they are embedded in physical structures or thrown into inhospitable terrain and then left *unattended*. Thus, wireless sensor networks differ from mobile ad-hoc networks (MANET), which usually have rechargeable and replaceable batteries. Given that battery failure leads to failure of the sensor node, which plays the dual role of data originator and data router [1], the depletion of node batteries in an area can potentially cause significant topological changes within the network due to broken connections. Such changes can lead to partitioning of the network into completely disconnected areas. In addition, if a subarea has no live sensor node due to battery depletion, creating a so-called void subarea, users cannot access information on the subarea and paths to users that pass through the subarea may be cut off. Sensor network users wish to keep receiving information from all subareas for the longest possible time.

Energy consumption by sensor nodes can be divided into three domains: sensing, communication, and processing. Here, we concentrate on energy consumption for communication because this domain is known to consume

* Corresponding author. Address: Engineering Research Center for Advanced Control and Instrumentation, Seoul National University, San 56-1, Shilimdong, Kwanakgu, Seoul 151-744, South Korea. Tel.: +82 2 873 2279.

E-mail addresses: jychoi@cisl.snu.ac.kr (J.Y. Choi), hskim@cisl.snu.ac.kr (H.S. Kim).

[1] Tel.: +82 1099814563; fax: +82 28788933.

0140-3664/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2004.11.005

the greatest amount of power in modern processors. For example, Berkeley motes [23] consume 8 mJ to transmit and 4 mJ to receive a byte, whereas the CPU requires about 0.8 mJ to execute 208 cycles (roughly 100 instructions). Thus, an energy-aware routing protocol that prevents energy consumption from concentrating in a few subareas (i.e. fair energy consumption) is required so that network connectivity is enhanced and access to some areas is not lost. Energy-aware routing protocols [3,4] for MANETs find a single optimal path for low energy consumption. In sensor networks, sensor nodes usually report data repeatedly to one or a few monitoring stations, known as *sinks*, along the path. Nodes along the path thus tend to consume more energy than other, idle nodes. Moreover, if multiple paths are used, nodes at the points of overlap among the paths will more quickly run out of energy, which may eventually lead to the creation of disconnected subnets. Another problem of existing power- or energy-aware routing protocols is that they do not take into account the delay from a source to a sink. If wireless sensor networks are constructed densely with a large number of nodes, packets can traverse the network in many short hops in order to save energy. However, such routing protocols entail unfeasibly long delays, and the addition of new sensor nodes to complement an existing sensor field will cause these delays to increase.

To tackle the problems outlined above, we propose the cell based energy density aware routing (CEDA), a routing protocol that uses *energy density* as a routing metric and is tailored to the unique features of wireless sensor networks. When the sensor field is divided into small subareas called cells, the energy density is defined as the weighted sum of the energies of all nodes in the cell. In each cell, a sensor node with a higher level of residual energy relays packets on behalf of other nodes in the same cell. This sensor node also forwards packets to the neighbor cell with the greatest residual energy density. This protocol ensures that energy is consumed more fairly by the sensor nodes, and thus that the sensor network runs longer. The lifetime of a cell is defined as the time required for all nodes in the cell to lose all of their energy. Increasing the cell lifetime decreases the frequency with which void cells appear, and thus helps to maintain the connectivity of the network and the ability to supervise all subareas. Because sensor nodes are intrinsically immobile and are used at densities several orders of magnitude higher than in MANETs, the CEDA is based on a *cellular addressing* scheme to group the sensor nodes. The cellular addressing approach has less overhead than a global addressing or position-based addressing [13] because it groups several nodes in the same area. Cellular addressing reduces communication burden due to sensor flooding. The CEDA routing protocol also guarantees the maximum hop count regardless of the node density of the network.

The remainder of this paper is organized as follows. Section 2 reviews related studies and compares other routing protocols with the CEDA protocol. The cellular addressing method and the basic service model of CEDA are introduced in Section 3. Section 4 provides a detailed description of the CEDA algorithms. Section 5 presents proof that CEDA guarantees the maximum hop count. A comparative performance evaluation using simulation is presented in Section 6. Section 7 concludes the paper.

2. Related work

Directed diffusion is a data-centric communication paradigm specifically designed for sensor networks [5]. Sinks use flooding to spread interests to the sensor network. Sensors matching an interest send their data to the sinks along multiple paths initially, and then gradually reinforce better paths. The use of multiple paths and the flooding of packets are likely to consume large amounts of energy, and do not take into account fair energy consumption. TTDD [20] is an overlay network protocol that exploits local flooding within a local cell of a grid that is proactively built by the sources. It exploits a geographic routing as the routing protocol. The source disseminates data along the nodes on the grid line to the sinks. SPEED is a routing protocol [22] designed for sensor networks, supplementing geographic routing protocols such as GPSR. It forwards packets to the neighbor nodes, with a packet delay that is proportional to the distance between the forwarding and neighbor nodes. As a result, the per-hop delay approaches a set value. Neither TTDD nor SPEED considers energy consumption or residual energy. Multi-path routing schemes designed for energy conservation have been proposed [17]. If multiple paths share the same intermediate nodes, the batteries of the sensors at those nodes are depleted faster.

Several studies have investigated energy-aware routing protocols for gathering data from a collection of sensor nodes [6,7]. All of these studies have used *data fusion* (also known as *data aggregation*) to reduce energy consumption. Two protocols incorporating data aggregation that have been developed are LEACH [6] and PEGASIS [7]. LEACH [6] partitions a sensor network into clusters and a randomly chosen node in each cluster aggregates the data of the cluster and transmits the aggregated data to the sinks. PEGASIS [7] is an improved version of LEACH that places the sensor nodes in a chain. By using a token passing approach, each node receives data from and transmits data to its close neighbors. LEACH and PEGASIS are based on a model in which every node is within the transmission range of every other node. However, a sensor network field may be much wider than the maximum transmission range of the sensor nodes within the network. Therefore, sensor networks are usually multi-hop networks and hence the network model presented here is a multi-hop sensor network.

