Available online at www.sciencedirect.com

ScienceDirect

Computer Networks 52 (2008) 531–541

Computer Networks

www.elsevier.com/locate/comnet

A color-theory-based energy efficient routing algorithm for mobile wireless sensor networks ☆

Tai-Jung Chang, Kuochen Wang *, Yi-Ling Hsieh

Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan, ROC

Received 25 January 2007; received in revised form 31 July 2007; accepted 11 October 2007
Available online 22 October 2007

Responsible Editor: X.S. Shen

Abstract

Wireless sensor networks (WSNs) with nodes spreading in a target area have abilities of sensing, computing, and communication. Since the GPS device is expensive, we used a small number of fixed anchor nodes that are aware of their locations to help estimate the locations of sensor nodes in WSNs. To efficiently route sensed data to the destination (the server), identifying the location of each sensor node can be of great help. We adopted a range-free color-theory based dynamic localization (CDL) [Shen-Hai Shee, Kuochen Wang, I.L. Hsieh, Color-theory-based dynamic localization in mobile wireless sensor networks, in: Proceedings of Workshop on Wireless, Ad Hoc, Sensor Networks, August 2005] approach, to help identify the location of each sensor node. Since sensor nodes are battery-powered, we propose an efficient *color-theory-based energy efficient routing* (CEER) algorithm to prolong the life time of each sensor node. The uniqueness of our approach is that by comparing the associated RGB values among neighboring nodes, we can efficiently choose a better routing path with energy awareness. Besides, the CEER has no topology hole problem. Simulation results have shown that our CEER algorithm can save up to 50–60% energy than ESDSR [Mohammed Tarique, Kemal E. Tepe, Mohammad Naserian, Energy saving dynamic source routing for ad hoc wireless networks, in: Proceedings of Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, April 2005, pp. 305–310] in mobile wireless sensor networks. In addition, the latency per packet of CEER is 50% less than that of ESDSR.

© 2007 Elsevier B.V. All rights reserved.

Keywords: Color-theory-based; Energy saving; Localization; Mobile wireless sensor network; Routing algorithm

1. Introduction

With the advance of recent wireless and VLSI technologies, small and low-cost sensor nodes have become feasible in our daily life. These sensor nodes sense data in the environment surrounding them and transmit the sensed data to the sink (or the server). The way to transmit sensed data to the server

---

☆ The insightful comments of the reviewers help to improve the quality of the paper. This work was supported by the National Science Council under Grant NSC94-2213-E-009-043 and NCTU EECS-MediaTek Research Center under Grant Q583.

* Corresponding author.
E-mail address: kwang@cs.nctu.edu.tw (K. Wang).

1389-1286/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2007.10.004

affects the power usage of each node deeply. Typically, wireless sensor networks (WSNs) contain hundreds or thousands of sensor nodes and these sensor nodes have the ability to communicate with either one another or directly to the destination node (sink). Using a larger number of sensor nodes allows for sensing over wider geographical regions with greater accuracy. Sensor nodes may use up their energy so that they become unavailable in a WSN during communications or measurements. With more and more sensor nodes becoming unavailable, the WSN may be separated into several sub-networks or become sparse, which is not desirable. Therefore, energy saving is an important issues in the WSN.

It is a great challenge for routing in a WSN due to the following reasons. First, since it is not easy to grasp the whole network topology, it is hard to find a routing path. Secondly, sensor nodes are tightly constrained in terms of energy, processing, and storage capacities. Thus, they require effective resource management policies, especially efficient energy management, to increase the overall lifetime of a WSN.

The proposed clustered-based *color-theory-based energy efficient routing* (CEER) algorithm is based on a range-free color-theory-based dynamic localization algorithm, CDL [1], in which the location of a sensor node is represented as a set of RGB values. With known RGB values for each sensor node, we can find out the most possible position of a node by looking up the location database in the server. To keep track of a sensor node's location, frequently updating the RGB values of each sensor node and delivering the update to the server is necessary. However, if battery-powered nodes frequently update and report their positions, they may consume energy quickly and also waste bandwidth. The CEER selects those cluster members that are closer to the anchor than itself as next possible hops by comparing their RGB values. Among the selected cluster members, the sensor node with the highest energy level is chosen as the next hop.

The remainder of this paper is organized as follows. Section 2 briefly introduces the CDL algorithm that is the basis of our routing algorithm. Section 3 introduces five related approaches and explains how they work. The network model and the proposed CEER algorithm are detailed in Section 4. Section 5 evaluates and compares our approach with a classical approach. Simulation results demonstrate the merits of our algorithm. Finally, concluding remarks are given in Section 6.

2. Preliminary of CDL

In this section, we introduce the color-theory-based dynamic localization (CDL) algorithm [1], which is the basis of our proposed routing algorithm. This centralized localization algorithm is based on the color theory to perform positioning in mobile wireless sensor networks. It builds a location database in the server, which maps a set of RGB values to a geographic position. And the distance measurements between sensor nodes are based on the DV-Hop [3]. After receiving an anchor' RGB values, a sensor node converts the RGB values into HSV values, using an algorithm *RGBtoHSV* [16]. Note that HSV stands for *Hue*, *Saturation*, and *Value* [17]. Based on color theory, only the lightness of color fades out with the increasing of propagating distances. That is, the V of HSV of an anchor, which is corresponding to the lightness, is decreased in proportion to the distance from the node to the anchor. With the new HSV values, the adjusted RGB values for the node can be obtained by using another algorithm, *HSVtoRGB* [16]. The node then calculates its own RGB values by averaging these adjusted RGB values, corresponding to the anchors. The node then sends its RGB values to the server so that the server can find its most probable location by looking up the location database.

