PAPER IEICE/IEEE Joint Special Section on Autonomous Decentralized Systems Theories and Application Deployments

Adaptive Routing Protocol with Energy Efficiency and Event Clustering for Wireless Sensor Networks

Vinh TRAN QUANG[†a)], ***Student Member and*** **Takumi MIYOSHI**[††], ***Member***

SUMMARY Wireless sensor network (WSN) is a promising approach for a variety of applications. Routing protocol for WSNs is very challenging because it should be simple, scalable, energy-efficient, and robust to deal with a very large number of nodes, and also self-configurable to node failures and changes of the network topology dynamically. Recently, many researchers have focused on developing hierarchical protocols for WSNs. However, most protocols in the literatures cannot scale well to large sensor networks and difficult to apply in the real applications. In this paper, we propose a novel adaptive routing protocol for WSNs called ARPEES. The main design features of the proposed method are: energy efficiency, dynamic event clustering, and multi-hop relay considering the trade-off relationship between the residual energy available of relay nodes and distance from the relay node to the base station. With a distributed and light overhead traffic approach, we spread energy consumption required for aggregating data and relaying them to different sensor nodes to prolong the lifetime of the whole network. In this method, we consider energy and distance as the parameters in the proposed function to select relay nodes and finally select the optimal path among cluster heads, relay nodes and the base station. The simulation results show that our routing protocol achieves better performance than other previous routing protocols.

key words: *energy efficiency, event clustering, routing protocol, wireless sensor networks*

1. Introduction

Wireless sensor networks are distributed event-based systems that consist of thousands of tiny and low-cost nodes with their limitations in terms of energy, computing power, and communication possibilities. WSNs differ from the traditional communication networks in several ways. First, because of the requirement of operation in an unattended mode or even potentially hostile locations, the battery of sensor nodes is not rechargeable. Secondly, since various sensor nodes often detect common phenomena, there is likely to be redundant low-rate data. Furthermore, a large number of sensor nodes are densely deployed in a random manner, and thus it is difficult to know network topology.

WSNs have been using to monitor many kinds of ambient conditions that include temperature, humidity, pressure, vehicular movement, soil makeup, and so on. There are many different applications of WSNs with different communication patterns. The main communication pattern is an asymmetric many-to-one data flow mainly from sensor nodes to a sink node. Therefore, most routing algorithms proposed for ad hoc networks are not directly applicable to WSNs without considerable modifications [1]. This leads to the main goal of achieving maximum throughput with minimal energy consumption in design and implement of routing protocols for WSNs [2], [3].

To model energy consumption, three basic states of a node can be identified: sensing, data processing and data communications. Among them, data communications consume significant amount of energy while transmitting or receiving packets. Minimizing the number of communications by eliminating or aggregating redundant sensed data will definitely save a large amount of energy, which ultimately leads to the network longevity.

Routing protocol for WSNs is very challenging because it should be simple, scalable, energy-efficient, and robust to deal with a very large number of nodes with their limitations, and also to be self-reconfigurable to node failures and changes of the network topology. Many researchers have focused on developing hierarchical protocols for WSNs [1]–[7]. Hierarchical algorithms separate the nodes into clusters. The communication among the nodes in different clusters is possible only through the selected leaders in the clusters (cluster heads). They are responsible to execute the managing operation and transmission of the collected data in their region. However, most protocols in the previous literatures cannot scale well to large sensor fields and accordingly they are hardly applied to the real applications. One of the main reasons is that nodes require knowledge of their locations, either by manual configuration or by using other location sensing techniques requiring extra hardware, for example equipped with GPS-capable antenna [19], [20]. On the other hand, the assumption that all sensor nodes are able to communicate directly with any sensor (i.e. one-hop transmission) in the field, including the base station [4]–[7], does not allow the network to be scalable. Cluster heads must spend more energy for longer-distance transmission directly to the remote base station. Such the transmission architecture will work fine on small networks, because every cluster head can reach the base station. On large networks, however, a cluster head cannot send data to a base station because of the distance between them. Moreover, the selection of cluster head is optimized with some probabilities, which does not always follow the geographical locations. As a result, there is a possibility that the cluster heads could be concentrated in one part of the network.

Manuscript received December 27, 2007.
Manuscript revised April 18, 2008.

[†]The author is with the Graduate School of Engineering, Shibaura Institute of Technology, Saitama-shi, 337-8570 Japan.
[††]The author is with the Department of Electronic Information Systems, College of Systems Engineering, Shibaura Institute of Technology, Saitama-shi, 337-8570 Japan.
a) E-mail: m706501@sic.shibaura-it.ac.jp

DOI: 10.1093/ietcom/e91–b.9.2795

In this paper, we propose a novel adaptive routing protocol for WSNs called ARPEES. The main design features of the proposed method are energy efficiency, dynamic event clustering, and multi-hop relay considering the trade-off relationship between the residual energy available of relay nodes and distance from the relay node to the base station. With the distributed and light overhead traffic approach, we spread energy consumption required for gathering, aggregating data and relaying data to the difference sensor nodes to prolong the lifetime of the whole network. In this method, we consider energy and distance as the parameters in the proposed function to select relay nodes and finally select the optimal path among cluster heads, relay nodes, and the base station.

The remainder of the paper is organized as follows: Section 2 discusses the related work. Section 3 describes a sensor network model that we consider. Section 4 presents the ARPEES algorithm in details. We use computer simulation to evaluate the performance of the proposed method in Sect. 5. We also discuss the system performance and mention the future work in Sect. 6. Finally the last section summarizes the conclusions of the paper.

2. Related Work

In recent years, many researchers have focused on developing routing protocols for wireless sensor networks. Based on the techniques, we can classify those routing protocols for WSNs as four main categories: data centric, hierarchical, location-based, and network flow with QoS awareness algorithm.

Data centric algorithms like Direct Diffusion [11], TTDD [12] and SPIN [13] protocols are based on network queries where the collected data is named and stored as the interest entry in local nodes to allow them to search and obtain only the desired information. The source sends data to a sink only when the sink has sent a query for the data. In Direct Diffusion [11], the interest entry contains a timestamp field and several gradient fields. As the interest is propagated throughout the sensor network, the gradients from the source back to the sink are set up. TTDD [12] constructs and maintains a uniform grid structure by dividing the field into small grid cells, and confines flooding within the local grid cells only. However, the grid construction and maintenance per each source leads to a considerable overhead. SPIN [13] uses high-level data descriptors, called meta-data, for negotiation. The adaptation to available resource on the sensor nodes helps them eliminate most of the redundant data transfer.

Hierarchical algorithms separate the nodes into clusters. The communication among the nodes in different clusters is possible only by the selected leaders in each cluster (cluster heads). They are responsible to execute the management and transmission of the collected data in their regions. Some hierarchical algorithms are LEACH [4], TEEN [5], APTEEN [6], PEGASIS [7], HPEQ [9] and OEDSR [10].

