

Research Article

A Diagnosis-Based Clustering and Multipath Routing Protocol for Wireless Sensor Networks

Wenjun Liu,¹ Shukai Zhang,^{1,2} and Jianxi Fan¹

¹ School of Computer Science and Technology, Soochow University, Suzhou 215006, China

² State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China

Correspondence should be addressed to Jianxi Fan, jxfan@suda.edu.cn

Received 29 March 2012; Accepted 29 May 2012

Academic Editor: Yong Sun

Copyright © 2012 Wenjun Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In wireless sensor networks, it is of great importance for fault diagnosis to ensure the gathering information accuracy and reduce energy additionally consumed by faulty nodes, for the deployment of a large number of sensor nodes in hostile environment. In this paper, we propose an energy-efficient data collection protocol which consists of clustering and multipath routing. Clustering based on fault diagnosis eliminates the possibility of cluster heads (CHs) acting by faulty nodes which reduce energy consumption and fault information transmission. Multipath routing provided by directed acyclic graph (DAG) increases system fault tolerance. Furthermore, clustering and multihop routing consider residual energy and routing cost, respectively; thus balanced energy consumption is achieved. Performance analysis shows that the message complexity disseminated in clustering and fault diagnosis is acceptable. Simulations demonstrate that the protocol has better energy efficiency compared with other related protocols.

1. Introduction

In recent years, Wireless Sensor Networks (WSNs) have become an attractive technology for a large number of applications, ranging from monitoring to event detection and target tracking [1]. To design and deploy successful WSNs, many issues need to be resolved such as deployment strategies, energy conservation, fault-tolerant routing in dynamic environments, localization, and fault diagnosis. To extend the network lifetime as long as possible, energy efficiency becomes one of the basic tenets in the WSNs protocol design. There are several possible solutions to balance energy consumption, such as deployment optimization [2], topology control [3], and data aggregation [4].

Among these schemes, clustering provides an effective way for promoting energy efficiency [5–9]. In clustering schemes, sensor nodes are organized into clusters and a main node is selected as the cluster head (CH) of a cluster, and the other nodes are called cluster members (CMs). Each CM collects local data from the environment periodically and then sends it to the CH. When the data from all the CMs arrives, the CHs aggregate the data and send it to the BS via single-hop or multihop. When the network is partitioned into clusters, data transmission can be classified

into two stages, that is, intra- and intercluster communication. Mhatre and Rosenberg have shown that multihop intercluster communication mode is usually more energy efficient because of the characteristics of wireless channel [10]. Thus it is better to let CHs cooperate with each other to forward their data.

Due to the low cost and the deployment of a large number of sensor nodes in uncontrolled or even harsh environments, it is common for nodes to become faulty. The existence of these faulty nodes in WSNs brings the data collection protocol many adverse effects such as nonuniform distribution of the clustering effect and inaccuracy of the information collected. In addition, too many faulty nodes directly affect the connectivity of the network, resulting in premature network partition, which is an important factor affecting network lifetime. How to identify faulty nodes and eliminate the impact of these nodes gradually attracts more and more attentions.

Multipath routing between a source and a destination is a promising routing scheme to achieve robustness, load balancing, bandwidth aggregation, congestion reduction, and security compared to the single shortest-path routing that is usually used in most networks [11]. Techniques developed for multipath routing are often based on employing multiple

spanning trees or directed acyclic graphs (DAGs) [12, 13]. Both of them offer resiliency to single-link failure. Due to the relative instability of communication links in WSNs, it is necessary to explore the feasibility of the multipath routing.

In this paper, we analyzed the opportunities and challenges of fault diagnosis based on comparison model in WSNs, and so far there is little work done for this owing to the inherent characteristic of WSNs. Furthermore, we design and implement a fault diagnosis-based clustering and multipath routing protocol (FDCM) for wireless sensor networks. FDCM mainly includes two phases: fault diagnosis-based clustering and multipath routing selection. The features making FDCM distinct are as follows. (i) Clustering based on fault diagnosis eliminates the possibility of CH acting by faulty nodes which reduce energy consumption and fault information transmission. (ii) The constructed DAG provides multipath routing approach, which increases system fault tolerance. (iii) Clustering and multihop routing consider residual energy and transmission cost, respectively, which balance energy consumption and promote network efficiency. Consequently, the communication overhead and network lifetime of FDCM are desirable.

The rest of the paper is organized as follows. The next section describes the related work. Section 3 introduces the network model and related terminologies at first, after which is the fault diagnosis model based on comparison model. Subsequently, it analyzes the fault diagnosis addressing WSNs and provides several requirements. The system design including clustering construction and multihop routing is detailed in Section 4. The simulation results are given in Section 5. Finally, the conclusion and the future work are drawn in Section 6.

2. Related Work

Clustering provides an effective way for prolonging the lifetime of WSNs. Heinzelman et al. [5] first proposed a clustering protocol called LEACH for periodical data gathering applications. It is an application-specific data dissemination protocol that uses clustering to prolong the network lifetime. HEED [6] introduced a variable known as cluster radius which defines the transmission power to be used for intracluster broadcast. EEUC [8] and EADUC [9] introduced cluster head competitive algorithms which extend LEACH and HEED by choosing CHs with more residual energy. Both of them achieve well distribution of CHs.

As faults are inevitable in every distributed computer system, especially in WSNs which consist of a large number of capacity-limited nodes, it is important to be able to determine which of them is working and which is faulty. Comparison-based diagnosis is a realistic approach to detect faulty nodes based on the outputs of tasks executed by system nodes. The model is based on comparisons of the outcomes returned by different units executing the same task and uses the invalidation rule of the generalized Maeng and Malek (gMM) model [14, 15]. Comparison-based diagnosis initially used for multiprocessor system has been firstly applied to mobile ad hoc networks (MANETs) by Chessa

and Santi [16]. Later, Elhadef et al. considered the problems of self-diagnosis of wireless mesh networks (WMNs) and MANETs using the comparison approach [17, 18].

For WSNs, traditional comparison-based fault diagnosis protocols for multiprocessor systems, WMNs and MANETs are not suitable without changing. To the best of our knowledge, so far fault diagnosis based on the comparison model has not yet been applied to WSN efficiently. Chen et al. proposed a distributed fault-detection algorithm to locate the faulty sensors [19]. It calculated the measurement difference between neighbor sensors at different times to find if the current measurement of a sensor is different from its previous measurement. Wang et al. provided a cluster-based real-time fault diagnosis aggregation algorithm for WSNs [20]. The protocol is based on the comparison approach aiming at achieving a correct and complete diagnosis for hierarchical WSNs. They assumed that each sensor can transmit data to any other sensor and can communicate directly with the BS, which is unrealistic in practice.