2.1. The information delivery of anchors [1]

In this section, we introduce some notations that are defined in CDL:

- D_{avg} is the average hop distance, which is based on the DV-Hop [3].
- h_{ij} is the hop count between nodes i and j.
- D_{ik} represents the hop distance from anchor k to node i: $D_{ik} = D_{\text{avg}} \times h_{ik}$.
  Each node i maintains an entry of (R_{ik}, G_{ik}, B_{ik}) and D_{ik}, where k represents the kth anchor.
- (H_{ik}, S_{ik}, V_{ik}) is a set of HSV values of anchor k, converted from (R_{ik}, G_{ik}, B_{ik}) by the ith node.
  Range represents the maximum distance that a color can propagate.
- (R_k, G_k, B_k) is a set of RGB values of anchor k.

The RGB RGB values of anchors are first assigned randomly from 0 to 1. After a sensor node

i obtains each anchor k's RGB values and hop count (h_{ik}), the RGB values are then converted to HSV values by Eq. (1):

$$(H_k, S_k, V_k) = RGBtoHSV(R_k, G_k, B_k). \tag{1}$$

With h_{ik}, D_{ik} can be computed. The updated HSV values corresponding to node i of anchor k are calculated by Eq. (2):

$$H_{ik} = H_k, \quad S_{ik} = S_k, \quad V_{ik} = \left(1 - \frac{D_{ik}}{\text{Range}}\right) \times V_k. \tag{2}$$

The RGB values of node i corresponding to anchor k are then calculated:

$$(R_{ik}, G_{ik}, B_{ik}) = HSVtoRGB(H_{ik}, S_{ik}, V_{ik}). \tag{3}$$

The RGB values of node i are the mean of the RGB values corresponding to n anchors, and are computed as follows:

$$(R_i, G_i, B_i) = \frac{1}{n} \times \sum_{k=1}^{n} (R_{ik}, G_{ik}, B_{ik}), \tag{4}$$

where n is the number of anchors that node i received their RGB values.

Fig. 1 illustrates how node i obtains its RGB values.

2.2. The establishment of location database [1]

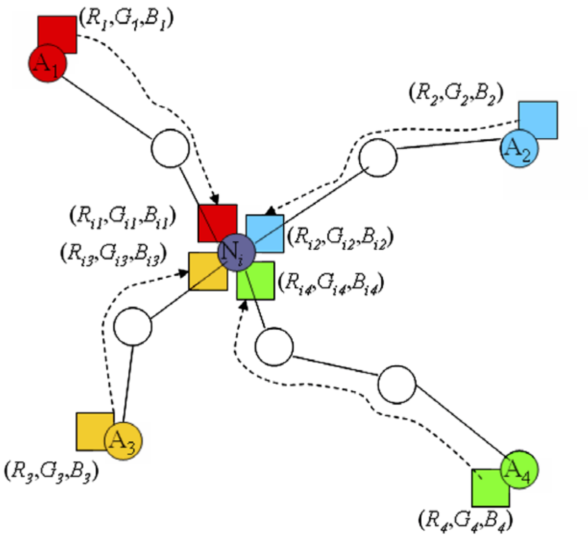
A location database is established when the server obtains the RGB values and locations of all anchors. The mechanism is based on the theorem of the mixture of different colors. With the RGB values of all anchors, the RGB values of all locations can be computed by exploiting the ideas of color propagation and the mixture of different colors. Note that our localization approach is a kind of localization fingerprinting approaches [18]. Based on the similar steps of how a sensor node obtains its RGB values, as described in Section 2.1, one can create a location database. The only difference is Euclidean distance instead of hop-count distance is used. Firstly, the Euclidean distance between each location i and anchor k is derived:

$$d_{ik} = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2}, \tag{5}$$

where (x_i, y_i) is the coordinate of location i, and (x_k, y_k) is the location of anchor k. First of all, we have to calculate the HSV values of each location i corresponding to anchor k:

$$(H_k, S_k, V_k) = \boldsymbol{RGBtoHSV}(R_k, G_k, B_k), \tag{6}$$

$$H_{ik} = H_k, S_{ik} = S_k, V_{ik} = \left(1 - \frac{d_{ik}}{\text{Range}}\right) \bullet V_k. \tag{7}$$



1. For each k, $(R_{ik},G_{ik},B_{ik}) \rightarrow (H_{ik},S_{ik},V_{ik}) \rightarrow (H_{ik},S_{ik},V'_{ik}) \rightarrow (R'_{ik},B'_{ik},G'_{ik})$
2. For node i (N_i), $(R_i,G_i,B_i) = \text{average}((R'_{i1},B'_{i1},G'_{i1}), (R'_{i2},B'_{i2},G'_{i2}), (R'_{i3},B'_{i3},G'_{i3}), (R'_{i4},B'_{i4},G'_{i4}))$

Fig. 1. An illustration of how node i obtains its RGB values.

The RGB values of location i corresponding to anchor k can be derived by Eq. (8):

$$(R_{ik}, G_{ik}, B_{ik}) = \boldsymbol{HSVtoRGB}(H_{ik}, S_{ik}, V_{ik}). \tag{8}$$

Then the RGB values of location i can be calculated by averaging all RGB values of location i corresponding to N anchors.

$$(R_i, G_i, B_i) = \frac{1}{N} \times \sum_{k=1}^{N} (R_{ik}, G_{ik}, B_{ik}), \tag{9}$$

where N is the number of anchors.