Heinzelman, et al. designed and implemented a cluster-based protocol called LEACH that uses randomization of local cluster heads to distribute the energy load among the sensors in the network [4]. In LEACH, the nodes organize themselves into local clusters, with one node acting as the local base station or a cluster head. The cluster heads collect data from all the sensor nodes in their own clusters, aggregate the collected data, and transmit the data directly to the base station. While this algorithm works well in small networks, it is not suitable for large networks under the assumption that all nodes can communicate directly with the base station. Nodes in TEEN and APTEEN are designed to respond to sudden changes in the sensed attribute when the attribute exceeds a user-defined threshold. On the other hand, PEGASIS protocol presented in [7] forms a chain including all nodes in the network using a greedy algorithm so that each node transmits to and receives from a neighbor. In each round, the randomly selected node in the chain takes turns to transmit the aggregated information to the base station. However, this algorithm suffers from the excessive delay created by the distant node in a single chain when the network size grows up. LEACH, TEEN, APTEEN, and PEGASIS assume that the position of the base station is fixed and every node in the sensor network can directly communicate to it. This assumption is not always valid in larger networks, because the base station could be out of range so that cluster heads have to adopt multi-hop routing protocols.

In HPEQ [9], Boukerche, et al. proposed a hierarchical, periodic, event-driven, query-based, and cluster-based routing protocol that groups sensor nodes to efficiently relay the sensed data to the sink. In HPEQ, nodes with more residual energy are selected as the aggregator nodes to relay data to the sink. HPEQ uses the publish/subscribe paradigm to disseminate requests across the network. However, this protocol has the shortcomings. First, a sink starts a flooding mechanism to configure the whole network. This method may increase the overhead due to many message transmissions in the network. Secondly, each node has to know the number of hops that represents how far the sink is, and also has to maintain routing table to find the neighbor closest to the sink in order to forward data to it. Routing tables for distributed sensor networks increase exponentially as nodes are added and, therefore, this process is complicated and accordingly consumes more energy.

OEDSR [10] is an on-demand routing protocol for WSNs that minimizes a different link cost factor, which is defined using available energy, end to end delay, and distance from a node to the base station. This information is treated as the link cost and it is recorded in the neighbor table for the one-hop neighbors. In OEDSR, clusters are formed in the sub-networks and the rest of the network outside the sub-network is treated as an ad hoc wireless network. Then the node with the highest energy available elects itself as the temporary sub-network head. The function of the temporary head is to calculate the required number of cluster heads and select the cluster-head nodes. Later, the sub-networks organize themselves into clusters and elect cluster heads in the sub-network portion. The weak point of OEDSR is that it requires maintaining the link cost factors as a routing table and need the temporary head to control cluster-head selection. Furthermore, the performance of the OEDSR protocol was compared with AODV, DSR, and Bellman Ford protocol in the literature [10]. Since those protocols are not designed for WSNs, the comparison might not be equitable.

Location-based algorithms such as GAF [14] and GEAR [15] use the information of node position, usually obtained by a global positioning system (GPS), to seek and transmit data in the desired network region. Lastly, network flow and QoS awareness algorithms use network traffic models and apply mechanisms to support the evaluation of QoS requirements in routing functions like SPEED [16] and SAR [17] protocols.

One of important issues in many sensor network algorithms is the distance estimation when the positions of nodes cannot be decided beforehand or, if nodes are mobile. Nodes could be equipped with GPS to provide them with absolute positions, but this is currently an infeasible solution with a network of thousands of nodes. RADAR is a user tracking system [18]. It uses empirical as well as mathematical models to estimate the distance between sender and receiver based on received signal strength indication (RSSI). The advantage of employing the RSSI values is that no extra hardware (e.g. GPS, ultrasonic or infra-red) is needed in small wireless devices and shows good characteristics with respect to power consumption, size and cost. Because of its attractiveness, researchers have extensively considered the use of this indication. A common distance estimation technique is to use the property of signal degradation while traveling in a space to determine the mutual distance. The method to estimate distance based on RSSI is that nodes exchange beacon messages. At the receiver, RSSI, supported by sensor node hardware such as the Berkeley Motes, is measured by the RF energy received and it is closely related to distance. The results of this paper showed the significantly accurate measurement of real distance between the sender and the receiver. In our method, we use RSSI to measure the distances between two sensor nodes and that between each node and the base station.

3. Sensor Network Model

3.1 Network Model

The wireless sensor network investigated in this paper is modeled as follows:

- Sensor nodes are quasi-stationary. This assumption is typical for most sensor network applications. The base station is fixed and located far from the sensors.
- Sensor nodes are homogeneous with the same capabilities (processing/communication) and energy-constrained since they remain unattended after the deployment. Therefore, recharging battery is not possible.
- Each node has the fixed number of transmission power levels and it can manage its amount of residual energy.
- Most sensor nodes can not communicate directly with the faraway base station. To obtain the distance from sensor node to the base station, we assumed that after deploying sensor nodes, the beacon messages from the base station, which is assumed to have unlimited energy and location awareness, are broadcasted to all the sensor nodes. Each sensor node that received the beacon from the base station can estimate its distance to the base station by measuring RSSI, mentioned in [18]. In our work, each sensor node can obtain the distance between each other by exchanging *REQ_CLUSTER* messages or *REQ_RELAY* messages described in Sect. 4.

3.2 Radio Model

For analyzing the transmission process, we use the same conditions in LEACH, an application-specific protocol architecture for wireless microsensor networks, with the simple first order radio model [4]:

- k is the number of bits per packet/transmission;
- $E_{Tx-elec} = E_{Rx-elec} = kE_{elec}$ is the power requirement on the electronics devices for transmitting and receiving the data;
- E_{amp} is the transmission amplification energy.
- Transmission cost to transmit k-bit message between any two nodes over distance d is given by the following equation:

$$E_{Tx}(k,d) = E_{Tx-elec}(k) + E_{amp}(k,d) \tag{1}$$

where E_{amp} varies according to the distance d between a sender and a receiver. $E_{amp} = E_{fs}$ in free space model when $d < d_0$; and $E_{amp} = E_{mp}$ in multi-path model when $d \geq d_o$. d_0 is the constant distance that depends on the environment. Thus the above equation can be transformed as follows:

$$E_{Tx}(k,d) = \begin{cases} kE_{elec} + kE_{fs}d^2, \ d < d_0 \\ kE_{elec} + kE_{fs}d^4, \ d \geq d_0 \end{cases} \tag{2}$$

To receive k-bit message, a node expends:

$$E_{Rx}(k) = E_{Rx-elec}(k) = kE_{elec} \tag{3}$$

The residual energy of every node after one data communication is calculated as follows:

$$E_{Res} = E_{Initial} - (E_{Tx} + E_{Rx}) \tag{4}$$

Thus among three domains, sensing, processing and data communication, a sensor node consumes the most energy in data communication. In our protocol, we aim to reduce the number of control messages, the length of each message, and the distance in the data communication function for less energy consumption and finally to improve network lifetime.

