Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2012, Article ID 156268, 13 pages
doi:10.1155/2012/156268

Research Article

A Self-Organized and Smart-Adaptive Clustering and Routing Approach for Wireless Sensor Networks

Kyuhong Lee and Heesang Lee

Department of Industrial Engineering, Sungkyunkwan University, Suwon 440-746, Republic of Korea

Correspondence should be addressed to Heesang Lee, leehee@skku.edu

Received 15 July 2011; Revised 1 October 2011; Accepted 6 October 2011

Academic Editor: Yuhang Yang

Efficient energy consumption is a critical factor for the deployment and operation of wireless sensor networks (WSNs). In general, WSNs perform clustering and routing using localized neighbor information only. Therefore, some studies have used self-organized systems and smart mechanisms as research methods. In this paper, we propose a self-organized and smart-adaptive clustering (SOSAC) and routing method, which performs clustering in WSNs, operates the formed clusters in a smart-adaptive way, and performs cluster-based routing. SOSAC is comprised of three mechanisms, which are used to change the fitness value over time, to back up routing information in preparation for any potential breakdown in WSNs, and to adapt to the changes of the number of sensor nodes for a WSN. We compared the performance of the proposed SOSAC with that of a well-known clustering and routing protocol for WSNs. Our computational experiments demonstrate that the network lifetime, energy consumption, and scalability of SOSAC are better than those of the compared method.

1. Introduction

A wireless sensor network (WSN) is an infrastructure composed of wireless sensor nodes, which perform sensing tasks and transmit the data to a base station (BS) that is the final processing node for the WSN. According to the features of the wireless network, WSNs are widely used in dynamic and hazardous regions as well as in many industries including production, logistics, distribution, transportation, and health [1].

WSNs have many characteristics that differ from conventional wireless networks. For example, WSNs have several inherent constraints that are unique to this type of wireless networks [2, 3]. First, a WSN consists of hundreds or thousands of wireless sensor nodes but each node in a WSN is constrained in terms of processing capability and storage capacity. Second, the power device of the sensor node in WSNs cannot be recharged or replaced so that energy efficiency is very important [4].

Routing protocols in a WSN should be developed and proposed with consideration for these unique characteristics. The routing decides the transmission route for a data packet from the sensor node to the final destination BS. The communication channels of WSNs can be configured through multihop mesh network called “flat routing”. In contrast, a “cluster-based routing” method divides a WSN into several clusters. In this cluster-based routing, each cluster is comprised of a “cluster head” (CH) and several “cluster members” (CMs). After formation of the cluster, hierarchical routing is performed, where CMs transmit data packets to its CH, and the CH integrates these data packets and transmits them to the BS directly or via other CHs [2].

Along with the development of communication networks, the routing paradigm has changed from a centralized system to a distributed system due to the demand for better scalability and simple installation. In WSNs, tens or thousands of sensor nodes may be needed to create one network, so when selecting the processor, memory, and power devices, economical devices with limited functions are used [1]. Accordingly, the distributed routing method is used rather than the centralized routing method for WSNs considering the processing ability of the sensor nodes [1]. When distributed processing is used, each sensor node is not allowed to use the information of the entire WSNs and must therefore perform clustering and routing using the localized neighbor information only. When the power device for sensor nodes is not rechargeable or replaceable, it is also important to consider energy efficiency in order to maximize the network lifetime of the sensor nodes by minimizing the energy consumption [4].

Self-organization, which is a further development of distributed processing, is a system that uses local information only through a distributed and peer-to-peer method. Many studies on self-organized systems are being conducted using only local information and simple rules because of the associated advantages such as reduction of overheads in communication traffic, scalability, and robustness. Also, self-organization has been applied to many areas in the wireless network field such as ad-hoc networks, WSNs, wireless LAN, in routing, forming clusters, MAC protocol design, and radio resources management. Some studies have been conducted on self-organization with the features of WSNs [5].

Recently "smart" characteristics of routing have also been suggested in telecommunication networking [6–8]. Here smart means the capability to describe and analyze a situation, taking decisions based on the available data in a predictive or adaptive manner, and thereby performing smart actions.

In this paper, we propose a self-organized and smart-adaptive clustering (SOSAC) and routing method comprised of three mechanisms for a WSN. First, in order to select an appropriate CH, which is the most important factor influencing the clustering performance, two types of performance measure are used. The proposed mechanism adjusts the weights of these two performance measures automatically to reflect the changes in the WSN. Second, the broadcasting range, which is another important factor influencing the performance of clustering in SOSAC, is predetermined as a function of the number of sensor nodes. Third, to overcome problems caused by possible breakdown, damage, or failure of sensor nodes in WSNs, a smart backup mechanism is established to monitor the state of the CH without heavy overhead and to restore the system automatically in the case of such problems.

This paper is organized as follows. Section 2 reviews previous studies on clustering, self-organization, and smart telecommunication networks and their implications for this research. Section 3 presents the radio model that is used and the basic assumptions used in this study. Section 4 explains the process of forming self-organized clusters and routing. Section 5 describes how to operate clustering in a smart-adaptive way by using three proposed mechanisms. In this section, we also compare the performance of the proposed SOSAC with that of another well-known model. The measures for comparison are network lifetime, residual energy after a certain time, and scalability of the model. The last section presents the conclusions of this study and proposes future research directions.

2. Related Works

Grouping sensor nodes into clusters has been widely pursued by the research community in order to achieve the network scalability objective. The objective of clustering is mainly to generate stable clusters in environments with sensor nodes. In addition to supporting network scalability, clustering has numerous advantages. It can localize the route set up within the cluster and thus reduce the size of the routing table stored at the individual sensor node [3, 9]. Clustering can also conserve communication bandwidth since it limits the scope of intercluster interactions to CHs and avoids redundant exchange of messages among sensor nodes [10]. Moreover, clustering can stabilize the network topology at the level of sensor nodes, and thus cut down on topology maintenance overhead. Sensor nodes are only affected by the connection with their CHs and not by changes at the level of inter-CH tier [11]. The CH can also implement optimized management strategies to further enhance the network operation and prolong the battery life of the individual sensor nodes and the network lifetime [10].

"Self-organization" is defined as the process where a structure or pattern appears in a system without intervention by external directing influences. It organizes through direct interaction in a peer to peer method [5]. The advantages of using the self-organized system are as follows [5, 12]. First, one of the most important characteristics of self-organization is the completely distributed control. Each participating system component acts on local decisions, that is, it is not possible to review the current global state and act accordingly. Second, an inherent feature of self-organizing systems is their capability to adapt to changing environmental conditions. This is a direct result of the distributed peer to peer working principle. Third, the robustness of the system prevents any problems due to breakdown, damage, or failure of individual elements. Fourth, scalability protects the system from degradation by increasing the number of individual elements in the system.

Several cluster-based and self-organized protocols for WSNs were studied as follows.