In this way, the location for each sensor node can be constructed in the location database by maintaining the coordinate (x_i, y_i) and the RGB values (R_i, G_i, B_i) at each location i. Then, the location of a sensor node can be acquired by looking up the location database based on the derived RGB values.

2.3. Mobility [1]

When a mobile node arrives at a new location, it sends an anchor information request to neighbor nodes. If a neighbor node has the anchors' RGB values, it transmits packets that contains the RGB values of each anchor and the hop count from the anchor to the node. After receiving the packets from neighbor nodes, node i compares and calculates the D_{ik} to the kth anchor and selects the smallest D_{ik}. With the RGB values and D_{ik} to all anchors, node i can update its RGB values using Eqs. (1)–(4). The new RGB values are then transmitted to the server and the position of node i will be updated in the location database.

3. Related work

3.1. Existing routing protocols

In this section, we review several routing protocols in WSNs, which can be categorized into *flat routing*, *hierarchical routing*, *location-based routing,* and *source routing*. Flat routing, which is a kind of data-centric routing, is that when a node queries for data in its communication range, neighbor nodes which have the data will transmit the data to that node. Example flat routing protocols include SPIN [4] and Direct Diffusion [11]. Hierarchical routing builds a hierarchical topology in the region. It performs data aggregation and fusion in cluster heads in order to decrease the number of transmitted messages to the base station. Example hierarchical routing protocols include TTDD [5], MECH [12], and PEGASIS [13]. As to the location-based routing, sensor nodes are addressed by their locations. We can use the direction of the destination to forward data to the nearest neighborhood recursively until reaching the destination. Example location-based routing protocols include GEAR [6] and SPAN [14]. Source routing is a routing technique in which the source node determines the complete sequence of nodes to forward the packet. The source node explicitly lists this route in the packet's header. It identifies each forwarding "hop" by the address or node's ID of the next node to transmit the packet on its way to the destination node. Example source routing protocols include DSR [7], ESDSR [2] and MSR [15]. In the following, we review a classical routing protocol from each category.

3.1.1. Flat routing – sensor protocols for information via negotiation (SPIN) [4]

SPIN disseminates all the information at each node to neighbor nodes in the network. A source node first advertises its data to neighbor nodes. If a node within the radio range wants the data advertised by the source node, it sends a reply packet to the source node. The source node will send data to the request node which queried for the data. This is illustrated in Fig. 2. The disadvantage of SPIN is

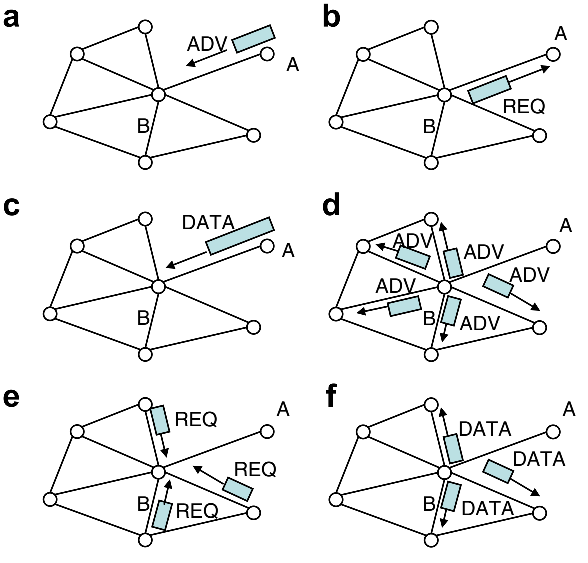


Fig. 2. Flat routing – SPIN. Node A starts by advertising its data to node B (a). Node B responds by sending a request to node A (b). After receiving the requested data (c), node B then sends out advertisements to its neighbors (d), who in turn send requests back to B (e and f) [4].

that it cannot guarantee the delivery of interested data. If a node is interested in the data that is far away from the source node and nodes between source and destination nodes are not interested in that data, data will not be delivered to the destination node at all.

3.1.2. Hierarchical routing – two tier data dissemination (TTDD) [5]

An approach that provides data delivery to multiple mobile sinks is called TTDD, is illustrated in Fig. 3, which is two tier data dissemination. Each data source proactively builds a grid structure that is used to disseminate data to the mobile sinks by assuming that sensor nodes are stationary and location aware. The sink chosen by the server may leave its old position to another position. Once this occurs, sensor nodes surrounding the sink process the signal sent by the sink and one of them becomes the new source node to generate data reports.

3.1.3. Location based routing – geographic and energy aware routing (GEAR) [6]

This algorithm, as illustrated in Fig. 4, discusses the use of geographic information while disseminating queries to appropriate regions since data queries often include geographic attributes. Each node in GEAR keeps an estimated cost and a learning cost of reaching the destination through its neighbors. The estimated cost is a combination of residual energy and distance to the destination. The learning cost is a refinement of the estimated cost that accounts for routing around holes in the network. A hole occurs when a node does not have any closer neighbor to the target region than itself. Two phases are used in this algorithm, forwarding packets towards the target region and forwarding packets within the region. In the first phase, when receiving a packet, a node checks its neighbors to see if one neighbor node is closer to the target region than itself. In the second phase, if the packet reaches the target region, it can be diffused in that region by using recursive geographic forwarding or restricted flooding.