4. Proposed Algorithm

4.1 Protocol Features

WSNs are supposed to have several hundred or thousand sensor nodes. Therefore, the protocols and algorithms should be scalable, which is a major problem observed in ad hoc networks. Additionally, because of fading wireless channels, malfunctioning of a few nodes, due to hardware or lack of energy, in the network will cause significant topology changes. WSNs are thus prone to failures or intermittent connectivity. The problem leads to greater consumption of energy in rerouting for transmitting packets. Therefore, energy-efficient schemes are very important feature in designing routing protocol for WSNs.

In this section, we propose a novel routing protocol for wireless sensor networks called ARPEES, Adaptive Routing Protocol with Energy efficiency and Event clustering for wireless Sensor networks. The main design features of ARPEES are adaptive among energy efficiency, dynamic event clustering, and multi-hop transmission. Our protocol aims to find the optimal relay path to transmit the aggregated data to the remote base station considering the tradeoff relationship between the residual energy available of relay nodes (energy efficiency) and the distance from a relay node to the base station (shortest path). In the protocol, we use energy and distance as the parameters to discover relaying paths. An example of the process to select the relay node (RN_1, RN_2, RN_3,) and to form relaying route from the cluster head (CH) to the remote base station (BS) is illustrated in Fig. 1. The relaying route is $CH - RN_1 - RN_2 - RN_3 - BS$.

The performance metrics of our method are as follows:

- Power consumption and load balance: We aims to reduce power consumption of sensors that have less residual energy by spread energy consumption required for gathering, aggregating, and relaying data to the different sensor nodes.
- Distributed and dynamic approach: Our goal is to design an on-demand distributed cluster formation algorithm, where the sensed attribute of the events or objects can be used to form clusters. A node can make the decision without any centralized control. With dynamic event-clustering method, clusters are formed based on where and when the event occurs in the environment. Therefore, the size of the active part in the network is dynamic and the number of clusters does not depend on that of sensor nodes in the network. Moreover, this method can save energy because only a portion of the network becomes active in response to an event.
- Optimal route: The definition of an optimal route is very important for the design of a routing protocol. In general, the number of hops on the path or the overall link cost is used as the metric to determine the optimal route.
- Scalability: One of the requirements for our protocol is scalability, meaning that routing protocol should scale well in large applications with rapid topology changes against link failures. The addition of more nodes to replace the dead or failure nodes in the existing network should not affect the functionality of the network.
- Control traffic overhead: The routing protocols must minimize the control traffic overhead required for forming clusters, selecting cluster heads, and processes on relay nodes.
- Efficiency: The routes selected by the routing protocol affect the performance of the network in terms of delay, throughput, and energy efficiency. Therefore, the routing protocol should take into consideration the limited resources such as battery power, memory, and bandwidth, and aim at improving the overall network efficiency.

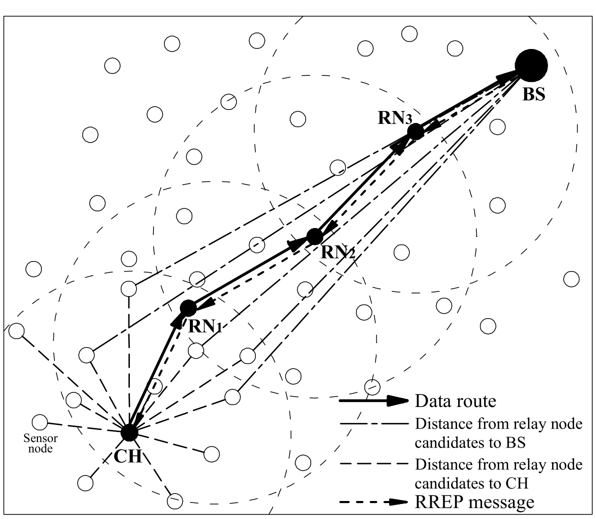

Fig. 1 Selecting the relay nodes by using the proposed function of residual energy, distance from the cluster head (CH) to a relay node (RN) and distance from RN to the base station (BS).

The operation of ARPEES is segmented into rounds, and each round has two stages shown in Fig. 2: Forming clusters and selecting each cluster head stage followed by data transmission stage when data is transferred to the base

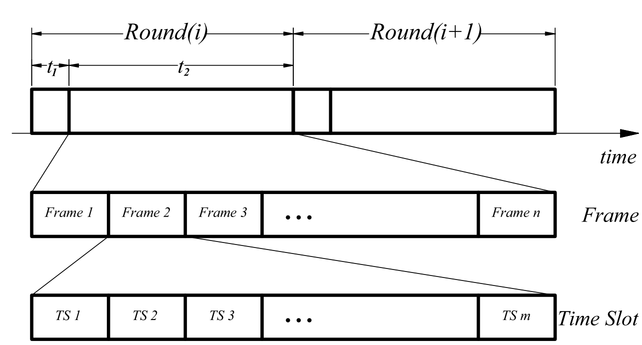

Fig. 2 Showing the time line of ARPEES protocol operation: t_1 is the first stage: forming cluster and selecting cluster head; t_2 is the second stage: selecting relay nodes and transmitting data.

station via relay nodes. In the following subsections, we discuss each of them in detail.

4.2 Forming Cluster and Selecting Cluster Head Algorithm

Initially, all nodes in the network are in sleep mode to save battery power. When an event is detected in the network, nodes near the event become activated and measure the specific sensed attribute. If the sensed attribute value is greater than a predefined threshold, then those nodes execute an algorithm to form a cluster and select a cluster head. Those nodes broadcast *REQ_CLUSTER* message to their neighbors. This message consists of the node ID, the amount of residual energy and descriptive information of the sensed data from the event:

$$REQ\_CLUSTER\{ID(i), E_{Res}(i), I(i)\} \quad (5)$$

where *ID(i)* and $E_{Res}(i)$ are the identification and residual energy of node i respectively. *I(i)* is descriptive information of the data sensed by node i. After that, nodes set their timer to t_1. During the period time t_1, each node will receive *REQ_CLUSTER* messages from all the other nodes within the cluster and executes the *Cluster_Head* function as follows:

$$\begin{aligned} &F_{CH}(i) = E_{Res}(i) \times I(i), \forall i \in X \\ &Max\, F_{CH}(i) \xrightarrow[set\,as]{\forall i \in X} Cluster\, Head \end{aligned} \quad (6)$$

where X is a set of nodes activated by the event.

When the timer t_1 expires, the node that has the maximum solution of $F_{CH}(i)$ mentioned in Eq. (6) sets itself as the cluster head. The cluster head stores the node ID of all the nodes in the set of active nodes X, and creates the TDMA schedule [4] to arranging each node when they can transmit their sensed data to the cluster head. The function of this schedule is to avoid the collision on data transmission and to keep the synchronization among all the nodes within the cluster. Other nodes set themselves as non cluster heads and wait for receiving the TDMA schedule from the cluster head. Meanwhile, they can turn their radio components off except for their own transmission period. The flow chat of the first stage is illustrated in Fig. 3.

