Journal of Systems Architecture 59 (2013) 260–270

Contents lists available at SciVerse ScienceDirect

Journal of Systems Architecture

journal homepage: www.elsevier.com/locate/sysarc

Informer homed routing fault tolerance mechanism for wireless sensor networks

Meikang Qiu [a], Zhong Ming [b,*], Jiayin Li [a], Jianning Liu [a], Gang Quan [c], Yongxin Zhu [d]

[a] Department of Electrical and Computer Engineering, University of Kentucky, Lexington, KY 40506, USA
[b] College of Computer Science and Software, Shenzhen University, Shenzhen, Guangdong 518060, China
[c] Department of Electrical and Computer Engineering, Florida International University, Miami, FL 33174, USA
[d] School of Microelectronics, Shanghai Jiao Tong University, Shanghai 200240, China

ARTICLE INFO

Article history:
Received 1 February 2012
Received in revised form 26 October 2012
Accepted 29 December 2012
Available online 18 January 2013

Keywords:
Fault tolerance
WSN
Routing algorithm
Energy saving
Robustness

ABSTRACT

Sensors in a wireless sensor network (WSN) are prone to failure, due to the energy depletion, hardware failures, etc. Fault tolerance is one of the critical issues in WSNs. The existing fault tolerance mechanisms either consume significant extra energy to detect and recover from the failures or need to use additional hardware and software resource. In this paper, we propose a novel energy-aware fault tolerance mechanism for WSN, called *Informer Homed Routing* (IHR). In our IHR, *non cluster head* (NCH) nodes select a limited number of targets in the data transmission. Therefore it consumes less energy. Our experimental results show that our proposed protocol can significantly reduce energy consumption, compared to two existing protocols: *Low-Energy Adaptive Clustering Hierarchy* (LEACH) and *Dual Homed Routing* (DHR).

1. Introduction

A wireless sensor network (WSN) consists of a large number of sensor nodes that collect meaningful environmental information and send them to a central repository. The evolution of communication technologies has motivated applications of WSNs and the development of wireless communications. Embedded system technologies make it possible to develop low-cost, low-power, and small-sized wireless sensor nodes [1–5]. WSNs have infiltrated every aspect of our daily life, such as home automation monitoring [6], medical monitoring [7], vehicle anti-theft monitoring [8], weather monitoring [9], building structures monitoring, and industrial plant monitoring [10–12].

Since WSNs become more and more popular, the quality of service provided by a WSN in the aspects of information integrity, data correctness and transmission in a timely manner have drawn more and more attention to researchers and system designers. However, nodes in WSNs are prone to failure due to the energy depletion, hardware failure, communication link errors, malicious attacks, etc. Therefore, fault tolerance is one of the most critical issues in WSNs. Fault tolerance is the ability of a system to deliver a desired level of functionality in the presence of faults [13]. Many fault tolerant mechanisms have been proposed and studied. However, these mechanisms either consume lots of extra energy to detect and recover failures or even need additional hardware and software resources [14,15].

In this paper, we design a fault tolerance mechanism for WSNs called *Informer Homed Routing* (IHR). This algorithm advances the existing fault tolerance mechnism, such as the *Dual Homed Routing* (DHR) [16] mechanism in the aspect that it reduces energy consumption, prolongs the lifetime of a WSN, and transmits more information in the situation when node faults happen. In our algorithm, instead of sending data information to the primary cluster head and making backup of cluster head simultaneously, the collector node only sends data when it founds the primary cluster head fails. This feature of IHR leads to less energy consumption compared to the DHR. In addition, it can still transmit information when cluster head faults happen.

We compare the effectiveness of IHR with DHR and *Low-Energy Adaptive Clustering Hierarchy* (LEACH) [1] mechanism in the aspect of energy consuming, throughput, and data loss rate. To achieve this task, we develop a customizable WSN simulator tool to support all of LEACH, DHR, and IHR and study the behavior of all these WSN fault tolerance mechanisms. We add configuration interface and user interface to this tool for users. The simulation tool can also be used for further WSN fault tolerance mechanism design and analysis.

The contributions of this paper can be summarized as the following:

* Corresponding author.
E-mail addresses: mqiu@engr.uky.edu (M. Qiu), mingz@szu.edu.cn (Z. Ming), jli6@engr.uky.edu (J. Li), jianning.liu@uky.edu (J. Liu), gang.quan@fiu.edu (G. Quan), zhuyongxin@sjtu.edu.cn (Y. Zhu).

1383-7621/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.sysarc.2012.12.003

- Based on the principles we summarized, we designed and implemented a novel mechanism called IHR derived from DHR mechanism used for WSN fault tolerant applications.
- We developed a WSN simulator to support all of LEACH, DHR, and IHR mechanisms and inject cluster head faults to observe the robustness of these mechanisms.

The organization of this paper is as follows. We describe related work and background in Section 2 and 3. In Section 4, we introduce communication procedures of the IHR mechanism and analyze the performance of it. The new features in our simulation tool are shown in Section 5. The evaluation results of IHR mechanism are presented in Section 6. Conclusions are provided in Section 7.

2. Related work

In this section, we present the related work of fault tolerance mechanism in WSN and discuss advantages and disadvantages of different WSN fault tolerance protocols. We also introduce the existing simulation tool. For the ease of reading, we list the abbreviations in Table 1.

There have been studies on fault tolerance in wireless sensor networks [17,18]. Most of current fault tolerance protocols introduce redundancy. For example, one of the techniques to tolerate wireless link failure is retransmission, which introduces extra traffic in the network and causes extra energy consumption. Receivers need to confirm the receipt of messages, which require extra energy compared to the scheme without retransmission is not required. when the number of retransmissions is large, the delay is significant and causes out-of-date information and meaningfulness lose.

A typical example is the fault tolerance mechanism used in Zigbee standard [19]. The IEEE 802.15.4/ Zigbee standard is a low-cost, low-power, wireless sensor networking stack that has been considered as a promising technology for WSNs. First, the low cost allows the technology to be widely deployed in wireless control and monitoring applications. Second, the low power-usage promises longer life with smaller batteries. However, the Zigbee protocol currently lacks of efficient fault tolerance mechanisms to support reliability for real-time applications.

IEEE 802.15.4/ Zigbee supports a native fault-tolerance mechanism called as the orphaned device realignment [20]. This recovery/repair procedure is activated when there are repeated communication failures in the request for data transmission (e.g. data frames sent without receiving the requested acknowledgment) between the device and its parent or when the device loses synchronization with its parent. When a device is found orphaned, a realignment or a channel re-scan process will be invoked.

A scheme in [21], is provided to tolerate random node failures, where sensor nodes are assigned two statuses, active and sentry, to save power. In this scheme, they introduced redundant nodes, sentry nodes, which periodically wake up as scheduled to detect whether the active node has failed or not. If any sentry node finds the active node has failed, it becomes active. Obviously, this scheme incurs extra traffic between sentry and active nodes due to the frequently scheduled detection. Besides, sentry nodes never collect data that will waste resources of nodes.

Table 1
The abbreviation.