3. System Models

3.1. Network Model. In this paper, we consider a sensor network consisting of N static and homogeneous sensor nodes uniformly deployed over a vast field to continuously monitor the environment. The communication topology of WSN is usually represented by the graph $G = (V, E)$, where each vertex $v \in V$ represents a sensor node and each edge $(u, v) \in E$ represents a communication link. For any vertex $v \in V$, $N(v)$ is the set of all vertices that are adjacent to v in G . We denote the i th sensor by s_i and the corresponding sensor node set $S = \{s_1, s_2, \dots, s_N\}$. Assume that links are bidirectional in nature, which may be realized using two unidirectional links. We denote a bidirectional link between nodes s_i and s_j as $s_i - s_j$, while the directed link from s_i to s_j is denoted by $s_i \rightarrow s_j$. When a link fails, it means that both directed edges have failed. For graph terminology and notation not defined here we refer the reader to [21]. Moreover, we make the following assumptions about the sensor nodes and the underlying network model.

- (1) There is a unique identifier for every node. The computing, storage, and energy power of sensors are limited. Nodes are capable of operating in an active mode or a low-power sleeping mode.
- (2) There is a stationary base station (BS) located far from the sensing field. BS distributes control messages in one-hop mode, and its energy and computing capability are not limited.
- (3) Nodes are location-unaware, but a node can compute the approximate distance to another node based on the received signal strength, if the transmitting power is known.
- (4) All nodes are static and homogeneous which are organized as clusters. CMs communicate with CH with one-hop manner, while the communication between CHs and BS is relayed by other CHs.

- (5) Proper data aggregation mechanism is adopted for energy saving, and there exists a MAC protocol which is executed to solve contentions, providing reliable one-hop broadcast over logical links.

All of these assumptions are typical for wireless sensor networks, which means that our model is general, that is, not unrealistic. We use a simplified model for the communication energy dissipation [22]. Both the free space (d^2 power loss) and the multipath fading (d^4 power loss) channel models are used, depending on the distance between the transmitter and receiver. The energy spent for transmission of an l -bit packet over distance d is

$$E_{Tx}(l, d) = \begin{cases} l \times E_{elec} + l \times \epsilon_{fs} \times d^2, & d < d_0 \\ l \times E_{elec} + l \times \epsilon_{mp} \times d^4, & d \geq d_0. \end{cases} \quad (1)$$

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

$$E_{Rx}(l) = l \times E_{elec}. \quad (2)$$

3.2. The Diagnosis Model. Each node in the system can be in one of two states: faulty or fault-free. There are different classifications for faulty type. Based on duration, faults can be classified as permanent, intermittent, and transient. A transient fault will eventually disappear without any apparent intervention, whereas a permanent one will remain unless it is repaired and/or removed by external administrator. Based on how a failed node behaves once it has failed, faults can be either hard or soft. When a node is hard-faulted, it cannot communicate with the rest of the system. In WSNs, a node can be hard-faulted either because it is crashed or due to battery depletion. Soft faults are subtle, since a soft-faulted node continues to operate and to communicate with the other nodes in the system, although with altered behaviors. In this paper, we utilize the invalidation rule of the gMM model [14, 15] that is summarized in Table 1.

In the gMM model, diagnosis is based upon comparison of the results generated by test tasks assigned to pairs of units with a common neighbor. Let u be a unit adjacent to both unit v and w . If nodes u, v , and w are fault-free, then the results agree and the comparison outcome is 0. If unit u is fault-free and any unit v or w is faulty, then the results disagree and the comparison outcome is 1. If unit u is faulty, then the comparison outcome may be not reliable (0 or 1), regardless of the state of v and w . Assuming that the topology of the network does not change during the diagnosis executing, comparison-based approach relies on the following operations.

(1) *Test Request Generation.* In order to test adjacent nodes, each node u generates a test sequence number i , a test task T_i , the expected result $R_{u,i}$ and sends the test request message TEST_REQ(u, i, T_i) to its neighbors $N(u)$ at time t . Then, node u sends a message T_{out} to initiate the timer. Node

TABLE 1: The invalidation rule of the gMM model.

u	v	w	Comparison result of v and w by u
Fault-free	Fault-free	Fault-free	0
Fault-free	Faulty	Fault-free	1
Fault-free	Fault-free	Faulty	1
Fault-free	Faulty	Faulty	1
Faulty	Any	Any	Not reliable

u expects to receive response message that comes from its neighbors within this time bound.

(2) *Test Request Reception.* Any node v in $N(u)$, upon receiving TEST_REQ, generates the test result $R_{v,i}$ for T_i and sends test response message TEST_RES($u, i, R_{v,i}$) to $N(v)$ at time t' , with $t < t' < t + T_{out}$. Note that (u, i) is known as header of message, which is used to uniquely identify the test task and the sender.

(3) *Test Response Reception.* Any node w in $N(v)$ at time t' , upon receiving TEST_RES($u, i, R_{v,i}$), does the following.

Case 1. If $w = u$, that is, w is the testing node itself, it compares $R_{v,i}$ with the expected result $R_{u,i}$ and generates the comparison outcome. Node v is diagnosed as fault-free if the outcome is 0, as faulty otherwise.

Case 2. If $w \neq u$, this means that w is not the testing node, we should check whether w is u 's neighbor. The following two cases arise.

Case 2.1. $w \in N(u)$ at time t . In this case, $w \in N(v) \cap N(u)$, that is, w and the tester node u share at least one common adjacent node v . Node w received the test request TEST_REQ from u and the test response TEST_RES from v , hence it can compare $R_{v,i}$ with $R_{w,i}$. Node v is diagnosed as fault-free if the comparison outcome is 0, as faulty otherwise. Figure 1(a) illustrates this case.

Case 2.2. $w \notin N(u)$ at time t . If there exists some $z \in N(u)$ such that $R_{z,i} = R_{v,i}$ then both nodes are diagnosed as fault-free; otherwise, if node z (node v) has been diagnosed as fault-free, then node v (node z) is diagnosed as faulty. Otherwise, the test result $R_{v,i}$ is stored. Figure 1(b) illustrates this case.

(4) *Timeout Reception.* After sending its test response message, node u initiates a timer to T_{out} in order to guarantee that its neighbors will response within this time bound. Once this bound expires, the testing node u receives the timeout message from the timer and diagnoses all the nodes that did not reply to the test request as faulty.