The sensor network is closely related to works on the wireless ad-hoc network. Numerous routing protocols have been proposed for wireless ad-hoc routing. However, ad-hoc network paradigms are not directly applicable to sensor networks because sensor networks have unique features not shared by ad-hoc networks.

Wireless ad-hoc network routing protocols can be categorized as either proactive [8] or reactive [9–11]. A typical example of a proactive routing protocol is destination-sequenced distance-vector (DSDV) routing [8]. In DSDV, each node continuously computes the cost of reaching every other node via the shortest path. Routing tables are periodically updated for consistency throughout the entire network. However, sensor networks are poorly suited to maintaining routing tables and transmitting frequent routing advertisements due to their special features such as many nodes, small memory, and limited battery energy.

Reactive routing protocols only initiate a route computation process when a packet needs to be sent to some other node. A comparative study [11] of proactive and reactive routing in simulated ad-hoc networks found that the reactive protocols introduced less communication overhead. One reactive protocol that has proved effective in wireless ad-hoc networks is dynamic source routing (DSR) [9]. In this protocol, the source routing finds the shortest path and each data packet carries the list of routers in that path. A route request packet is repeatedly flooded to the network until it reaches its destination. This approach may, however, prove impossibly unwieldy for sensor networks, which tend to consist of numerous nodes. In such large networks, packets must pass through many nodes, making it impractical for each data packet to contain a list of all routers. Also, the source routing in which sensor nodes repeatedly broadcast their requests is energy inefficient. Another well-known reactive protocol is the ad-hoc on-demand distance-vector (AODV) routing [10]. In this source routing protocol, a source node broadcasts a route request when it requires a route to a destination. However, the large numbers of sensor nodes in sensor networks mean that the AODV approach may still cause communication overhead and large energy consumption. The routing protocols described above do not include a metric for energy conservation or residual energy.

Several routing protocols based on the geographic locations of the nodes have been developed for wireless ad-hoc networks. In geographic routings, a router determines a path by using information on the geographic locations of nodes. Karp [12] and Ko [19] proposed localized routings based on geographic information. Their routing protocols forward a packet to a neighbor if among all neighbor nodes, the neighbor has the shortest geographic distance to the destination, provided this distance is less than the distance from the forwarding node to the destination.

Stojmenovic [14] suggested a localized routing for wireless ad-hoc networks that focused on minimizing the transmission power by considering the geographic distance of each hop. In this protocol, the node having a data packet forwards the data to a neighboring node with the minimized transmission power. In sensor networks with densely deployed nodes, however, this scheme may incur a delay because the data packet must traverse many short hops. Repetitive reporting to sinks through a fixed path causes more rapid depletion of the energy of nodes in a specific subarea. Another tool that has been developed to minimize power consumption is geographical adaptive fidelity (GAF) [18], which identifies nodes that are equivalent from a routing perspective and then turns off unnecessary nodes in order to decrease energy loss due to overhearing. GAF is not a routing algorithm but a medium access control (MAC) algorithm, and hence it should be combined with routing algorithms for wireless ad-hoc networks.

The method proposed here, CEDA, is an energy-aware routing algorithm that uses a cell-based grouping for the sensor network. It groups a set of sensor nodes inside the cell and evades the lower energy density of regions.

3. Addressing and service model

3.1. Addressing

Traditional networks assign unique global addresses (e.g. IP addresses of the Internet) to nodes. Sensor nodes, however, may not have global identification addressing because of the large amount of overhead and the large number of nodes in a sensor network [1]. The user or monitoring station typically requires information from a specific local area. Thus, it is helpful that the nodes are identified using their geographic positions. In fact, this geographic information can be exploited in the routing protocol to guide communication pathways.

CEDA uses a geographic addressing wherein a cell specifies a geographic subarea in space where sensor nodes are deployed. Sensor nodes in the same cell are grouped and each of them is distinguished using an address (x, y, n), where x and y are the coordinates of the cell and n is the local address assigned to nodes in the cell. For example, if there are 10 nodes in cell (x, y), the addresses of the nodes within the cell will be $(x, y, 1)$, $(x, y, 2)$,…, and $(x, y, 10)$.

When sensors are deployed by an aircraft, the nodes will tend to be located at random positions rather than in a uniform arrangement. Under these circumstances, a cell may contain no nodes. Fig. 1 shows an example of a network of randomly deployed sensor nodes and their cell group addressing. Under position-based addressing [12], if the distance between two sensor nodes is very small as shown in Fig. 2, they may have the same address. Each sensor node is assumed to be aware of its own geographic location. The network can use location services such as those reported in [15,16] to estimate the locations of the individual nodes. These location estimation techniques do not require a GPS at every node. Class-based addressing [17] may also assign the same addresses to sensors of equivalent type at similar locations. In cellular addressing, however, a pair of nodes of the same type at similar locations are assigned the distinct addresses (x, y, m_1)

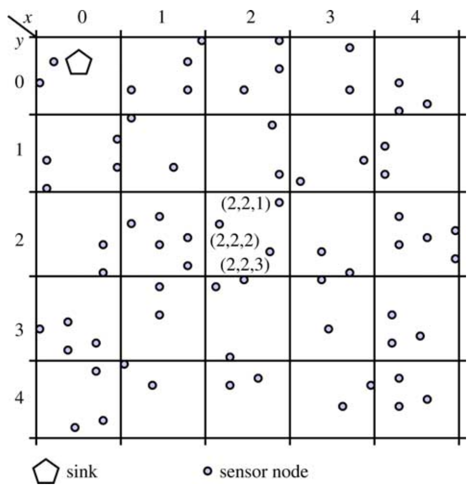

Fig. 1. An example of cellular addressing in a network of randomly deployed sensor nodes.

and (x, y, m_2). This approach to node assignment prevents multiple sources with the same address from sending redundant data.