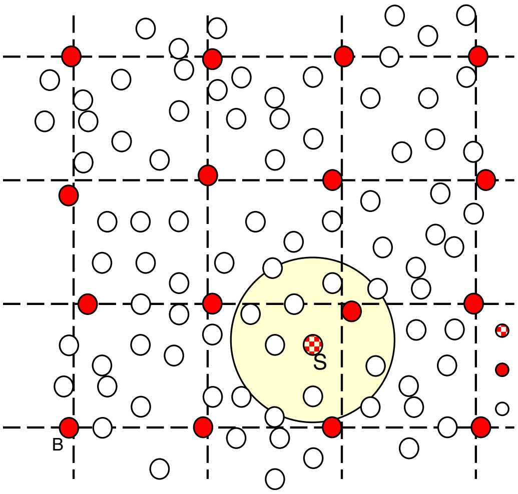


Fig. 3. Hierarchical routing – TTDD: one source B and one sink S [5].

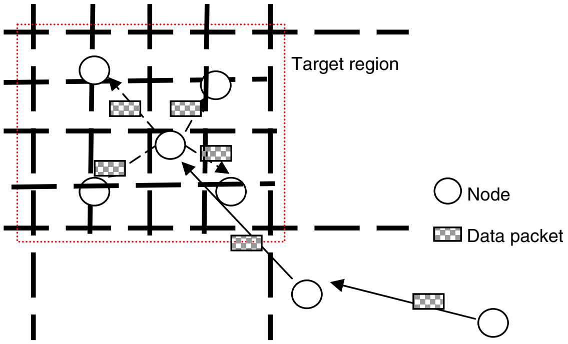


Fig. 4. Location based routing – GEAR. Recursive geographic forwarding: data disseminate to four sub-regions recursively until regions with only one node are left [6].

3.1.4. Source routing – dynamic source routing (DSR) [7] and ESDSR [2]

The DSR routing [7] is based on the source routing. It uses each node's cache to store information about the routing path which will be maintained by each sensor node if the routing path is broken or not. Two operations are used to build a routing path, *route discovery* and *routing maintenance*. Route discovery is the mechanism by which a source node finds a route to the destination. When a source wants to send information to the destination, it searches its own cache to find a routing path. If the source node can not find a routing path to the destination, it starts to perform route discovery. First of all, the source node initiates a local broadcast to start the routing. If the node which receives the packet is the route destination, it will route back to the source node. If it is not the destination, it also broadcasts a packet to its neighbors until reaching the destination. The source will choose the shortest path – the smallest hop-count path – to route to the destination.

The route maintenance operation is the mechanism which a node is able to detect any changes in the network topology. While a node is unavailable, its one hop neighbor nodes around the routing path should route back to the source such that every node's cache can update the route information.

ESDSR [2] is an energy saving DSR protocol. It changes the way of choosing a routing path from the shortest path in DSR to the maximum expected life of the path. Once a source node setups a route discovery, the return packet of each route path will contain the minimum expected battery life. The source node chooses a routing path with the maximum expected battery life among all routing paths. Each node also has its power table which contains its neighbor's power state information so that it will choose a more energy efficient path to the destination.

3.2. Comparison of different routing protocols

In terms of the following metrics: *energy cost*, *mobility*, *position awareness*, and *localization*, the above mentioned approaches are compared, as shown in Table 1. The proposed color-theory-based energy efficient routing (CEER) is also included in the table, which will be described in Section 4. First, the metric of energy cost indicates the average energy cost of the protocol for a packet sending from the source to the destination. CEER and ESDSR have the lowest energy cost. These two algorithms will be further evaluated quantitatively in our simulations later. Mobility indicates whether nodes in the WSN are fixed or mobile. DSR, ESDSR, and the proposed CEER can support mobility. The position awareness indicates that if a protocol needs each node to know its position, e.g., using GPS. TTDD and GEAR require this information to route data to the destination. The last metric, localization, indicates that the proposed CEER can perform localization because the source node uses RGB values to assist routing [1].

Table 1
Comparison of six routing protocols, including the proposed CEER protocol

Approach	Energy cost	Mobility	Position awareness (e.g., using GPS)	Localization
SPIN [4]	Mid	N/A	No	No
TTDD [5]	Mid	Low	Yes	No
GEAR [6]	Mid	Low	Yes	No
DSR [7]	Mid	Mid	No	No
ESDSR [2]	Low	Mid	No	No
CEER (proposed)	Low	Mid	No	Yes

4. Color-theory-based energy efficient routing algorithm

In this section, we propose a color-theory-based energy efficient routing (CEER) algorithm for WSNs, which is based on the CDL [1]. The network model is first described. The routing process can be organized into three phases: setup phase, data dissemination phase, and refinement phase, which will be described next.

4.1. Network model

The network model for our routing protocol is described as follows:

- A server is used for building a location database, localization, and collecting sensed data.
- The area is divided into clusters and a cluster head for each cluster is selected by the server.
- There are four anchors, which are placed in the four corners of the area [9]. Anchors collect aggregate data received from cluster heads.
- All sensor nodes have a uniform energy at the beginning.
- Each node wakes up at a fixed rate to check if its RGB values have changed.
- All sensor nodes are mobile.
- CSMA/CA is used to avoid collision of packets.