Name	Meaning
LEACH	Low-Energy Adaptive Clustering Hierarchy
DHR	Dual Homed Routing
IHR	Informer Homed Routing
CH	Cluster Head
PCH	Primary Cluster Head
BCH	Backup Cluster Head
NCH	Non Cluster Head
BS	Base Station

In [1], the authors proposed a clustering hierarchy, LEACH, for micro-sensor networks. This is an effective way to avoid out-of-power node failures. They developed algorithms for adapting clusters and rotating cluster head positions to evenly distribute the energy load among all nodes. By maintaining the fairness in the whole network, the possibility of some sensors being out-of-power long before others has been reduced. Thus extend lifetime of the system. However, this method can only postpone most out-of-power failures, but cannot reduce the data loss when node failures happen.

In some other fault tolerance mechanism, redundant paths are used to transmit data to destination, such as 1 + 1 DHR in [16]. In DHR protocol, each cluster has two dedicated cluster-heads and the data is sent to both the primary cluster-head (primary home) and the backup cluster-head (backup home) simultaneously. The cluster heads aggregate data and relay data to the *base station* (BS) independently. Although this method can improve system's dependability, transmitting data twice is energy consuming.

3. Model and background

3.1. The network topology model

The network topology model used in our simulation tool can be depicted in Fig. 1. The network has only one sink node. Without lost of generality, we refer it as BS in the rest of the paper. This sensor network has N sensor nodes uniformly deployed over a square area. *Cluster heads* (CHs) are selected with a defined probability, which equals the ratio of the expected number of CHs to N. CHs not only forward data, but also sense the environmental information and aggregate their own data with the information collected from their children. Other sensor nodes can be associated with CHs. These sensor nodes are called as *Non Cluster Heads* (NCH). NCHs can communicate with BS through single or multi-hops routing. Bi-directional symmetrical links are assumed. The maximum number of hops in this network topology model is two.

3.2. The energy model

Our tool uses the same radio energy model as that in [1]. The energy dissipation $E_T(k,d)$ of transmitting k-bit data between two nodes separated by a distance of d meters is given as follows:

$$E_T(k,d) = \begin{cases} k(E_{elec} + \varepsilon_{FS} * d^2) & (d < d_0) \\ k(E_{elec} + \varepsilon_{MP} * d^4) & (d > d_0) \end{cases} \quad (1)$$

where $d_0 = \sqrt{\varepsilon_{FS}/\varepsilon_{MP}}, E_{elec}$ denotes electronic energy, ε_{FS} and ε_{MP} denote transmit amplifier parameters corresponding to the free-space and the two-ray models. The energy dissipation of the receiver is

$$E_R(k) = k * E_{elec} \quad (2)$$

Also, the energy dissipation of fusing k-bits data is

$$E_F(k) = k * E_{df} \quad (3)$$

where E_{df} is the energy dissipation of fusing one bit data. The parameters used in this paper are given below: E_{df} = 5 $nJ/bit, \varepsilon_{FS}$ = 10 $pJ/bit/m^2, E_{elec}$ = 50 nJ/bit, ε_{MP} = 0.004 $pJ/bit/m^4$, and $d_{toBS} > d_0$.

3.3. Fault model

In order to evaluate whether a mechanism used for a WSN is fault tolerant or not, we need to observe how the network reacts when faults happen. The existing tools do not support any fault

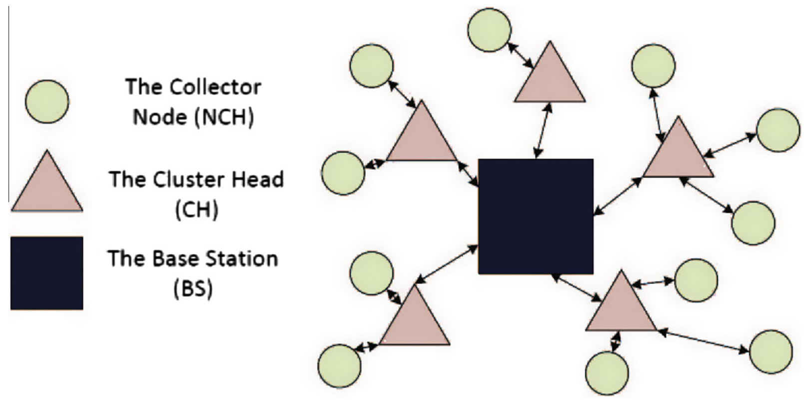

Fig. 1. The network topology model.

injection to a network. The only reason a fault happens in the existing tools is battery depletion. However, there are several other reasons that cause a WSN fails. There are different ways to classify these faults into different categories based on different criteria. According to the layer in the network architecture where the fault happens, the faults could be divided into hardware layer faults, software layer faults, network communication layer faults, and application layer faults [22]. According to the time the faults last, the faults could be divided into ephemeral faults, intermittent faults, and permanent faults.

In our paper, we tackle hardware faults happen at cluster head. The hardware faults may be caused by battery depletion, hardware deterioration, transmitter failure, malicious human behavior, etc. The impact of this kind of fault is severe, because when a cluster head fails, its children are cut off from the cluster tree, resulting in the lost of communication with the outside. This will significantly reduce the availability of the sensor network.

To evaluate the robustness and behavior of WSN when faults happen, we add a fault model in our simulation tool to inject CH faults into WSN network. The fault generating scheme is as the following:

For every cluster head, if the time arrival of the k_{th} fault is T_k, then the inter-arrival times are defined as follows:

$$X_1 = T_1, X_k = T_k - T_{k-1}; \quad for\ k = 2, 3, \ldots \tag{4}$$

We suppose X_i is independent, identical distributed random variable, and belongs to the classical exponential distribution with rate parameter λ:

$$f_{T_i}(t) = \lambda * e^{-\lambda * t}, \quad t > 0 \tag{5}$$

The inter-arrival time stream, $X_1, X_2, X_3, \ldots$, actually forms a Poisson process. $E(X_i)$ equals $1/\lambda$, which is the expected value of X_i. Apparently, the larger the λ, the less frequent the faults happen.

To generate the Poisson process, we set the time interval between two consecutive faults at $-1/\lambda lnx$, where x is uniformly distributed over 0 and 1.

3.4. LEACH and DHR

The LEACH mechanism was invented in 2002 by Heinzelman [1]. This mechanism was aimed to prolong network lifetime by changing clusters distribution to keep the fairness of energy consumption in the whole network. This mechanism reduced the possibility of some sensors being out-of-power long before others. This method has been proven to be effective. A lot of advanced fault tolerance mechanisms are designed and developed based on this mechanism architecture. In the LEACH mechanism, each NCH choose the nearest CH to join. And there are two phases in each data transmision: sending data from NCH to their respective CHs, and sending data from CHs to BS.

Note that the data transmission phase of LEACH mechanism includes two stages: one is from NCHs to their corresponding CHs, another stage is from CHs to BS. To simulate cluster heads faults, we inject cluster heads faults in this phase by using a Poisson distribution random number generator.

The DHR fault tolerant routing mechanism used for a WSN is derived from the DHR mechanism used for Internet. The basic idea is each collector node has two CHs associate with it, one is PCH, and the other is BCH. The collector sensor node sends data information to both PCH and BCH at the same time. If PCH fails, the data information can still be transmitted through the BCH. The problem of this method is that it wastes energy. In 2008, researchers investigated and reported the performance of this mechanism used in a WSN [16]. Their results showed that 1 + 1 DHR minimizes the average loss probability, until the time of cluster head failures. However, due to higher energy expenditure, it tends to decrease average throughput, average end-to-end delay, and network lifetime of a WSN.