Wireless communication of sensor nodes is bidirectional and homogeneous, and hence their radio ranges are assumed to be equal. Let L denote the proximity level of the cells. At $L=1$, a cell has a total of eight adjacent cells, and at $L=2$ it has 24 adjacent cells. The number of cells G that are reachable by a single hop under proximity level L is computed as

$$G = 4L(L+1). \tag{1}$$

Let u_x denote the unit horizontal length of a cell and u_y be the unit perpendicular length of a cell, as shown in Fig. 2. To ensure that any node in a given cell can communicate with every other node in the G cells within level L, u_x and u_y must satisfy the following

$$u_x^2 + u_y^2 \le r_{\max}^2/(L+1)^2 \tag{2}$$

where $r_{\max}$ is the maximum distance that the sensor node can transmit a data packet. When $L=1$, $r_{\max}$ should be larger than the distance between nodes a and b in Fig. 2. When $L=2$, $r_{\max}$ should be larger than the distance between a and c. For example, when $r_{\max}=20$ m, Eq. (2) indicates that if L is set to one, then $u_x=5$ m and $u_y=5$ m are acceptable choices as the unit lengths of the cell. Under these settings, an arbitrary cell can communicate with eight adjacent cells. However, if $u_x=8$ m and $u_y=7$ m are chosen, a given cell cannot communicate with all of the adjacent cells.

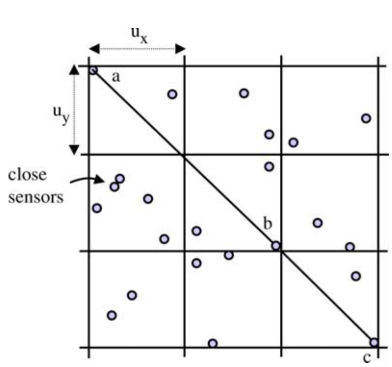

Fig. 2. Neighbor level and unit length of the cells.

When sensors are initially deployed, the local addresses of the nodes in each cell are determined by the id of each node (i.e. (x, y, n), where n is the node id). The neighbor cells of cell c are defined as adjacent cells in which all nodes are within radio range of cell c. Among the sensor nodes within each cell of $u_x \times u_y$, one node is selected as a router node to route packets to neighbor cells. This router node also relays packets for the other sensor nodes in the same cell. The exceptions to this rule are source nodes and sinks, which are not required to relay data packets via the routing node. This scheme avoids unnecessary energy consumption.

Each node has an attribute table containing the attributes of the sensors in its cell. Query packets for discovering the route are received by the router node that has the attributes of other nodes in the cell. When the router node receives a packet, it looks up the attribute table and relays the packet to the nodes matching the query. This scheme dramatically reduces communication overhead caused by packet flooding in attribute-based routing protocols. The node with the highest energy level is selected as the router node by comparing the residual energy levels of all nodes in the cell. However, frequent exchanges of packets giving current energy levels may cause excessive communication overhead. To avoid this problem, the residual energy of each node is expressed by a small number of energy levels. Whenever the energy level of a particular node changes, the node advertises its new level to the other nodes in its cell and they update their energy tables. Each node has an energy table listing the energy levels of its neighbor nodes. This table includes the information (*cell* (x, y), *local id*, *energy level*). The router node always has a node id of 1 (i.e. (x, y, 1)). The node ids are used to break ties of their energy levels. If the router node changes its energy level, it can be replaced with another node having more residual energy. If it is replaced by another node, the new and old router nodes swap node id values such that the new router node has the address (x, y, 1). Hence, the cell does not have to concern about a change of the router node of neighbor cells due to its failure or battery depletion that represents another advantage of the cellular addressing approach.

3.2. Service model

A sensor network comprises a number of sensor nodes and one to several monitoring stations as sinks connected by wireless links. It is assumed that the wireless links are bidirectional and all the sensor nodes have the same transmission range. The sensor nodes are randomly deployed over an area. Each sensor monitors its vicinity and generates information. In most scenarios currently envisioned, the sensor nodes are immobile and the sink(s) can be fixed or moveable monitoring stations. We assume that moveable sinks move slowly, and thus that the communication overhead associated with searching for the moving sink is negligible. When a moving sink changes its position to another cell, it will send a new query for routing.

Sources should send messages to the sink at regular intervals or whenever an event occurs. Sensor network applications of the sink may submit queries through a set of query service APIs. The API provides high-level abstraction to applications by hiding the specific location and status of each sensor node. The sink sends a query including attributes to obtain information from the sensor nodes. The sensor nodes matching the query sent from the sink then send sensing data to the sink. The query may be of the form *query* (*attribute*, *interest area*, *query generator loc*). If, for example, the query is 'Send me average temperature data in an area (x_1, y_1, x_2, y_2)', the *attribute* is the average temperature and the *interest area* of the query is the area (x_1, y_1, x_2, y_2). The *query generator loc* is the location of the sink sending the query. Because the query contains the location of the sink, the sensing data can be sent back using a geographic routing protocol. If necessary, the query may include an update period.

The battery of each sensor node (x, y, n) has a finite energy $E(x, y, n)$ that cannot be replenished. Whenever the sensor node transmits or receives a data packet, it consumes energy from the battery. In contrast, sinks are typically managed by users (e.g. a workstation or laptop), and hence their energy sources can be replenished. Therefore, compared to the sensor nodes, the sinks are assumed to be able to run much longer without running out of energy. To evaluate fair energy consumption, we define the lifetime of a cell, T_c, as the time until all nodes in a cell first run out of energy. The CEDA routing specifies how data packets from the sensors are routed to the sink to maximize T_c. The longer the value of T_c for a cell, the longer the network connectivity is maintained and the longer the period for which the sink can access sensing data from the cell.

4. Cell based energy density aware routing algorithm

We now outline the CEDA algorithm, which consists of two phases, a route discovery phase and a reconstruction phase. During the route discovery phase, the sink disseminates a query including an attribute and its location. The sensor nodes matching the query send data packets to the location of the sink through a path chosen based on the energy density. During the reconstruction phase, routes can be changed to avoid uneven draining of the energy levels of the batteries in the various nodes.