Figure 4 illustrates the forming cluster based upon the event occurrence and selecting cluster head algorithm. In Fig. 4(a), nodes from n_1 to n_9 detected the event and changed to active mode. And then they broadcast *REQ_CLUSTER* message to exchange their information. In Fig. 4(b), we illustrate only nodes n_2 and n_5 broadcasting their information; other nodes do the same process. After a given timer t for calculating *Cluster_Head* function, in this case node n_5 that has maximum solution sets itself as cluster head, and sends the TDMA schedule to neighbors as shown in Fig. 4(c). Following the TDMA schedule, each node transmits sensed data to the cluster head n_5. The cluster head n_5 gathers data from its member and performs data aggregation as shown in Fig. 4(d).

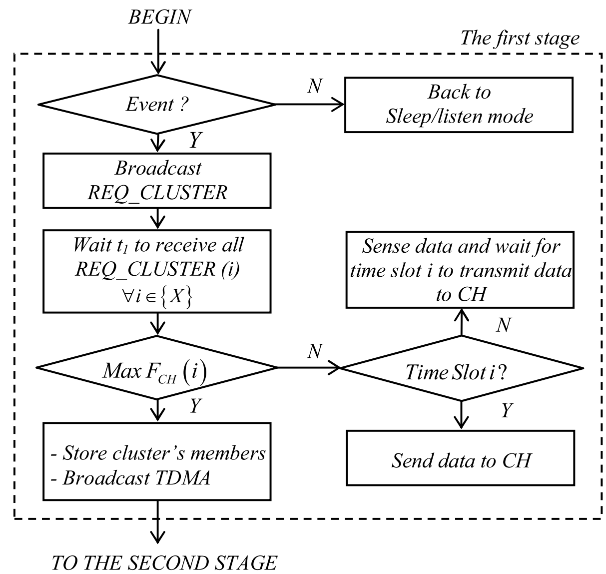

Fig. 3 Flow chat of the ARPEES protocol in the first stage.

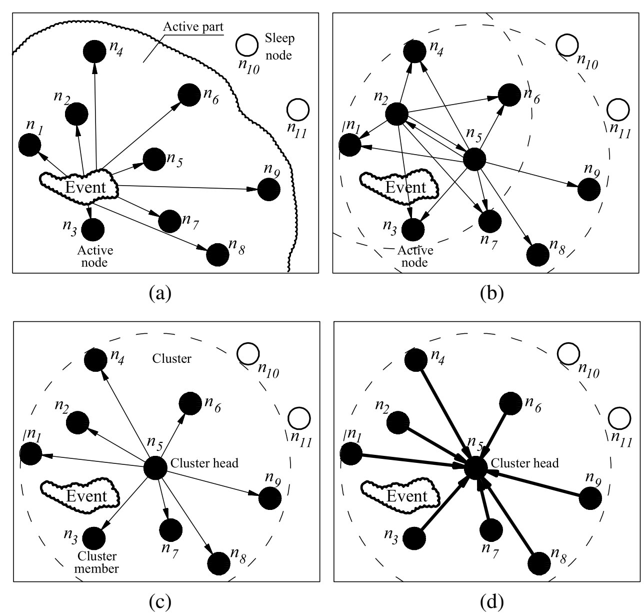

Fig. 4 Forming cluster and selecting cluster head algorithm based on event/target occurrence: (a) the nodes in a part of network become active when detected the event. (b) Active nodes exchange *REQ_CLUSTER* message. (c) Node n_5 becomes cluster head with maximum solution of $F_{CH}(n_5)$ function and broadcasts TDMA schedule. (d) Cluster members transmit sensed data to the cluster head in their time slot.

Our algorithm ensures that the node with maximum energy available and nearest the event is selected as the cluster head. Furthermore, in this algorithm we use only one message type to create clusters and select cluster heads. Therefore, with the light-weight routing protocol we can reduce not only the number of overhead messages but also that of the data packets transferred from nodes to their cluster head that will be described in the second stage later, because a cluster head is the node near the event and it already has more sensed data than the other nodes faraway from the event.

4.3 Data Transmission Stage

In this stage, we use intermediate nodes to relay packets from a cluster head to the base station. Such the intermediate nodes (relay nodes), in turn, have to decide to which neighbor to forward an incoming packet that is not destined for themselves. The data transmission stage consists of three major activities.

(a) Gathering data within cluster

Using the TDMA schedule scheme described above, each sensor node transmits the sensed information to its cluster head during their allocated transmission period. The most simple form for power saving is to turn the transceiver off when it is not required and to turn it on only when desired. The radio of each non-cluster-head node can be turned off until the allocated transmission time, and thus energy consumption will be minimized in these nodes. The cluster head must keep on its receiving device to receive all the data from the nodes in the cluster. A key issue here is that sensor nodes are grouped into a cluster surrounding the event: These transmissions consume minimal energy due to small spatial separations between the cluster head and the sensing nodes.

Another key issue to be addressed here is the priority of each node in the TDMA schedule. As mentioned above, a cluster head has information about data descriptors in the messages of all other nodes, *I(i)*, within their own clusters. Thus, it can arrange the order and a period of time for each node to transmit data. The node that has more data descriptors will transmit first with more time slots than the others. According to the arrangement, all nodes will be allocated in the particular time slot in which they sense the environment and transmit the sensed data to their cluster head.

(b) Performing data aggregation

Energy expended in data processing is much less compared to data communication. Hence, data aggregation by local processing is very important to minimize power consumption. To avoid redundant data transfer, the cluster head performs data aggregation function on the collected data, and accordingly reduces the amount of raw data that need to be sent to the base station. The compressed data, along with the information required by the base station, are then transmitted to the base station via multi-hop scheme.

(c) Selecting relay nodes and creating a route

In our protocol, the cluster head that has a data packet ready to be transmitted selects relay nodes to send it to the base station with a multi-hop route, instead of transmitting the packet directly such as LEACH [4].

Initially, the cluster head broadcasts a *REQ_RELAY* message to all nodes within its range to search the relay node. Each node that received the *REQ_RELAY* message calculates its residual energy and distance to the base station, puts the results into an *ACK_RELAY* message, and sends it back to the cluster head. The cluster head waits for receiving all *ACK_RELAY* messages from relay node candidates and checks if it can transmit data to the base station directly. Otherwise the cluster head performs *Relay_Node* function to choose the relay node. The desired relay node should satisfy the following features:

- The relay node should have the maximum amount of residual energy.
- The relay node should be located as near the base station as possible. This means that it has the maximum distance from the cluster head and the minimum distance to the base station.
- The multi-hop path should be almost straight between the cluster head and the base station.