In the LEACH protocol [13], the basic idea is to select sensor nodes randomly as CHs. Random selection of a CH is good for self-organization of the cluster configuration. Cluster configuration is repeated at each round, and the round is divided into two phases: the "set-up phase" and the "steady phase". In the set-up phase, LEACH selects a CH candidate with a threshold for a random number. In this phase, each sensor node compares a random number with the threshold $T(n)$ to elect itself to a CH, and this process is performed independently for each cluster. In the steady phase, the sensor nodes sense environment and transmit data, and the CHs aggregate the data before sending the data to the BS.

LEACH-Energy Distance (LEACH-ED) [14] is another self-organized protocol that is based on LEACH. It uses a different threshold from LEACH. The ratio between the residual energy of a sensor node and the total current energy of all of the sensor nodes in the network is used for the first threshold of LEACH-ED. It also uses the distance threshold as the second threshold. If the distance between a sensor node and an existing CH is less than the distance threshold, the sensor node cannot be elected as a CH.

The hybrid-energy efficient distributed protocol (HEED) [15] selects CHs by combining the residual energy of the sensor and the communication cost for selection of the CH in a WSN. The communication cost is calculated using the average minimum reachability power or the neighbor sensor node degree as the secondary parameter. Once the CH is selected, HEED uses a hierarchical routing protocol with a 3-tier structure, where first each CM transmits data to its CH, second each CH sends data to one special CH by using the breadth-first search tree, and finally the special CH sends all data to the BS.

Robust energy efficient distributed clustering (REED) [16] is a self-organized clustering method which constructs k independent sets of CH overlays on the top of the physical network to achieve fault tolerance. Each sensor must reach at least one CH from each overlay. The method of selection of CH is same as HEED.

Distributed, energy-efficient, and dual-homed clustering (DED) [17] is a self-organized clustering method that achieves fault tolerance by providing an alternative route from sources to the BS. Each regular sensor node has a primary CH and a secondary backup CH. DEED and HEED are the same method for CH selection but use different parameters for the selection.

In telecommunications, although some researchers have studied smart systems for the formation or operation of communication networks, each study has used different methods and different definitions, as shown in the related studies described below.

In traditional networks, passive means were used to send packets or signals to each termination, but active networks have recently been suggested that allow nodes to customize computation on message flowing [18] or a new smart network framework with not only active services but also the concepts of contextawareness and userawareness [19]. In this network, the nodes of the network have the capability to sense, to reason and to be aware of the context and behavior of users, and automatically provide active services to users according to their situation and context knowledge [20].

Stone et al. [21] have suggested a smart network which requires three types of intelligence. First, it is an inference system for collecting information about the user's interaction with the network. Second, it is a framework associated with mobile agents for routing through the network of processing messages and servers. Third, it is a gateway to allow access to the network.

Gelenbe et al. [6, 7] have suggested a cognitive packet network which provides intelligent capabilities for routing and flow control to packets instead of the nodes or protocol in the wired network or wireless ad hoc network. This has enabled the realization of a smart routing system that allows the cognitive packets to find their own route between source and destination in the packet switched networks.

3. Network Modeling

3.1. First Order Ratio Model. A sensor node consumes energy when transmitting and receiving data packets in a WSN. In wireless data transmission, energy consumption is correlated to the data packet size and the distance between the two sensor nodes. Extensive research has been conducted in the area of low-energy radios. Different assumptions about the radio characteristics, including energy dissipation in the transmission and received modes, will change the advantages of different protocols. In our work, we assume the following first-order radio model as our radio energy consumption model.

(i) Transmitting the data packet: a sensor node consumes $\varepsilon_{\text{elec}} = 50\,\text{nJ/bit}$ at the transmitter circuitry and $\varepsilon_{\text{amp}} = (100\,\text{pJ/bit})/\text{m}^2$ at the amplifier.

(ii) Receiving the data packet: a sensor node consumes $\varepsilon_{\text{elec}} = 50\,\text{nJ/bit}$ at the receiver circuitry.

(iii) A k-bit data packet is transmitted from sensor node to sensor node, and d_{ij} is the distance between the two sensor nodes i and j; the energy consumption of sensor node i is given by $T_{ij} = \varepsilon_{\text{elec}} \times k + \varepsilon_{\text{amp}} \times d_{ij}^2 \times k$.

(iv) The sensor node receives data packet: the energy consumption of sensor node is given by $R_i = \varepsilon_{\text{elec}} \times k$.

3.2. Assumptions and Definitions for the Model. We assign the following properties and make the following assumptions for modeling and simulation for the WSN.

(i) The location of all sensor nodes and the BS are fixed.

(ii) The location of the BS (25 m, 150 m) is known in advance in the 50 m*50 m sensor field.

(iii) The data packet size is 1,000 bits, and signal packet size is 50 bits.

(iv) All sensor nodes have an initial energy of 0.5 J.

(v) Period: in the data transmission phase, a CM creates information sensed by itself into a data packet and transmits this packet to its CH. The CH aggregates the data packets received from its CMs and transmits them to the BS through the intercluster route. A cycle of this process is defined as a "period".

(vi) Round: some number of periods carried out within a data transmission phase can be defined as a "round". In our experiments, we assume that 1 round is made up of 10 periods.

(vii) Network lifetime: the periods until a certain number of sensor nodes drained of its energy can be defined as the "network lifetime".

4. Self-Organized and Smart-Adaptive Clustering (SOSAC): The Proposed Clustering and Routing Protocol

This section explains the procedure of the SOSAC proposed herein. SOSAC decides the CHs every round, and it is comprised of three phases, as set forth in Sections 4.2, 4.3, and 4.4. In Section 4.1, the state of each sensor node of the WSN is explained to help understand the SOSAC operations.

4.1. States of Sensor Nodes. Each sensor node of SOSAC performs its duties while being changed into the following five states depending on the roles. Figure 1 indicates the state transition diagram of SOSAC.

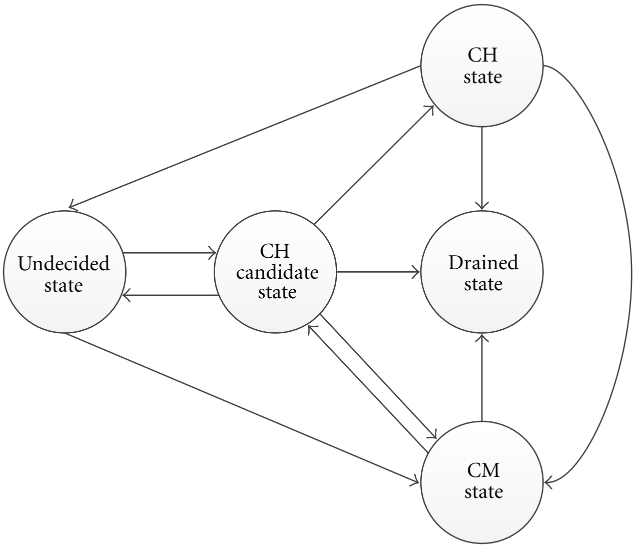

Figure 1: Transition diagram of sensor node states.