In our paper, the DHR mechanism is implemented based on the LEACH mechanism and implement the mechanism round by round. In each round, the algorithm goes through the LEACH mechanism. We find a backup cluster head for each collector sensor node. In the DHR mechanism, each CH will broadcast its status to others. Every NCH will choose it nearest CH, while each CH will choose the NCH with the most energy left as its BCH. This information will be forwarded to this CH's child nodes. There are also two phases in data transmission: sending data from the NCH to the corresponding CH and BCH both, and sending data from CHs and BCHs to BS.

4. IHR fault tolerance protocol for WSN

In this section, we first provide an overview of the IHR mechanism. Then describe how the IHR algorithm is implemented. At last we will give the theoretical analysis of IHR mechanism to show how it can improve LEACH and DHR in the aspects of dependability and energy dissipation.

4.1. Our IHR mechanism

To design a good fault tolerant mechanism for a WSN, we need to keep low data loss rate, maintain minimum energy cost, while guarantee high throughput and short latency to achieve the goal of prolonging network's lifetime.

After studying the LEACH and DHR mechanisms, we identify the spot that consumes energy in DHR (a mirror backup is very resources consuming) and the spot that is less robust in LEACH.

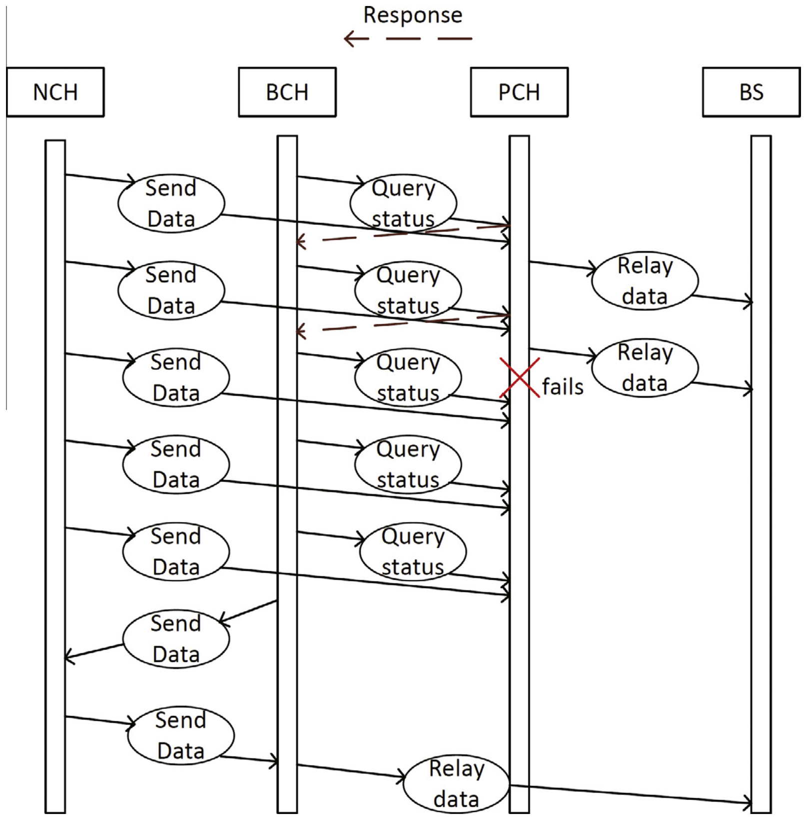

Fig. 2. The IHR mechanism communication procedure.

LEACH only prolongs network lifetime by making different nodes to work as a CH, but it lacks a mechanism to monitor the aliveness of CH.

Based on the above findings, we designed our own mechanism IHR, which is derived from DHR, the sensors will dissipate energy. The DHR sends data to PCH and BCH at the same time, half of the energy cost is wasted. In our IHR mechanism, instead of sending data to PCH and BCH at the same time, the collector nodes only send data to PCH in the regular runtime. In each data transmission round, the BCH will check the aliveness of PCH based on the beacon message it receives from PCH. After three rounds, if it cannot receive any respond message from PCH, it will declare that the PCH has failed and inform NCHs to transmit data to BCH. In this design, we can achieve at least the same dependability as DHR. While the energy cost decreased, it can improve the network throughput and prolong network lifetime and result in better data loss rate. Because the beacon packet size is much smaller than the data packet size, the communication overhead is much less than the data communication energy consumption in DHR.

4.2. IHR Algorithm

In our paper, the IHR mechanism is also implemented based on LEACH mechanism. We implement the mechanism round by round. In each round, the algorithm goes through the LEACH mechanism, finding a BCH for each NCH. The difference between DHR and IHR is that the NCH nodes only send data to one of the CHs at a time. The following pseudo-code in Algorithm 4.1 shows how the IHR algorithm is implemented. We use the same fault model as that in LEACH to inject CH faults into network.

One of the communication scenario is shown in Fig. 2. From the figure we can see that the NCH sends data information to only one of its CHs, either PCH or NCH. Thus the sensor nodes save a lot of energy dissipation. As long as one of its CHs is alive, NCH can find a path to deliver the message it has collected to BS.

Algorithm 4.1 IHR Algorithm

Input: Network setting configuration, number of rounds M, number of times for data transmission N
Output: A IHR routing scheme
1: Initialize the network topology based on network setting configuration
2: **for** $rounds\_index = 1 \rightarrow M$ **do**
3: Select CHs for Clusters.
4: Each CH will broadcast its CH status to other nodes. All other nodes keep their receivers ON during this phase.
5: The NCHs will choose the nearest CH to join.
6: The CH will choose the node with the most energy left among all the NCH as the BCH.
7: After deciding upon the backup clusters, each CH will transmit this information to its children nodes. All other nodes keep their receivers ON during this phase.
8: **for** $times\_index = 1 \rightarrow N$ **do**
9: BCH sends query message to its corresponding PCH to check the aliveness of PCH.
10: Increment BCH's query counter.
11: PCH receives message from BCH.
12: **if** PCH is alive **then**
13: PCH sends respond message to confirm its aliveness to BCH.
14: Upon receiving the respond message from PCH, BCH decrements its query counter.
15: **end if**

(*continued on next page*)

Algorithm 4.1 IHR Algorithm

```
    if query counter > 3 then
        BCH decides the PCH has failed and sends inform message to its children to make its NCH to send data to itself next time.
    end if
    if NCH receives the informer message from BCH then
        NCH sends data to its BCH.
    else
        NCH sends data to its PCH.
    end if
  end for
end for
```

4.3. Performance analysis

In this section we give the theoretical analysis of IHR mechanism to show how it can improve LEACH and DHR in the aspects of energy dissipation and network dependability.