3.3. Problems and Requirements. The inherent characteristics of WSNs make the direct applications of existing fault diagnostic models that have been pervasively applied in

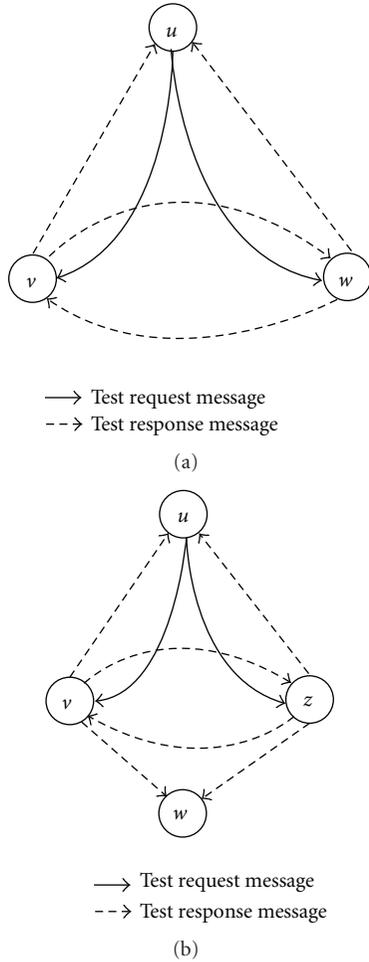


FIGURE 1: (a) Node w received u 's test task and v 's corresponding response reply. (b) Node w received v and z 's response corresponding to u 's test task.

traditional multiprocessor systems, WMNs, and MANETs are unrealistic. In the following, we sum up these problems and give the corresponding analysis.

On the one hand, we require a suitable fault diagnosis mechanism for WSNs which is energy-efficient and message complexity acceptable. Sensor nodes are limited in battery energy, computing, and storage capacity, and the protocol design need to consider the energy efficiency of the network nodes. That is, the amount of calculation performed by nodes should be moderate and the traffic generated by the exchange of messages should be reasonable. Clearly, there should be some mechanisms to isolate messages flooding in the networks. Owing to instability of nodes, the number of faulty nodes is far more than the traditional multiprocessor systems or MANETs. Besides, constrained by wireless communication capacity, the structure properties in terms of regularity, connectivity, and so forth, of the communication topology are worse than interconnection networks. As a result, the traditional diagnostic measures cannot meet the fault diagnosis requirements in WSNs.

On the other hand, an excellent clustering mechanism provides a good basis for fault diagnosis. Clustering WSNs, CMs only perform the sensing tasks, while in addition to collecting data from CMs, CHs also have to fulfill a variety of other tasks, such as data aggregation, cluster maintenance, and communication with BS via multihop or one-hop way. Considering the efficient mechanisms for CHs to reduce and balance energy consumption and to ensure the CHs fault-free is crucial, especially in fault diagnosis applications. Besides, nodes are left unattended after deployment requires adaptive node fault processing mechanism. For instance, the fault conditions can be transmitted to the BS, which are uniformly processed according to the actual requirements.

In this paper, we summarize the protocol design as clustering problem based on fault diagnosis and multipath routing problem based upon DAG. Since the networking nodes in WSNs are very limited in resources, clustering should not only have small size, but also be constructed with low communication overhead and computation cost. In addition, the amounts of communication and computation should be scalable as the networks are typically deployed with large network size. For the multipath routing problem, the backbone network composed of CHs can be abstracted as an edge and vertex weighted graph $G(V, E, W, R)$. We view the CHs as vertexes set V , the communication links as edges set E , the communication cost (edge weight) as $W: E(G) \rightarrow R^+$, and the residual energy (vertex weight) as $R: V(G) \rightarrow R^+$. In order to provide solutions for the two problems, we believe the following requirements should be met.

- (1) Clustering should be completely distributed with accepted message complexity. Each node independently makes its decisions based on local information and results into well-distributed CHs over the sensing field.
- (2) At the end of clustering, each node is either a cluster head or a member node.
- (3) Using message complexity acceptable diagnostic mechanism to eliminate the harmful influence and to avoid unnecessary energy consumption imposed by faulty nodes.
- (4) Utilizing multipath routing mechanism to make the gathering data transmission reliable. In particular, it is necessary to select the cost-aware or energy-aware communication path among all of the possible paths.
- (5) Avoiding excessive energy consumption of CHs and maintaining the energy consumption balance of network nodes.

4. System Design

Our fault diagnosis-based clustering and multipath routing data collection protocol (FDCM) can be divided into two phases mainly as follows. Phase (I): clustering construction based on fault diagnosis; Phase (II): resilient multipath routing selection. In the following, we explain how FDCM works in detail.

4.1. Clustering Construction Based on Fault Diagnosis. In network deployment phase, BS broadcasts a HELLO message to all the nodes in the network at a certain power level which includes a certain number of candidate CHs selected in advance. After receiving this message each sensor node checks whether it is a candidate CH. Each selected CH computes the approximate distance to the BS based on the received signal strength and then executes a distributed cluster head competitive algorithm similar to EEUC [8]. Our CH competition is primarily based on the fault status and residual energy of candidate CHs. The size of cluster is controlled by competition radius R , which is a constant tuned by typical situation. In addition, let s_i denote a CM and c_i represents any cluster i , respectively.

Definition 1. Candidate CH s_i 's adjacent CH set N_{CH} is given by $s_i.N_{CH} = \{s_j \mid s_j \text{ is candidate CH, and } \text{dist}(s_i, s_j) < R\}$. Furthermore, if the nodes in $s_i.N_{CH}$ are fault-free, then the set is denoted by $s_i.N_{FF_CH}$, otherwise, $s_i.N_{F_CH}$. Obviously, $s_i.N_{CH} = s_i.N_{FF_CH} \cup s_i.N_{F_CH}$.

Before clustering, BS selects predefined candidate CHs randomly on certain probability to compete for final CHs. For the sake of saving energy, nodes that fail to be candidate CHs keep sleeping until the cluster head competition stage ends.

The distributed clustering algorithm which is initiated by BS and executed by each candidate CH is presented in Algorithm 1. First, each candidate CH broadcasts a COMPETE_CH(s_i .ID, s_i .R, s_i .RE) message which contains its node ID, competition radius R , and residual energy RE. After the construction of N_{CH} has finished in lines 2–4, each candidate CH checks the fault status of its N_{CH} based on comparison model approach in lines 5–24. Candidate CH s_i generates a test sequence number i and the correspondent test task T_i and sends a test request message TEST_REQ(s_i .ID, i , T_i) to its adjacent candidate CH set $s_i.N_{CH}$. Node s_i waits for the responses of $s_i.N_{CH}$ and diagnoses their status according to the comparison model. Lines 25–36 describe the CH competition process. The candidate CHs with faulty status (soft-faulty nodes) have no qualification for the competition. If s_i belongs to $s_i.N_{CH}$ and s_i receives a FINAL_CH message from s_j , then s_i will give up the competition immediately. After that the fault-free candidate CH makes a decision whether it can act as a final CH. In particular, if the constructed adjacent set N_{CH} is null, then the candidate CH becomes final cluster head immediately. Once fault-free s_i finds that its residual energy is more than all the nodes in its S_{CH} , it will win the competition.

