Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2010, Article ID 201416, 11 pages
doi:10.1155/2010/201416

Research Article

Security Topology Control Method for Wireless Sensor Networks with Node-Failure Tolerance Based on Self-Regeneration

Liang-Min Wang,[1, 2] Yuan-Bo Guo,[3] and Yong-Zhao Zhan[1]

[1] *School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, China*
[2] *School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore 639798*
[3] *School of Electronic Technology, Information Engineering University of PLA, Zhengzhou 450004, China*

Correspondence should be addressed to Liang-Min Wang, lmwang@ntu.edu.sg

Received 18 September 2009; Accepted 15 February 2010

Academic Editor: Amiya Nayak

Wireless sensor networks are often deployed in hostile and unattended environments. The nodes will be failure by fault, intrusion, and the battery exhaustion. Node-failure tolerance is an acceptable method to improve the networks' lifetime. Then two key problems for topology control are presented: first, how to get a node-failure topology when there is intrusion from the nodes of hostile enemies? second, how to sustain this node-failure topology with all deployed nodes being exhausted ultimately? We propose a novel approach for topology control and prove that it is node-failure tolerant. The approach contains three phases: topology discovery, topology update, and topology regeneration. A tricolor-based method is proposed to build architecture with high tolerance ability and some security protocols are employed to preclude the hostile nodes in discovery phase. In update and regeneration phases, the newly deployed nodes are regarded as renewable resource to fill in the consumed energy, enhance the debased node-failure tolerance ability, and prolong network lifetime, and a security protocol with forward and backward secrecy is devised to adapt the topology changed by node failure and node joining. Some attributes of the presented method are shown by simulations, and differences are given by comparison with related work.

1. Introduction

The topology of wireless networks is the basis of data transmitting. The topology control should consider how to construct and sustain a connected sparse network. Of course, it should also pay attention to communication efficiency. Moreover, the topology of wireless sensor networks should have node-failure tolerance ability for being disposed in hostile region, where the sensors are unattended and will be failure by fault, intrusion, and energy exhausting. Firstly, the sensor nodes are limited by the bandwidth, memory, and computational capability which make the traditional asymmetrical cryptographic methods unavailable to protect the intrusion. Further, the unattended sensors are doomed to failure because the battery will be exhausted. The failure nodes have great influence on the network topology, because they often divide the communication connection of nodes, deduce the availability of network, and lead to network failure. So we try to propose a new method for topology control to improve the topology survivability of WSN in hostile region.

Early studies of topology control in wireless sensor networks try to balance energy loss and network lifetime [1], such as the methods of power control method [2, 3], hierarchy models [4, 5], and working on duty. Later on, some fault tolerant methods [6, 7] are proposed with consideration of the paucity of node failures, in which multiplepoints connectivity and multi-edge connectivity [8] are defined as the tolerance ability of topology. Recently, security becomes the hot topic of wireless sensor networks, and the differences between fault and intrusion are discovered and noticed. Faults are brought by random events, but intrusions are produced by malice adversary. Wang et al. [9] discussed the differences between fault tolerance and intrusion tolerance and pointed out what kinds of topology structure have higher capability of intrusion tolerance in theory. Following the concept of node-failure tolerance presented by Wang et al. [9], this paper focuses on two key problems: how to get a tolerant topology and how to sustain this tolerant topology when all deployed nodes will be failure ultimately? In this paper, we focus on these two questions. In the following text, we use failure nodes as the unified name of the fault nodes, the intruded nodes, and the exhausted nodes. We only consider the intrusion nodes in security architecture for the hostile and malice attack to be remarkably different from the fault and exhaustion in this scenario.

The organization of this paper is as follows. In Section 2, a topology discovering approach is proposed, in which the discovered topology is highly node-failure tolerant. In the discovery phase, we employ a secure architecture based on symmetric cryptology to exclude the outside intrusion, and the secure architecture provides some nodes authentication, communication encryption, and certain intrusion tolerance. In Section 3, we propose a self-regeneration method for topology reconfiguration. The newly deployed nodes are regarded as renewable resource to form new network and prolong network lifetime. A security protocol with forward and backward secrecy is devised to adapt the topology changed by node joining and failure. In Section 4, the simulations show the performances of topology control, followed by comparison and analysis of the characteristics of this paper.

2. Topology Discovery with Node-Failure Tolerance

In this section, we give the network model and our assumptions at first. Then we propose a tricolor-based algorithm to generate a topology structure with high ability of node-failure tolerance. A lightweight security structure is used to prevent outside attack. In the end of this section, we present the node-failure tolerance ability of the topology and the security structure by theorem.

2.1. Network Model. We study the intolerance topology control in a two-dimension static sensor network where the location of sensors does not change after deployment. We assume that all direct communication links between nodes are bidirectional. We also assume that every sensor node has a unique ID number in the network which is assigned by the network operator before deployment.

We assume that all the sensors are dropped from an airplane or spread out by a roboticized device with very high density which provides redundancy to tolerant failure nodes. The density is determined by the network operators' requirement. If t nodes tolerance is required, then the density should be high enough to provide at least t neighbors for each node. We call that the network is *satisfyingly redundant* or has *satisfying redundancy* if the node density can satisfy the requirement of the operator.

We also assume that the sensors are deployed in hostile environment and it is unattended. So the battery will be exhausted after long-time using and every node will be failure ultimately, and the adversary can deploy their sensors in the same area to launch outside attack. Moreover, the adversary can capture some nodes and compromise them. Then can intrude the network to launch the inside attacks.

2.2. Tricolor-Based Topology Discovery. In our Tricolor-Based topology discovery algorithm, the sensors have three working states: initial, dormant, and running. We denote the nodes in these three states by white, grey, and black, respectively. Algorithm 1 gives the process of generating topology.

Algorithm 1 (Tricolor based Topology Discovery Algorithm). (1) Network initializes and all nodes are dyed to white

(2) A white node is arbitrarily selected and dyed black, named as A; if there is no white nodes, go to(6)

(3) The newly dyed black node initiates Neighbor Query Protocol (Protocol 1) with radius r_1 dyes all the neighbors grey and preserves the record of these neighbors

(4) The newly-dyed black node initiates Neighbor Nodes Query Protocol (Protocol 1) with communication radius r_2, arbitrarily selects a white node from A's r_2 neighbors, named B, and dyes B black, then the algorithm go to (3); if it cannot find node B, goes to(2)

(5) The algorithm outputs the result and is terminated.

In Algorithm 1, r represents communication radius of sensors, the value of r_1 and r_2 must meet the following:

$$a_1 r \leq r_1 \leq a_2 r \leq a_3 r \leq r_2 \leq a_4 r, \tag{1}$$

where a_1, a_2, a_3, and a_4 are undetermined parameters, and meet $a_1 \leq a_2 < a_3 \leq a_4 \leq 1$. The Neighbor Query Protocol is given as Protocol 1.