(i) *Undecided state*: when a sensor node has been scattered first in the sensor field, after a CH has completed its duty, and after a sensor node has dropped out of the node with CH candidate state, the sensor node becomes a node with undecided state. A node with undecided state does not belong to any cluster, yet.

(ii) *CH candidate state*: when the sensor nodes are scattered in the sensor field at period zero, all sensor nodes become a node with CH candidate state. Some of sensor nodes with a CM state also can become nodes with CH candidate state. Only a node with CH candidate state can compete for selection of CHs.

(iii) *CH state*: one node with CH candidate state within a cluster becomes a node with CH state, that is, a CH in that period. As a CH, it collects and aggregates information from its CMs and transfers data packets to other CHs or the BS. Also, a CH decides its nearby nodes as the nodes with CH candidate state for the next round.

(iv) *CM state*: when a CH is decided, all other sensor nodes in the same cluster become nodes with CM state, that is, CMs. A CM periodically sends the sensed information to its CH.

(v) *Drained state*: when a sensor node cannot function anymore because all its energy has been drained or it has broken down, it becomes a node with drained state.

SOCAC is comprised of a clustering phase that forms the clusters, an intercluster routing phase that decides on the transmission route among the CHs, and a data transmission phase that sends/receives the data packets.

Figure 2 indicates the progress of SOSAC in a flowchart.

4.2. Clustering Phase. SOSAC needs a clustering phase for the hierarchical routing. The clustering phase commences if the sensor nodes are scattered first in the sensor field or after the "data transmission phase" has finished. The clustering phase is comprised of three steps: Broadcasting step, CH selection step, and clustering step as shown in Algorithm 1.

4.2.1. Broadcasting Step. When the sensor nodes are scattered in the sensor field at period zero, no sensor node belongs to any cluster, thus all sensor nodes are assumed in the same cluster at period zero. The CHs broadcast a CH-change-signal packet only within the broadcasting range in procedure 2.1 of Algorithm 1 because since a CH was at a good position when it was selected as the CH, it is highly likely that its neighbors are also at good location for being CHs. By limiting the broadcasting range, the overhead of the clustering phase can be reduced because of the proximity of the nodes with the CH candidate state.

4.2.2. CH Selection Step. In procedure 3.2 of Algorithm 1, we update the counter, which is the number of received CH-candidate-signal packets. This counter will be used for calculating the fitness value in procedure 5 of Algorithm 1. A CH-candidate-information-signal packet, which a node with CH candidate state broadcasts in procedure 4.1 of Algorithm 1, contains the energy state of the sending sensor node. Information of the received CH-candidate-signal packets is also used to calculate the fitness value in procedure 5 of Algorithm 1. The CH-candidate-information-signal packets are broadcast within 2*(broadcasting range) for the following reasons. All nodes with the CH candidate state of a cluster must share the neighborhood information with all nodes with CH candidate states within the same cluster. This can be achieved by sending the CH-candidate-information-signal packets within two times of the broadcasting range for

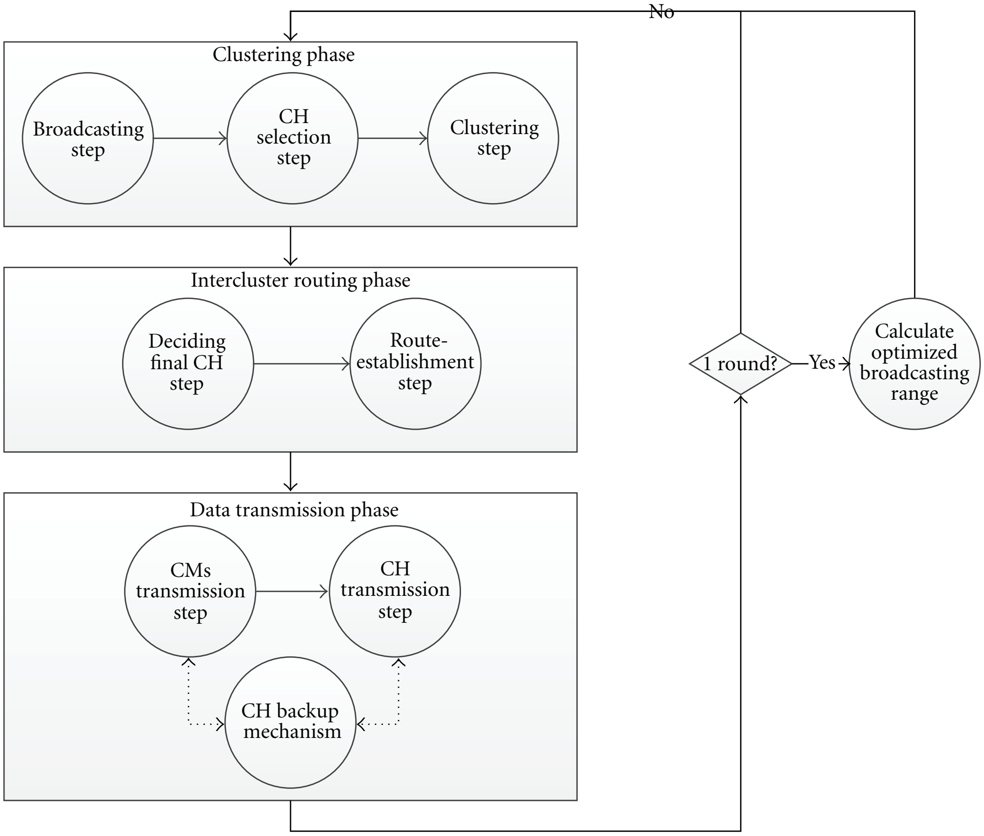

Figure 2: Flowchart of SOSAC.

two nodes with the CH candidate state that are farthest away from each other.

4.2.3. Clustering Step. A node with CH state in procedure 8.1 of Algorithm 1 broadcasts a CH-signal packet to the entire sensor field to form its cluster. The sensor nodes in all states except nodes with CH state or nodes with drained state that receive these signal packets select the nearest CH and become a CM of that cluster.

4.3. Intercluster Routing Phase. After the clustering phase, the CHs create an intercluster routing tree and select a CH as the "master CH". Each CH sends packets to the master CH and the master CH transmits the packet to the BS directly. Transmission of data packets through the intercluster routing tree, and its master CH can save energy compared to the direct transmission of data packets from all CHs to the BS.

In the process of creating a tree for intercluster routing, the CHs perform their duties while changing into the following three states depending on the roles.

(i) *Initial state*: if the intercluster routing phase starts, all CHs are initialized as the CHs with initial state. If a CH with initial state receives a route-broadcast-signal packet, it transmits ACK back to the CH that sent the route-broadcast-signal packet to the CH.