After all the final CHs have been elected, immediately, previous sleeping nodes now are waked up and each CM chooses their closest CH with the largest signal strength received. All the CMs register with the CH by sending a JOIN_CLU message. In order to determine the status of CMs in each cluster, the CH sends a test quest TEST_REQ to its member nodes. These nodes compute the tasks and feed back the results to the sender. The lower layer fault diagnosis algorithm based on comparison protocol is presented in Algorithm 2. Once the faulty nodes are determined, they will

be ordered to turn dead. The final CH sets up a TDMA schedule and transmits it to the nodes in the cluster. After the TDMA schedule is known by all nodes in the cluster, the clustering phase is completed and the data transmission stage begins. Based on the execution of Algorithm 1, we can draw the following theorem.

Theorem 2. In clustering stage, FDCM has a message exchange complexity of $O(1)$ per node and $O(N)$ for entire network.

Proof. During the execution of clustering algorithm, each candidate CH sends a COMPETE_CH message at first. In order to identify the faulty candidate CHs, which will be deprived of the eligibility of final CHs, each of them sends TEST_REQ and receives TEST_RES message one after another. If it becomes a final CH then broadcasts a FINAL_CH message to declare its win, otherwise broadcasts a QUIT_ELECT message to exit. After the declaration of winning the election, each regular node broadcasts a JOIN_CLU message. So each node has the message complexity of $O(1)$.

Assume that network size is N , the number of candidate CHs is M and the number of final CHs/clusters is N_c ($N_c \leq M \leq N$). In clustering stage, the overall message overhead is $3M + N_c + (M - N_c) + (N - N_c) = 4M + N - N_c = O(N)$.

The theorem is proved. \square

From Theorem 2 we can conclude that the clustering stage has a low message complexity both for individual node and entire network, thus requirement (1) is satisfied.

Theorem 3. At the end of clustering phase, any fault-free node either is a final CH or a CM. Furthermore, each CM exactly belongs to a cluster, and only one final CH is allowed in each competition range.

Proof. During the execution of Algorithm 1, for the nodes in sensor network there are at most four states in total: *RegularNode*, *CandidateCH*, *FinalCH* and *deadNode*. Here the status of *RegularNode* and *FinalCH* means it is a CM and CH, respectively.

In the following we first show that any node is either a final CH or a CM after execution of Algorithm 1. Initially, in addition to the selected candidate CHs in advance, the remainder nodes are all regular nodes. For candidate CHs, in lines 5–24, each of them knows the fault status of its adjacent CHs. If the node is determined as faulty nodes, then they quit the competition process, immediately. In lines 33–36, the faulty candidate CHs are ordered to turn dead (*deadNode*), and they do not participate in the subsequent work any more. For any candidate CH s_i , if it has not any adjacent node, then Algorithm 1 executes line 26 and s_i becomes a final CH at once. Furthermore, in lines 25–36 s_i either becomes a final CH (*FinalCH*) or becomes a CM (*RegularNode*) mutually exclusive.

After the election of final CHs has finished, each CM registers with only one CH based on received signal strength, thus each CM exactly belongs to a cluster. The competition process shows that for any candidate CH s_i 's adjacent CH

```

1. Candidate CH  $s_i$  broadcasts a competition message
   COMPETE_CH( $s_i$ .ID,  $s_i$ .R,  $s_i$ .RE);
2. On receiving a COMPETE_CH from  $s_j$ ;
3. if ( $\text{dist}(s_i, s_j) < R$ ) then
4.    $s_i.N_{CH} \leftarrow s_i.N_{CH} \cup s_j$ ;
5.  $s_i$  generates test sequence number  $i$  and the test task  $T_i$ ;
6.  $s_i$  broadcasts a TEST_REQ( $s_i$ .ID,  $i$ ,  $T_i$ );
7. On receiving a TEST_REQ( $s_j$ .ID,  $j$ ,  $T_j$ ) from  $s_j$ ;
8.    $s_i$  generates the test result  $R(s_i, j)$  for task  $T_j$ ;
9.    $s_i$  broadcasts the Test_RES( $s_j, j, R_{si}$ ) to  $s_i.N_{CH}$ ;
10. On receiving a TEST_RES from  $s_j$  initiated by  $s_k$ 
11.  if ( $s_k = s_j$ ) then
12.    if ( $R(s_j, j) = R(s_i, j)$ ) then
13.       $s_i.N_{FF,CH} \leftarrow s_i.N_{FF,CH} \cup \{s_j\}$ ;
14.    else if  $s_i.N_{F,CH} \leftarrow s_i.N_{F,CH} \cup \{s_j\}$ ;
15.    else if ( $s_k \in s_i.N_{CH}$ ) then
16.      if ( $R(s_j, j) = R(s_i, j)$ ) then
17.         $s_i.N_{FF,CH} \leftarrow s_i.N_{FF,CH} \cup \{s_j\}$ ;
18.      else  $s_i.N_{F,CH} \leftarrow s_i.N_{F,CH} \cup \{s_j\}$ ;
19.    else if ( $s_k \notin s_i.N_{CH}$ )
20.      if ( $s_i$  has received a TEST_RES from  $s_z \in s_i.N_{CH}$ 
        and  $R(s_j, j) = R(s_z, j)$ ) then
21.         $s_i.N_{FF,CH} \leftarrow s_i.N_{FF,CH} \cup \{s_j, s_z\}$ ;
22.      else if ( $s_i$  received a TEST_RES from  $s_z \in s_i.N_{CH}$ 
        and  $R(s_j, j) \neq R(s_z, j)$  and  $s_z \in s_i.N_{FF,CH}$ ) then
23.         $s_i.N_{F,CH} \leftarrow s_i.N_{F,CH} \cup \{s_j\}$ ;
24.      else store the test response;
25. if ( $s_i.N_{CH} = \text{NULL}$ ) then  $s_i$ .state  $\leftarrow$  finalCH;
26. while ( $s_i$ .state = candidateCH)
27.  if ( $s_j \in s_i.N_{FF,CH}$  and  $s_i.E_{RE} > s_j.E_{RE}$ ) then
28.    broadcast FINAL_CH( $s_i$ .ID);
29.     $s_i$ .state  $\leftarrow$  finalCH;
30.  On receiving a FINAL_CH from  $s_j$ ;
31.  if ( $s_j \in s_i.N_{CH}$ ) then
32.    broadcast QUIT_ELECT( $s_i$ .ID);
33.     $s_i$ .state  $\leftarrow$  RegularNode;
34.  On receiving a QUIT_ELECT from  $s_j$ 
35.  if ( $s_j \in s_i.N_{CH}$ ) then
36.    remove  $s_j$  from  $s_i.N_{CH}$ ;
37. end while

```

ALGORITHM 1: The distributed CH election algorithm based on fault diagnosis executed by candidate CH s_i .

set N_{CH} if s_i wins the competition, then the nodes in $s_i.N_{CH}$ quit competition. Otherwise, if s_i receives a message, it quits competition too, so only one final CH is allowed in each competition range.