In CEDA, each node has two tables: the neighbor node table and neighbor cell table (shown in Tables 1 and 2, respectively). The neighbor node table consists of a list of neighbor nodes, the cell with which each neighbor node is affiliated, the energy levels of the nodes, and attributes such as light, acoustic, etc. The neighbor cell table consists of a list of neighbor cells. It is used as a type of forwarding table for the routing. The size of table is proportional to the number of neighbor node, the number of neighbor cell, and the number of destination sink. The number of neighbor node is denoted as n_{nn} and the number of destination sink is n_{ns}. Let us denote that the size of node address is s_{addr} bits. Then the size of neighbor node table can be represented as $n_{\mathrm{nn}} \times (s_{\mathrm{addr}} + \lfloor \log_2 G \rfloor + \lfloor \log_2 F \rfloor)$(bits). $\lfloor \log_2 G \rfloor$ means the bit conversion of the number of neighbor cell and $\lfloor \log_2 F \rfloor$ is the bit calculation of the energy level. Also the size of the neighbor cell table can be shown as $n_{\mathrm{ns}} \times G \times 2$(bits). Tables 1 and 2 can be referred. If there may be tens of neighbor nodes, the size of the tables may be only tens of bytes. It is small value in comparison with the general memory size of the sensor network nodes.

Table 1
An example of the neighbor node table

Neighbor node	Cell	Energy level	Attributes
(22,24,1)	Same cell	4	Light
(22,24,2)	Same cell	3	Acoustic
(22,25,1)	Neighbor cell 2	4	N/A
(21,24,1)	Neighbor cell 3	2	N/A
(21,24,2)	Neighbor cell 3	0	N/A
…	…	…	…

4.1. Route discovery phase

The route discovery phase consists of the sink issuing a query packet and the sensors replying to the query. When a sink needs specific data from the sensor nodes, it generates a query packet. The query packet includes attributes and location information (x, y) regarding the position of the sink. The sink broadcasts the packet only to the router nodes in the network because the router nodes have records of the attributes of all other nodes in their respective cells. Each router node then looks up the attributes in its neighbor node table (e.g. Table 1) and feeds the packet to nodes matching the query. This scheme involves much less communication overhead than flooding the query packet to all nodes.

The routing is carried out using information on both the position of the sink and the remaining energy levels of the nodes. For brevity, the cell id i is defined to express

Table 2
An example of the neighbor cell table: case of two sinks D_1 and D_2

Neighbor cell	ANC to D_1	FC to D_1	ANC to D_2	FC to D_2
1	○	×	○	×
2	○	○	×	×
3	○	×	○	×
4	×	×	○	×
5	○	×	×	○
…	…	…	…	…

ANC, available neighbor cell; FC, forwarding cell.

cell (x, y) as

$$i = y \times X + x, \quad x = 0, 1, \ldots, X, \quad y = 0, 1, \ldots, Y. \tag{3}$$

Let S_i be the set of all sensor nodes in cell i and V_i be the set of nodes that are reachable by the nodes in cell i with a certain power level within its radio range. Wireless link $l_{i,j}$ exists if $j \in V_i$. Let each node $(i, n) \in S_i$ have battery energy $E(i, n)$, which is assumed to have several discrete levels defined as F (full energy), $F-1$, $F-2, \ldots, 0$ (completely depleted).

The routing discovery phase consists of the following steps.

Step 1. A sink disseminates a query packet into the network. Router node i receives the packet and broadcasts it to the other router nodes in neighboring cells. It also stores a list of available neighbor cells (ANC) A_i from which it can receive query packets among the cells included in V_i, that is, $A_i \subset V_i$. Set A_i is recorded in the neighbor cell table. Sensor nodes matching the query receive the packet from the router node in the cell. The sensor node should find, through Step 1, the next step, a neighbor cell to which the data packet is forwarded so that the data packet may arrive at the destination node.

Step 2. Suppose that intermediate cell i has a data packet. Let D denote the cell that contains the destination node that sent the query. Then, cell D is reachable from cell i by a single hop if the following is true

$$D \in V_i, \tag{4}$$

or

$$d(i, D) \le \sqrt{2}L \tag{5}$$

where L is the proximity level (defined in Eq. (2)), and $d(i, D)$ is the distance between cell i and cell D. In that case, cell i forwards the packet directly to destination node (D, n) without relaying it via the router node of cell D. Otherwise, Step 3 is executed to find a cell that can forward the packet toward the destination.

Step 3. If intermediate cell i has destination cell D in the forwarding table, it forwards the packet to the cell. However, if cell D is not in the table, cell i selects from among its neighbor cells V_i a set J of cells j which satisfy the following conditions:

$$j \in A_i \subset V_i, \tag{6}$$

$$d(i,j) \le \sqrt{2}L, \tag{7}$$

and

$$d(i, D) > d(j, D). \tag{8}$$

Condition 6 means that cell j must be one of the available neighbor cells A_i filtered at Step 1. This condition precludes the production of routes for which no path exists to destination D. Condition 7 restricts the neighbor cell j to be within the distance for transmission of a data packet from cell i in a single hop. The packet should be forwarded along the direction of the destination by (8).

Step 4. Node i forwards the packet to neighbor cell j among a set of cells J obtained at Step 2 to minimize the metric

$$M(i,j) = \alpha d(j, D) + \beta \left\{ \sum_{m \in S_j} W_{E(j,m)} E(j, m) \right\}^{-1} + \gamma d(i,j) \tag{9}$$

where $W_{E(j,m)}$ is a weight factor for different energy levels. For example, consider two systems of four nodes in a cell, one with node energy levels of (1, 1, 1, 1) and the other with (4, 0, 0, 0). In both systems, the sum of the energy levels of the four nodes is four; however, the weight factors make the metrics different. The second term of Eq. (9) is the inverse of the energy density of the cell j. If the energy densities of all neighbor cells are the same, the forwarding cell is chosen as the cell for which the distances $d(i, j)$ and $d(j, D)$ are smallest. Each of the terms of (9) has a weight factor: α, β, and γ, and $\beta \gg \alpha$, γ is suitable for the purpose of CEDA.