4.2. Setup phase

In this phase, we need to choose a sensor node to be the cluster head in each cluster so that data collection can be started. First, we use a sensor node's radio range R such that the grid size is $R \times R$. All sensor nodes in a cluster (grid) are within their cluster head's radio range. Secondly, each anchor floods its RGB values and average hop distance to each sensor node so that each sensor node can calculate its hop count to the anchor and adjust its RGB values based on the hop count. Then, the server will receive the RGB values from each sensor node. By looking up the location database, the server can calculate each sensor node's location and choose a cluster head which is close to the center of the grid, as shown in Fig. 5. Finally, the server uses the

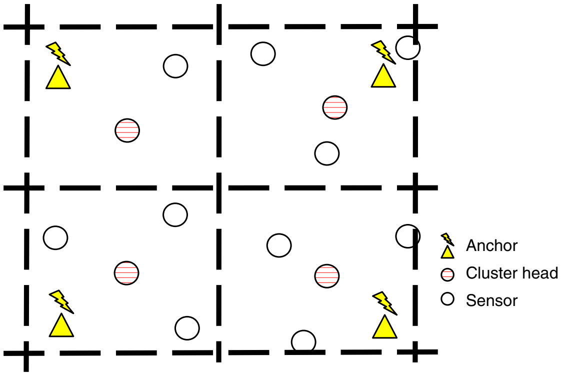


Fig. 5. Setup step 1: Choose a cluster head which is close to the center of the grid.

closest anchors to send information to the selected sensor nodes to let them know that they are cluster heads, as shown in Fig. 6.

4.3. Data dissemination phase

After the setup phase, each cluster head will receive its cluster member's information. If a sensor node's position changes, its hop counts to the anchors may also change. The node then updates its RGB values and transmits the RGB values as well as collected data (if available) to its cluster head. When a cluster head receives information from its cluster member, it will setup a timer and wait for other possible information from other members until the timer expires. The advantage of setting up a timer is that we can wait for other cluster members to see if they have data to send. In this way, communication cost can be reduced.

When a cluster head wants to forward the aggregate data toward the server, two steps are performed in CEER. In the first step, it selects its one hop neighbors that are closer to a nearby anchor than itself as the next possible hop by comparing their respective RGB values, as shown in Fig. 7. In the second step, among those selected cluster members, the sensor node with the highest energy level is selected as the next hop, as shown in Fig. 8. The selected node will receive aggregated data and again follows these two steps to transmit the data to its next hop until reaching the server via the anchor.

Our CEER algorithm does not have the topology hole problem that often occurs in location-based routing [6]. The CEER is based on the CDL [1] and is basically a hop-count-based routing approach, not a location-based routing approach. We use Fig. 9 to explain how our routing approach will not encounter the topology hole problem. In Fig. 9, it shows a topology hole caused by an obstacle. For node 2, in a location-based routing

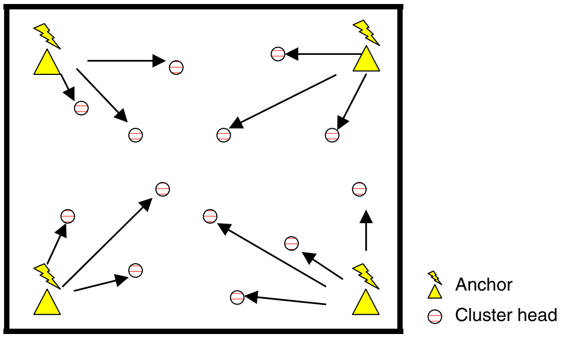


Fig. 6. Setup step 2: The server uses the closest anchors to send information to the selected sensor nodes to let them know that they are cluster heads.

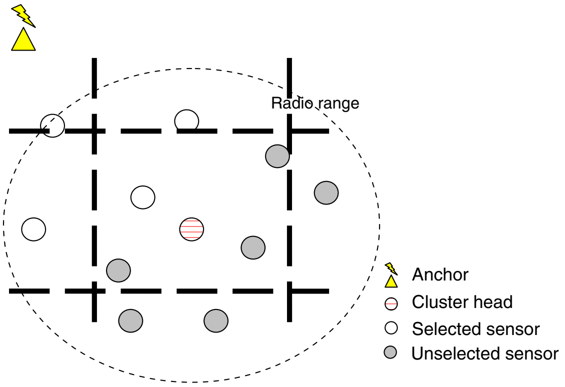


Fig. 7. Data dissemination step 1. Select one hop neighbor nodes closer to the anchor.

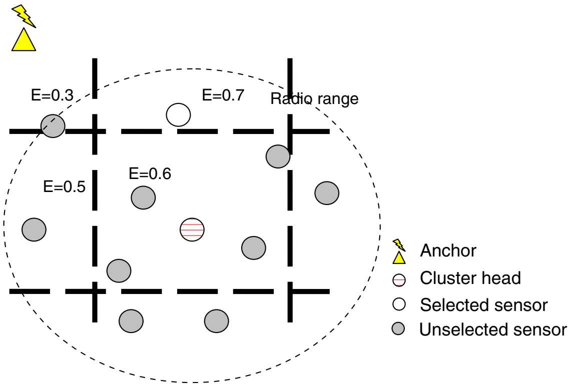


Fig. 8. Data dissemination step 2. Choose the node with the maximum energy level from the selected nodes.