Protocol 1 (Neighbor Query Protocol). (1) Node A broadcasts query message for neighbor nodes with radius r, in which A's ID is included in the message

(2) After receiving the query message from A. node B writes A's ID. and sends its own ID to A as response.

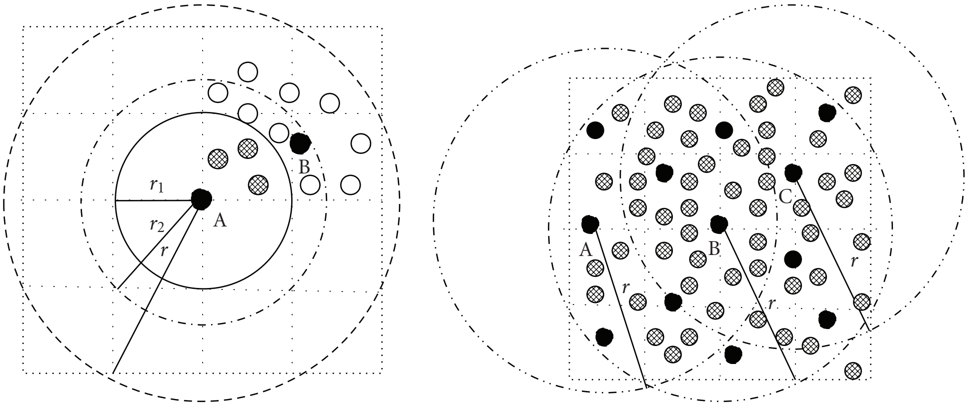
(3) After receiving the response from B, node A records the ID of node B.

When Algorithm 1 is finished, the black nodes are used as cluster head whose communication module is open to receive and forward messages with other nodes in a range of communication radius. The grey nodes in adjacent range of black nodes are set in dormant state, in which communication module is turned off to save energy, and it will not be turned on until there are new sensor apperceiving data.

Lemma 1. *If the sensor nodes are uniformly distributed with satisfying redundant density, each black node has no less than t black neighbors in their neighbor area after Algorithm 1 is terminated under constraint of Formula* (1). *Then it holds that*

$$\frac{2\pi a_1 + a_4}{{a_4}^3} < t < \frac{2\pi a_4 + a_1}{a_1 {a_3}^2}. \tag{2}$$

Proof. Figure 1(a) shows the instance when Algorithm 1 completes the first cycle. There is not black node in range



(a) black node

(b) self-generated topology

Figure 1: Tricolor-Based Topology Discovery: Black nodes are filled in the black, and the grey nodes are shaded by the gird instead of being filled in grey for obviously discrimination.

of A's r_1 neighbor area, then new black node is selected in the annular area with point A and radius $(r_2 - r_1)$. After Algorithm 1 is completed, we denote that the number of the nodesin the annular area is n. With the assumption that the networks have satisfying redundancy, we consider two boundary cases:

When all the black nodes are located at the circumference with radius r_1 and the distance between every two nodes is r_2, we obtain the minimum value of n. The maximum value of n is obtained in the case that the black nodes are located at the circumference with radius r_2 and the distance between every two nodes is r_1. Thus we have

$$\frac{2\pi r_1}{r_2} < n < \frac{2\pi r_2}{r_1}. \tag{3}$$

That is to say, all the grey nodes in A's r_1 neighbor cycle area are shared by $n + 1$ black nodes. Let t be the number of black neighbors of an arbitrarily selected node, then we have

$$t = (n+1)\frac{r^2}{{r_2}^2}. \tag{4}$$

From Formulas (1), (2), (3), and (4), we conclude that

$$\frac{2\pi a_1 + a_4}{{a_4}^3} < t < \frac{2\pi a_4 + a_1}{a_1 {a_3}^2}. \tag{5}$$

□

Deduction 1. If the sensor nodes are uniformly distributed with satisfying redundant density, the topology generated by Algorithm 1 under constraint of Formula (1) is t_1 node-failure tolerance for external attack, in which t_1 satisfies

$$t_1 > \left[\frac{2\pi a_1 + a_4}{{a_4}^3}\right] - 1. \tag{6}$$

Proof. According to [9], the topology is t_1 node failure tolerance for external attack if it is still connected after arbitrarily selected t_1 nodes are deleted. In satisfying redundant network, if t_1 grey nodes are deleted, the network is still connected. If t_1 black neighbors of a cluster-head are deleted, then the cluster-head has still no less than $[(2\pi a_1 + a_4)/{a_4}^3] + 1$ black neighbors according to Lemma 1. Even if $[(2\pi a_1 + a_4)/{a_4}^3] - 1$ neighbors of a black cluster-head are deleted, the node still has a neighbor and network is still connected. So the topology is t_1 node-failure tolerance. □

2.3. Security Assurance for Topology Discovery. The message of Neighbor Query Protocol must be encrypted and authenticated to exclude outside attack. The best authentication scheme is public key cryptography which provides asymmetric authentication for query broadcast authentication from the initial black node. However, public key system does not fit the wireless sensor networks because of its weak computing capability, narrow communication bandwidth, and extremely limited energy. A symmetric key system with short key t and small calculating requirements is used in the authentication of our query broadcast.

Firstly, a pair-key negotiation scheme based on the Blom Key Predistribution Scheme [10] and the Blundo Method [11] is proposed, which makes two nodes share a symmetric key. Before deployment, each node gets a unique identity ID and a polynomial $f(x, y)$ from center authentication (CA), in which $f(x, y)$ is a symmetric bivariant polynomial defined in a infinite field with degree t_2. After deployment, a pair of nodes, A and B, can negotiate a master key K_{AB} to communicate with Protocol 2.

Protocol 2 (Master Key Generation Protocol). (1) Node A sends a request message to node B, in which A's identity ID_{A} is contained

(2) Node B sends its identity ID_{B} as a reply message, lets $x = \mathrm{ID}_{\mathrm{A}}$, ID = ID_{B}, then fills the value of x, ID in Formula (7), and uses the computed value as the sharing key K_{BA};

(3) Node A accepts ID_{B}, lets $x = \mathrm{ID}_{\mathrm{B}}$, ID = ID_{A}, and computes the sharing key K_{AB} from formula (1) too.

In Protocol 2, sharing keys K_{BA} and K_{AB} are computed with Formula (7):

$$g_{\mathrm{ID}}(x) = f(x, \mathrm{ID}), \tag{7}$$

where x is the counterpart's identity ID, $f(x,y)$ are Formula (8) shared by all nodes which is set by the operator before deployment

$$f(x, y) = \sum_{i,j=0}^{t_2} a_{ij}x^i y^j \left(a_{ij} = a_{ji}\right). \tag{8}$$

Theorem 1. *Node A and node B obtain the secret master key shared by each other from Protocol 2, that is,* $K_{\mathrm{AB}} = K_{\mathrm{BA}}$.