After receiving the *REQ_RELAY* messages from all the candidates, the cluster head will have the needed information about the one-hop topology. Next, the cluster head calculates the *Relay_Node* function to choose the best relay node among all candidates. The function is defined as follows:

$$\begin{aligned} &F_{RN}(j) = E_{Res}(j) \times \tfrac{d(CH,j)}{d(j,BS)} \times \cos\alpha_j, \forall j \in Y \\ &Max\, F_{RN}(j) \xrightarrow[set\ as]{\forall j \in Y} Relay\ Node \end{aligned} \tag{7}$$

where $E_{Res}(j)$ is energy available of sensor node candidate j, $d(CH, j)$ and $d(j, BS)$ are distance from the cluster head to node j and distance from node j to the base station, respectively. Y represents a set of candidates for evaluating the relay node within the radio range of cluster head. α_j is the critical angle value created by node j, the cluster head, and the base station. $\cos\alpha_j$ can be obtained by the following geometric calculation:

$$\cos\alpha_j = \frac{d(CH, j)^2 + d(CH, BS)^2 - d(j, BS)^2}{2d(CH, j)d(CH, BS)} \tag{8}$$

The node that has the maximum value of $F_{RN}(j)$ is selected as the relay node. In the next hop, the relay node in turn served as the cluster head to find a next relay node. This process for searching the relay node is repeated until reaching the base station. Finally, the optimal transmission path will be created by relay nodes between the cluster head and the base station. Figure 5 presents the flow chat of the ARPEES protocol in the second stage. In Fig. 6 we illustrate an example of selecting relay node function. Cluster head n_1 broadcasts a *REQ_RELAY* message to search relay node. Sensor nodes n_2, n_3, n_4 within n_1's transmission range will receive the message and they reply an *ACK_RELAY*. After receiving all replies, the cluster head n_1 performs the *Relay_Node* function. In the example, sensor node n_3 has maximum solution of the *Relay_Node* function and is chosen as the relay node.

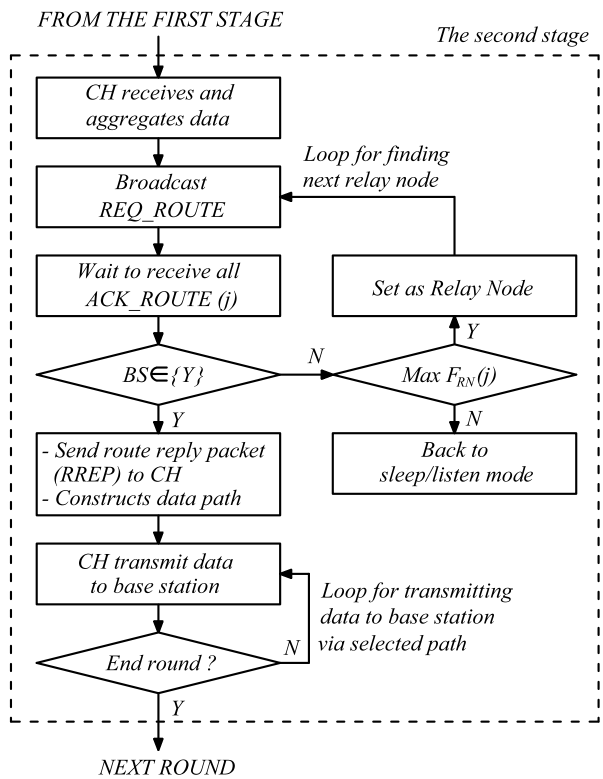

Fig. 5 Flow chat of the ARPEES protocol in the second stage.

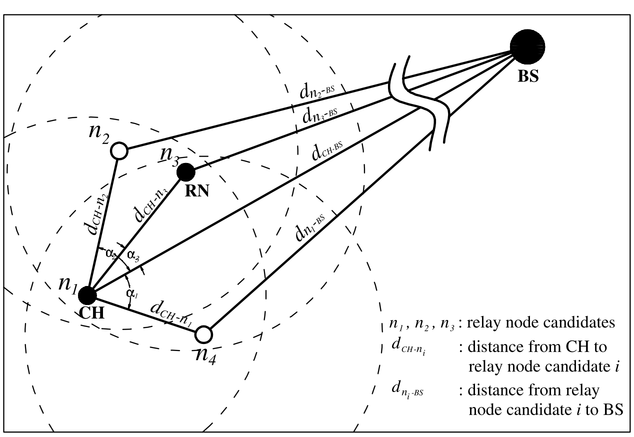

Fig. 6 An example of relaying node selection.

5. Performance Evaluation

The proposed ARPEES algorithm was implemented in the OMNeT++ simulator [21], which is a public-source, component-based, modular and open-architecture simulation environment with strong GUI support and an embeddable simulation kernel. Simulations were performed by varying network area and with variable numbers of nodes in the network. The LEACH, OEDSR, and HPEQ routing protocols were also employed for the comparison.

Table 1 Simulation parameters.

Parameter	Value
Initial energy($E_{initial}$)	2 *Joule*
Data packet size	500 *byte*
Broadcast packet size	25 *byte*
Packet header size	25 *byte*
Dataframes	30
Energy of transceiver electron(E_{elec})	50 nJ/bit
Energy for transmission in free space model(E_{fs})	10$pJ/bit/m^2$
Energy for transmission in multi-path model(E_{fs})	0.0013 $pJ/bit/m^4$
Threshold distance(d_0)	75 m
Sensing range	60 m

5.1 Simulation Method and Metrics

The ARPEES protocol concentrates on the most crucial aspect: energy efficiency. Our primary performance is measured by quantitative metrics of the average amount of residual energy, the number of living nodes, and network life time due to energy awareness. Moreover, we are interested in how this protocol scales to large sensor networks.

In our simulations, we used network model and radio model mentioned in Sect. 2. We assume that sensor nodes that initially have 2-Joule energy are uniformly dispersed onto a field with square dimension. The number of data frames to be transferred in each round is set to 30. The data message size for all simulations is fixed at 500 bytes, including a 25-byte packet header. We stop the simulation after the two-thirds nodes are dead. Table 1 shows more detail about the simulation parameters. Figure 7 illustrates a snapshot of routing paths in the ARPEES simulation program. The cube-shaped nodes are cluster heads, and the nodes with rectangle shape are relay nodes. The thin arrows are data packets from cluster members to their cluster head. The thick arrows are the routes selected by the ARPEES protocol to transmit aggregated data from the cluster heads to the base station.

5.2 Result Analysis

To measure the metrics of the average amount of residual energy and the number of nodes alive, we ran the simulation with 1,200 sensor nodes uniformly dispersed onto a 500 × 500 meters square field in the first simulation. The base station was located at co-ordinate (250, 600), and 100-meter away from the closest sensor node.