4.3.1. Energy dissipation analysis

The analysis of energy dissipation is based on the energy model introduced in Section 2. We use d_{X-Y} to denote the transmission distance from node X to node Y. Therefore, d_{N_i-PCH} means the distance between node i to node PCH, d_{PCH-BS} means the distance between PCH to BS, etc. Given m is the number of sensor nodes in a cluster, k is the size of data message, b is the size of beacon message, the energy dissipation of IHR for each cluster in one data transmission round can be described by the following equations:

When PCH is alive:

$$E_{IHR} = (m-1)(E_T(k, d_{N_i-PCH}) + E_R(k) + E_F(k)) + E_T(k, d_{PCH-BS}) + 2(E_T(b, d_{BCH-PCH}) + E_R(b)) \tag{6}$$

When PCH is dead:

$$E_{IHR} = (m-2)(E_T(k, d_{N_i-BCH}) + E_R(k) + E_F(k)) + E_T(k, d_{BCH-BS}) \tag{7}$$

Communication cost:

$$E_{Cost} = 3 * (E_T(b, d_{BCH-PCH}) + E_R(b)) + (m-2)(E_T(b, d_{BCH-N_i}) + E_R(b)) \tag{8}$$

In Eq. (6), the first line represents the energy dissipation between NCHs and PCH, i.e., the communication process includes NCHs sending data to PCH, PCH receiving data, and fusing the data into one data package. The second line is the energy dissipation between PCH and BS. The communication cost generated by the faults detection and recovery process in IHR mechanism is described in Eq. (8). If the ratio of k/b is very large, this communication cost can be omitted. The Eq. (7) is the energy dissipation after NCH's transmitting path switching to BCH. There is no more communication cost occurred in this phase.

By using the same energy model, we can deduct the energy dissipation for LEACH and DHR mechanisms are as Eq. (9) and (10).

$$E_{LEACH} = (m-1)(E_T(k, d_{N_i-CH}) + E_R(k) + E_F(k)) + E_T(k, d_{CH-BS}) \tag{9}$$

$$E_{DHR} = (m-2)(E_T(k, d_{N_i-PCH}) + E_R(k) + E_F(k)) + E_T(k, d_{PCH-BS}) + (m-2)(E_T(k, d_{N_i-BCH}) + E_R(k) + E_F(k)) + E_T(k, d_{BCH-BS}) \tag{10}$$

Comparing among all the energy dissipation equations, let's assume k/b is very large, m is very large, $d_{PCH-BS} = d_{BCH-BS} = d_{CH-BS}$ and for every i in a cluster $d_{N_i-PCH} = d_{N_i-BCH} = d_{N_i-CH}$, we can conclude that $E_{IHR} \approx E_{LEACH} = 1/2 E_{DHR}$.

The number of selected CHs do impact the battery operation time of the CHs. In extreme cases where there is few CHs selected, the battery of each CH may run out quickly. Therefore, the selection of the number of CHs is a tradeoff between total energy consumption and the lifetime of the selected CHs, while a large number of CHs certainly bring large energy consumption overhead and a small number of CHs limit the lifetime of batteries of CHs. However, since this tradeoff exists in both DHR and IHR, our proposed method still has the smaller energy consumption than that of DHR.

4.3.2. Dependability analysis

If a WSN has n sensor nodes, c of them are CHs, considering CHs' faults only, if each CH's fault rate is ε, we can define the dependability of this network as $1 - \varepsilon^c$. This means if there is no CH fault, the network is 100% dependable. Otherwise, the possibility of the network to be dependable is $1 - \varepsilon^c$. If the dependability is 0, it means the network is dead. The definition is under the assumption that as long as one of the nodes in a cluster can work fine, we can get satisfactory information around that area.

Based on the above definition, we can define the dependability of a network that uses IHR mechanism as the following:

$$D_{IHR} = \sum_{i=0}^{R-1} (1 - \varepsilon^{2c_{iIHR}})/R \tag{11}$$

In Eq. (11), R is the number of rounds during the whole network time, c_{iIHR} is the number of CHs in a cluster at round i, and ε is the probability a CH in this network fails. The reason we calculate ε^2 is that because in IHR, we can find a backup path in each cluster, as shown in Eqs. (12) and (13).

In the same way, we can get the dependability of network which use LEACH and DHR mechanisms

$$D_{LEACH} = \sum_{i=0}^{R-1} (1 - \varepsilon^{c_{iLEACH}})/R \tag{12}$$

$$D_{DHR} = \sum_{i=0}^{R-1} (1 - \varepsilon^{2c_{iDHR}})/R \tag{13}$$

Because the number of CHs in each round is highly related with the energy dissipation, based on the conclusion we have in the energy dissipation analysis section, we can have $c_{iIHR} \approx c_{iLHR} > c_{iDHR}$. Thus, we can conclude that $D_{IHR} > D_{DHR} > D_{LEACH}$.

5. New features in our simulation tool

We have developed a simulation tool in order to set network parameters easily, view the topology changes over the lifetime of a WSN, observe the behavior of WSN when cluster head faults are injected, and generate visualized results automatically. One of the objectives in this design is to provide a user-friendly interface to change the network parameters. Another objective is that to provide the functionality of observing the topology changes over the whole simulation period. Most of all, the major objective of this design is to provide fault models that generated in the failure to support us to study the characteristic of fault tolerance mechanisms in WSN.

Based on the above reasons, we customize our simulation tool to support the requirement for WSN fault tolerance analysis. Our tool has the following advantages:

- A fault model that can be used to study the effectiveness of fault tolerance mechanisms in WSN.
- A user friendly configuration interface that can decrease the chance of human error.
- Multiple tabs to display results automatically.

- A function to visualize topology so that user can observe the network changes during the whole simulation period.

In this section, we introduce the configuration interface and presentation interface of our tool for end users and some function interfaces of our tool for developers to show how to adapt the current tool to support new mechanisms. Also we present the fault model and other mechanisms we add to this tool.

5.1. User interface

5.1.1. The configuration interface

We use C# Windows form project to implement this simulation tool. C# provides interfaces to draw image. We can run simulation and see topology changes in the lifetime of a WSN and see the results immediately. By providing a user interface, it is easier to change the setup of the WSN parameters and decrease the configuration faults dramatically. Our user interface is shown in Fig. 3.

We provide 12 parameters for users to configure the network setup. The meanings of the parameters are as the followings:

- The number of nodes. This is to define the size of the network, i.e., how many nodes are there in the WSN. All the nodes are uniformly distributed in a 600×600 pixels picture box area. The ratio of pixels to meters is 600/N, where N is the number of nodes. For example, if we set the network size as 100 nodes, then the 100 nodes are uniformly distributed in a 100×100 squared meters area. Each of six pixels represents one meter.
- Number of cluster heads. This parameter is to define the number of cluster nodes in a WSN. We do not expect the CH/nodes ratio is too large. If it is too large, the distance between each cluster is too short, which will increase interference. We accept the ratio to be less than 0.2.
- Base station position X. This parameter is the X coordinate on the picture box control component on the form. It should be within the rage of 0 to 600.
- Base station Position Y. This parameter is the Y coordinate on the picture box control component on the form. It should be within the rage of 0 to 600.
- Rounds to Run: This parameter is to determine the duration of a simulation process.
- Cluster head fault rate: This parameter is to determine the value of λ to simulate the CHs faults.
- Threshold D0: This parameter is to decide the distance threshold in the energy model.
- Hearing distance: This parameter is to define the broadcast distance, measured in meter unit. Even when there is no channel established between two nodes, the beacon message can be received within this distance.
- Data Packet Size: This parameter is to determine the size of the data information sensed by collector nodes, measured in bytes.
- Beacon Packet Size: This parameter is to determine the size of the beacon message, measured in bytes. The beacon message can represent the status of a node. All the query messages and respond messages about aliveness of a sensor node can be encapsulated in a beacon packet.
- Initial energy: This parameter is to define the energy of a node at the very beginning of a WSN simulation process.
- Algorithms to Simulate: We support three mechanisms LEACH, DHR and IHR. Users can check the algorithms to simulate.