Step 5. After the packet is forwarded to the node selected in Step 3, the process returns to Step 1 and is repeated until the packet arrives at the destination, as described in Step 1.

4.2. Reconstruction phase

CEDA is a routing protocol that takes into consideration the features of sensor networks. CEDA is not an optimal routing but a localized one in that each node stores only information about its neighbors and the nodes to which it forwards data packets, as shown in Tables 1 and 2. CEDA avoids wasteful flooding of data by grouping nodes on the basis of cells. It additionally prevents the network from being disconnected by utilizing sensor nodes in areas densely populated by nodes as routers instead of nodes in sparse areas. If a cell has a lower residual energy density than other candidates for the forwarding, the packet is redirected around the cell. Below we outline the steps involved in path reconstruction.

Step 1. Whenever a node changes its energy level during communication, the node advertises its new energy level to its neighbor cells and the nodes in its cell. This scheme involves very little communication overhead. For example, if the energy of a node is expressed as five levels (i.e. 0 (zero energy), 1, 2, 3 and 4), each node will advertise its energy only four times during its lifetime. Thus, provided the number of energy levels is not large, the advertisement step consumes only a small amount of energy. The neighbor cells update their energy table with the new energy level of the node, as mentioned in Section 2. Suppose that cell i sends a data packet to neighbor cell j stored in the forwarding table. In the same manner, whenever a node in cell j changes energy level and the energy density of cell j is decreased, the node advertises its new energy level to its neighbor cells V_j

and the nodes in its own cell. When cell i receives the advertisement, it performs Step 2.

Step 2. If metric $M(i, j)$ satisfies the condition

$$M(i,j) \le \min_{k\in(A_i-j)} M(i,k) \tag{10}$$

where $M(i, j)$ is the metric of the established link $l_{i,j}$, then link $l_{i,j}$ continues to be used as a forwarding link. Thus, node i can still deliver the packet to node j. However, if Eq. (10) is not satisfied, the cell minimizing metric 9 should be chosen to forward the packet. This new cell is stored in the forwarding table in the neighbor cell table (Table 2).

Step 3. If the intermediate cell i receiving the packet has no node in the forwarding table, computation of metric $M(i, j)$ using Eq. (9) is repeated and the packet is forwarded into the neighbor node with the minimum energy density metric. When an intermediate cell already has a cell in its forwarding table, it forwards the packet to that cell. If the route to the destination is broken, causing the timer of the destination to expire, or the destination moves into another cell, the sink executes Step 1 of the route discovery phase to obtain the new available neighbor set A_i.

5. Guaranteeing the maximum hop count

A query may include an update period that limits the maximum delay from the sensor to the sink. Some power- or energy-aware routing protocols in wireless ad-hoc networks use paths consisting of short hops to save energy. However, such protocols can potentially suffer from very long delays because they do not limit the maximum number of hops allowed. In the case of wireless sensor networks, which typically consist of many densely distributed nodes, this problem would become worse. In this section we prove that CEDA can guarantee the maximum hop count. The fact that the maximum hop count can be guaranteed implies that CEDA is loop-free.

Lemma 1. *Let $h(S, D)$ be the maximum hop count between source S and destination D. If S is in cell (x, y) and D is in cell $(x+m, y)$ or $(x, y+m)$, the path generated by CEDA always has the maximum hop count*

$$h(S,D) = \begin{cases} |m|, & |m| \le 1, \\ 2|m|-2, & |m| > 1. \end{cases} \tag{11}$$

Proof. To find the path with the maximum hops between S and D, each cell should forward the packet to the neighbor cell that is farthest from the destination, where the neighbor cells are those satisfying the geometric conditions in Eqs. (5)–(8). It is certain that the maximum hop count $h(S, D)$ is zero when m is zero and $h(S, D)=1$ when $m=1$ or $m=-1$. In the case of $|m|>1$, as shown in Fig. 3(a), the distance $d(G_1, D)$ is always greater than $d(G_2, D)$, and thus a longer route is constructed as $S\to G_1\to G_2$ and, in the same

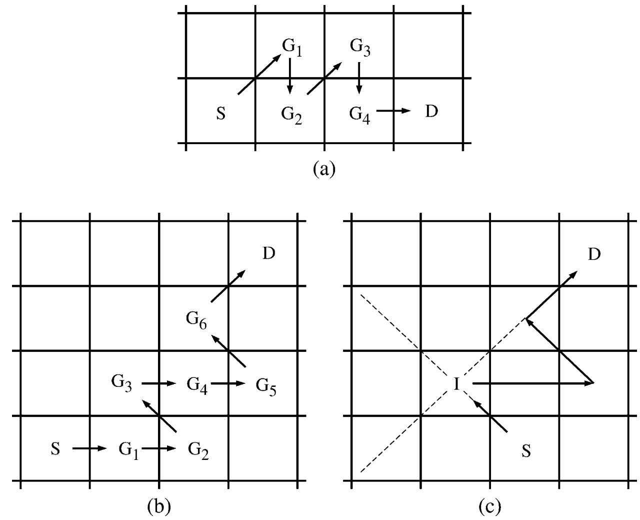

Fig. 3. Examples of computing the maximum hop count.

manner, it is repeated in $G_2\to G_3\to G_4$. Therefore, two hops are added whenever the packet goes from cell $(x+n, y)$ to cell $(x+n+1, y)$. This process gives

$$h(S,(x+n+1,y)) = h(S,(x+n,y)) + 2. \tag{12}$$

Because $h(S, D)=2$ at $|m|=2$, from Eq. (12), $h(S,D)=2|m|=-2$ when $|m|>1$. □

Lemma 2. *Let $h(S, D)$ be the maximum hop count between source S and destination D. If S is in cell (x, y) and D is in cell $(x+m, y+m)$, the path generated by CEDA always has the maximum hop count*

$$h(S,D) = \begin{cases} 0, & m = 0, \\ 3|m|-2, & m \ne 0. \end{cases} \tag{13}$$