The results of this simulation show that network lifetimes are 585, 224, 519, and 489 rounds for ARPEES, LEACH, OEDSR and HPEQ, respectively. ARPEES outperforms LEACH, OEDSR and HPEQ as the aspect of network lifetime with more two times longer than LEACH, 13% and 20% longer than OEDSR and HPEQ respectively in this network scenario. Figure 8 shows the variation of the total amount of residual energy over the network lifetime of

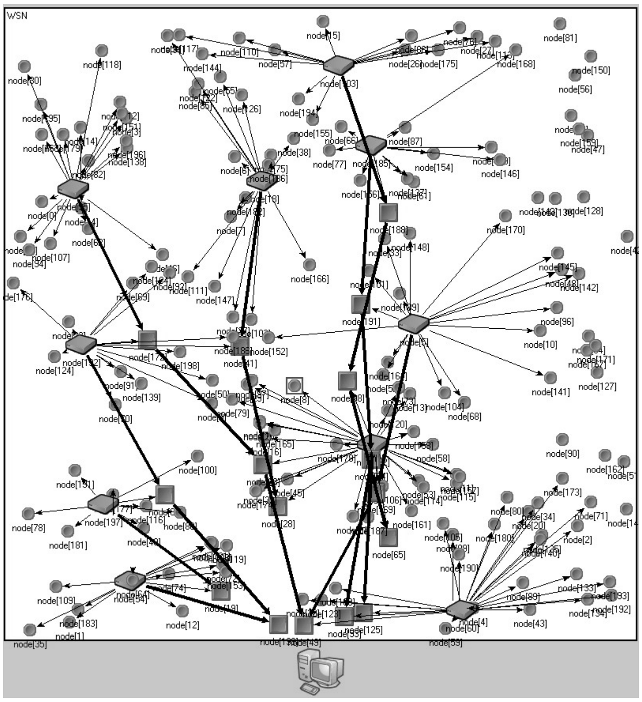

Fig. 7 A snapshot of ARPEES routing protocol in the simulation program: The cube nodes are cluster heads; rectangle nodes are relay nodes; thin arrows are data transmissions from cluster members to their cluster heads; thick arrows are relaying paths created by the ARPEES protocol.

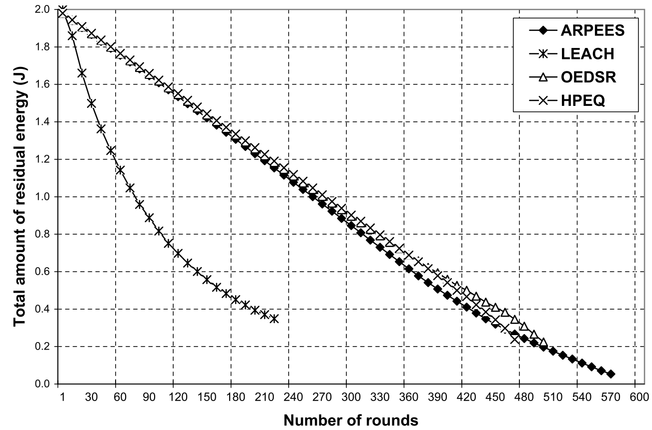

Fig. 8 The total amount of residual energy over the number of rounds (network lifetime) with four methods in the 500 × 500 meters network area of 1200 sensor nodes.

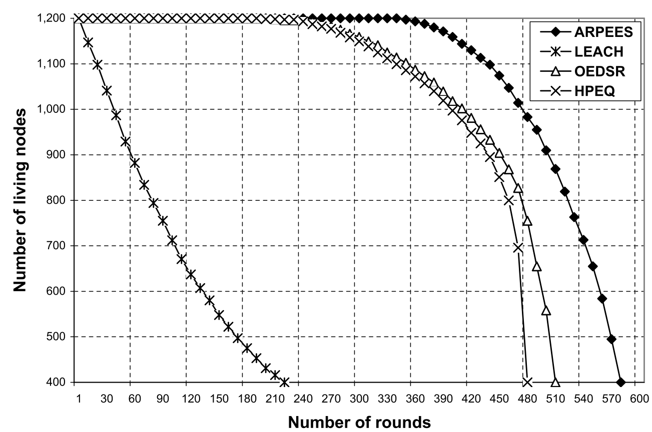

Fig. 9 The number of living nodes over the number of rounds with four methods in the 500 × 500 meters network area of 1200 sensor nodes.

the four methods. Clearly, ARPEES performs better than LEACH as it consumes less energy than LEACH. This is why ARPEES prolongs the network lifetime over two times longer than LEACH mentioned above.

(1) Load balance

Clearly, where many nodes are dead or density of nodes decreases significantly, we will get low network connectivity. One of our main goals in this proposal is to avoid this problem. We aim to balance the energy consumption among the nodes to avoid *hot spot*, which causes quick deaths of the nodes due to their overload. This improvement gained through ARPEES is further exemplified by comparing the number of nodes that remain alive after each round activities plotted in Fig. 9. This plot shows that ARPEES has more than twice the number of living nodes than LEACH in this network scenario. Furthermore, the energy distribution can be evaluated by the factor of the first node dead. In our simulations, a node is considered as the dead node if it consumes 90% of its initial energy. With the ARPEES protocol, the cluster heads and relay nodes are selected depending upon their energy. Therefore, the energy consumption for transmitting data to the faraway base station is divided and well balanced among the relay nodes. As the results plotted in Fig. 9, ARPEES performs 350-round activities before the first node dead, while LEACH does only 7 rounds. Those metrics are 200 and 189 rounds for OEDSR and HPEQ, respectively. With the number of living nodes larger than three other comparators, we can maintain higher connectivity to the base station than LEACH and other comparators.

Our method can not only decrease energy consumption for the cluster-head selection and data transmission but also achieve load balancing, for example, to evenly scatter energy consumption required for relaying data to different nodes. Figure 10 shows the remaining energy of the 200 randomly selected sensor nodes in the first simulation after 220 rounds and Fig. 11 measures the same metrics at the end of the simulation. In both figures, the horizontal axes are the sensor node ID, the vertical axes are the value of remaining energy of the corresponding sensor nodes. Clearly, in ARPEES method, energy is consumed smoothly and equally among sensor nodes. In LEACH method, energy consumption is very different with each node, almost the three-fours nodes run out of energy while others still have high energy level. This problem illustrated clearly in Fig. 11. OEDSR and HPEQ perform load balance better than LEACH but worse than ARPEES. HPEQ protocol uses a shortest path algorithm to construct relaying path. In the practice, however, the shortest path is not always the optimal path.