5.1.2. The presentation interface

Our simulation tool can display the topology for different mechanisms. Fig. 4 and Fig. 5 show the topologies at the beginning phase when using LEACH mechanisms and IHR mechanism, respectively. We do not show the topology of DHR mechanism, because the DHR and IHR use the same method to form clusters and to find back up cluster heads. The beginning phase of DHR will not be much different from that of IHR.

Our tool also supports to view the different phases of a specific mechanism. Fig. 6 shows the 1/4 phase of IHR mechanism. From the figure we can see, at the 1/4 phase, there are only 36 alive nodes left, and there are still 10 cluster heads. We can only see 6 clusters in the figure, because there are 4 CHs do not have any children. We provide this function for the user to observe the topology changes during the whole WSN network lifetime.

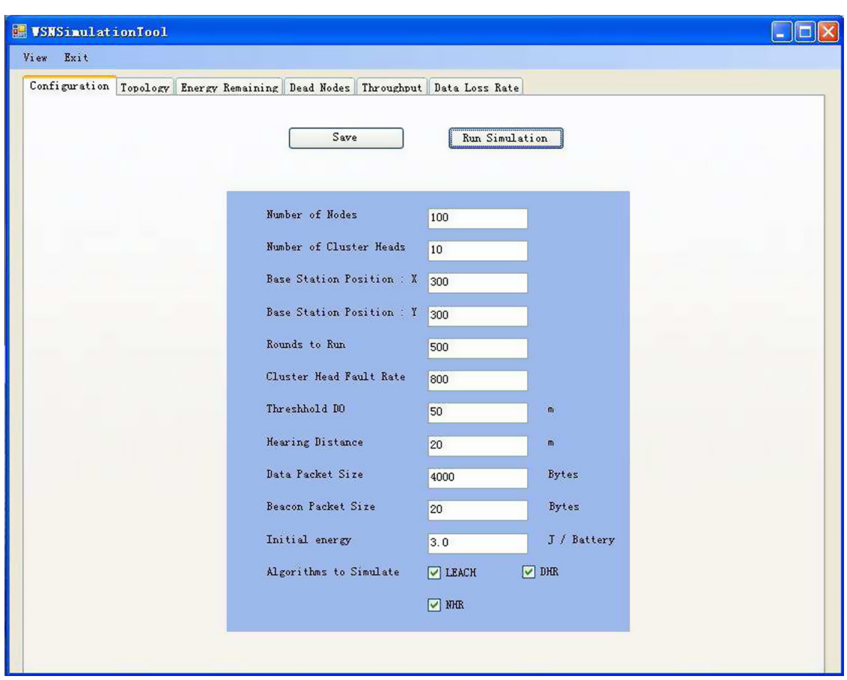

Fig. 3. Network configuration user interface.

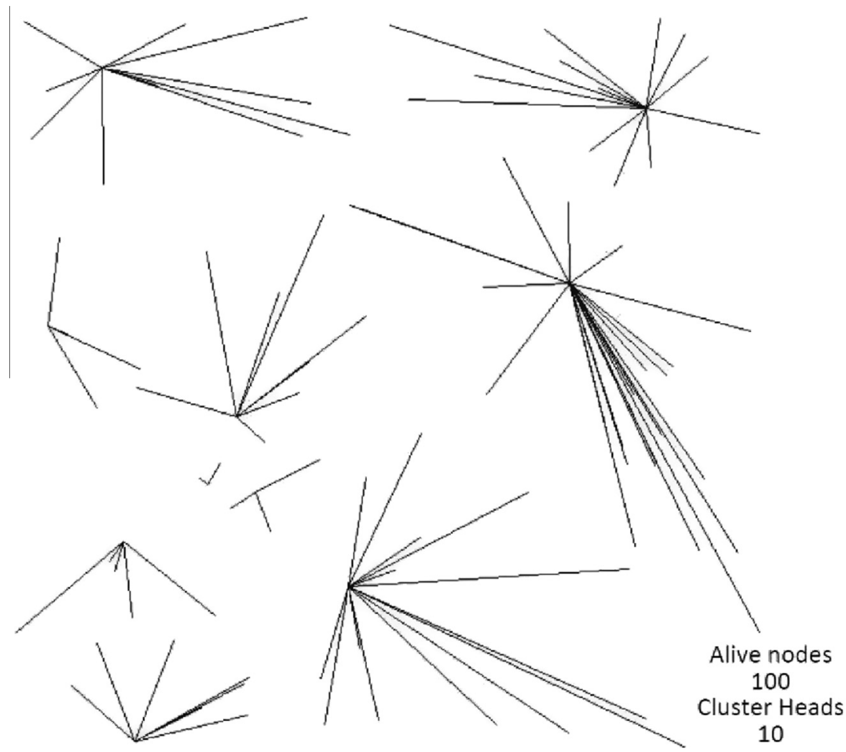

Fig. 4. The LEACH topology at the beginning phase.

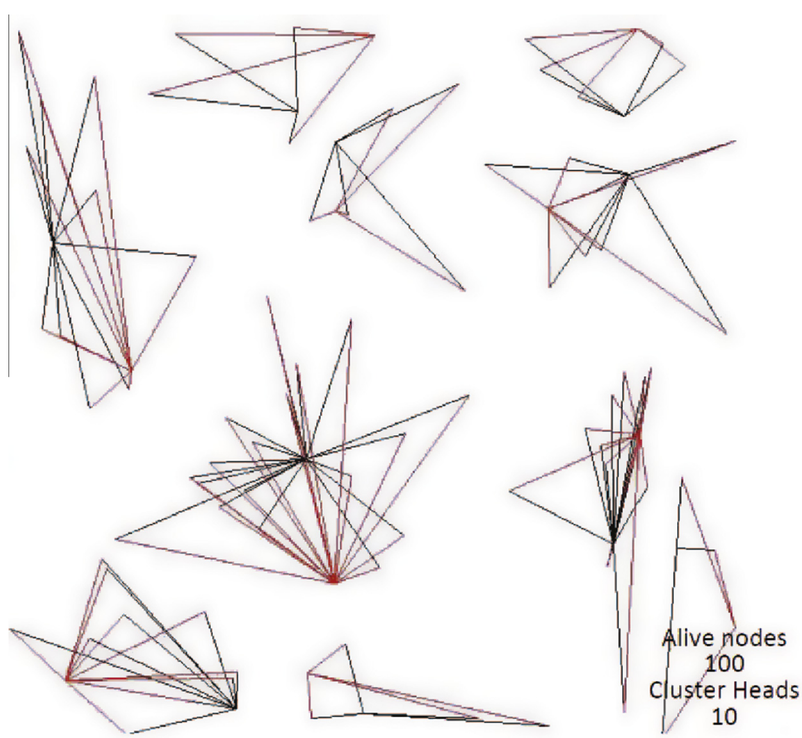

Fig. 5. The IHR topology at the beginning phase.