To sum up, the theorem is proved. \square

Fault diagnosis in this paper refers to all faulty nodes within the sensor network are identified correctly and these faulty nodes are ordered to stop working, and the fault conditions about each cluster are reported to BS via data transmission by CHs. Upon receiving fault information, BS takes appropriate actions, such as forbid faulty nodes taking part in the final CH election in the next round. According to the description and analysis above, the fault diagnosis process consists of two phases. First, it eliminates the candidate CHs to participate in the final CH competition in the process

```

1. generate test task  $T_j$  with sequence number  $j$ ;
2. broadcast test request TEST_REQ( $c_i$ .CH,  $j$ ,  $T_j$ ) to its
   member nodes  $c_i$ .CMs;
3. generate the expected result  $R(c_i$ .CH,  $j$ ) for  $T_j$ ;
4. set the timer to be  $T_{out}$ ;
5. Initialize the faulty CM set  $c_i.S_{F,CM}$  and fault-free
   CM set  $c_i.S_{FF,CM}$  with NULL;
6. Each CM sends test response TEST_RES( $c_i$ .CM,  $j$ ,  $T_j$ )
   in random back-off time  $t' < T_{out}$ ;
7. while receiving test response TEST_RES from any
   node  $c_i$ .CM in  $c_i$ .CMs
8.  if ( $R(c_i$ .CH,  $j$ ) =  $R(c_i$ .CM,  $j$ ) then
9.     $c_i.S_{FF,CM} \leftarrow c_i.S_{FF,CM} \cup \{c_i$ .CM $\}$ ;
10.  else
11.     $c_i.S_{F,CM} \leftarrow c_i.S_{F,CM} \cup \{c_i$ .CM $\}$ ;
12.  end while
13. send TURNDEAD message to  $c_i.S_{F,CM}$ ;

```

ALGORITHM 2: Lower layer diagnosis algorithm executed by cluster head c_i .CH.

of clustering. Secondly, after the clustering is finished, the fault diagnosis is done based on special comparison model between CH and CMs. These two phases together complete the diagnosis of all faulty nodes in the network correctly. A diagnostic message can be a test request, a test response, or a timeout message. The message complexity of diagnostic algorithm is presented in Theorem 4.

Theorem 4. *The communication complexity of our comparison-based fault diagnostic approach is $O(N+M(1+d_{\max}))$, where M and d_{\max} denotes the number of candidate CH and its maximum degree, respectively.*

Proof. According to Theorem 3, at the end of the clustering stage, any fault-free node is either a final CH or a CM. Each candidate CH s_i generates at most one test request TEST_REQ. In turn, the test request generates at most $|s_i.N_{CH}| < d_{\max}$ test responses. The communication complexity during CH competition is $O(M(1+d_{\max}))$. The fault diagnosis of CMs is done by CH generates and sends test request TEST_REQ. The test request sent by CH generates N_c test responses. CHs obtain the fault status of their CMs by comparison of test tasks. The communication complexity of each cluster is $O(N_c)$ and for entire network, it is $O(N)$. Thus, the overall communication complexity is $O(N+M(1+d_{\max}))$. \square

4.2. Resilient Multipath Routing. Before delivering their data to BS, each CH first aggregates the sensing data from its CMs, and then sends the data packet plus the fault information to BS via a multihop fault-tolerant path. We assume that any two CHs within their communication scope can communicate with each other and consider them as neighboring. Each CH has the information about its neighbors. The distributed multipath routing consists of three subphases: Construct Connected Network, Choose Next-hop Neighbor and Route Maintenance.

4.2.1. Construct Connected Network. Assume $c_i.CH$ is the source CH, from where the sensing data and fault information is aggregated; $c_j.CH$ is a candidate relay CH. The factors that are taken into account by each CH for constructing communication path are summarized as:

Condition 1. $\text{dist}(c_i.CH, c_j.CH) < 2R$: the communication radius is defined as two times of cluster radius. The distance relation ensures that connectivity of constructed network (graph), since any two CHs have communication link (edge) only if they meet with this condition.

Condition 2. $\text{dist}(BS, c_j.CH) < \text{dist}(BS, c_i.CH)$: the distance relation ensures that the constructed routing paths moving toward BS from the source $c_i.CH$. The selection of any relay CH always approaches BS geographically, which provides direction for data transmission.

Condition 3. $c_i.CH.E_{RE} > E_{relay}$: the energy relation ensures that the relay nodes should have enough residual energy for data transmission in practice. Relay nodes' residual energy is greater than the energy sum of receiving and sending data packets. Therefore, for balancing network energy consumption, it is necessary to protect the relay nodes' residual energy and give priority to the use of the relay nodes with more remaining energy.

According to the model described in Section 3, we have mapped multipath routing problem into finding communication path in weighted graph $G(V, E, W, R)$. On the basis of this mapping, we assume that there is a logical communication link between any two nodes which meet conditions 1–3 synchronously. Note that the links direction is always from sources to BS, thus a directed connected cost network (vertex and edge weighted digraph) is built. In order to model and depict mapped multipath routing problem, the definition of DAG is given followed by a theorem which satisfies requirement (4).

Definition 5. A directed acyclic graph (DAG) is a directed graph with no directed cycles. We say that a DAG is rooted at r if it is the only node in the DAG that has no outgoing edges. Every other node has at least one outgoing edge.

Theorem 6. *The mapped weighted graph $G(V, E, W, R)$ which meets Conditions 1 and 2 synchronously is a connected DAG rooted at BS. That is, the graph is connected, directed and without directed cycles. Furthermore, for any vertex in G , it is BS reachable.*

Proof. As stated before, there is an edge between any two vertices which meet with Condition 1. For any two adjacent vertices u and $v \in V$, $\text{edge}(u, v) \in E$. Condition 2 ensures that the direction of edge (u, v) is always from u to v if $\text{dist}(u, BS) \geq \text{dist}(v, BS)$, according to distance relation. Therefore, graph G is directed.

Suppose that the mapping produces at least two connected components $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ such that any $v_1 \in V_1$ cannot communicate with any $v_2 \in V_2$.

Note that a connected component is a maximal connected subgraph of G . Without loss of generality, assume that V_2 lies on the right of V_1 and $BS \in V_2$, we have the following two cases.

Case 1. There is not any relay CH in V_2 such that v_1 reaches BS via it. There must be a cluster head $v_1 \in V_1$ which is able to communicate with $v_2 = BS \in V_2$.

Case 2. There exists at least a relay CH $v_2 \in V_2$ such that v_1 reaches BS via v_2 . Then v_1 communicates with it firstly.

So both cases there must be at least a directed edge (v_1, v_2) connects these two connected components, which contradicts with the initial assumption that a cluster head in one component cannot communicate with the one in the other component. Therefore, V_1 and V_2 are connected.