(ii) *Route broadcasting state*: after finding the master CH, that CH becomes a CH with route broadcasting state. The master CH sends route-broadcast-signal packets to CHs with initial state. A CH with initial state that received a route-broadcast-signal packet becomes a CH with route broadcasting state.

(iii) *Route-established state*: the CH with route broadcasting state broadcasts route-broadcast-signal packet to establish an intercluster route and then becomes a CH with route-established state. When all CHs with initial state become CHs with route-established state, all CHs become CHs with route-established state.

Figure 3 indicates the transition diagram of CHs in the intercluster routing phase.

The intercluster routing phase is divided into the deciding master CH step and the route establishment step.

4.3.1. Deciding Master CH Step. The expected residual energy in procedure 1 of Algorithm 2 is the expected remaining energy of each CH, which is calculated as the following.

$$\text{The expected residual energy} = \text{current energy} - \gamma, \qquad (1)$$

where γ is the expected consuming energy which sends data packets to BS during the next round. CH broadcasts a CH-expected-residual-energy-signal packet to the whole area of the sensor field in procedure 2.1 of Algorithm 2. We use this simple broadcasting technique because any CH

```
Clustering phase
    (1) Broadcasting step
(1)     If (Period = 0), every sensor nodes changes its state as a node with CH candidate state.
        Else
(2.1)       CH broadcasts a CH-change-signal packet within broadcasting range. It changes its state
            as a node with undecided state.
(2.2)       CM receives the CH-change-signal packet from one's CH. It changes its state as a node
            with CH candidate state.
    (2) CH selection step
        (For a node with CH candidate state)
(3.1)       Broadcast a CH-candidate-signal packet.
(3.2)       Receive the CH-candidate-signal packets:
                If two nodes are at the same cluster, increase by 1 to the counter.
                Else discard the received signal packet.
(4.1)       Broadcast a CH-candidate-information-signal packet within 2*(broadcasting range).
(4.2)       Receive CH-candidate-information-signal packets:
                If two nodes are at the same cluster, store the received packet.
                Else, discard the received packet.
(5)         Calculate the fitness value.
(6.1)       Broadcast a fitness-value-signal packet.
(6.2)       Receive fitness-value-signal packets:
                If two nodes are at the same cluster, store the received signal packet.
                Else, discard the received signal packet.
(7)         Compare the received fitness values:
                If one's fitness value is Max, the maximum sensor node changes its state as a node
                with CH state.
                Else, the sensor node changes its state as a node with undecided state.
    (3) Clustering step
(8.1)   CH broadcasts a CH-signal packet to the whole area of the sensor field.
(8.2)   Others decide one's CH by finding the closest CH.
```

ALGORITHM 1: The clustering phase algorithm.

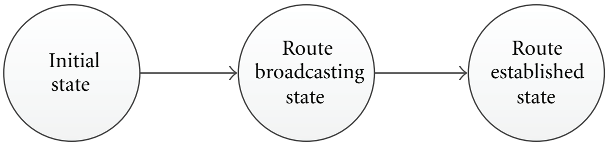

FIGURE 3: Transition diagram of CH states in the intercluster routing phase.

can get energy information for all CHs in the sensor field and compare it with its own energy information since each CH broadcasts CH-expected-residual-energy-signal packets to the whole area of the sensor field. This does not generate a heavy overhead since the signal packets are small and we assume that the sensor field is small (between 50 m*50 m and 150 m*150 m) in this study. When the sensor field is very large, we need to modify this intercluster routing procedure by using some flat routing techniques.

4.3.2. Route Establishment Step. A CH with initial state that receives the route-broadcast-signal packets sets the nearest CH with route established state as the first section of its transmission route. The CHs with route broadcasting state broadcast route-broadcast-signal packets in procedure 4 of Algorithm 2. The initial broadcasting range is δ m. (In our computational implementation, we use $\delta = 25$.) If a CH with initial state receives a route-broadcast-signal packet, it transmits ACK to the CH that sent a route-broadcast-signal packet. If a CH with route broadcasting state does not receive any ACKs during given time interval, it extends the broadcasting range by δm repeatedly until the broadcasting range is wider than (width of the sensor field $*\sqrt{2}$). Within this broadcasting range, if a CH with route broadcasting state does not receive any ACK, it becomes a leaf CH of the intercluster routing tree. This route establishment method yields a spanning tree that includes all CHs. This spanning tree is built as a breadth-first search tree while the CHs exchange the route-broadcast-signal packets and ACKs. While the breadth-first search tree is made, a CH tries to include its neighbor CHs by increasing the broadcasting range δ when the CH does not receive any ACKs, which is a kind of stop-and-wait method.

```
Intercluster routing phase
    (1) Deciding master CH step
        (For a CH with initial state)
(1)             Calculate expected residual energy.
(2.1)           Broadcast a CH-expected-residual-energy-signal packet to the whole area of the sensor field.
(2.2)           Receive CH-expected-residual-energy-signal packets.
(3)             Compare CH expected residual energy:
                    If one's CH expected residual energy is Max, the maximum CH changes its state as
                    a CH with route broadcasting state.
    (2) Route establishment step
        While (All CHs with initial state become CHs with route established state)
            (For a CH with route broadcasting state)
(4)             Broadcast a route-broadcast-signal packet within the intercluster broadcasting range. If it receives
                ACK, it changes its state as a CH with route established state.
                If it does not receive any ACK during given time interval, it extends the broadcasting range. If it does
                not receive any ACK within maximum broadcasting range, that CH becomes a leaf CH.
            (For a CH with initial state)
                If it receives route-broadcast-signal packets:
(5)                 Find the closest CH that sent route-broadcast-signal packets and establish intercluster route by
                    sending ACK to the closest CH. It changes its state as a CH with route broadcasting state.

        End while
```

Algorithm 2: The intercluster routing phase algorithm.

4.4. Data Transmission Phase. Once the clusters and the intercluster routing tree have been created, information sensed by each sensor node is transmitted to the BS using data packets in the data transmission phase. The data transmission phase is divided into 2 steps: CM transmission step and CH transmission step.

In Algorithm 3, a CM transmits data packets, which are created by sensing the surroundings in each period, to its CH. Each CH aggregates the received data packets and transmits the aggregated data packet to the BS through the route that is set in the intercluster routing phase. Sensing and data transmission of each sensor node are repeated during one round.

5. Smart Mechanisms and Computational Experiments of SOSAC

5.1. Smart Mechanisms in SOSAC

5.1.1. Smart Fitness Value Adaptability Mechanism. The most important factor affecting the clustering performance is the CH selection method. SOSAC uses the following fitness comparison procedure to select the appropriate CHs.