6. Performance evaluation of IHR protocol

In this section, we present the evaluation results of the IHR mechanism. We evaluate the IHR in the aspects of energy consumption, dead nodes, throughput, and data loss rate. We compare these results to those of LEACH and DHR, the simulation results are consistent with our theoretical analysis showing that IHR performs better in the aspect of energy dissipation than DHR and better in the aspect of dependability than LEACH.

6.1. Environment setup

In this subsection we present the environment setup for the evaluation of the IHR mechanism. The simulation is running for

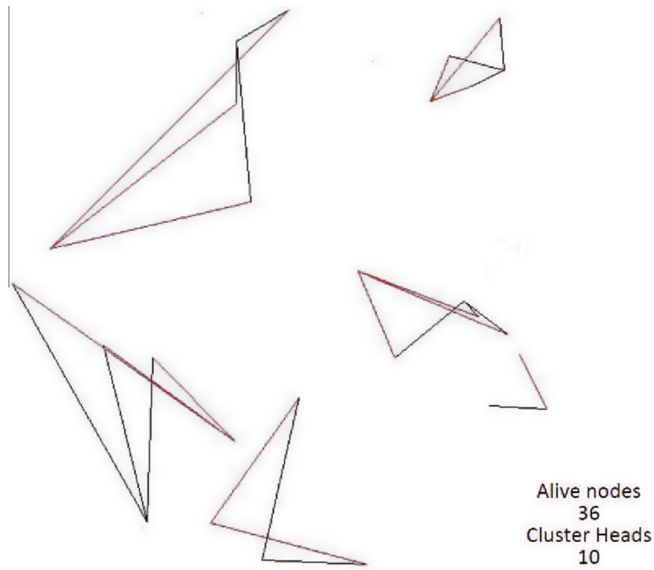

Fig. 6. The IHR topology at the second phase.

500 rounds; the network size is 100 nodes; we have 10 clusters; data packet size is 4000 bytes; beacon (include both query message and respond message) message size is 20 bytes; the d0 in the energy model is 50 m; the BS is at (300, 300) on the picture box control component, and the initial energy for each node is 3.0 J/Battery.

To simulate faults, we use Poisson distribution to schedule a time *T* when CH faults is expected to inject into the network. The fault rate at 0.5 means at time T, 50 percent of CHs will fail.

6.2. Simulation results

In this subsection we will present the simulation results. Fig. 7 shows the Energy Remaining in each round. In Fig. 7, there are 6 curves. We show two different situations for each algorithm. LEACH: 0/DHR: 0/ IHR: 0 is the situation when no CH fault is injected. {LEACH: 0.5/ DHR: 0.5/ IHR: 0.5} is the situation when CH faults are injected at rate 0.5.

From the figure we can see, when there is no fault injected, the energy remaining of DHR is much lower than that of the other two algorithms. The reason is that, in DHR, sending data to two cluster heads at the same time is too energy consuming. When the cluster heads fault rate is at 0.5, IHR is better than LEACH, and LEACH is better than DHR, because the collecting mechanism in IHR saves a lot of energy dissipation.

Table 2
Energy remaining value without fault injection.

LEACH	DHR	IHR
79.59 J	18.87 J	89.13 J

Table 2 shows the energy remaining value in the last round when there is no fault injected to the network.

From the table we can see the energy remaining in IHR is much more than that of in DHR. The result confirms our conclusion that the IHR can save energy dramatically than that of DHR, while it has nearly the same energy dissipation rate as that in LEACH.

Fig. 8 shows the number of dead nodes in each round. There are 6 curves. We show two different situations for each algorithm. {LEACH: 0/ DHR: 0/ IHR: 0} is the situation when no CH fault is injected. {LEACH: 0.5/ DHR: 0.5/ IHR: 0.5} is the situation when CH faults are injected at rate 0.5.

From the figure we can conclude, when there is no fault injected, the number of dead nodes of DHR is much larger than the other two algorithms. This also demonstrates that DHR is energy consuming and the lifetime of it is shorter than the other two. When the cluster heads fault rate is at 0.5, all nodes are dead at round 320 when using DHR algorithm, while all nodes are dead at round 370 for IHR and LEACH. This shows that IHR has longer lifetime than that of DHR.

Table 3 shows the dead nodes value in the last round when there is no fault injected to the network. From the table we can see the number of dead nodes in IHR is much less than that of in DHR. The result also confirms our conclusion that the IHR can save energy dramatically than DHR.

Fig. 9 shows the throughput in each round. The value of throughput is calculated as the accumulative size of unique data transmitted to BS successfully at each round. So in DCH, the received data by BS is only calculated once. We show two different situations for each algorithm. LEACH: 0/DHR: 0/ IHR: 0 is the situation when no CH fault is injected. LEACH: 0.5/DHR: 0.5/ IHR: 0.5 is the situation when CH faults are injected at rate 0.5.

To calculate the throughput, we only count the unique data collected from each data transmission process. DHR sends duplicate data, but we only count once. This is the reason the throughput of DHR is much lower than the other two algorithms. When CH faults are injected, the throughput of IHR is much higher than the other two. The throughput of IHR is higher than LEACH is because it provides alternative path for NCHs when CHs faults happening, while LEACH does not do this. Thus, more data information can be delivered to BS, resulting in higher throughput in this mechanism. The throughput of IHR is higher than DHR, be-

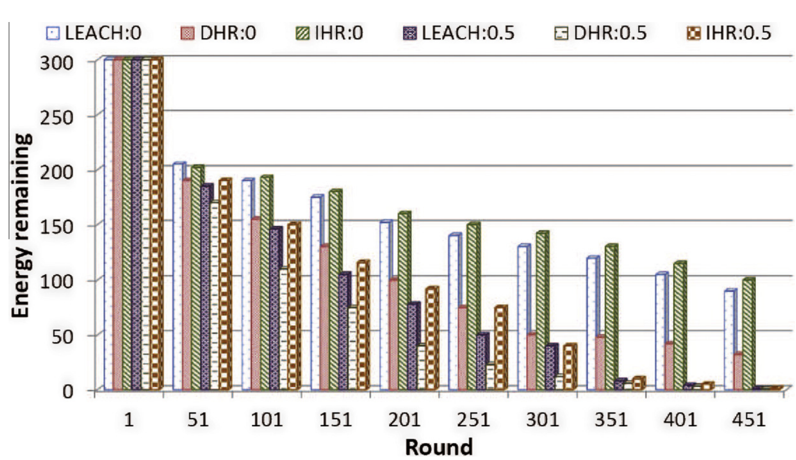

Fig. 7. Energy remaining in each round.

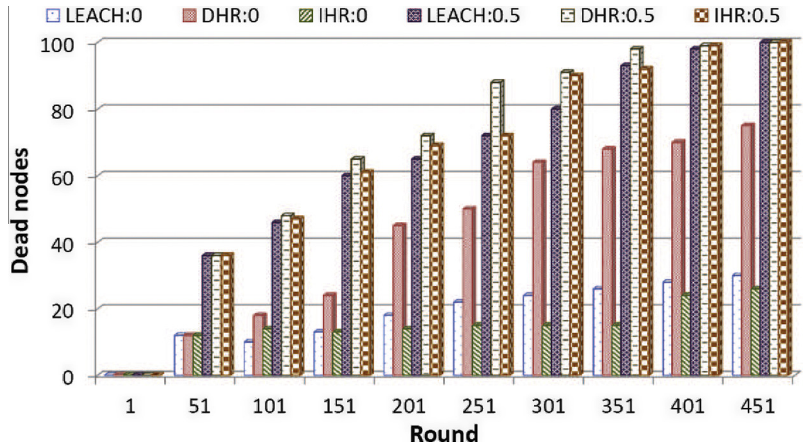

Fig. 8. Number of dead nodes in each round.