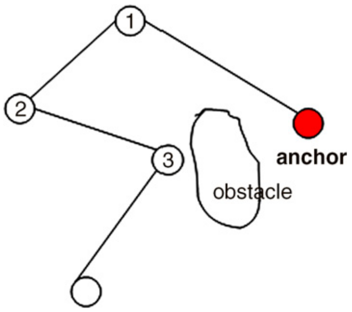


Fig. 9. Topology hole problem.

approach, it will choose node 3 as the next node since node 3 is closer to the destination in terms of Euclidean distance. Thus, it will encounter the topology hole problem. However, in the CEER, node 2 would choose node 1 as the next hop because the color of node 1 is closer to the anchor (destination) than node 3, even if node 3 is geographically closer to the anchor. This is because if a node with a color closer to the color of the destination, it also implies the node has a smaller hop-count distance to the destination based on color theory. Since each sensor node can receive the hop-count information from adjacent sensor nodes, we can always find a sensor node with RGB values closer to that of the anchor. Therefore, the CEER will not encounter the topology hole problem when routing data to the anchor.

4.4. Refinement phase

There are several situations that a cluster head may fail to do its job. For example, if a cluster head has low power, it may not live long enough to be a cluster head. On the other hand, if a cluster head moves away from the grid, it may lose contact with some cluster members. In these situations, it is necessary to choose another sensor node to become a new cluster head so that the collected information can still be forwarded to the server. In our approach, the server is responsible to choose cluster heads because the server knows the speed, position, and energy level of each node. However, if we change cluster heads too often, it will waste energy and induce extra communication cost to switch the cluster head's job from one sensor node to another. To save energy, under the following two conditions, the role of a cluster head will not be replaced:

- A cluster head with an energy level higher than half of the original battery capacity.
- A cluster head with an energy level lower than half of the original battery capacity but higher than the average energy level.

If a cluster head with low energy needs to be replaced, the server collects information of the sensor nodes in the corresponding grid. Then the server sorts each node's speed from low to high and chooses a node if its energy level is higher than half of the original battery capacity or higher than the average energy level. When a sensor moves away from its cluster head, it broadcasts a packet to its neighbor nodes to ask for a new cluster head and hop counts. If there are more than two cluster heads for the sensor node to choose, the cluster head with RGB values closer to itself will be chosen.

5. Simulation results and discussion

In this section, we compare the proposed CEER with the ESDSR [2] by measuring their total energy consumption and latency per packet with respect to various numbers of sensor nodes. Since our routing algorithm is based on the CDL [1], which was implemented in C++. We also used C++ to construct the simulation environment.

5.1. Simulation model

Our simulation model follows that of CDL [1] since our routing algorithm is based on the CDL. All nodes were randomly placed in a 500 m × 500 m area and a modified random waypoint models [8] were used to simulate the mobility characteristic of each sensor node. In addition, a sensor node selects a moving destination and a velocity randomly. After reaching the destination, the sensor node pauses for a period of time (*pause time*). The node speed is uniformly distributed within $[V_{\min}, V_{\max}]$. The simulation parameters are shown in Table 2. We used the same energy consumption model as that of ESDSR [2]. The energy consumption per data-packet of size D bytes over a given link can be modeled as

$$E(D, P_t) = K_1 P_t D + K_2,$$

where P_t is the minimum transmit power required for successful reception, and the typical values of constant K_1 and K_2 in IEEE 802.11 MAC environ-

Table 2
Simulation parameters

Parameter	Value
Area size	500×500 m^2
Node speed	Randomly choose from $[V_{\min}, V_{\max}]$
Node transmission range (R)	50 m
Pause time	0
Measurement period	50 t_u
Update interval	5 t_u
Time slot length (time unit)	t_u

ment at 2 Mbps are 4 µs per byte and 42 µJ, respectively.

The two performance parameters are defined as follows:

- *Total energy consumption* (mJ): summary of all sensor nodes' energy consumption after 50 time slots (time units).
- *Latency per packet* (s): average time spent to send a packet with aggregate data, such as RGB values and collected data, from a sensor node to the server.

In [9], it has shown that the location estimated error is minimized when anchors are placed at the corners. So we put four anchors (A_1–A_4) at the four corner of the area, as shown in Fig. 10, where S represents a sensor, a solid line represents a direct link with one hop, and a dotted line represents an indirect link with more than one hop. In addition, Fig. 11 shows the impact of seed (anchor) density on location estimated error [1]. Note that anchor density is the average number of anchors in one hop transmission range [1]. This figure shows that the more anchors (more specifically, the higher anchor density), the better position accuracy (the lower estimated error).

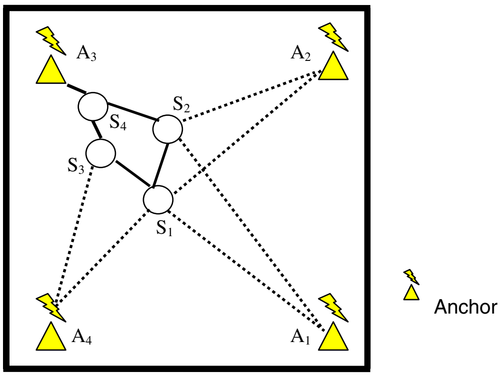


Fig. 10. Anchor deployment.

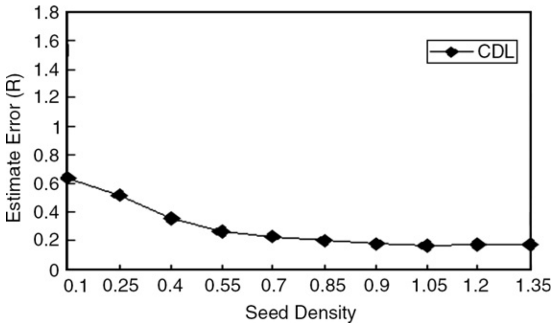


Fig. 11. Impact of seed density on location estimated error [1].