First, check the location of each node with CH candidate state for the fitness value. SOSAC uses the number of the received CH-candidate-signal packets to check the appropriateness of the location of a sensor node. In other words, each node with CH candidate state collects information on the locations of some neighboring nodes with CH candidate state (in the same cluster) as the number of received CH-candidate-signal packets and transmits it to the neighboring sensor nodes to compare the number of CH-candidate-signal packets. A CH with many neighboring nodes with CH candidate state is considered as a good candidate for the CH of a cluster since it is located in the center of the cluster. Accordingly, SOSAC decides on the neighborhood degree fitness of each node with CH candidate state based on the ratio of the number of neighbors of v to the maximum number of neighbors of the cluster where the sensor node belongs to. This fitness value indicates the location appropriateness of the node with CH candidate in the current cluster. We define "neighborhood degree fitness" of a sensor node v that has CH candidate state of the cluster as follows. The "neighborhood degree fitness" consists of the first component of the fitness value.

$$\text{Neighborhood degree fitness of } v = \left(1 + \frac{\text{The number of neighbor of } v}{\text{Max. number of neighbor of cluster}(v)}\right)^{\alpha}, \qquad (2)$$

where cluster(v) is the cluster that sensor node v belongs to, and α is the average ratio of remaining energy to initial energy of cluster(v).

Second, "energy state fitness" of a node with CH candidate state should be considered as the second component of the fitness value. Since CH consumes more energy than the CMs, SOSAC gives a priority to a node with CH candidate state that has more residual energy. Therefore, the energy state is used as the second component of the fitness value, and we define energy state fitness as follows:

$$\text{Energy state fitness of } v = \left(1 + \frac{\text{residual energy of } v}{\text{Max. residual energy of cluster}(v)}\right)^{1-\alpha}. \qquad (3)$$

```
Data transmission phase
  While (1 round)
    (1) CM transmission step
      (For a node with CM state)
          Sense environment.
          Make a data packet.
          Transmit a data packet to its CH.
    (2) CH transmission step
      (For a node with CH state)
          Aggregate received data packets from CMs
          Transmit an aggregated data packet using the established path.
  End while
```

Algorithm 3: The data transmission phase algorithm.

Since each component of the two fitness components has values between 1 and 2, the values of the two fitness components are inherently normalized. Also we use α and $1-\alpha$ as exponent parameters, each of which decides the weight of the respective fitness component. A total fitness value can be calculated by adding these two components together as follows.

$$\text{Total fitness value} = \text{neighborhood degree fitness of } v + \text{energy state fitness of } v. \tag{4}$$

If the total fitness value is to reflect the state of the networks, then it should be adjusted in a smart way to consider the change in the environment. In the beginning of networking, each sensor node has sufficient residual energy so that the energy state fitness is not very important. Therefore, the neighborhood degree fitness should be weighted more. This can be done by using the suggested total fitness value since the neighborhood degree fitness becomes more important than the energy state fitness as α remains relatively large. As time goes by, however, if the sensor nodes consume more energy, the scarcity of the energy state should be reflected. This can be done by using the suggested total fitness value since the energy state fitness becomes more important than the neighborhood degree fitness as $(1-\alpha)$ becomes large. Hence, SOSAC tries to achieve energy balance by increasing the weight of the energy state fitness.

For SOSAC implementation, we constructed a source code using Visual C++ of Visual Studio 2008 and conducted a computer simulation. The sensor field for the test was 50 m*50 m in size, and the simulation was carried out under the assumption that 100 sensor nodes with an initial energy of 0.5 J were distributed. The test used the average of the performances of the 10 different sensor distributions in the 50 m*50 m sensor field.

Figure 4 is an experiment to examine the effects of self-adjusting the weight α of the fitness value in a smart way. We compared the original SOSAC that self-adjusts the weights of the fitness value as time passes with three variations of SOSAC that are given with the fixed weights (0 : 1, 1 : 1, 1 : 0) of the two fitness value components. Here 0 : 1 means that SOSAC uses energy state fitness only, 1 : 1 that equal weights of two fitness value components are used, and 1 : 0 that SOSAC uses neighborhood degree fitness only.

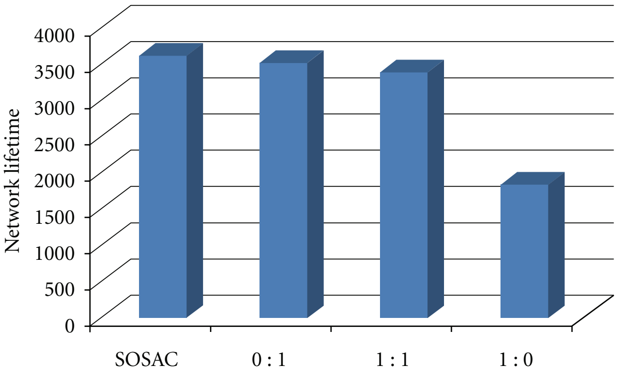

Figure 4: Effect of the fitness value with self-adjustment of its weights.

In Figure 4, the Y axis indicates the network lifetime and the X axis indicates the ratio of weights. The network lifetime of the automatically adjusted SOSAC is longer than those of the fixed weight (0 : 1, 1 : 1, and 1 : 0) SOSACs. This experiment shows that SOSAC adjusts suitable weights by itself in a smart way rather than providing weight parameters as inputs.

5.1.2. Smart Adjustment of Broadcasting Range. Along with the selection of an appropriate CH in the hierarchical routing, deciding on the number and size of the clusters is one of the important factors that determines the energy consumption of the sensor nodes. In this section, we suggest a method for deciding the appropriate number and size of clusters.

The number of clusters in SOSAC is decided by the broadcasting range of the sensor nodes. As the broadcasting range widens, the number of sensor nodes with CH candidate states, which participate in the election of a CH, increases. Consequently, a large cluster is formed with fewer clusters and CHs. With the fewer CHs, the energy consumption of the CHs for transmitting data packets over a long distance is also reduced. However, because there are more CMs, the CH has to receive more data packets from them. Therefore, the energy consumption is greater when receiving the data packets in a large cluster. On the contrary, if the broadcasting range narrows, the number of clusters and CHs is increased due to the formation of small clusters. With an increasing number of CHs, the energy consumption for receiving is reduced, whereas the energy consumption of the CHs for transmission increases. Therefore, the broadcasting range of SOSAC must be optimized to increase the network lifetime of the WSNs.

Since we can enumerate many values for the broadcasting range of a sensor field in a computer simulation, it is not difficult to optimize the broadcasting range during computer simulation. However, it is impossible for the sensor nodes, which have poor calculation capability, to determine the optimized broadcasting range in real time in the sensor field. Hence we try to find the optimized broadcasting range as the function of the number of sensors that varies from 50 to 250 in the 50 m*50 m sensor field using computer simulations for SOSAC. Hence we make the following predictive formula that calculates the optimized broadcasting range. The formula (5) was made from the following nonlinear regression model that has the adjusted coefficient of determination = 0.977:

$$y = \frac{0.000034x^2 - 0.02783x + 9.677}{\text{an area of sensor field}/2500\ (50\,\text{m} \ast 50\,\text{m})}, \qquad (5)$$

where y is the broadcasting range(m) and x the number of sensors in the sensor field.