Table 3
Number of dead nodes without fault injection.

LEACH	DHR	IHR
33	72	31

Table 4
Throughput With Fault Rate 0.5.

LEACH	DHR	IHR
333952 KB	307276 KB	388520 KB

cause the collecting mechanism of it saves a lot of energy dissipation. Thus the nodes in IHR work longer than those nodes in DHR, and more data information can be collected and relayed to BS.

Table 4 shows the throughput value in the last round when CH faults are injected to network at rate 0.5. From the table we can see the throughput of IHR is higher than the other two. This confirms our conclusion that IHR has the best dependability among the three mechanisms we implement. Thus its throughput is the highest during the whole simulation time.

Fig. 10 shows the data loss rate when different CH faults are injected at different rates. The data loss rate is calculated as the following formula:

$$\frac{ideal\_throughput - actual\_throughput}{ideal\_throughput} \tag{14}$$

The *ideal_throughput* is the throughput when no node is dead during the whole simulation period. From Fig. 10, we can see that IHR has the lowest data loss rate, while DHR has the highest data loss rate. This is because DHR is energy consuming and more nodes are dead in DHR collecting mechanism.

The simulation results confirm our theoretical analysis that IHR can perform better in the aspect of energy dissipation than that of DHR and performs better in the aspect of dependability than that of LEACH. The disadvantage of IHR protocol is that it will bring communication overhead. However, the experimental results show that the energy consumption of the communication overhead can be compensated by the energy saved from avoiding the redundant transmission. Another concern of the IHR protocol is that how long it will take the backup cluster head to find out that its primary cluster head has failed. The longer it takes, the more meaningful sensed data will be lost.

In some extreme cases we know, it is possible that certain NCHs cannot find and join a CH. This is a reliability issue in the design. But the chance is very very low. There are many general techniques

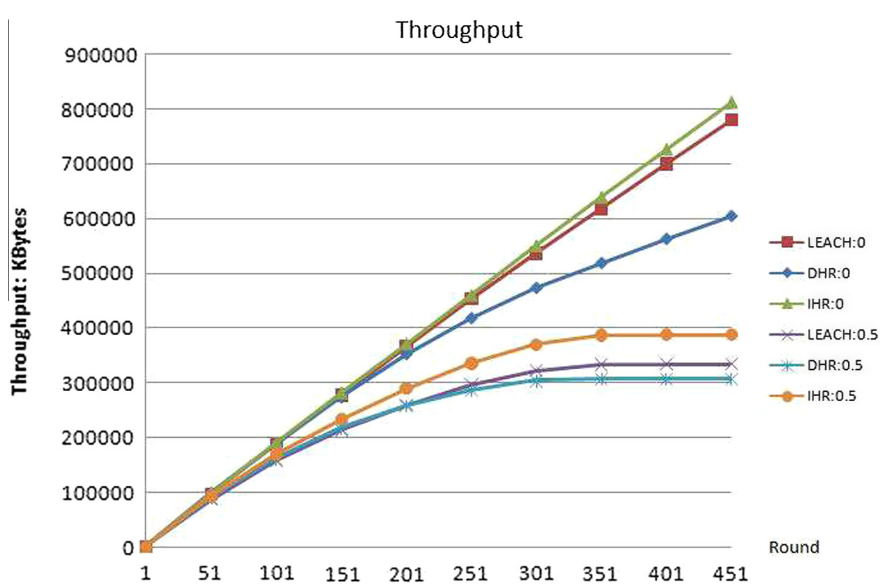

Fig. 9. Throughput in each round.

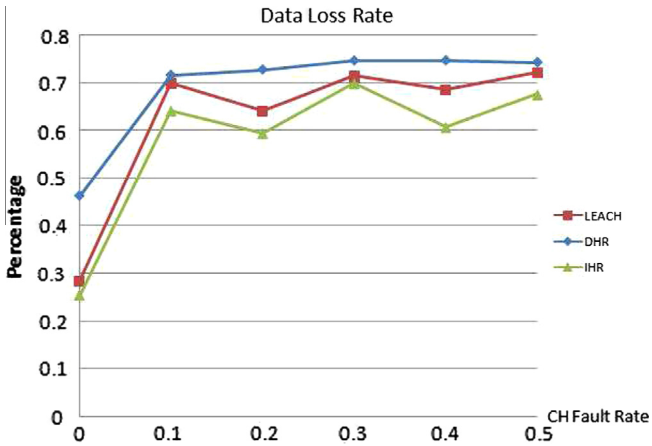

Fig. 10. Data loss rate at different cluster head fault rate.

for enhancing the system reliability. For a specific high performance design on this scenario, we will investigate and solve it in our future work.

The failure rate of the sensor in the wireless sensor network varies. In our approach, the collector only send data to the PCH, instead of sending data to both PCH and BCH. And the BCH will check the aliveness of PCH in every period of time. In the case that the PCH has failed, the data will be send directly to the BCH. Meanwhile, when the BCH has failed, the sub-nodes in this cluster will be re-cluster to one cluster in the neighborhood by using the LEACH mechanism. The delay of the end to end communication when the PCH has failed can be limited by reducing the period time between two aliveness checks of a PCH, since the shorter period leads to a smaller set of data that needs to be re-send.

7. Conclusion and future work

In this paper, we summarized the principles to design fault tolerance mechanisms for a WSN. Based on the principles, we designed and developed our own fault tolerance mechanism, called IHR. We developed a customized simulation tool to support these mechanisms, i.e., LEACH, DHR and IHR, with a fault injection function in order to evaluate the robustness of our mechanism. The experimental results showed that compared to DHR and LEACH, IHR can save energy consumption significantly and result in lower data loss rate.

In our proposed method, we assume that NCHs are able to find their nearest CH. However, in some extreme cases, some of the NCHs cannot find and join a CH. This may lead some reliability issues in the proposed method. In the future work, we will further investigate the method to solve this issue. Meanwhile, message may be corrupted or lost, because of imperfections in point-to-point communication, especially if the communication medium is wireless. Certain level of relief against this issue can be provided by link layer protocols. For example, we can implement a simple *automatic repeat request* (ARQ) scheme as the link layer packet lost recovery scheme in our proposed method. Such an evaluation will be conducted in the future work.

Acknowledgements

This work was supported in part by the NSF CNS-1249223, NSFC 61071061, the Univ. of Kentucky Start Up Fund; NSFC 61170077, SZ-HK Innovation Circle project ZYB200907060012A, NSF GD:10351806001000000, S & T project of SZ JC200903120046A, S & T project of GD 2012B091100198; the NSF CNS-0969013, CNS-0917021, and CNS-1018108; The National High-Tech. R & D Program of China (863 Program 2009AA012201) and the Shanghai International S & T Collaboration Program (09540701900).