Proof. Obviously, function g_{ID} is a univariate polynomial about x with degree t_2 by filling $f(x,y)$ of Formula (8) in Formula (7). Operator distributes secret material (Coefficient of polynomial g_{ID}) to sensor nodes before they are disposed. Each node fills the other's ID as the value of x in $g_{\mathrm{ID}}(x)$ and obtains the secret value by computing. We denote ID_1 and ID_2 as the identities of node A and node B, respectively; we can obtain the value from the following formulas

$$\begin{aligned} g_{\mathrm{ID}_1}(\mathrm{ID}_2) &= f(\mathrm{ID}_2, \mathrm{ID}_1), \\ g_{\mathrm{ID}_2}(\mathrm{ID}_1) &= f(\mathrm{ID}_1, \mathrm{ID}_2). \end{aligned} \tag{9}$$

We can obtain the following equation by symmetry of bivariate polynomial $f(x,y)$ in Formula (8):

$$f(\mathrm{ID}_2, \mathrm{ID}_1) = f(\mathrm{ID}_1, \mathrm{ID}_2), \tag{10}$$

thus $g_{\mathrm{ID}_1}(\mathrm{ID}_2) = g_{\mathrm{ID}_2}(\mathrm{ID}_1)$, that is, $K_{\mathrm{AB}} = K_{\mathrm{BA}}$. □

Black head node often broadcasts messages to the nodes in its cluster. A secure session key is required for communication between head and its neighbors. It can be used to encrypt and authenticate message of the communication. Now node A and B share a secret K_{AB} and take it as the master key. If node A accepts a message from B, but node A can not decrypt and authenticate the message, then node A uses Protocol 3 to obtain session key by negotiation.

Protocol 3 (Session Key Establishment Protocol). (1) Request: node A computes the sharing key K_{AB} with source node by Formula (7) then chooses a stochastic number K_{A} as session key. Node A sends a message M_1 to node B, which contains ID_{A} and K_{A} and is encrypted by K_{AB}. We denoted that $M_1 = (K_{\mathrm{AB}}\ (K_{\mathrm{A}}), \mathrm{ID}_{\mathrm{A}})$.

(2) Reply: node B computes K_{AB} by ID_{A} and decrypts M_1 to obtain K_{A}, then node B replies to node A with its own session key K_{B} which is encrypted by K_{AB}. M_2 is the message that node B sends to node A and $M_2 = (K_{\mathrm{AB}}\ (K_{\mathrm{B}}), \mathrm{ID}_{\mathrm{B}})$.

In Protocol 3, A and B send automatically their own session keys and the session keys are encrypted by the sharing key in the message.

Lemma 2. *Supported by Protocol 2, Protocol 3 accomplishes negotiating authentication key and communication encryption, which exclude the external attacks.*

Proof. Node authentication is accomplished by Master Key Generation Protocol. Authentication is implicitly included in key exchange process. Because external nodes can not get the key material (Coefficient of polynomial) from the operator (or an offline CA for the network) which distributes the key material (Coefficient of polynomial) before network deployment, malicious nodes can not achieve the session key sent by other nodes. Only legal nodes which are disposed by legal CA can join communication.

Secrecy is achieved by message encryption. It prevents all illegal leakage of message. As a result, violent attack is the only way to achieve content of message.

We can use a counter and add an index number to every message encoded by session key. Receiver can effectively prevent replay attack by checking the used counting value. □

From Lemma 2, the security architecture provides node authentication, communication encryption, and message authentication which excludes the attack from external nodes and makes external attack restricted to physical destroy. Physical attack usually can be divided into two forms. One is simple passive attack which makes nodes invalid; the other is physical capture and node replica which can launch active inside attack. However, higher cost of physical attack for sensor node in bad environment. Theorem 2 analyzes the precaution ability for physical attack of the topologies discovered by Algorithm 1.

Theorem 2. *If a network is deposed with sufficiently redundant density and supported by the security architecture defined in Protocols 2 and 3, then the network topologies discovered by Algorithm 1 are t-level node-failure tolerant, that is, it is unconditionally secure when captured nodes are no more than t if we have*

$$t = \min(t_1, t_2), \tag{11}$$

where the values of t_1 *and* t_2 *are given by Formula* (6) *and* (8).

Proof. Firstly, security architecture of protocol can effectively prevent external attack which is proved by Lemma 2. Now we consider only the physical attack from external force.

The physical attack can be divided into two forms. One is simple passive destroy, in which nodes are made invalid and intruder can not damage message confidentiality and network security. The worst influence of this attack is damaging the network connectivity, but it can be tolerated. We show the proof in deduction 1.

The other is physical capture in which the captured nodes are compromised to implement insider attack. A compromised node can take part in communication and achieve message in a cluster. If it is a grey dormant node, it is easy to be found by cluster head with data fusion when it forges sensor data. If it is the cluster head, its influence will be greatly increased from eavesdropping to injecting false sensor data. But the backbone network depends on multi-path data forwarding. If the number of captured nodes is not more than t, communication between normal nodes is still secure and the whole security structure works well. It is difficult to obtain $(t_2 + 1)$ undetermined coefficients in Formula

(8) when captured nodes are fewer than t_2. That is to say, backbone network depends on multipath data forwarding. If the number of captured nodes is not more than t, the confidentiality of network can not be damaged in cooperative sensor network.

So we conclude that the topology from Algorithm 1 is t node-failure tolerance where $t = \min(t_1, t_2)$. □

We should notice two points. Firstly, the value of t is determined by the degree of early symmetric bivariant polynomial and coefficient a_i in Formula (8). There is no linear dependence with network scale. Secondly, it is impossible to compromise many nodes by physical attack, because the cost of physical attack is even higher than the attack of large-scale wireless communication interference.

3. Topology Updating Method

In this section, we solve the problem of how to maintain a survivable topology when nodes are exhausted. The method can be divided into three steps. The first step is that cluster head serves on duty to balance energy cost in cluster. In the second step, new nodes join the network to prolong the survival time of each cluster; the third step balances energy between cluster by topology reconstruction.

3.1. Topology Updating and Topology Reconstruction. Energy of black running nodes will be consumed fast in data forwarding backbone network. After running a period of time, the energy of some running nodes is lower than a threshold. Then these black running nodes can not fit for running as cluster head. And new head is required to be selected in the clusters by Protocol 4.

Protocol 4 (Cluster Head (Black Node) Updating Protocol). (1) Black node A selects a grey node B which does not act as cluster head in range of r_1. Node B begins running as cluster head node.

(2) Node A broadcasts ID of node B in range of r in backbone network, and A claims that B is the head candidate of its cluster in the next period.