If the sensor nodes sense the number of sensor nodes, an optimal broadcasting range can be adjusted in a smart way using the above formula, and the number of sensor nodes can be computed after the data packets are transmitted in the first round. In other words, respective CHs can determine the number of their CMs using the number of data packets transmitted by their CMs in the first period. This information is transmitted to the master CH through the intercluster routing, and the master CH can determine the number of sensor nodes scattered in the sensor field by summing them. Therefore, in the clustering phase of the first round of SOSAC, clusters are formed in the broadcasting range with the initial value to perform routing. When the first round is over, the master CH calculates an optimized broadcasting range and broadcasts this information to the entire sensor field so that all the sensor nodes can be adjusted in the optimized broadcasting range at the start of the second round.

Figure 5 compares the optimized broadcasting range calculated by an enumerative method without the smart adjustment mechanism with the network lifetime calculated with the smart adjustment mechanism. In Figure 5, the Y axis indicates the network lifetime when 20% of the sensor nodes have become drained in periods, and the X-axis shows five different experimental sizes. Figure 5 shows that the network lifetimes when SOSAC uses an optimized broadcasting range by finding an enumerative method are longer than those when SOSAC uses the smart adjustment mechanism of formula (5). In fact, the smart adjustment of the broadcasting range of formula (5) shows almost identical performance with the enumerative optimum values of the broadcasting range, which cannot be easily implemented in real WSNs. This demonstrates that the smart adjustment mechanism of predictive formula (5), which can be easily implemented in real WSNs, can provided good estimation of the optimal range of broadcasting range.

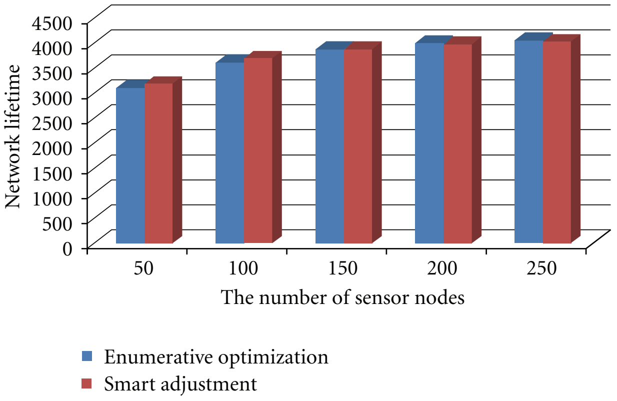

Figure 5: Effect of optimized broadcasting range.

5.1.3. Smart Backup Mechanism. A WSN can be used in inaccessible and dangerous areas such as battlefields and hazardous regions. Also some of the WSN sensor nodes can be broken down by vicious physical attack or cyber-attack before their energy is used up. Therefore, the robustness of WSNs is important for dealing with potential problems such as breakdown, failure, or attack of the sensor nodes.

When cluster-based routing is used, all information on the clusters may be lost if the sensor nodes in the node with CH state have become drained or failed due to breakdown, failure, or attack. Transmission of the entire network may fail depending on the location of the CH on the tree in the intercluster routing.

In order to avoid these risks, SOSAC has a “smart backup mechanism” that allows all the CMs of each cluster to recognize their CH’s states. If a CH losses its function unexpectedly, SOSAC has to select a new CH to minimize the transmission failures in the network. SOSAC can sense all CMs within its cluster range because it has to transmit the data packets that are broadcast by the CH to the CH of the neighboring cluster during intercluster routing. No additional overheads for sensing CHs or CMs are needed because the sensor nodes with CH state can be recognized depending on whether the data packets have been received or not. The smart backup mechanism of SOSAC selects a new CH when the CH has lost its function, and SOSAC needs overheads only to reconnect the network without using failed CHs. The smart backup mechanism is comprised of the following 2 steps: CH reelection step and rejoin intercluster routing tree step.

CH Reelection Step. The CH reelection step is carried out immediately when the CMs recognize a failure of the CH.

```
CH backup mechanism
        (1) CH re-election step
            If (A node with CM state = TRUE)
(1)                 If (CM does not receive a data packet from its CH)
                        Changes its state as a node with CH candidate state.
(2.1)               Broadcast a fitness-value-signal packet, within 2*(broadcasting range)
(2.2)               Receive fitness-value-signal packets
                        If (Two nodes at the same cluster) store
                        Else discard
(2.3)               Compare fitness value
                        If (One's fitness value is Max) changes its state as a node with CH state
                        Else changes its state as a node with undecided state
(3)             CH broadcasts a CH-signal packet to the whole area sensor field
(4)             Others receive CH-signal packets and set CH
        (2) Re-join intercluster routing tree step
            If (A node with CH state = CH)
                If (Re-elected CH = TRUE)
(5)                     Broadcast a re-join-signal packet to the whole area sensor field
                If (Not re-elected CH = TRUE)
(6)                     If (Receive re-join-signal packets) re-route establishment.
```

Algorithm 4: The CH backup mechanism algorithm.

In procedure 2.1 of Algorithm 4, the CMs broadcast the fitness values calculated in the previous CH selection step of clustering phase within 2*(broadcasting range) to restore the system as soon as possible when a CH failure occurs. Each CM compares the received fitness values with its own fitness value to select a new CH. The selected CH broadcasts the CH-signal packets (procedure 3). When the CMs receive the CH-signal packets, they select this CH as their new CH (procedure 4).

Rejoin Intercluster Routing Tree Step. All the CMs of the cluster assume that they can obtain information on the neighboring CHs on the intercluster routing tree of their own CH during the clustering step in the clustering phase. The new CH elected in procedures 5 and 6 of Algorithm 4 broadcasts a rejoin-signal packet to its neighboring CHs on the tree to restore the network to a stable state again. The smart backup mechanism of SOSAC is able to restore an unstable network to a stable state by using unused information without incurring additional overhead to the recognition of nodes with CH state.

The smart backup mechanism should be able to restore the network, not only if an error has occurred in the CH, but also if several sensor nodes have lost their functions in a certain concentrated area or sporadic failures of each sensor node in scattered areas. In order to check if the smart backup function works properly, we can conduct two types of experiments. The basic setting for the experiments is the same as the experiment for smart adaptability of the fitness value.

The first experiment observes how long the network is able to maintain itself in the event that all sensor nodes have lost their function to sense any objects within 10 m from a certain point by a physical or cyber error in a certain area. This experiment assumes that the sensor nodes within 10 m of the center of a coordinate which changes randomly have failed unexpectedly. In order to calculate an average value, we carried out the same experiments 10 times repeatedly and compared the average value with the normal state. To determine the occurrence of errors, we experimented with 3 cases of failure at the 501st, 1001st, and 1501st periods.

Figure 6 indicates that the earlier an error occurs, the shorter the network lifetime is. However, the maximum reduction rate of failure at the 501st period was only 1.8%, which reveals that the smart backup mechanism of SOSAC can cope with external errors in a smart way.