Condition 2 ensures that the route moving toward BS from source CHs via relay CHs. Without loss of generality, we suppose that there is a directed circle $C = \langle v_i \rightarrow v_j \rightarrow \dots \rightarrow v_k \rightarrow v_i \rangle$ in the digraph, so we have $\text{dist}(v_i, v_i) = 0$. Due to condition (2), we have $\text{dist}(v_i, v_j) > 0, \dots, \text{dist}(v_k, v_i) > 0$. The total distance on the directed circle C is $\text{dist}(v_i, v_j) + \dots + \text{dist}(v_k, v_i) > 0$. Since $\text{dist}(v_i, v_j) + \dots + \text{dist}(v_k, v_i) = \text{dist}(v_i, v_i)$, thus we have $\text{dist}(v_i, v_i) > 0$. This is a contradiction.

Therefore, the theorem is proved. \square

4.2.2. Next-Hop Neighbor Choosing. The reachability relation in a DAG forms a partial order, with which a routing path can be constructed, that is, the DAG provides a multipath routing mechanism. With Theorem 6, any CH can transmit its data along relay CHs to BS. Any node or link fails, based on the candidate nodes and links in DAG another one will be chosen. According to different requirements, the selection of the next hop node gives priority to the minimum energy strategy or the maximum residual energy strategy or any other factors. The former forwards packets along the minimum energy path to BS; while the latter forces packets to move toward the BS considering more residual energy of the node on routing path. The decision is made depending on the connected network model and the information gathered in clustering stage.

As an example, the distributed algorithm looking for next-hop neighbor for any cluster head $c_i.CH$ is presented in Algorithm 3. Each CH chooses its next-hop neighbor independently according to the distance to BS. Initially, $c_i.CH$ chooses a neighbor, which is the nearest to BS within its communication range. In lines 5–9, if more than two neighbors have the same distance, then algorithm selects the one with more residual energy for the sake of balancing energy once again. When the cluster head cannot choose its neighbor any more, the network becomes partitioned. From the view of entire network, a spanning tree with root BS which has minimum hop counts to BS is received.

Figure 2 illustrates the communication path selection process. Each cluster head has $2J$ initial energy, and the communication cost is denoted by edge weight, the residual energy is represented by vertex weight. The DAG shows the available communication paths. For sensor node v_4 , if the minimum energy first, it selects v_3, v_6 as the next hop node

```

1. while network is not partitioned;
2.   for any  $c_j.CH \in c_i.N_{FF,CH}$ 
3.      $c_k.CH = \min(\text{dist}(c_j.N_{CH}, BS))$ ;
4.      $c_i.CH.nextHop \leftarrow c_k.CH$ ;
5.     if  $\text{dist}(c_j.CH, BS) = \text{dist}(c_k.CH, BS)$  then
6.       if  $(c_i.CH.E_{RE} > c_k.CH.E_{RE})$  then
7.          $c_i.CH.nextHop \leftarrow c_j.CH$ ;
8.       else
9.          $c_i.CH.nextHop \leftarrow c_k.CH$ ;
10.  end while

```

ALGORITHM 3: Distributed multihop routing selection for cluster head $c_i.CH$.

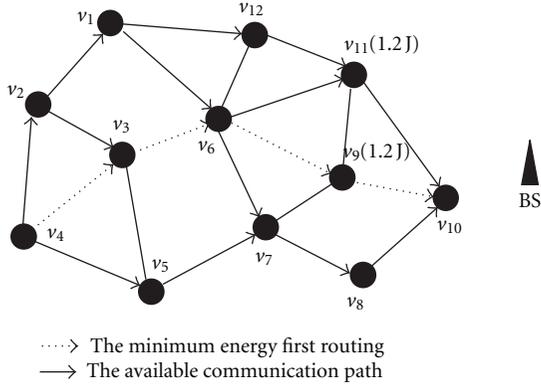


FIGURE 2: Intercluster multihop paths selection among DAG.

one after another. Note that when node v_6 looking for its next hop node and find that there are two nodes v_9 and v_{11} with the same distance to BS. At this moment, it selects node v_9 which has more residual energy. On the contrary, if the highest residual first, algorithm not necessarily selects the paths which approach BS more quickly but have more residual energy. Finally, from the global view, algorithm outputs a relative optimal communication spanning tree about the graph G whose root is BS.

4.2.3. Route Maintenance. As stated previously, in the connected network each CH always chooses the neighboring CH with the minimum distance to BS as the next-hop routing node independently. The rest neighbors are maintained in its routing table in order of their distance to BS. If the optimal neighbor is unavailable due to node or link failure, then the node chooses a suboptimal one in its routing table, thus providing a robust routing. In addition, this multipath routing selection mechanism to some extent guarantees that the malicious attacker in network cannot obtain the communication path by listening in the signal simply. After each CH determined its next-hop neighbor, CHs are ready to start transmitting sensing data.

In intercluster multihop routing stage, Algorithm 3 finds a routing path, which approaches BS more quickly among all the available paths. While in clustering stage, candidate CH competition takes into residual energy into account. In round based protocol, both stages progress alternatively and

TABLE 2: Simulation parameters.

Parameter	Value
Sensor field	100 m \times 100 m
BS location	(170,50)
Number of nodes	100
Initial energy of nodes	2 J
Data packet size	500 bytes
E_{elec}	50 nJ/bit
ϵ_{fs}	10 pJ/(bit \cdot m ²)
ϵ_{mp}	0.0013 pJ/(bit \cdot m ⁴)
R	30 m
d_0	86 m

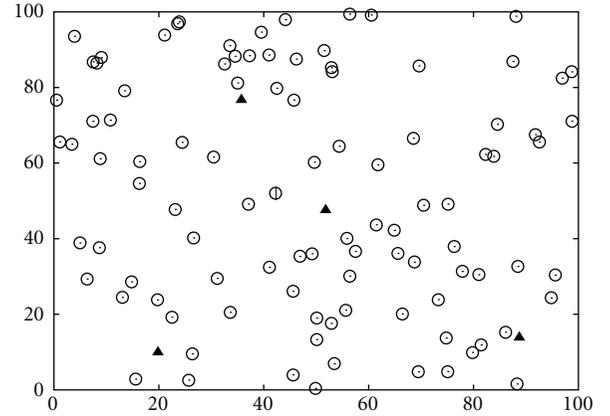


FIGURE 3: Wireless sensor network deployment. Hollow circles denote fault-free nodes, and triangles show faulty nodes.

thus provide network balanced energy consumption among all sensor nodes which meets requirement (5).

5. Simulations

5.1. Simulation Settings. In this section, we evaluate the performance of FDCM with simulations. Because LEACH [5], HEED [6], and EEUC [8] are the most similar clustering protocols, we use them for comparisons. For fault diagnosis-based clustering, CRFDA [20] is compared. One hundred of sensor nodes are randomly distributed over the region of 100 m \times 100 m as showed in Figure 3. The number of candidate CHs is set as 20% of the total nodes. The BS is located far away from the region, at point (50, 175). The simulation parameters are listed in Table 2.