(3) Nodes in $\{\mathrm{ID}\}_{\mathrm{A}}$ (neighbor black node set of node A) decrypt the message from node A and preserve ID of node B.

(4) Node B broadcasts its own ID. If black nodes in its neighbor area do not only receive this message but also receive the message from node A, then these black nodes transmit their own ID for node B.

(5) Node B encrypts ID set of received black nodes $\{\mathrm{ID}\}_{\mathrm{B}}$ and transmits it to node A

(6) Node A checks black node set $\{\mathrm{ID}\}_{\mathrm{B}}$ from node B with its own set $\{\mathrm{ID}\}_{\mathrm{A}}$, the result has three conditions

(i) If $|\{\mathrm{ID}\}_{\mathrm{B}}|$ is less than a threshold, and node B has no enough direct neighbors to communicate with, that is to say, the topology can not meet the demand of node-failure tolerance, so we start topology reconstruction algorithm.

(ii) If $\{\mathrm{ID}\}_{\mathrm{B}} \not\subset \{\mathrm{ID}\}_{\mathrm{A}}$ and $\{\mathrm{ID}\}_{\mathrm{B}} \neq \{\mathrm{ID}\}_{\mathrm{A}}$, it is denotes that hostile nodes in node B or ($\{\mathrm{ID}\}_{\mathrm{B}}$-$\{\mathrm{ID}\}_{\mathrm{A}}$). if the suspicious hostile node is recorded, go to (1).

(iii) If $\{\mathrm{ID}\}_{\mathrm{B}} \subseteq \{\mathrm{ID}\}_{\mathrm{A}}$ and $|\{\mathrm{ID}\}_{\mathrm{B}}|$ is higher than a threshold, go to (8);

(7) Node A notifies nodes in set $\{\mathrm{ID}\}_{\mathrm{B}}$ that node B is running node in the next period. These nodes replace A with B in their own neighbor black node list;

(8) Node A transmits ID of neighbor nodes in range of r_1 to node B, transmits ID of node B to neighbor nodes, and announces that cluster head is updating success.

Cluster head updating protocol (Protocol 4) runs in the cluster formed by Algorithm 1 and it only requires 5 communications. It costs little energy and has little influence on the network life in global range in which old running node implements four broadcast communications. In Protocol 4, all communications are encrypted and authenticated with the security architecture provided by Protocol 2 and Protocol 3, in which key negotiation function between new cluster head and nodes is provided by Protocol 3.

After a period of running, the whole remaining energy is lower than a threshold and it is required to add new nodes to the network. For example, if there is no node with enough energy to act as cluster head, it is necessary to add new nodes. Protocol 5 gives the method of new nodes joining a specific cluster. The method uses key negotiation protocol based on symmetric bivariant polynomial and accomplishes node authentication, and the clock parameter in message prevents forwarding attack.

Protocol 5 (New Node Joining Protocol). (1) N is an arbitrarily selected node which is laid in second generation. N broadcasts its own ID in range of r_1.

(2) Black node A which receives broadcast message from node N sends message $((K_{\mathrm{A}})K_{\mathrm{AB}}, \mathrm{ID}_{\mathrm{A}}, \mathrm{S})$ to N, where S is a count of message.

(3) Node N computes K_{AB} by ID_{A}, decrypts K_{A} by K_{AB}, and then sends message $((K_{\mathrm{N}})K_{\mathrm{AB}}, \mathrm{ID}_{\mathrm{N}}, \mathrm{S})$ to node A.

(4) Node A adds ID and key of node N to grey neighbor node list.

From Protocol 4, obviously, if ($\{\mathrm{ID}\}_{\mathrm{A}}$-$\{\mathrm{ID}\}_{\mathrm{B}}$) is not empty, the neighbors of new cluster head B are less than those of old cluster head A, which make the ability of node-failure tolerance of topology drop. With several running of Cluster Head Updating Protocol, the ability of node-failure tolerance is lower than a setting threshold, so it is required to run topology self-generation algorithm for network reconstruction.

Because the cost of network reconstruction is very large, it is required to vote before running regeneration protocol. Topology regeneration protocol is run only if the number of agreement nodes reaches a given threshold.

Before starting topology regeneration protocol, grey nodes are initialized into white by their old cluster head. Then a reliable and trustable node is elected as protocol launcher. All the cluster heads send their blacklist of hostile nodes to the launcher and then become grey. Then the launcher runs topology regeneration algorithm.

Algorithm 2 (Topology Self-Regeneration Algorithm). (1) The launcher marks itself as black node.

(2) The new black node initiates Neighbor Query Protocol (Protocol 2) with radius r_1, dyes grey all its white neighbor nodes in radius r_1, excludes nodes in blacklist, and records the ID of the rest of grey nodes in grey neighbor node list

(3) The new black node initiates Neighbor Query Protocol with radius r_2. A node is selected as B from the white neighbor nodes in radius r_2 excluding nodes in blacklist. B is marked as black, and the new black node sends the blacklist of hostile nodes to node B, then goes to (2), until all white nodes can not meet the demand.

3.2. Key Updating. Grey nodes communicate with the nearest black node, join the cluster, and take the black node as cluster head. Then the cluster architecture is constructed. Cluster head (black node) sends the session key to grey nodes in its cluster, and all nodes in a cluster share a session key. Because there are many conditions such as topology updating, new nodes joining, and nodes being captured, it is necessary to update session key for forward security and backward security. We use the count threshold T to limit the number of messages encrypted by a session key and proactively update the session key by taking T as a cycle. Moreover, the session key may be changed after a node joins in or depart off.

Protocol 6 (Key Updating Protocol with Proactive and Responsive Model). (1) The black cluster head sustains a time counter S_j. It is the only order number connected with message.

(2) When the number of broadcast is bigger than the given threshold T (T is the cycle of message number for proactive key updating), we clear counters and update the session key.

(3) When the member of cluster node changed leads to update communication cluster, we clear counters and update the session key.

Theorem 3. *Security architecture provided by Protocols 2, 3, and 6 is fit for changing topology, and it has forward security and backward security.*

Proof. There are only two messages used in Protocol 2 when session key is established between a pair of nodes. It is convenient for newly joined node to set its own session key. It is fit for not only the initialization stage of topology establishment, but also the stages of new nodes joining network and the stages of topology updating and topology regeneration.

Session key is updated periodically as the number of message and topology changes. When node fails by physical attack or energy exhausted, session key is updated. Even if the intruder has the key of some captured nodes, he can not decrypt the message sent in the network for session key updating, so security architecture has backward security. When new nodes are added, session key is updated. Thus new nodes ca not decrypt foregoing message. As a result, the forward security is provided. □

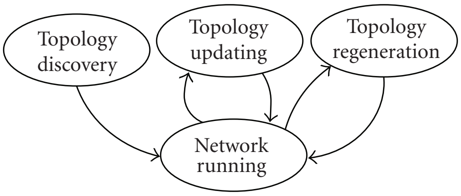


Figure 2: Phases of topology control.