From the above analysis, we find out that the short life-

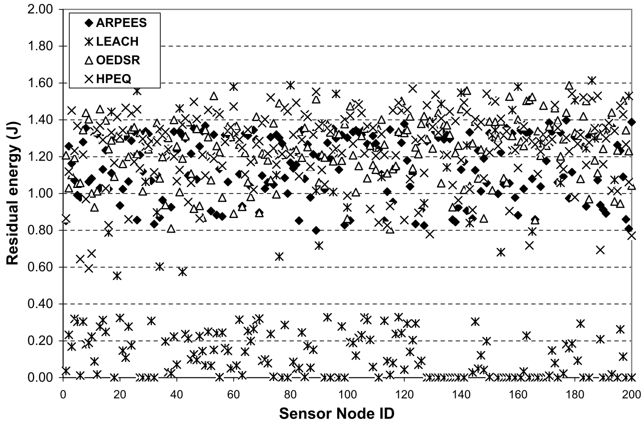

Fig. 10 The residual energy of 200 randomly selected sensor nodes after 220 rounds with four methods in the 500 × 500 meters network area of 1200 sensor nodes.

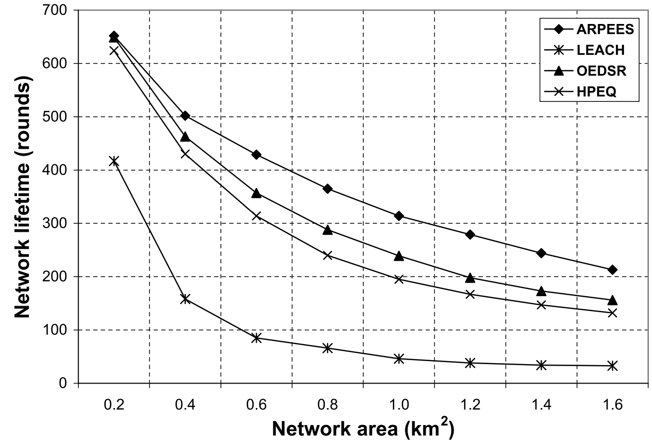

Fig. 12 Network lifetime over increasing network area with four protocols with 1500 sensor nodes.

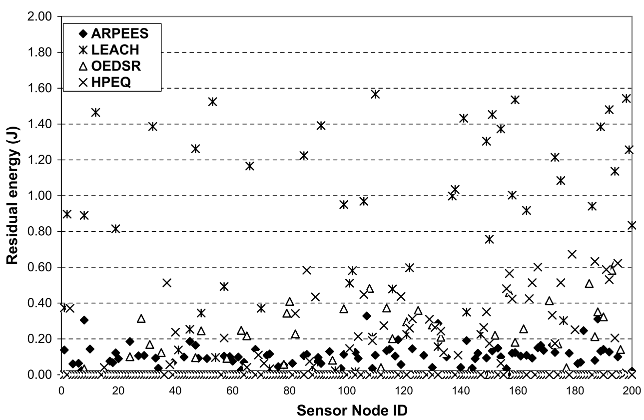

Fig. 11 The residual energy of 200 randomly selected sensor nodes at the last round of simulation: after 585, 224, 519, and 489 rounds for ARPEES, LEACH, OEDSR, and HPEQ, respectively.

time of the whole sensor network is cause by the unbalance load to the different sensor nodes. Although some nodes still have a lot of energy remaining, the sensor network cannot operate correctly. Thus, we should consider maximizing the lifetime of whole sensor network rather than just the lifetime of some individual sensor nodes.

(2) Scale to large sensor network

In the second simulation, we evaluated the performance of the ARPEES protocol as the area of sensing field increases. We ran the simulation with 1,500 sensor nodes uniformly dispersed in a square field of varying network areas. The other parameters are the same as the first experiment. Figure 12 plots the network lifetime of the four routing protocols with increasing network area. The network lifetime of our method is longer than those of three other comparators, especially comparing with LEACH. The lifetime of LEACH decreases quickly because the cluster heads waste the considerable amount of energy for transmitting their data to the faraway base station. ARPEES shares energy consumption among the several relay nodes or takes advantage of load balancing by adopting multi-hop scheme with energy efficiency in selecting relay node function. Therefore, this plot clearly shows that ARPEES operates much longer than LEACH, especially as the area of sensing field increases.

6. Discussion and Future Work

Our approach differs from the previous hierarchical protocols. First, we do not divide all sensor nodes in the whole network into clusters, only a part of the network where and when the event appears will be clustered. Therefore, the number of clusters and of course the number of cluster heads are decided depending on the scope of the event occurrence (event clustering). The data from the sensor node toward the base station are transmitted whenever an event occurs. Using event-clustering technique, our protocol avoids the explosion of messages due to clustering the entire network and also avoids wasting energy in some parts of the network where sensor nodes have nothing to transmit. Therefore, the system can reduce the total energy consumption especially flexible to the enlargement of the network scale and adapt to network dynamics.

Furthermore, multi-hop scheme adopted to relay the aggregated data for a long distance, which is the main advantage of the proposed algorithm, allows balancing the energy consumption onto some relay nodes. This algorithm ensures selecting the optimal relay path between the cluster head and the base station. The routes can be recomputed dynamically in each round to reflect the changes in network topology. In this paper, for the term "optimal relay path," we consider the trade-off between the residual energy available of the relay node and distance from the relay node to the base station. This idea is expressed in the Eq. (7). As a result, we can choose the relay node that satisfies two conditions: (1) to have enough residual energy for acting as the relay node or for carrying aggregated data (energy efficiency); (2) as farther from the cluster head or nearer to the base station as possible. The second condition is to make sure that the final relaying path will have small number of hops (shortest or shorter path). Furthermore, the optimization generally focused on improving one or two aspects of performance: in this work we aim to improve energy efficiency and network lifetime. Together with the method to select the cluster head, also based on its energy in the first stage, we distribute data load or balance energy required for transmitting aggregated data to different nodes. The optimal relay path in our method has been evaluated by comparing with three other methods. From the results shown in Fig. 9, ARPEES has the longest network lifetime among four compared methods. Especially, our method achieved energy balance as having more number of living nodes than the others have; Energy of nodes reduces smoothly in ARPEES while it is very different in the others (Figs. 10 and 11).

The protocol also does not make the assumptions that all sensor nodes are capable of communicating directly with any sensor in the field, including the base station. Moreover, we do not require special sensor nodes equipped with extra hardware to be aware of their locations. Those are very important issues to realize the sensor network and scale with the enlargement of the network with only small and cheap sensor nodes.

RSSI is measured by the RF energy received at the receiver and this function supported by sensor node hardware such as the Berkeley Motes. Normally, sensors consume energy for three main domains: sensing, processing, and data communication. Among them, a sensor node consumes the most energy in data communication. In our method, we use *REQ_RELAY* message not only for searching relay node but also as a beacon for measuring RSSI. Therefore, the overhead for processing RSSI is very small and it does not affect to the overall results of the method.