In our paper we make the following assumptions. (1) Nodes that are detected as faulty will turn into dead mode, that is, they will no longer generate information and consume energy. (2) During the network lifetime, nodes may be faulty at any time. The data sending by soft-faulty nodes is invalid. (3) Sensor nodes have idealized sensing capabilities. Ideal MAC layer conditions are assumed, that is, perfect transmission of data on a node-to-node wireless link. (4) In diagnosis process, we use the uniform rules to generate test Task T_i , and ignoring the energy consumed by its implementation.

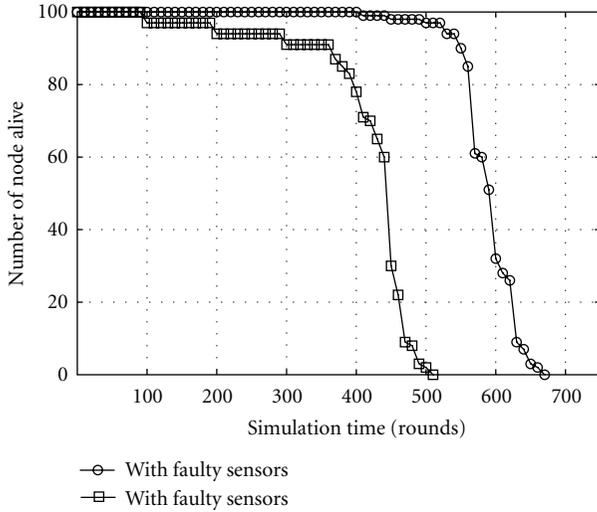


FIGURE 4: The number of fault-free nodes changes with time.

5.2. Simulation Results. Since our protocol is round based, we assume that sensor network randomly generates certain number of faulty nodes when it is running in a certain round. In ideal situation, we expect that the number of the fault-free nodes in network decreases correspondingly comparing with previous round. In our simulation, assume that 3 faulty nodes present in 100th, 200th, 300th round, respectively, and then the result of algorithms execution is shown in Figure 4. One of the curves indicates the result when there are faulty nodes in the network; while the other is fault-free case. From the figure, we can see that the number of fault-free nodes is reduced with faulty nodes arise when the network is running in each round. This means FDCM can correctly detect the faulty nodes in the network. Besides, by comparing, the existence of faulty nodes decreases the network lifetime observably.

Communication complexity is an important measurement for fault diagnosis efficiency. That is one of reasons why fault diagnosis mechanisms with high complexity in traditional MANETs are unrealistic using for WSNs directly. By means of introducing fault diagnosis based on the comparison model in the process of the candidate CHs running for final CHs, FDCM eliminates the chance of faulty nodes to participate in the election; Then, within each cluster only $O(N)$ message complexity needed to complete the fault diagnosis of the cluster members. The isolation of diagnosis boundary avoids large-scale message diffusion throughout the entire network, which made the message complexity reduced significantly. Figure 5 confirmed this by comparing the message complexity of different protocols. It shows that the message complexity of FDCM increases nearly linearly. When faulty nodes come into being, if they do not be diagnosed and excluded from the network, they will consume additional energy of other fault-free CHs. Then, it will shorten the lifetime of the entire network. In simulation, we monitor the number of nodes alive changing with time (round). We find that the network lifetime of FDCM may be influenced by various factors in different situations. The

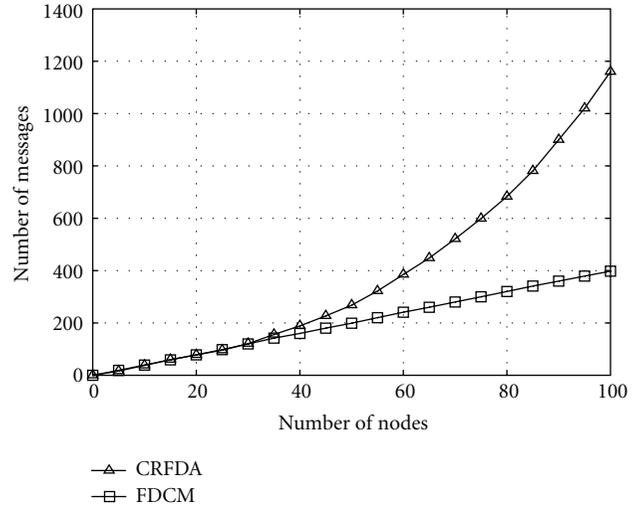


FIGURE 5: Comparison of communication complexity.

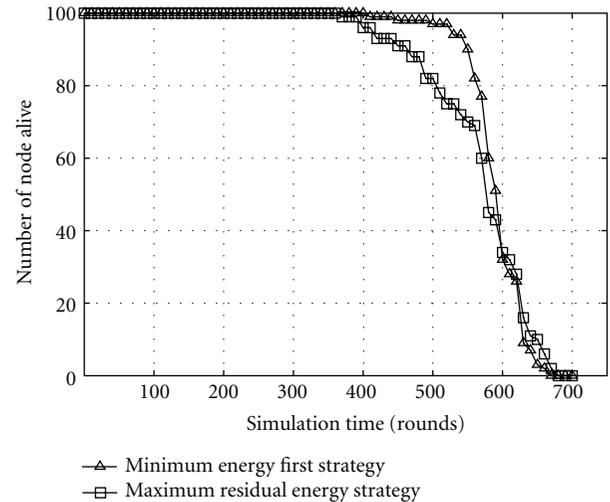


FIGURE 6: Comparison of network lifetime with different routing modes.

evaluation of the lifetime comparison between the cost-aware and the energy-aware is shown in Figure 6.

In contrast with similar clustering protocols, we run LEACH, HEED, and EEUC to compare their performance in network lifetime. As shown in Figure 7, FDCM and EEUC perform far better than LEACH and HEED in prolonging network lifetime attributed to the consideration of energy conservation. In FDCM, a certain amount of energy is spent by the nodes involving in fault diagnosis; however, this eliminates additional energy consumption caused by the faulty nodes. More importantly, the existence of fault diagnosis ensures the correctness of the information collected.

