Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 973163, 11 pages
http://dx.doi.org/10.1155/2014/973163

Research Article

Localized and Energy-Efficient Topology Control in Wireless Sensor Networks Using Fuzzy-Logic Control Approaches

Yuanjiang Huang, José-Fernán Martínez, Vicente Hernández Díaz, and Juana Sendra

Centro de Investigación en Tecnologías Software y Sistemas Multimedia para la Sostenibilidad (CITSEM), Campus Sur Universidad Politécnica de Madrid (UPM), 28031 Madrid, Spain

Correspondence should be addressed to Yuanjiang Huang; yuanjiang@diatel.upm.es

Received 28 November 2013; Revised 2 April 2014; Accepted 4 April 2014; Published 24 April 2014

Academic Editor: Rongni Yang

The sensor nodes in the Wireless Sensor Networks (WSNs) are prone to failures due to many reasons, for example, running out of battery or harsh environment deployment; therefore, the WSNs are expected to be able to maintain network connectivity and tolerate certain amount of node failures. By applying fuzzy-logic approach to control the network topology, this paper aims at improving the network connectivity and fault-tolerant capability in response to node failures, while taking into account that the control approach has to be localized and energy efficient. Two fuzzy controllers are proposed in this paper: one is Learning-based Fuzzy-logic Topology Control (LFTC), of which the fuzzy controller is learnt from a training data set; another one is Rules-based Fuzzy-logic Topology Control (RFTC), of which the fuzzy controller is obtained through designing if-then rules and membership functions. Both LFTC and RFTC do not rely on location information, and they are localized. Comparing them with other three representative algorithms (LTRT, List-based, and NONE) through extensive simulations, our two proposed fuzzy controllers have been proved to be very energy efficient to achieve desired node degree and improve the network connectivity when sensor nodes run out of battery or are subject to random attacks.

1. Introduction

The advent of Wireless Sensor Networks (WSNs) [1] stimulates a tremendous number of applications, such as forest monitoring, factory automation, secure installation, and battlefield surveillance. Unlike other conventional network devices, the nodes in WSNs are more likely to be disconnected from each other. On the one hand, the sensor nodes are usually battery powered, so they are prone to loss connectivity due to battery depletion. On the other hand, the sensor nodes are subject to unpredictable node failures, for example, deployment in a hostile environment. The WSNs operate properly only when all nodes are reachable to each other. Therefore, one of the major concerns when planning WSNs is to make sure that all nodes in a network are, directly or indirectly, connected together.

In addition, the WSNs are expected to be able to tolerate a certain amount of node failures. From the graph theory point of view, the fault-tolerant problem is a *k-connected* network problem, where k indicates that there are at least k distinct paths from one node to any other node. A *k-connected* network is able to be constructed, maintained, or improved by means of topology control [2–4].

Furthermore, the energy issue is usually taken into account to make the sensor nodes functional as long as possible. Any algorithm or control system running in WSNs nodes is expected to be localized, because the cost of gathering global information is very time consuming and energy consuming.

In order to study the challenges mentioned above, this paper aims at developing an energy-efficient adaptive technique to improve the connectivity by means of adjusting the communication range under difficult events sensors may suffer from, such as battery depletion and malicious attacks. Some computational intelligence algorithms are applied to WSNs, such as fuzzy-logic, neural networks, and evolutionary algorithms [5]. The fuzzy-logic control is a very powerful technique that helps designers to construct a control system, regardless of the lack of mathematical models to completely describe network dynamics. More specifically, by using the fuzzy-logic controller to adaptively adjust the communication range of each node, this paper aims at maintaining desired node degree (namely, the number of neighbors a node has), thus improving network connectivity and fault-tolerant capability in response to node failures in WSNs, while at the same time taking into account that the control approach has to be localized and energy efficient. In this paper, in the case that the node degree is characterized by the mathematical model, the Learning-based Fuzzy-logic Topology Control (LFTC) is proposed to learn the dynamics to construct a fuzzy controller; we also propose another fuzzy controller, named Rules-based Fuzzy-logic Topology Control (RFTC), which is dependent on a heuristic approach to design the membership functions and if-then rules. LFTC and RFTC are both localized, because all information the fuzzy-logic controllers needed can be obtained from the node itself and its one-hop neighbor(s).