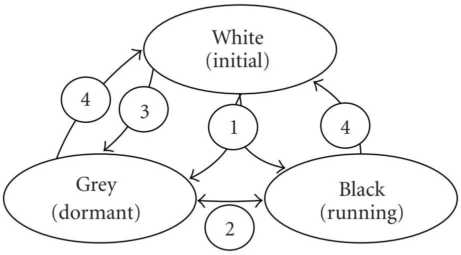


Figure 3: States of sensor nodes.

3.3. Topology Control Based on Regeneration. The topology control includes three phases: topology discovery, topology updating, and topology regeneration. Figure 2 shows the process. After the topology is discovered, a connected network is self-organized. Then the network turns into the running phase. After a period of network running, topology is updated to solve the problem of exhaustion, fault, and intrusion, then the network turns continuously into the phase of network running. When many nodes require to be updated, network begins reinitialization and uses topology regeneration algorithm with considering of running conditions. Then the regenerated network turns into the phase of second-generation network running.

There are three states for nodes in this paper, which are initial state, dormant state, and running state. These three states are denoted, respectively, by white, grey, and black in our tricolor-based topology control method. Figure 3 shows the operation process of node-state change in topology control algorithm: (1) denotes topology discovery phase, in which white original nodes turns into grey or black nodes in a cluster or connected structure; (2) denotes topology updating phase; in this phase, the running nodes will turn into dormant state, and a suitable node will be selected as running node; (3) denotes the phase that new nodes joining the network; after new nodes are deployed, they are in dormant state firstly; (4) denotes topology regeneration phase. When many cluster heads are required to be changed and updated, network begins reinitialization and runs topology regeneration algorithm then regenerates the second-generation connected network.

In the whole time of network running, new nodes are added as the supplements for energy exhaustion and failure nodes. They join the networks in white dormant state firstly,

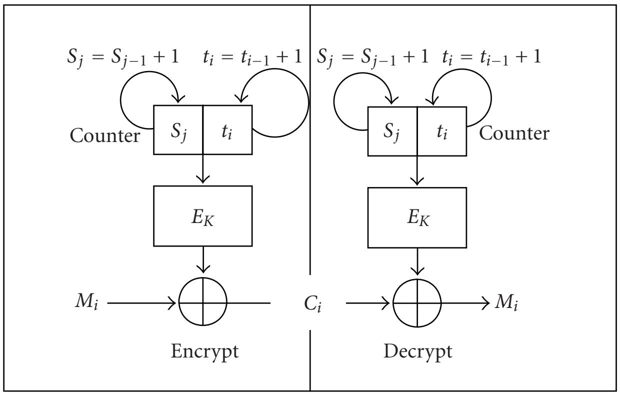


Figure 4: Clock operation model.

then they become grey or black by topology updating and topology regeneration.

4. Experimental Analysis and Related Work

The second section is devoted to solve how to securely form a topology with high node-failure tolerance, and the third section is devoted to use new nodes joining the network to sustain the node-failure tolerant topology. A security structure with identity authentication and security communication is given in Protocols 2, 3, and 5. Then some attributes of the security structure are proved in Theorems 1 and 3. Theorem 2 gives the capability of node-failure tolerance with the help of the security structure. This section analyzes the validity of topology control algorithm by simulation and introduces related work.

4.1. Experimental Analysis. In our experiment, the original value of ID belongs to $0 < \mathrm{ID} < 2^{16}$. The network has 500 nodes initially, then we add nodes 20 times, 250 nodes each time. Each node computes secret value in Formula (7) by Qin Jiushao's algorithm of Formula (12), which makes the times of multiplications reduce from $O(t^2/2)$ to $O(t)$

$$\begin{aligned} &a_t\mathrm{ID}^t + a_{t-1}\mathrm{ID}^{t-1} + \cdots + a_1\mathrm{ID} + a_0 \\ &\quad = (a_t\mathrm{ID} + a_{t-1})\mathrm{ID}^{t-1} + a_{t-2}\mathrm{ID}^{t-2} + \cdots + a_0 \\ &\quad\ \vdots \qquad\qquad\qquad\qquad \vdots \\ &\quad = (\cdots((a_t\mathrm{ID} + a_{t-1})\mathrm{ID} + a_{t-2})\mathrm{ID} + \cdots + a_1)\mathrm{ID} + a_0. \end{aligned} \tag{12}$$

For communication key update in cluster, a time counter S_j is sustained by the cluster head, in which the count value is dealt to zero-length by clock operation model in Figure 4, that is, the whole field for the encryption is not required for encoding the count value in each message, and communication bandwidth and energy consumption are saved.

We consider the relation between r_1/r_2 (it provides basis for value range of a_i in Formula (1)) and cluster head distribution. Figure 5 shows the number of cluster head changing with the value of r_1/r_2 when the number of node is 500 and 900, respectively. From Figure 5, the big curve slope with that the value of r_1/r_2 indicates that the value of r_1/r_2 significantly affects the number of cluster head when r_1/r_2 is small. And the low curve slope with the value of $r_1/r_2 > 0.5$ indicates the value of r_1/r_2 has little effect on the number of cluster head when $r_1/r_2 > 0.5$. Furthermore, two curves almost become one when the value of r_1/r_2 is more than 0.5, this indicates that the number of node has little effect on the number of cluster head when the value of r_1/r_2 is more than 0.5.