Fig. 10 is also used to illustrate the difference between CEER using RGB and a revised CEER using (absolute distance, hop count). In Fig. 10, if sensor node S_1 intends to send data to anchor A_3, for a revised CEER using (absolute distance, hop count), it will select either S_2 or S_3 as the next hop, as both sensor nodes have the same hop count to A_3. However, for CEER using RGB, it will always select S_3 as the next hop since S_3's color is closer to A_3's color than S_2's color. This is because CEER using RGB has better location accuracy than CEER using (absolute distance, hop count), as shown in [1]. That is, CEER using RGB always chooses a path with a shorter actual distance. Therefore, the direct benefits of using RGB are to have lower total energy consumption and lower latency per packet in CEER.

5.2. Comparison with ESDSR [2]

Kim et al. [10] mentioned that a well dissemination routing protocol should have three characteristics: *energy efficiency*, *self-configuration*, and *scalability*. By simulations, we have shown that our algorithm is better than ESDSR in terms of energy efficiency. Our algorithm has the characteristic of self-configuration in terms of cluster head selection, as described in Section 4.4. As to the scalability, our algorithm is more scalable than ESDSR when the area is bigger, which will be described at the end of this section.

As shown in Fig. 12, our approach (CEER) saves more energy than the ESDSR when the number of nodes increases. This is because the CEER uses a cluster head to aggregate data and adopts dynamic path finding to find a neighbor node with maximum energy to route data to a nearby anchor. The

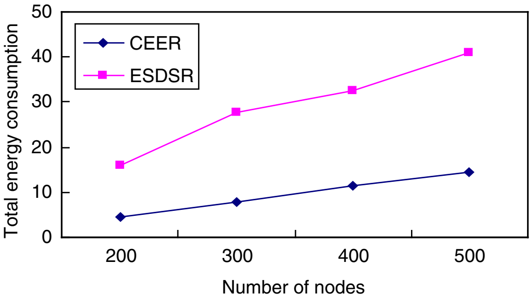


Fig. 12. Total energy consumption vs. number of nodes (with cluster heads).

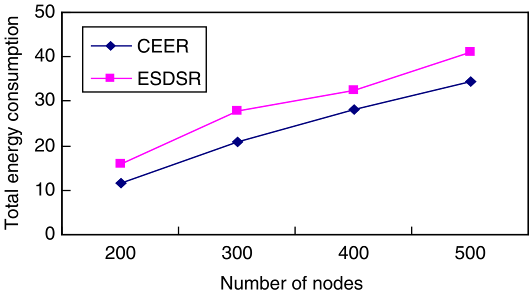


Fig. 14. Energy consumption vs. number of nodes (without cluster heads).

ESDSR uses local broadcast to find a path and chooses the best route with the maximum expected life, but the source node does not aggregate data of its neighbor nodes to send data to the destination.

In Fig. 13, the CEER has less latency per packet than the ESDSR because the ESDSR uses the path with the maximum expected life in order to balance the load of the entire sensor network. In contrast, our approach selects a next hop node to deliver data based on the RGB values and the maximum energy levels of neighbor nodes, which is more efficient.

We also simulated our routing algorithm without cluster heads to verify that our routing algorithm also works well without clustering. We used the two data dissemination steps in Section 4.3 so that each node will route its data to the nearest anchor without relying on a cluster head. Fig. 14 shows that our algorithm also works well without cluster heads and saves more energy than the ESDSR.

We also used the same network model as that of ESDSR [2] to validate the correct implementation of the ESDSR and CEER. Forty sensor nodes were randomly placed in an area sized from 200 m × 200 m, 300 m × 300 m, 400 m × 400 m, to 500 m × 500 m, respectively, while keeping the number of sensor nodes (40) and connection (20) constants. Fig. 15 shows that, in a small area, our approach did not perform well compared to the ESDSR in terms of number of dead nodes at the end of the simulation. This is because we used a cluster head to forward collected data, even if a node is within the transmission range of an anchor. When the area grows bigger, the routing path is long enough so that the overhead of routing packets to the cluster head can be reduced. Therefore, the CEER is more *scalable* than the ESDSR when the area grows bigger, as illustrated in Fig. 15.

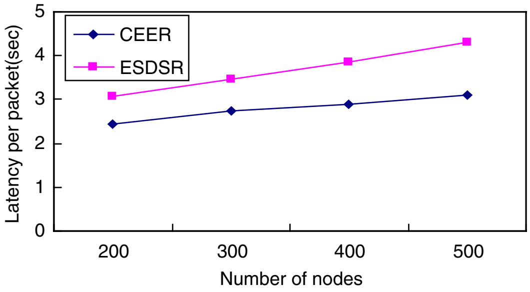


Fig. 13. Latency per packet vs. number of nodes.

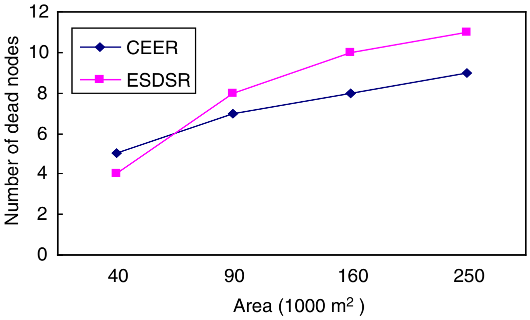


Fig. 15. Number of dead nodes at the end of the simulation.