Proof. As described in Lemma 1, we find the maximum hops of the path satisfying the geometric conditions in Eqs. (5)–(8). It is certain that the maximum hop count $h(S, D)$ is zero when $m=0$ and $h(S, D)=1$ when $m=1$ or $m=-1$. As shown in Fig. 3(b), $d(G_2, D)$ is greater than $d(G_3, D)$, and a longer route can be constructed by the hops $S\to G_1\to G_2\to G_3$ and it is repeated by the hops $G_3\to G_4\to G_5\to G_6$. Therefore, three hops are added whenever the packet goes to cell $(x+n+1, y+n+1)$ from an arbitrary cell $(x+n, x+n)$. This process gives

$$h(S,(x+n+1,y+n+1)) = h(S,(x+n,y+n)) + 3. \tag{14}$$

Because $h(S, D)=2$ at $|m|=1$, from (14), $h(S, D)=3|m|-2$ when $|m|>1$. □

Lemma 3. *Suppose that an intermediate cell (x, y) has a packet and the destination is in cell $(x+m, y+n)$, where $x\ge 0$ and $y\ge 0$. The condition where the intermediate cell (x, y) can forward the packet to $(x-1, y+1)$ or $(x+1, y-1)$ is $n>m+1$ or $n<m-1$, respectively.*

Proof. We still find the maximum hop count of the path satisfying the geometric conditions in Eqs. (5)–(8). As shown in Fig. 3(b), G_2 is further from the destination than G_3. Cell (x, y) can forward the packet to cell $(x-1, y+1)$ if $d((x+m, y+n), (x-1, y+1))$ is less than $d((x+m, y+n), (x, y))$, that is

$$(m+1)^2 + (n-1)^2 < m^2 + n^2, \tag{15}$$

which is the same as $n > m+1$. A similar process is followed for the case of cell $(x+1, y-1)$. □

Theorem 1. *In CEDA, the route from source cell* (x, y) *to destination cell* $(x+m, y+n)$ *guarantees the maximum hop count.*

Proof. Only destination cell $(x+m, y+n)$, where $m>0$, $n>0$, and $n>m$, will be considered in this proof because, by symmetry, the argument presented here is also valid for the remaining areas. First, to deliberately create the path with the maximum hop count, the packet is forwarded to $(x-1, y+1)$ if m and n satisfy the conditions of Lemma 3, or otherwise it is forwarded to either $(x+1, y)$ or $(x, y+1)$. When the intermediate cell satisfies Lemma 1 or 2, the maximum hop count can be directly calculated from the lemmas. If $(n-m)$ is an even number, the path of maximum hops has the intersection point $(n-m/2, n-m/2)$ between $y=x+(n-m)$ and $y=-x$, as shown in Fig. 3(c). Therefore, the maximum hop count is

$$\frac{n-m}{2} + 3\left(n - \frac{n-m}{2}\right) - 2 = 2n + m - 2 \tag{16}$$

If $(n-m)$ is an odd number, the result computed at $(x+m-1, y+n)$ plus one produces the same equation (Eq. (16)). Therefore, the maximum hop count of the route from source cell (x, y) to destination cell $(x+m, y+n)$ is

$$h((x,y),(x+m,y+n)) = \begin{cases} 0, & m=n=0, \\ 1, & 0<m^2+n^2<1 \\ 2|n|+|m|-2, & |n|\geq|m|, \\ 2|m|+|n|-2, & |n|<|m|. \end{cases} \tag{17}$$

Therefore, CEDA can guarantee the maximum hop count. □

6. Evaluation

To test whether the CEDA method actually conserves energy and to ascertain how long each monitored area is available in a finite-energy sensor network under a given set of conditions, we carried out simulations of sensor networks under various conditions. CEDA was simulated in the ns-2.26 simulator [21]. The simulation model was a network consisting of 200 randomly distributed acoustic sensor nodes and two sinks in an area of 100 m×100 m. The sinks could move at low speed. The sensor were set and sent a data packet every 5 s.

The MAC algorithm was based on IEEE 802.11 and the packet size was kept as 52 bytes. The radio range of each node was set to 20 m and neighbors were taken as the set of nodes within that range. Energy consumption for transmission was set to 1 μJ/bit and that for reception energy was set to 0.5 μJ/bit. The energy consumed by the microprocessor was considered to be a fixed overhead. All sensor nodes were given the same initial energy of 4 J. The energy remaining in each sensor node was expressed in terms of five energy levels (i.e. 0, 1, 2, 3 and 4). Each sink was set to have a high energy (e.g. 100 J) to reflect the fact that sinks can generally be replenished. The metric for selecting the path in Eq. (9) has weight factors $W_{E(\,)}$, which were all set to one in this simulation. The other weight factors α, β, and γ were assigned values of $\alpha=0.01$, $\beta=100$, and $\gamma=0.01$, respectively. Energy consumption was measured in Joules.

Since cellular addressing is used in the simulations, it is assumed that each node knows the location of the cell to which it belongs. In addition, it is generally assumed that wireless sensor nodes are static. So it is sufficient for sensor networks to find out geographic information only when network is initialized because this information has little chance to be changed. Thus the overhead to find the location is very small in comparison with the overall communication. Each cell was set to 10 m×10 m; hence there were a total of 100 cells in the field. The network layer for the routing takes a packet and processes it, and then sends it to a neighbor node or to the application layer. It also maintains the neighbor cell table and the neighbor node table, which include the energy table and the forwarding table with the list of its neighbors. The neighbor node table and neighbor cell table are small in size. Since CEDA uses geographical addressing, the node id is small in size. The ANC to D and FC to D entries of the neighbor cell table use a 1 bit Boolean flag each. In this simulation, the sizes of the neighbor cell table and neighbor node table do not exceed 100 bytes.