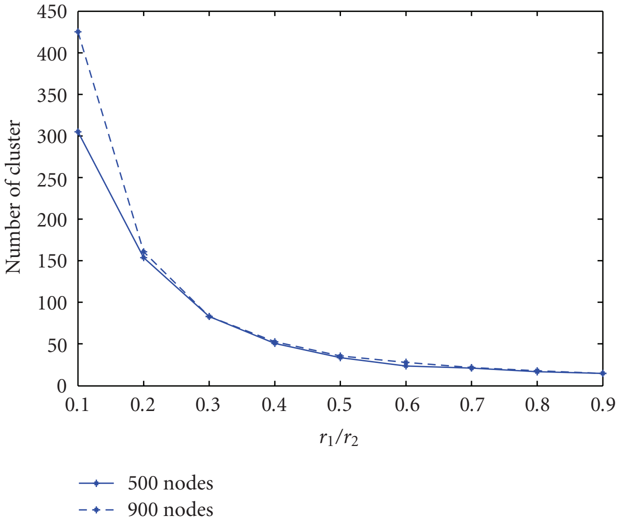
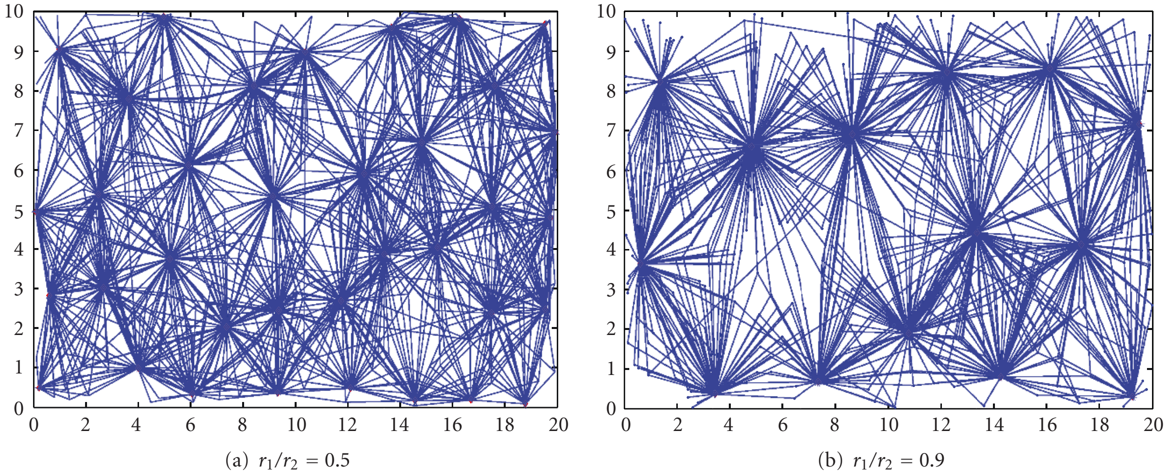


Figure 5: Relation between r_1/r_2 and number of cluster heads.

From Figure 5, we think that the value of r_1/r_2 should be bigger than 0.5, so the value of r_1/r_2 is set as 0.5 and 0.9, respectively in Figure 6. In Figure 6(a), we show cluster distribution in which 500 nodes are simulated, $r_1/r_2 = 0.5$, and cluster heads distribute well and it ensures better node-failure tolerance ability. But in Figure 6(b), 500 nodes are simulated with $r_1/r_2 = 0.9$, where the network is sparse and many cluster heads gather the edge zone. The distribution which some clusters have more nodes than the others has lower ability of node-failure tolerance. So we conclude that $r_1/r_2 = 0.9$ is too big for Algorithm 1. In following experiments, we let $r_1/r_2 = 0.6$.

We mainly study the effects of node distribution density on the number of cluster head in Figure 7. We begin the simulation with 500 nodes, then add 250 nodes each time, and note the number of cluster head. Figure 7 shows that the number of cluster head changes from 33 to 42, in which the changing range is very small compared with the change of the number of node from 500 to 8000. That is to say, the cluster number does not change obliviously with the node density. In Figure 7, there are 33 clusters with 2,500 nodes and 42 clusters with 3,000 nodes, in which the changing from 33 to 42 is caused by random distribution of nodes deployment. Figure 7 indicates that the number of cluster head is insensitive to node density in our algorithm. So our algorithm lets more nodes dormant and preserves energy of



(a) $r_1/r_2 = 0.5$

(b) $r_1/r_2 = 0.9$

Figure 6: Relation between r_1/r_2 and cluster head distribution

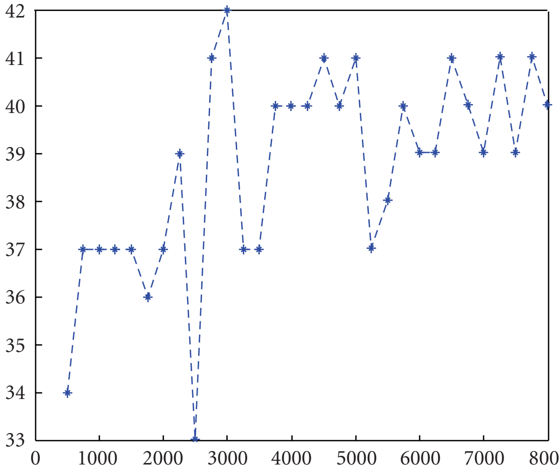


Figure 7: Relation between node density and number of cluster heads.

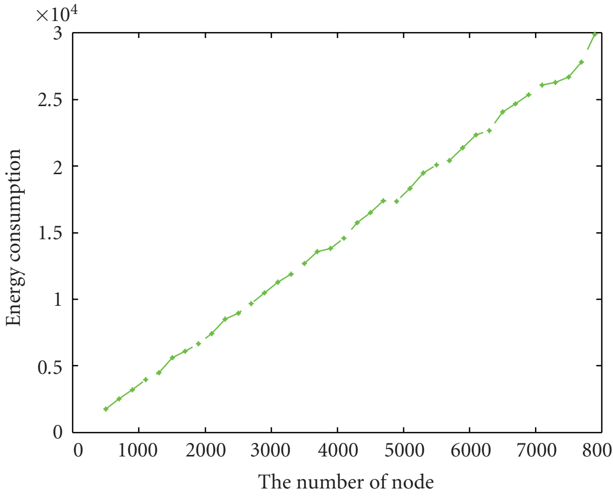


Figure 8: Relation between total energy consumption and number of nodes.

network with big density. It still discovers a sparse network topology with good node-failure tolerance.

We focus on the relation between total energy consumption and node deployment density in topology discovery algorithm. Figure 8 shows that energy consumption in topology discovery varies approximately linearly with the increase of node density. It indicates that topology discovery algorithm has better extending ability.

We compare energy consumption with security operator and without security operator by simulation in Figure 9. We find that the energy consumption is not obviously increased in running period when the nodes are running as cluster head. But the consumption increases obviously in original topology discovery period and cluster head update period. We think the increased energy consumption is caused by key negotiation and communication key update when topology is changing. But the security operation such as encryption and decryption consumes little energy. In simulation, we change cluster head after each node in the cluster that sends a message, and we pay our attention to the added energy consumption caused by security operation. Figure 9 shows that additional operation and extra communication only shorten 30% of the node life. Considering importance of security, the extra 30% cost is reasonable.

We compare the energy consumption in regeneration with and without the security algorithms in Figure 10. We calculate the rest energy in simulation in which we take 3 times node adding and one time topology regeneration. Figure 10 shows that total rest energy increases significantly with nodes adding. The detailed data show that energy consumption after new nodes joined is larger than the first 500 nodes in the same frequency of message exchanges. Figure 10 also shows that the security algorithm has effects on energy consumption. As we know, security operator costs large amount of energy when topology changes. So security operator has marked influence on energy consumption after deploying new nodes. Whether security operator is used, Figure 10 shows that topology regeneration can effectively

Table 1: Security protocol structure.