In the second experiment, we tested the result if 5, 10, or 15 sensor nodes stopped their functions sporadically in the WSNs in comparison with the result in the normal state.

Figure 7 implies that the occurrence of sporadic errors barely influenced the network lifetime by using a smart back up mechanism. The results of the two experiments show that the smart backup mechanism of SOSAC is a robust smart mechanism capable of coping with some changes in the external environment.

5.2. Performance Comparison with HEED.

5.2. Performance Comparison with HEED. In this section, the performance of SOSAC is compared with that of HEED, which is a well-known self-organized clustering and routing method. HEED maintains or improves the network performance by comparing the previous dispersing type methods using the self-organized method. In constructing the source code and simulation for SOSAC and HEED, we used Visual C++ of Visual Studio 2008.

5.2.1. Network Lifetime Comparison. In this experiment, all the cases were compared, ranging from the case in which the network lifetime of the WSN ceases as the energy of the first sensor node is drained to the case in which the network

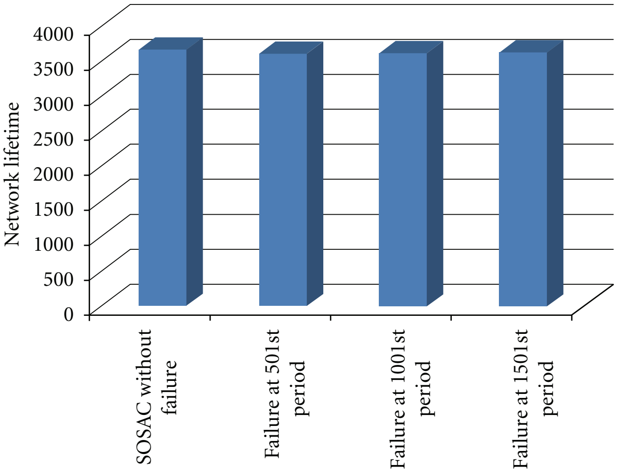

Figure 6: Experiment on concentrated failures.

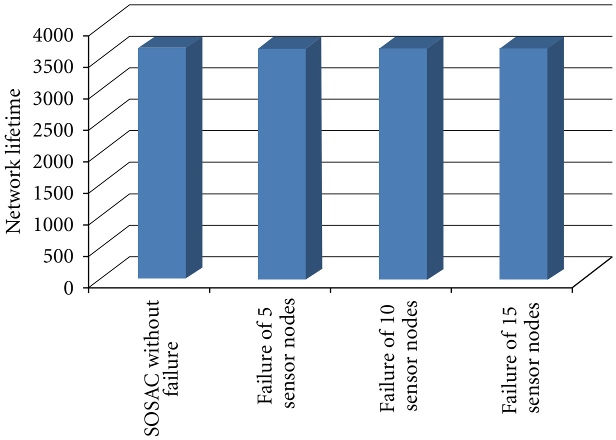

Figure 7: Experiment on sporadic failures.

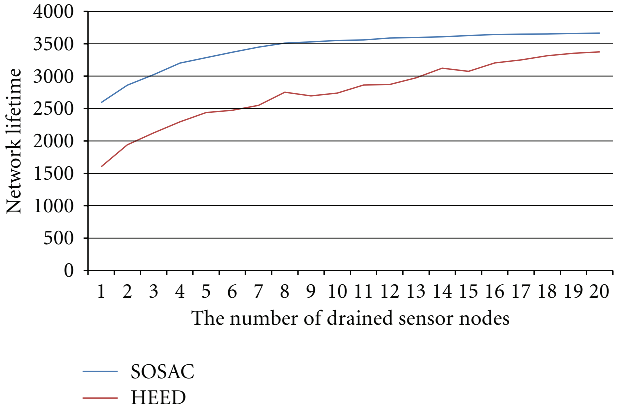

Figure 8: Comparison of network lifetime of SOSAC and HEED.

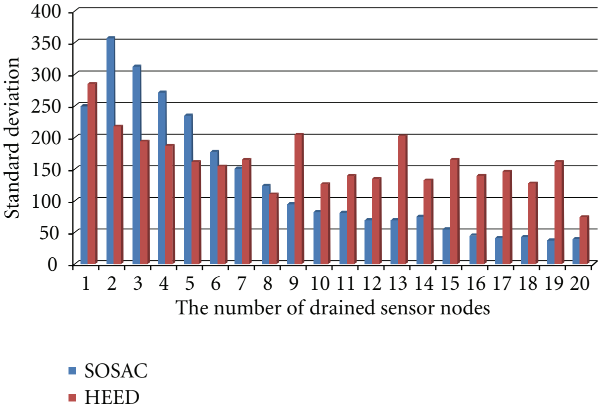

Figure 9: Comparison of standard deviations of network lifetimes for SOSAC and HEED.

lifetime of the WSN ceases as the energy of the 20th sensor node is drained.

Figure 8 shows that the network lifetime of SOSAC is between 61.7% (when the 1st sensor node is drained) and 8.5% (when the 20th sensor node is drained) longer than that of HEED. The network lifetimes until 20 sensor nodes had become drained were compared because it is impossible to observe the WSNs normally in the event that over 20% of the sensor nodes lose their functions due to the large vacuum in the sensing function.

We calculated the standard deviations of network lifetimes of SOSAC and HEED in Figure 9. On average, the standard deviation of SOSAC is 18.9% less than that of HEED, indicating that SOSAC shows more uniform performance than HEED.

5.2.2. Residual Energy Comparison. We compared the ratios of the residual energies in the WSNs to the network lifetimes of SOSAC and HEED.

In Figure 10, the Y-axis indicates the ratio of the residual energy in the WSN, and the X-axis indicates the number of drained sensor nodes. Figure 10 shows that the residual energy of SOSAC is between 16.9% and 52.7% shorter than that of HEED. The comparison of the two experiments (Figures 8 and 10) indicates that SOSAC has a longer network lifetime than HEED, and that the sensor nodes of SOSAC consume energy more uniformly than those of HEED when the network lifetime of WSNs ceases. This implies that SOSAC controls certain sensor nodes so as not to consume energy quickly by balancing the energy consumptions of the sensor nodes based on the appropriate fitness value, which gives a relatively long network lifetime.

5.2.3. Scalability Comparison. We compared network lifetimes of SOSAC and HEED for different size sensor fields and different numbers of sensor nodes, while maintaining the same density of sensor nodes. For example, the 100 m*100 m sensor field with 400 sensor nodes has four times as many as the 50 m*50 m sensor field that has only 100 sensor nodes. Here the network lifetime of the WSN is defined for the 20% of sensor nodes that are drained.