6. Conclusion

In this paper, we have presented an efficient color-theory-based energy efficient routing (CEER) algorithm based on a color-theory-based dynamic localization (CDL) algorithm. The key idea of the CEER is that it selects those cluster members that are closer to the anchor than itself as next possible hops by comparing their RGB values. Among the selected cluster members, the sensor node with the highest energy level is chosen as the next hop. Simulation results have shown that our routing algorithm can save up to 50–60% energy compared to the ESDSR [2] in mobile wireless sensor networks.

In addition, the latency per packet of the CEER is 50% less than that of the ESDSR.

References

[1] Shen-Hai Shee, Kuochen Wang, I.L. Hsieh, Color-theory-based dynamic localization in mobile wireless sensor networks, in: Proceedings of Workshop on Wireless, Ad Hoc, Sensor Networks, August 2005.

[2] Mohammed Tarique, Kemal E. Tepe, Mohammad Naserian, Energy saving dynamic source routing for ad hoc wireless networks, in: Proceedings of Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, April 2005, pp. 305–310.

[3] J.G. Lin, S.V. Rao, Mobility-enhanced positioning in ad hoc networks, in: Proceedings of the IEEE Wireless Communications and Networks Conference, vol. 3, March 2003, pp. 1832–1837.

[4] W. Heinzelman, J. Kulic, H. Balakrishnan, Adaptive protocols for information dissemination in wireless sensor networks, in: Proceedings of the 5th ACM/IEEE MOBICOM, Seattle, WA, August 1999, pp. 174–185.

[5] F. Ye, et al., A two-tier data dissemination model for large-scale wireless sensor networks, in: Proceedings of the ACM/IEEE MOBICOM, 2002, pp. 148–159.

[6] Y. Yu, D. Estrin, R. Govindan, Geographical and Energy-Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks, UCLA Comp. Sci. Dept. Tech. Rep., UCLA-CSD TR-010023, May 2001.

[7] J. Borch, D.B. Johnson, D.A. Maltz, The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks, IETF Internet Draft, draft-ietf-manet-dsr-00.txt, March 1998.

[8] J. Yoon, M. Liu, B. Noble, Random waypoint considered harmful, in: Proceedings of the IEEE INFOCOM. April 2003, pp. 1312–1321.

[9] Hongyi Wu, Chong Wang, Nian-Feng Tzeng, Novel self-configurable positioning technique for multihop wireless networks, IEEE/ACM Transaction on Networking 13 (3) (2005) 609–621.

[10] Sooyeon Kim, Sang H. Son, John A. Stankovic, Yanghee Choi, Data dissemination over wireless sensor networks, in: IEEE Communication Letters, September 2004, pp. 561–563.

[11] C. Intanagonwiwat, R. Govindan, D. Estrin, Direct diffusion: a scalable and robust communication paradigm for sensor networks, in: Proceedings of the ACM MobiCom, 2000, pp. 56–67.

[12] R.-C. Chang, C.-J. Kuo, An energy efficient routing mechanism for wireless sensor networks, in: Proceedings of the Advanced Information Networking, and Applications, April 2006.

[13] S. Lindsey, C. Raghavendra, PEGASIS: power-efficient gathering in sensor information systems, Proceedings of the IEEE Aerospace Conference 3 (9–16) (2002) 1125–1130.

[14] B. Chen et al., SPAN: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks, Wireless Networks 8 (5) (2002) 481–494.

[15] L. Wang et al., Adaptive multipath source routing in ad hoc networks, in: Proceedings of the IEEE International Conference on Computer Communications, 2001, pp. 867–871.

[16] E. Vishnevsky, RGB to HSV & HSV to RGB. <http://www.cs.rit.edu/~ncs/color/t_convert.html>.

[17] DevX.com, “Color Models”. <http://www.devx.com/projectcool/Article/19954/0/page/7>.

[18] A. Taheri, A. Singh, A. Emmanuel, Location fingerprinting on infrastructure 802.11 wireless local area networks (WLANs) using Locus, in: Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks, November 2004, pp. 676–683.

Tai-Jung Chang received the B.S. degree in Computer and Information Science from the National Chiao Tung University, Taiwan, in 2002, and the M.S degree in Computer Science and Engineering from the same university in 2006. He is currently a System Engineer in Sunplus Technology Co., Ltd. He served in the army as a second lieutenant artillery platoon leader from 2002 to 2004. His research interests include wireless sensor networks and network file systems for consumer electronic products.

Kuochen Wang received the B.S. degree in control engineering from the National Chiao Tung University, Taiwan, in 1978, and the M.S. and Ph.D. degrees in electrical engineering from the University of Arizona in 1986 and 1991, respectively. He is currently a Professor in the Department of Computer Science, National Chiao Tung University and a Acting Director of the Computer and Network Center at the university. From 1980 to 1984, he was a Senior Engineer at the Directorate General of Telecommunications in Taiwan. He served in the army as a second lieutenant communication platoon leader from 1978 to 1980. His research interests include wireless (ad hoc/sensor) networks, mobile computing, and power management for portable wireless/multimedia devices.

Yi-Ling Hsieh received the B.S. and M.S. degrees in computer and information science from the National Chiao Tung University, Taiwan, in 2003 and 2005, respectively. She is currently a Ph.D. student in the Department of Computer Science, National Chiao Tung University. Her research interests include wireless ad hoc/sensor networks, mobile computing, and efficient energy management for portable wireless/multimedia devices.