6. Conclusions

In wireless sensor network, it is of great importance for fault diagnosis to ensure the gathering information accuracy and

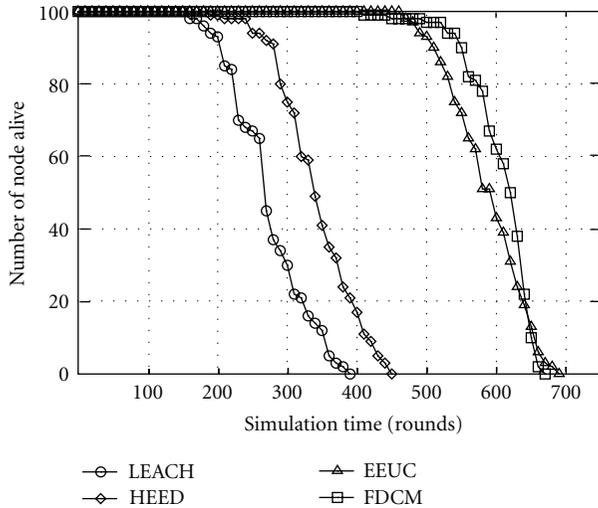


FIGURE 7: Comparison of Network lifetime with different protocols.

reduce energy additionally consumed by faulty nodes, for the deployment of a large number of sensor nodes in hostile environment. For the inherent characteristics of sensor networks, this paper analyzes the issues and challenges of comparison based-fault diagnosis model for wireless sensor networks and gives the relevant design requirements.

As a complete data collection protocol, the proposed protocol mainly consists of fault diagnosis-based clustering and multipath routing. During clustering stage, a fault diagnosis approach based on comparison model is introduced. Fault diagnosis of the network nodes consists of two phases. At first, it eliminates the faulty candidate CHs to participate in the final CH competition in the process of clustering. Secondly, after the clustering is finished, the fault diagnosis is done based on special comparison model between CH and CMs. CH sends a test request message to its members and according to their responses to determine the fault status of these nodes, failure nodes are ordered to turn dead. These two phases together complete the diagnosis of all faulty nodes in the network.

In Multipath routing stage, communication characteristics impose certain conditions, which map the original abstract communication graph into the DAG. The new graph determines the feasible multipath communication path of any node to transfer data to the BS. In particular, we give an algorithm greedy select next hop neighbor which has the minimum distance to BS. When multiple nodes are optional, then the node with the highest residual energy is preferential. If any node in the routing path fails, then select an available path in the DAG depending on the highest residual energy or have the minimum distance to BS until the data transfer to the BS. Note that the transmitted data including node fault status, which can be used in the next round as a basis of cluster first election, that is, faulty nodes will lose the possibility of acting as the candidate CH.

For future work, we will consider two new directions. First, we intend to improve our algorithm effectiveness and obtain better performance, such as more accurate

diagnosis and lower response time. Second, on condition that acceptable message complexity, we will study the possibility of new diagnosis approach which is appropriate for dynamic topology.

Acknowledgments

This work is supported by National Natural Science Foundation of China (nos. 61070169, 61170021), Natural Science Foundation of Jiangsu Province (no. BK2011376), Specialized Research Foundation for the Doctoral Program of Higher Education of China (no. 20103201110018), Application Foundation Research of Suzhou of China (nos. SYG201034, SYG201240), and sponsored by Qing Lan Project.

References

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.
- [2] J. Lian, K. Naik, and G. B. Agnew, "Data capacity improvement of wireless sensor networks using non-uniform sensor distribution," *International Journal of Distributed Sensor Networks*, vol. 2, no. 2, pp. 121–145, 2006.
- [3] H. M. Ammari and S. K. Das, "Promoting heterogeneity, mobility, and energy-aware Voronoi diagram in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 7, pp. 995–1008, 2008.
- [4] H. Zhang and H. Shen, "Balancing energy consumption to maximize network lifetime in data-gathering sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 10, pp. 1526–1539, 2009.
- [5] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.
- [6] 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.
- [7] S. Soro and W. B. Heinzelman, "Prolonging the lifetime of wireless sensor networks via unequal clustering," in *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS '05)*, April 2005.
- [8] G. Chen, C. Li, M. Ye, and J. Wu, "An unequal cluster-based routing protocol in wireless sensor networks," *Wireless Networks*, vol. 15, no. 2, pp. 193–207, 2009.
- [9] J. Yu, Y. Qi, G. Wang, and X. Gu, "An energy-aware distributed unequal clustering protocol for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2011, Article ID 202145, 8 pages, 2011.
- [10] V. Mhatre and C. Rosenberg, "Design guidelines for wireless sensor networks: communication, clustering and aggregation," *Ad Hoc Networks*, vol. 2, no. 1, pp. 45–63, 2004.
- [11] Z. Ye, S. V. Krishnamurthy, and S. K. Tripathi, "A framework for reliable routing in mobile ad hoc networks," in *Proceedings of the 22nd Annual Joint Conference on the IEEE Computer and Communications Societies (IEEE INFOCOM)*, pp. 270–280, April 2003.
- [12] J. Tsai and T. Moors, "A review of multipath routing protocols: from wireless ad hoc to mesh networks," in *Proceedings of the*

ACoRN Early Career Researcher Workshop on Wireless Multihop Networking, pp. 17–22, July 2006.

- [13] T. Elhourani and S. Ramasubramanian, “Independent directed acyclic graphs for resilient multipath routing,” *IEEE/ACM Transactions on Networking*, vol. 20, no. 1, pp. 153–162, 2012.
- [14] M. Malek, “A comparison connection assignment for diagnosis of multiprocessor systems,” in *Proceedings of the 7th International Symposium on Computer Architecture*, pp. 31–36, 1980.
- [15] A. Sengupta and A. T. Dahbura, “On self-diagnosable multiprocessor systems: diagnosis by the comparison approach,” *IEEE Transactions on Computers*, vol. 41, no. 11, pp. 1386–1395, 1992.
- [16] S. Chessa and P. Santi, “Comparison-based system-level fault diagnosis in ad hoc networks,” in *Proceedings of the 20th IEEE Symposium on Reliable Distributed Systems (SRDS '01)*, pp. 257–266, New Orleans, La, USA, October 2001.
- [17] M. Elhadef, A. Boukerche, and H. Elkadiki, “Self-diagnosing wireless mesh and ad-hoc networks using an adaptable comparison-based approach,” in *Proceedings of the 2nd International Conference on Availability, Reliability and Security (ARES '07)*, pp. 983–990, April 2007.
- [18] M. Elhadef, A. Boukerche, and H. Elkadiki, “A distributed fault identification protocol for wireless and mobile ad hoc networks,” *Journal of Parallel and Distributed Computing*, vol. 68, no. 3, pp. 321–335, 2008.
- [19] J. Chen, S. Kher, and A. Somani, “Distributed fault detection of wireless sensor networks,” in *Proceedings of the Workshop on Dependability Issues in Wireless Ad Hoc Networks and Sensor Networks (DIWANS '06)*, pp. 65–71, September 2006.
- [20] W. Wang, B. Wang, Z. Liu, and L. Guo, “A cluster-based real-time fault diagnosis aggregation algorithm for wireless sensor networks,” *Information Technology Journal*, vol. 10, no. 1, pp. 80–88, 2011.
- [21] D. B. West, *Introduction to Graph Theory*, Prentice Hall, New York, NY, USA, 2001.
- [22] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless micro-sensor networks,” in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences (HICSS-33)*, p. 10, January 2000.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