In Figure 11, the Y axis indicates the network lifetime in periods, and the X axis is the size of the sensor fields. Figure 11 shows that the network lifetime of SOSAC was a maximum of 10.7% and a minimum of 2.2% longer than

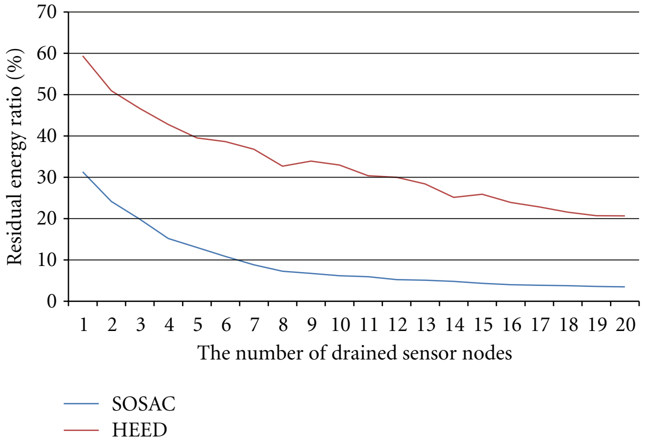

Figure 10: Experiment on residual energy ratios of SOSAC and HEED.

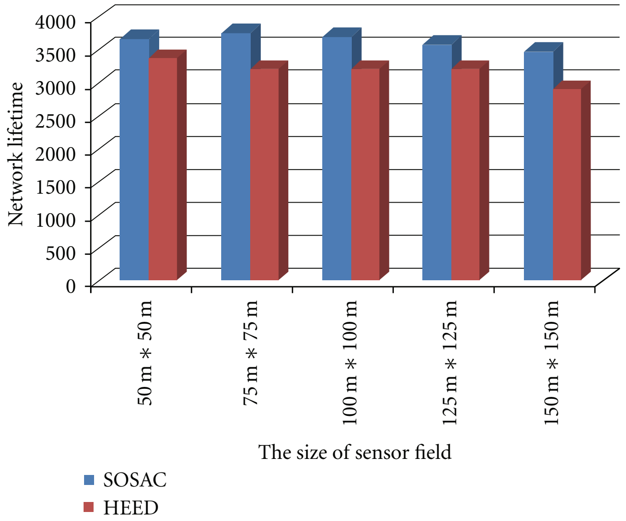

Figure 11: Experiment on scalability of SOSAC and HEED.

the network lifetime of HEED in different sensor field sizes and different numbers of sensor nodes. Hence, SOSAC shows good scalability in a smart way, which enables it to form clusters suitable for different sensor field sizes.

6. Conclusion

This paper has proposed a hierarchical clustering and routing model capable of maximizing the network lifetime through the decisionmaking of each sensor node based on local information by adopting a self-organized and smart-adaptive system in the design of the clustering and routing model of a WSN. The proposed method enables the sensor nodes to form clusters without a server or any external assistance, and the subsequent routing is performed based on it. The key advantage of this model is its ability to sense any environmental disturbances such as time changes, number of sensor nodes, and failures of sensor nodes using three smart adaptive mechanisms. The proposed method also demonstrated superior performance compared to that of an existing self-organized clustering method.

A smart mechanism for WSNs that can cope with diverse changes in the environment needs to be developed in future study. Appropriate research examples are changes in the operation or mobile environment. In addition, research on self-organized and smart-adaptive communication methods for other communication networks holds the promise of valuable results in the near future.

Acknowledgment

This paper is the result of the research with the support of National Research Foundation of Korea (2009-0074081) using the funds from the Ministry of Education, Science and Technology in 2009.

References

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[2] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6–28, 2004.

[3] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, vol. 3, no. 3, pp. 325–349, 2005.

[4] S. Mahfoudh and P. Minet, "Survey of energy efficient strategies in wireless ad hoc and sensor networks," in *Proceedings of the 7th International Conference on Networking (ICN '08)*, pp. 1–7, Cancun, Mexico, April 2008.

[5] F. Dressler, *Self-Organization in Sensor and Actor Networks*, WILEY, New York, NY, USA., 2007.

[6] E. Gelenbe, Z. Xu, and E. Seref, "Cognitive packet networks," in *Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence (ICTAI '99)*, pp. 47–54, November 1999.

[7] E. Gelenbe, R. Lent, M. Gellman, P. Liu, and P. Su, "CPN and QoS driven smart routing in wired and wireless networks," *Lecture Notes in Computer Science*, vol. 2965, pp. 68–87, 2004.

[8] M. D. Santo, A. Pietrosanto, P. Napoletano, and L. Carrubbo, "Knowledge based service systems," in *System Theory and Service Science: Integrating Three Perspectives in a New Service Agenda*, Social Science Electronic, New York, NY, USA, 2011.

[9] N. Israr and I. U. Awan, "Multilayer cluster based energy efficient routing protocol for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 4, no. 2, pp. 176–193, 2008.

[10] M. Younis, M. Youssef, and K. Arisha, "Energy-aware management for cluster-based sensor networks," *Computer Networks*, vol. 43, no. 5, pp. 649–668, 2003.

[11] Y. T. Hou, Y. Shi, H. D. Sherali, and S. F. Midkiff, "On energy provisioning and relay node placement for wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 4, no. 5, pp. 2579–2590, 2005.

[12] C. Prehofer and C. Bettstetter, "Self-organization in communication networks: principles and design paradigms," *IEEE Communications Magazine*, vol. 43, no. 7, pp. 78–85, 2005.

[13] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual*

Hawaii International Conference on System Siences (HICSS-33), p. 223, January 2000.

[14] Y. Sun and X. Gu, "Clustering routing based maximizing lifetime for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 5, no. 1, p. 88, 2009.

[15] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.

[16] O. Younis, S. Fahmy, and P. Santi, "An architecture for robust sensor network communications," *International Journal of Distributed Sensor Networks*, vol. 1, no. 3-4, pp. 305–327, 2005.

[17] M. M. Hasan and J. P. Jue, "Survivable self-organization for prolonged lifetime in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2011, Article ID 257156, 11 pages, 2011.

[18] D. L. Tennenhouse and D. J. Wetherall, "Towards an active network architecture," in *Proceedings of the Multimedia Computing and Networking (MMCN 96), and Computer Communcation Review*, vol. 26, no. 2, pp. 5–18, April 1996.

[19] E. Amir, S. McCanne, and R. Katz, "An active service framework and its application to real-time multimedia transcoding," *Computer Communication Review*, vol. 28, no. 4, pp. 178–189, 1998.

[20] A. Ren and G. Q. Maguire Jr., "A smart network with active services for wireless context-aware multimedia communications," in *Proceedings of the Wireless Communications and Systems, Emerging Technologies Symposium*, pp. 17.1–17.5, April 1999.

[21] S. Stone, M. Zyda, D. Brutzman, and J. Falby,, "Mobile agents and smart networks for distributed simulations," in *Proceedings of the 14th Workshop on Standards Interoperability of Distributed Simulations*, pp. 909–917, Institute for Simulation and Training, Orlando, Fla, USA, March 1996.