Related work	Reference	Security protocol structure		
		Mutual authentication	Broadcast authentication	Intrusion tolerance
Security algorithm	SPINS [12]	no	have	no
	Eschenauer and Gligor [13]	all	have	no
	Liu and Ning [14]	part	have	have
Research on intrusion tolerance	INSENS [15]	have	have	no
	Albert et al. [16]	No consider security structure		
	Wang et al. [9]			
	The paper	part	have	have

Table 2: Topology control method.

Related work	Reference	Topology control method				
		Fault/intrusion tolerance	Judgment basis	Discovery algorithm	Sparse network	Regeneration network
Fault-tolerant topology	Jia et al. [7]	Fault tolerance	Vertex connectivity	have	no	no
	Lee [8]		Edge connectivity	have	no	no
	ASCENT [17]		Communication signal	no	yes	have
	ART [18]		Multipath effects	have	res	no
Research on intrusion tolerance	INSENS [15]	Path intrusion tolerance	Available paths	no	yes	Path regeneration
	Albert et al. [16]	Intrusion tolerance	Residual connections	No consider practical problem		
	Wang et al. [9]		Available nodes			
	The paper		Available nodes	have	yes	have

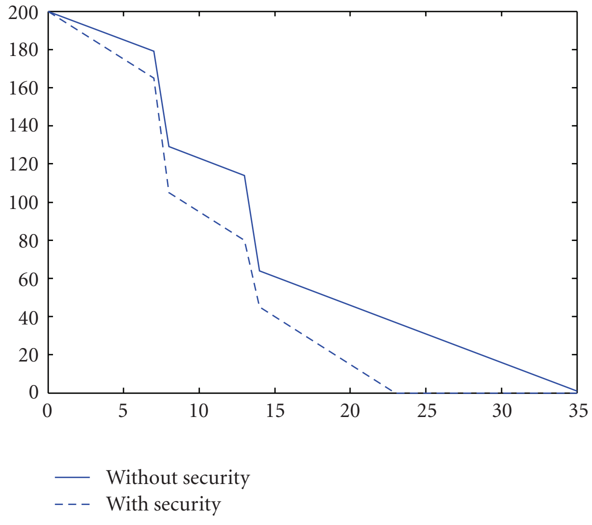


Figure 9: Security operator on rest energy.

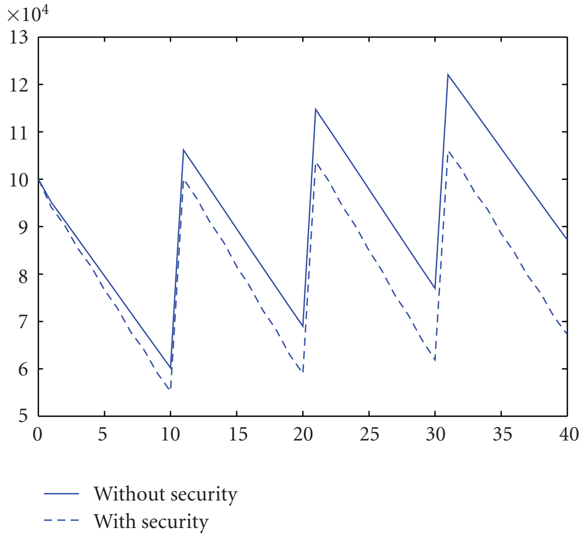


Figure 10: Regeneration for rest energy.

supply energy and prolong network life with adding nodes.

4.2. Related Work. Topology research is becoming a hot topic in wireless sensor network [18–20]. This section introduces some related works about security protocol structure, topology control with fault tolerance, and some intrusion tolerance and regeneration technology methods for wireless sensor network. Tables 1 and 2 give comparison of these related works.

On the aspect of security protocol structure in sensor network, SPINS presented by Perrig et al. [12] has large influence in this area. The security structure of SPINS is composed by SNEP protocol and μTESLA protocol, SNEP provides data security, data authentication each other, and data freshness; μTESLA provides broadcast authentication. But this security structure excessively depends on sink node which is assumed to be entirely secure. Moreover, it provides authentication only between nodes and sink node, there is no authentication and communication between nodes and nodes. Eschenauer and Gligor [13] present a random key predistribution method and accomplish successfully authentication between node and node, but it needs at least additional exchanged 32-byte information, and it has not the ability of intrusion tolerance. That is, if an intruder compromises a node, he confirms whether it shares the same polynomial communicating with other nodes. Furthermore, he can actively attack nodes with the same polynomial by this method. Liu and Ning [14] combines Blundo et al. [11] method and random key predistribution method [13]. Liu's method has certain ability of intrusion tolerance, and the method deploys more than one private polynomial in pool for each node. Each node has redundant identity by storing several polynomials. So node has the ability to resist possible attack. The disadvantage of this method is that the communication is limited in certain partial nodes. If the method provides a sharing pairwise key for every two nodes at least, it is the same as the method of established master key method in this paper.

On the aspect of fault tolerant topology discovery method, fault tolerance topology is often considered equivalent to a multiconnected graph. And the work for topology control tries to find an approximate algorithm for power control to meet k-vertex-connectivity [6] and k-edge-connectivity [7]. It is proved that the optimum multiconnected graph is an NP-hard problem. Moreover, the topology produced by these methods is often too dense. ASCENT [17] uses theory of dormancy and duty, which changes the number of active nodes in network according to communication signal. Thus it forms a relative sparse network. Hackmann et al. [18] proposed Adaptive and Robust Topology control with consideration of multipath effects in indoor environments, it can form sparse network and each node has enough robust link with the network. But these topology discovery methods did not consider security problem and authentication to distinguish intrusion node from ordinary inside nodes. That is to say, they do not consider about the nodefailure caused by intrusion.

Eariler document about intrusion tolerance of topology is written by Albert et al. [16], in which statistical analysis is used to discuss the tolerance ability of the topology in complex networks. INSENS [15] is the earlier document about intrusion tolerance in wireless sensor network, which mainly considers resisting attack and intrusion by redundant path. Wang et al. [9] and Albert et al. [16] studied the node-failure tolerance of topology in wireless sensor network based on available links, in which they considered tolerance ability of topology. But they did not present the methods to discover or maintain the high tolerant topology.

Regeneration technology [21–23] is called the fourth-generation security technology [22] after the third-generation survivability technology. The key problem of this regeneration technology is substitutable resources [23]. The prepared redundant paths are taken as substitutable resource for failure path in INSENS [15]. And the dormant nodes are regarded as regeneration resource to prolong network life in the field of WSN topology and communication [17, 21, 22]. In this paper, we take the newly added nodes as substitutable resource, and use these new nodes to form second-generation networks, increase rest energy, and prolong network life.

5. Conclusion

There are many failure nodes in wireless sensor networks disposed in hostile and unattended environments. These failure nodes are caused by random fault, malice intrusion, or exhausted battery. In this paper, we focus on the topology discovery and topology configuration of these unattended wireless sensor networks. Our goal is to achieve a topology with high node-failure tolerance with the existence of intrusion nodes and maintain a survivable topology by using the continuously deployed nodes.