The simulation starts with the sinks disseminating queries to the sensor nodes. Two acoustic sensor nodes far from the sink generate data every 5 s. We set these two paths to cross at the center of the field to create a hot region where traffic is concentrated. For example, sources are placed at (5, 5) and (95, 5), and sinks are placed at (5, 95) and (95, 95), creating a hot region at the center of the field. Sensor nodes in the hot region are expected to run out of energy first. If more functional sources are present in the simulation, the sensor nodes around the sink will become the hottest region. Because nodes around a sink will inevitably consume more energy, we focused on a single hot region created by path crossing in order to show that routes detour around the hot region.

They keep the rate constant throughout this simulation. The performance of CEDA is compared with the performances of three other routing protocols: directed diffusion (DD), ad-hoc on-demand distance-vector (AODV), and

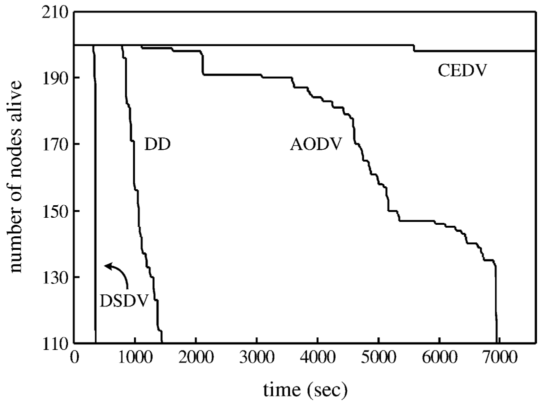

Fig. 4. The number of live nodes as a function of time for simulations using four different routing protocols.

destination-sequenced distance-vector (DSDV). AODV and DSDV are representative ad-hoc network routing protocols, and directed diffusion is well-known sensor network routing protocol. Under the various protocols, the energy efficiency and the network lifetime were evaluated. The same environment was employed in the simulations using the different routing protocols.

Fig. 4 shows a plot of the number of nodes that remain alive after t seconds for the four different routing protocols. This graph also gives the time at which a sensor node first runs out of energy. Losing a sensor node means that some information may not be gathered around it unless there is another sensor node in the near vicinity. CEDA forwards the packet to a high energy density cell, thereby creating a path that evades low energy density cells. In the simulation using CEDA, a node runs out of energy for the first time at $t=5590$; this time is called the system lifetime of the node. In the AODV- and DD-based simulations, on the other hand, the first node is lost at $t=1120$ and $t=790$, respectively. The system lifetime is shorter for AODV and DD than for CEDA because AODV and DD use the same path until a node along the path runs out of energy.

Fig. 5 shows the sum of the residual energies of all nodes in the network as a function of time. The simulation results indicate that the CEDA and AODV protocols consume less energy than the DD and DSDV protocols. The CEDA protocol consumes less energy primarily due to its lower communication overhead, which is achieved by using cells for energy conservation and localized path reconstruction. Since DSDV is a proactive routing protocol, it consumes the most energy among the four routing protocols.

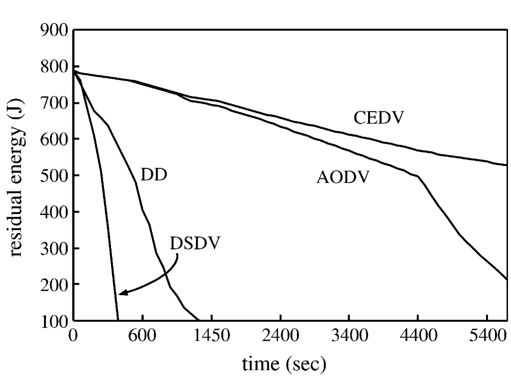

Fig. 5. Energy consumption over time for simulations using four different routing protocols.

DD is not a proactive routing protocol. It initially uses multiple paths to the sink and then gradually finds the best path. As a result, it quickly depletes the energy levels of nodes. At the early stage, the energy consumption of AODV is similar to that of CEDA. When AODV starts losing nodes, however, it tries to find a new path. AODV is an optimal path routing for delay, but it also uses broadcasting to find a route. Since it was originally developed for mobile nodes, it needs periodic reconstruction and route maintenance between the source and the destination. On the other hand, CEDA leaves reconstruction and route maintenance to the local cell. AODV considers the hop count, not the energies of the nodes along the path.

In dense sensor networks, each sensor node can exploit its adjacent nodes as redundant router nodes. However, if all nodes in a cell run out of energy, the sink will be unable to obtain information from the cell and therefore the user will no longer be able to monitor the cell. If critical phenomena occur in such an unavailable area, a disaster may ensue. Therefore, we introduce a measure of energy consumption of a cell referred to as the cell lifetime (T_c), which is defined as the time required for all nodes in the cell to run out of energy. Table 3 lists the cell lifetimes for CEDA and the other three routing protocols for networks containing 50, 100 or 200 nodes. Under the CEDA protocol, the cell lifetime increases markedly as the number of nodes in the network is increased. The cell lifetime obtained under AODV also increases with increasing node number, but to a lesser degree than under CEDA because AODV keeps using nodes along only the chosen path without replacing them with nodes in other more vital areas. When the number of nodes is 50 or 100, the density of nodes in the 10 m × 10 m cell is not more than one. When there are 200 nodes, the node density is two and hence a node has a good possibility of having one or more redundant nodes. Under DSDV, increasing the number of nodes in the network actually causes more rapid depletion of the energy of the cell, because the nodes must spread routing information to more nodes. Under DD, the number of multiple paths will increase in proportion to the number of nodes in the network. Thus, consistent with the results in

Table 3
Cell lifetime T_c: the time when all nodes of a cell have run out of energy

Routing	The number of nodes		
	50	100	200
CEDA	1195.13	1770.21	7740.20
AODV	1111.52	1100.15	2086.52
DD	846.70	1036.51	787.29
DSDV	730.43	611.45	320.77

Table 3, the cell lifetime of DD is not expected to show a simple trend with increasing node density. Therefore, among the four routing protocols, CEDA gives the longest cell lifetime. Moreover, the cell lifetime under CEDA increases with increasing node density. The cell lifetime achieved using CEDA could potentially be enhanced by further subdivision of the energy levels, provided the use of more levels does not cause too great an increase in the overhead associated with updating the energy table. Thus, the results of the evaluation indicate that CEDA is the most appropriate routing protocol for sensor networks because, compared with other protocols, it enables every area to be supervised for a longer time.