References

[1] W.R. Heinzelman, A.P. Chandrakasan, H. Balakrishnan, An application-specific protocol architecture for wireless microsensor networks, IEEE Transactions on Wireless Communications 1 (4) (2002) 660–670.
[2] M. Qiu, C. Xue, Z. Shao, M. Liu, E.H.-M. Sha, Energy minimization for heterogeneous wireless sensor networks, Journal Embedded Computing 3 (2) (2009) 109–117.
[3] M. Qiu, C. Xue, Z. Shao, Q. Zhuge, E.H.-M. Sha, Efficient algorithm of energy minimization for heterogeneous wireless sensor network, in: IEEE EUC, 2006, pp. 25–34.
[4] M. Qiu, E.H.-M. Sha, Cost minimization while satisfying hard/soft timing constraints for heterogeneous embedded systems, ACM Transactions on Design Automation Electronic Systems 14 (2) (2009) 1–30 (TODAES 2011 Best Paper Award).
[5] M. Qiu, W. Gao, M. Chen, J. Niu, L. Zhang, Energy efficient security algorithm for power grid wide area monitoring systems, IEEE Transactions on Smart Grid 2 (4) (2011) 715–723.
[6] L. Yang, M. Ji, Z. Gao, W. Zhang, T. Guo, Design of home automation system based on ZigBee wireless sensor network, in: International Conference on Information Science and Engineering, Nanjing, China, 2009, pp. 2610–2613.
[7] S. Nourizadeh, C. Deroussent, Y.Q. Song, J.P. Thomesse, Medical and home automation sensor networks for senior citizens telehomecare, in: IEEE International Conference on Communications Workshops, Dresden, Germany, 2009, pp. 1–5.
[8] H. Song, S.C. Zhu, G.H. Cao, SVATS: A sensor-network-based vehicle anti-theft system, in: INFOCOM, Phoenix, AZ, USA, 2008, pp. 2128–2136.
[9] R. Szewczyk, A. Mainwaring, J. Polastre, An analysis of a large scale habitat monitoring application, in: ACM SenSys'04, Baltimore, Maryland, USA, Nov. 3–5, 2004, pp. 214–226.
[10] G. Toll, J. Polastre, R. Szewczyk, A macroscope in the redwoods, in: ACM SenSys'05, San Diego, CA, USA, Nov. 2005, pp. 51–63.
[11] P. Sikka, P. Corke, P. Valencia, Wireless adhoc sensor and actuator networks on the farm, in: ACM SenSys'06, Boulder, CO., USA, Nov. 2006, pp. 492–499.
[12] G. Werner, P. Swieskowski, M. Welsh, Demonstration: real-time volcanic earthquake localization, in: ACM SenSys'06, Boulder, CO., USA, Nov. 2006, pp. 357–358.
[13] M. Demirbas, Scalable design of fault-tolerance for wireless sensor networks, Ph.D. thesis, The Ohio State University, Columbus, OH, 2004.
[14] N. Ramanathan, K. Chang, R. Kapur, L. Girod, E. Kohler, D. Estrin, Sympathy: a debugging system for sensor networks, in: IEEE International Conference on Local, Computer Networks, 2004, pp. 554–555.
[15] M. Ringwald, K. Romer, A. Vitaletti, SNIF: Sensor network inspection framework, Tech. rep., ETH Zurich, Switzerland, 2006.
[16] N. Jain, V.M. Vokkarane, J.P. Wang, Performance analysis of dual-homed fault-tolerant routing in wireless sensor networks, in: IEEE Conference on Technologies for Homeland Security, Waltham, MA, 2008, pp. 474–479.
[17] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, A survey on sensor networks, IEEE Communications Magazine 40 (8) (2002) 102–116.
[18] N. Xu, S. Rangwala, A wireless sensor network for structural monitoring, in: ACM SenSys'04, Baltimore, Maryland, USA, Nov. 2004, pp. 13–24.
[19] S.B. Attia, A. Cunha, A. Koubaa, M. Alves, Fault-tolerance mechanisms for Zigbee wireless sensor networks, Tech. rep., IPP-HURRAY Research Group, Polytechnic Institute of Porto, 2007.
[20] ZigBee Alliance, Zigbee specification. Available from: <www.zigbee.org>.
[21] S.C. Yu, Y.Y. Zhang, R-Sentry: providing continuous sensor services against random node failures, in: IEEE International Conference on Dependable Systems and Networks, Edinburgh, UK, 2007, pp. 235–244.
[22] S. Misra, I. Woungang, S.C. Misra, Guide to Wireless Sensor Networks, Springer, 2009.

Meikang Qiu received the B.E. and M.E. degrees from Shanghai Jiao Tong University, China. He received the M.S. and Ph.D. degrees of Computer Science from University of Texas at Dallas in 2003 and 2007, respectively. He had worked at Chinese Helicopter R&D Institute and IBM. Currently, he is an assistant professor of ECE at University of Kentucky. He is an IEEE Senior member and ACM Senior member. He has published 160 papers. He is the recipient of the ACM Transactions on Design Automation of Electronic Systems (TODAES) 2011 Best Paper Award. His paper about cloud computing has been ranked #1 in Most Downloaded Paper 2012 of JPDC journal (Elsevier). He also received four other best paper awards (IEEE EUC'09, IEEE/ACM GreenCom'10, IEEE CSE'10, and IEEE ICESS'12) and one best paper nomination (IEEE EmbeddedCom'09). He also holds 2 patents and has published 3 books. He has also been awarded Navy summer faculty award in 2012 and SFFP Air Force summer faculty in 2009. He has been on various chairs and TPC members for many international conferences. He served as the Program Chair of EM-Com'09. His research interests include embedded systems, computer security, and wireless sensor networks.

Zhong Ming is a professor at College of Computer and Software Engineering of Shenzhen University. He is a member of a council and senior member of China Computer Federation. His major research interests are software engineering and embedded systems. He led two projects of National Natural Science Foundation, and two projects of Natural Science Foundation of Guangdong province, China.

Jiayin Li received the B.E. and M.E. degrees from Huazhong University of Science and Technology (HUST), China, in 2002 and 2006, respectively. In May 2012, he has obtained his Ph. D. degree in the Department of Electrical and Computer Engineering (ECE), University of Kentucky. His research interests include software/hardware co-design for embedded system and high performance computing.

Jianning Liu got her B.S. degree in 2002 at Beijing University of Posts and Telecommunications in China, major in telecommunication engineering. She got her master degree in computer science department of University of Kentucky in 2011. Currently she is an employee in Cummins, Inc. working on CAN controller development.

Gang Quan is currently an Associate Professor with the Electrical and Computer Engineering Department, Florida International University, Miami. He received the B.S. degree from the Tsinghua University, Beijing, China, the M.S. degree from the Chinese Academy of Sciences, Beijing, and the Ph.D. degree from the University of Notre Dame, Notre Dame, IN. His research interests include real-time system, power/thermal aware design, embedded system design, advanced computer architecture and reconfigurable computing. Prof. Quan received the NSF CAREER award in 2006.

Yongxin Zhu (Winson) is an Associate Professor with the School of Microelectronics, Shanghai Jiao Tong University, China. He is a Senior Member of China Computer Federation and a senior member of IEEE. He received his B. Eng. in EE from Hefei University of Technology, and M. Eng. in CS from Shanghai Jiao Tong University in 1991 and 1994 respectively. He received his Ph.D. in CS from National University of Singapore in 2001. His research interest is in computer architectures, embedded systems, medical electronics and multimedia.