Firstly, the paper proposes a secure topology control method, which generates a topology structure with high ability of node-failure tolerance. We employ a security algorithm in the method which implements mutual authentication, recognizes internal nodes, and excludes external nodes. So the outside intrusions are excluded. The security algorithm is proved to have the ability of intrusion tolerance. That is to say, the whole structure is still secure when several nodes are captured and compromised.

Secondly, a topology updating method is proposed, in which the newly deployed nodes are regarded as renewable resource to replace the failure nodes and charge energy. When the newly deployed nodes join the networks, they make the networks have sufficient redundancy to sustain the node-failure tolerant ability. And the joined nodes with nonused battery will prolong network life. It is proved that the security architecture has forward security and backward security, so it is still secure after new nodes join the networks or the exhausted or intruded nodes depart.

When considering geometry architecture of topology, we regard fault, intrusion, and energy exhausting as node failure. But in security operation, all the failure nodes are regarded as intrusion because that is the worst case. It is derived from theory that the presented method for topology control is node-failure tolerant. The simulation shows the influences on topology control method made by node density and dynamic radius proportion and shows the performance of the method in saving energy and prolonging network life. In the end, some related publications are compared and analyzed.

Acknowledgments

The first author is supported by China Scholarship for visiting scholarship under Grant 200832053, Special Funding

Scheme of China Postdoctoral Science Foundation under Grant no. 200801357, and "Qing Lan Project" for Excellent Youth Scholars of Jiangsu Province and Talents Foundation of Jiangsu University under Grant no. 07JDG080. The work of this paper is supported by National Natural Science Foundation of China (NSFC: 60703115, 60633020), China Postdoctoral Science Foundation (CPSF: 20070420955), and Postdoctoral Science Foundation of Jiangsu Province (0702003B).

References

[1] P. Santi, "Topology control in wireless ad hoc and sensor networks," *ACM Computing Surveys*, vol. 37, no. 2, pp. 164–194, 2005.

[2] M. Kubisch, H. Karl, A. Wolisz, L. C. Zhong, and J. Rabaey, "Distributed algorithms for transmission power control in wireless sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '03)*, pp. 558–563, New York, NY, USA, March 2003.

[3] N. Li, J. C. Hou, and L. Sha, "Design and analysis of an MST-based topology control algorithm," in *Proceedings of the 22nd Annual Joint Conference on the IEEE Computer and Communications Societies (IEEE INFOCOM '03)*, vol. 3, pp. 1702–1712, San Francisco, Calif, USA, March-April 2003.

[4] 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.

[5] 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.

[6] Y. Chen and S. H. Son, "A fault tolerant topology control in wireless sensor networks," in *Proceedings of the 3rd ACS/IEEE International Conference on Computer Systems and Applications*, pp. 269–276, Cairo, Egypt, January 2005.

[7] X. Jia, D. Kim, S. Makki, P.-J. Wan, and C.-W. Yi, "Power assignment for *k*-connectivity in wireless ad hoc networks," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM '05)*, vol. 3, pp. 2206–2211, Miami, Fla, USA, March 2005.

[8] H. Lee, "SEEMLESS: distributed algorithm for topology control of survivable energy efficient multihop wireless sensor networks using adjustable transmission power," in *Proceedings of the 6th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing and 1st ACIS International Workshop on Self-Assembling Wireless Networks (SNPD/SAWN '05)*, pp. 268–273, Towson, Md, USA, May 2005.

[9] L.-M. Wang, J.-F. Ma, C. Wang, and A. C. Kot, "Fault and intrusion tolerance of wireless sensor networks," in *Proceedings of the 20th International Parallel and Distributed Processing Symposium (IPDPS '06)*, pp. 7–14, Rhode Island, Greece, April 2006.

[10] R. Blom, "An optimal class of symmetric key generation systems," in *Proceedings of the workshop on Advances in Cryptology: Theory and Application of Cryptographic Techniques (EUROCRYPT '84)*, vol. 209 of *Lecture Notes in Computer Science*, pp. 335–338, Springer, 1985.

[11] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly-secure key distribution for dynamic conferences," in *Proceedings of the 12th Annual International Cryptology Conference (CRYPTO '92)*, vol. 704 of *Lecture Notes in Computer Science*, pp. 471–486, Springer, Santa Barbara, Calif, USA, August 1992.

[12] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "SPINS: security protocols for sensor networks," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MOBICOM '01)*, pp. 189–199, ACM Press, Rome, Italy, July 2001.

[13] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pp. 41–47, Washington, DC, USA, November 2002.

[14] D. Liu and P. Ning, "Establishing pairwise keys in distributed sensor networks," in *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS '03)*, pp. 52–61, ACM Press, Washington, DC, USA, October 2003.

[15] J. Deng, R. Han, and S. Mishra, "INSENS: intrusion-tolerant routing for wireless sensor networks," *Computer Communications*, vol. 29, no. 2, pp. 216–230, 2006.

[16] R. Albert, H. Jeong, and A.-L. Barbara, "Error and attack tolerance of complex networks," *Nature*, vol. 406, no. 6794, pp. 378–382, 2000.

[17] A. Cerpa and D. Estrin, "ASCENT: adaptive self-configuring sensor networks topologies," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pp. 272–285, 2004.

[18] G. Hackmann, O. Chipara, and C. Lu, "Robust topology control for indoor wireless sensor networks," in *Proceedings of the ACM Conference on Embedded Networked Sensor Systems (SenSys '08)*, November 2008.

[19] C.-F. Huang, Y.-C. Tseng, and H.-L. Wu, "Distributed protocols for ensuring both coverage and connectivity of a wireless sensor network," *ACM Transactions on Sensor Networks*, vol. 3, no. 1, article 5, 2007.

[20] Y. Mengjie, H. Mokhtar, and M. Merabti, "Fault management in wireless sensor networks," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 13–19, 2007.

[21] S. Parvin, D. S. Kim, and J. S. Park, "Towards survivable sensor networks using self-regenerative rejuvenation and reconfiguration," in *Proceedings of International Conference on Computational Intelligence and Security Workshops (CIS '07)*, pp. 546–549, Harbin, China, December 2007.

[22] K. M. M. Aung, K. Park, and J. S. Park, "Survivability analysis of a cluster system with 4th generation security mechanism: regeneration," *International Journal of Network Security*, vol. 3, no. 3, pp. 271–278, 2006.

[23] H. Chen, Y. B. Al-Nashif, G. Qu, and S. Hariri, "Self-configuration of network security," in *Proceedings of the 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC '07)*, pp. 97–108, IEEE Press, Annapolis, Md, USA, October 2007.