7. Conclusions

This paper has described a new energy-aware routing protocol, cell based energy density aware routing (CEDA), which is specifically designed for wireless sensor networks. It uses energy density as a routing metric and finds forwarding paths using information on geographic locations. A key concept of CEDA is the creation of simple routes by grouping nodes within uniform subareas (or cells). Another key concept of the proposed protocol is avoiding the concentration of energy consumption in particular nodes by forwarding packets to the cell with the greatest energy density. We evaluated CEDA by quantifying the residual energy, the cell lifetime, and the system lifetime. The cell lifetime is a useful variable because the presence of vacant cells may hinder communication and subarea monitoring in wireless sensor networks. CEDA prolongs the cell lifetime of the wireless sensor network, thereby enabling the monitoring stations to cover the entire sensor field for a longer time. CEDA guarantees the maximum hop count regardless of the number of nodes in the network.

The CEDA routing protocol is the first protocol to take into account the unique features of sensor networks, including small resources, location information, group management, attributes of sensing data, changeable topology, and uniform subarea monitoring. In the future, it is likely that networks of mobile sensor nodes will be developed. Hence, in future work it is planned to investigate the effects of sensor mobility in maintaining the geographical grid system. In addition, data aggregation is needed to reduce the energy consumption of the wireless sensor networks. Data aggregation is not considered in this paper. We are planning to cover this topic in future work.

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, A survey on sensor networks, IEEE Communication Magazine August (2002) 102–114.

[2] K. Kalpakis, K. Dasgupta, P. Namjoshi, Maximum lifetime data gathering and aggregation in wireless sensor networks, in: Proceedings of IEEE International Conference on Networking, Atlanta, GA, August, 2002.

[3] S. Singh, M. Woo, C.S. Raghavendra, Power-aware routing in mobile ad hoc networks, in: Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking, 1998 pp. 181–190.

[4] V. Rodoplu, T.H. Meng, Minimum energy mobile wireless networks, IEEE Journal of Selected Areas in Communication 17 (8) (1999) 1333–1344.

[5] C. Intanagonwiwat, R. Govindan, D. Estrin, Directed diffusion: a scalable and robust communication paradigm for sensor networks, in: Proceedings of ACM Mobicom 2000, Boston, 2000 pp. 56–67.

[6] W.R. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in: Proceedings of the Hawaii International Conference on System Sciences, Maui, Hawaii, January, 2000.

[7] S. Lindsey, C. Raghavendra, K.M. Sivalingam, Data gathering algorithms in sensor networks using energy metrics, IEEE Transactions on Parallel and Distributed Systems 13 (9) (2002) 924–935.

[8] C. Perkins, P. Bhagwat, Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers, in: Proceedings of the ACM SIGCOMM, October, 1994.

[9] D.B. Johnson, D.A. Maltz, Dynamic source routing in ad hoc wireless networks, Mobile Computing, Kluwer, Dordrecht, 1996. pp. 153–181.

[10] C. Perkins, E. Royer, Ad hoc on demand distance vector routing, in: Proceedings of IEEE Workshop on Mobile Computing, Systems and Applications, February, 1999.

[11] J. Broch, D.A. Maltz, D.B. Johnson, Y.C. Hu, J. Jetcheva, A performance comparison of multi-hop wireless ad-hoc network routing protocols, in: Proceedings of the ACM/IEEE Mobicom, October, 1998.

[12] B. Karp, H.T. Kung, Greedy perimeter stateless routing for wireless networks, in: Proceedings of the Sixth ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM 2000), Boston, August, 2000 pp. 243–254.

[13] J. Heidemann, F. Silva, C. Intanagonwiwat, R. Govindan, D. Estrin, D. Ganesan, Building efficient wireless sensor networks with low-level naming, in: Proceedings of the Symposium on Operating Systems Principles, Banff, Canada, October, 2001.

[14] I. Stojmenovic, X. Lin, Power-aware localized routing in wireless networks, IEEE Transactions on Parallel and Distributed Systems 12 (11) (2001) 1122–1133.

[15] J. Albowitz, A. Chen, L. Zhang, Recursive position estimation in sensor networks, in: Proceedings of the Ninth IEEE International Conference on ICNP, 2001 pp. 35–41.

[16] N. Bulusu, J. Heidemann, D. Estrin, GPS-less low cost outdoor localization for very small devices, IEEE Personal Communications Magazine, Special Issue on Smart Spaces and Environments October (2000).

[17] R.C. Shah, J.M. Rabaey, Energy aware routing for low energy ad hoc sensor networks, in: Proceedings of the IEEE Wireless Communication and Network, 2002 pp. 350–355.

[18] Y. Xu, J. Heidemann, D. Estrin, Geography-informed energy conservation for ad hoc routing, in: Proceedings of the International Conference on Mobile Computing and Networking, 2001, pp. 70–84.

[19] Y.B. Ko, N.H. Vaidya, Location-aided routing (LAR) in mobile ad hoc networks, in: Proceedings of Mobile Computing, MOBICOM, 1998, pp. 66–75.

[20] F. Ye, H. Luo, J. Cheng, S. Lu, L. Zhang, A two-tier data dissemination model for large-scale wireless sensor networks, in: Proceedings of MOBICOM 2002, 2002.

[21] The ns-2 simulator, http://www.isi.edu/nsnam

[22] T. He, J. Stankovic, C. Lu, T. Abdelzaher, SPEED: a stateless protocol for real-time communication in sensor networks, in: Proceedings of International Conference on Distributed Computing Systems, Rhode Island, USA, May, 2003.

[23] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D.C.K. Pister, System architecture directions for networked sensors, in: Proceedings of the Ninth International Conference on Architectural Support for Programming Languages and Operating Systems, November, 2000.