In the future work, we will study how the ARPEES protocol can be implemented in the mobile network scenario where sensor nodes or sink nodes have moving abilities. We also take account of sensing coverage problem in WSNs. The idea is that a small number of delegated sensor nodes should be selected by identifying and removing redundant nodes in high-density networks and assigning them an off-duty operation under the guarantee where the whole area is *k*-covered. That scheme makes sure that all events occurred in the monitored area can be accurately and timely detected, while we can also extend the network lifetime.

7. Conclusion

We proposed an adaptive routing protocol with energy efficiency and event clustering for wireless sensor networks called ARPEES. We demonstrated the features that adaptive mechanism among energy efficiency, event clustering, and multi-hop relaying transmissions achieves both objectives of maximizing the network lifetime and minimizing the total consumed energy. Through the computer simulation, we showed that proposed protocol can significantly reduce the energy consumption of each node, can balance the energy required to transmit data to other nodes, and therefore prolongs the whole network lifetime, especially as the network size becomes large.

References

[1] A. Visvanathan, J.-H. Youn, and J. Deogun,“Hierarchical data dissemination scheme for large scale sensor networks,” IEEE Int'l Conf. Commun. (ICC 2005), vol.5, pp.3030–3036, May 2005.
[2] V. Tran Quang and T. Miyoshi, “ARPEES: Adaptive routing protocol with energy-efficiency and event-clustering for wireless sensor networks,” 4th Int'l Conf. Ubiquitous Robots and Ambient Intelligence (URAI 2007), pp.95–100, Nov. 2007.
[3] V. Tran Quang and T. Miyoshi, “ARPEES: Adaptive routing protocol for large scale wireless sensor networks,” Proc. Commun. Conf. IEICE 2007, BS-12-3, pp.S-154–S-155, Sept. 2007.
[4] W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless microsensor networks,” IEEE Trans. Wireless Commun., vol.1, no.4, pp.660–670, Oct. 2002.
[5] A. Manjeshwar and D.P. Agrawal, “TEEN: A routing protocol for enhanced efficiency in wireless sensor networks,” 15th Int'l Parallel and Distributed Process. Symp. (IPDPS'01) Workshops, p.30189a, April 2001.
[6] A. Manjeshwar and D.P. Agrawal, “APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks,” Int'l Parallel and Distributed Process. Symp. (IPDPS'02) Workshops, p.0195b, April 2002.
[7] S. Lindsey and C.S. Raghavendra, “PEGASIS: Power-efficient gathering in sensor information systems,” IEEE Aerospace Conf., vol.3, pp.3-1125–3-1130, March 2002.
[8] S.D. Muruganathan, D.C.F. Ma, R.I. Bhasin, and A.O. Fapojuwo, “A centralized energy-efficient routing protocol for wireless sensor networks,” IEEE Commun. Mag., vol.43, no.3, pp.8–13, March 2005.
[9] A. Boukerche, R.W.N. Pazzi, and R.B. Araujo, “HPEQ — A hierarchical periodic, event-driven and query-based wireless sensor network protocol,” IEEE Conf. Local Comput. Networks, pp.560–567, Nov. 2005.
[10] S. Ratnaraj, S. Jagannathan, and V. Rao, “Optimal energy-delay subnetwork routing protocol for wireless sensor networks,” IEEE Conf. on Networking, Sensing and Control, pp.787–792, April 2006.
[11] C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed diffusion: A scalable and robust communication paradigm for sensor networks,” Int'l Conf. Mobile Computing and Networking (MobiCom'00), Aug. 2000.
[12] F. Ye, H. Luo, and J. Cheng, “A two-tier data dissemination model for large scale wireless sensor networks,” Int'l Conf. Mobile Computing and Networking (MobiCom'02), pp.148–159, Sept. 2002.
[13] W. Heinzelman, J. Kulik, and H. Balakrishnan, “Adaptive protocols for information dissemination in wireless sensor networks,” Int'l Conf. Mobile Computing and Networking (MobiCom'99), pp.174–185, Aug. 1999.
[14] Y. Xu, J. Heidemann, and D. Estrin, “Geography-informed energy conservation for ad hoc routing,” Int'l Conf. Mobile Computing and Networking (MobiCom'01), pp.70–84, July 2001.
[15] L. Li and J.Y. Halpern, “Minimum-energy mobile wireless networks revisited,” IEEE Int'l Conf. Commun. (ICC'01), vol.1, pp.278–283, June 2001.
[16] T. He, J.A. Stankovic, C. Lu, and T.F. Abdelzaher, “SPEED: A stateless protocol for real-time communication in sensor networks,” Int'l Conf. on Distributed Computing Syst. (ICDCS'03), pp.46–55, May 2003.
[17] K. Sohrabi, J. Gao, and V. Ilawadhi, “Protocols for self-organization of a wireless sensor network,” IEEE Pers. Commun., vol.7, no.5, pp.16–27, Oct. 2000.
[18] P. Bahl and V.N. Padmanabhan, “RADAR: An in-building RF-based user location and tracking system,” IEEE INFOCOM 2000, vol.2,

pp.775–784, March 2000.

[19] G.T. Shi and M.H. Liao, "A hierarchical architecture for energy-efficient information dissemination in sensor networks," 2nd Int'l Conf. Embedded Software and Syst., Dec. 2005.

[20] Z.-W. Zheng, Z.-H. Wu, and H.-Z. Lin, "Clustering routing algorithm using game-theoretic techniques for WSNs," Int'l Symp. Circuits and Syst. (ISCAS'04), vol.4, pp.IV-904–907, May 2004.

[21] OMNeT++, A discrete event simulation system, Version 3.3, retrieved from official website: http://www.omnetpp.org/

Vinh Tran Quang received his B.E. (2000) and M.S. (2003) degrees in Electronics and Telecommunications from Hanoi University of Technology, Vietnam. Currently, he is doing his research as a Ph.D. student at the Graduate School of Engineering, Shibaura Institute of Technology, Japan. His current interests include ad hoc network and wireless sensor network. He received the IEEE Section Prize Student Paper Award in 2008. He is a student member of IEEE.

Takumi Miyoshi received his B.Eng., M.Eng., and Ph.D. degrees in electronic engineering from the University of Tokyo, Japan, in 1994, 1996, and 1999, respectively. He was a visiting associate from 1994 to 1996 and an Internet technical staff from 1996 to 1997 at the Institute for Monetary and Economic Studies, Bank of Japan. He was also a research associate at Global Information and Telecommunication Institute, Waseda University, from 1999 to 2001, and a research fellow at Telecommunications Advancement Organization of Japan from 1998 to 2003. He is presently an associate professor at Department of Electronic Information Systems, College of Systems Engineering, Shibaura Institute of Technology, Saitama, Japan. His research interests include multimedia communication technologies, mobile ad hoc and sensor networks, and online learning systems. He received the IEICE Young Investigators Award in 2004, the IEICE Information Network Research Award in 2001, 2004 and 2006, the IEICE Communications Society Distinguished Contributions Award in 2006 and 2007, and the Ericsson Young Scientist Award in 2002. He is also a member of IEEE.