By comparing LFTC and RFTC with other three algorithms (Local Tree-based Reliable Topology (LTRT) [6], List-based topology control [7], and NONE) through extensive simulations, our two proposed fuzzy controllers are proved to be very energy efficient to improve network connectivity when node failure occurs (running out of battery and random attacks). First, due to the close-loop feedback of the control system, our two proposed approaches are able to trace the desired node degree as node density changes, while others are not able to do so; second, the average communication range, which is proportional to energy consumption, is lower than other algorithms, implying that our approaches are more energy efficient; third, our proposals are totally localized and the inputs that the fuzzy controller needs are very easy to obtain; fourth, the simulation results show that our approaches are able to respond to network dynamics, because the network is still able to maintain reasonable connectivity in the presence of random node failures. In short, our two proposed approaches are able to react to the network dynamics and outperform other three algorithms.

The main contributions of this paper are summarized as follows. (1) We have presented two control approaches based on the fuzzy logic to deal with network connectivity problem in WSNs. We have first presented the fuzzy controller based on the training dataset, called Learning-based Fuzzy-logic Topology Control (LFTC), and then proposed another fuzzy controller based on designing membership functions and if-then rules, called Rules-based Fuzzy-logic Topology Control (RFTC). (2) We have performed extensive simulations to compare LFTC and RFTC with other algorithms, and also the comparison between LFTC and RFTC was made. (3) The simulation results show that our two proposed fuzzy controllers are proved to be very energy efficient to achieve desired node degree and improve network connectivity when the sensor nodes run out of battery and suffer random attacks.

The rest of this paper is organized as follows. An introduction of related works is provided in Section 2. Section 3 presents our two fuzzy control approaches in detail. In Section 4, this paper evaluates the proposals by comparing them with other three representative algorithms. The potential applications of this work are discussed in Section 5. Section 6 concludes our work.

2. Related Works

The WSNs fault-tolerant and energy problems can be solved by means of appropriate topology control. For instance, deploy nodes in specific positions control the number of nodes deployed in the field or control communication range or transmission power of each node, and so forth. In a wireless environment, such as WSNs, adjusting communication range or transmission power is a very common approach. From the graph theory point of view, the fault-tolerant problems are *k-connected* network problems. Unfortunately, many of them are proved to be NP-complete or NP-hard problems even when k is very small, which means that the optimal solutions do not exist. For instance, the following optimization problems are NP hard: the minimum number of links required to obtain a 2-*connected* network [8], minimizing the power while maintaining *k-connected* network [9], minimizing the number of node placement for *k-connected* network [10], minimizing the number of relay nodes for 2-*connected* network [11, 12], and so forth. Therefore, heuristic algorithms are needed to obtain near-optimal performance, which is the main goal of this paper. Some of existing solutions are nonlocalized, so they are very unlikely to be applied to WSNs due to limited processing capability of WSNs nodes. Some of other solutions, such as Local Tree-based Reliable Topology (LTRT) [6] and List-based topology control [7], are two reprehensive heuristic localized approaches to achieve desired connectivity. Similar to LTRT and list-based topology control, our goals are to propose localized and energy-efficient control algorithms to manage the communication range, in order to maintain the network connectivity. Our proposals are evaluated through computer-based simulation and the simulation scenarios are relatively realistic, because we considered various network configurations that may influence the results, for example, energy dissipation model, undirected links, heterogeneous network, routing algorithm, random failures, and so forth.

From the graph theory point of view, the conditions of a network being *k-connected* have been studied. The upper bound of k is given by Menger's theorem [13]: $k \leq 2|E|/|V|$, where $|E|$ is the number of links and $|V|$ is the number of vertices in a graph. The upper bound can be achieved by constructing the Harary graph. Assuming that all nodes in a network are randomly and uniformly deployed, the asymptotical condition that a network is 1-*connected* with high probability is the node degree of each node at least bigger than $0.5193 \log n$ as $n \rightarrow +\infty$, where n is the total number of nodes in the network [14]. Once the network is 1-*connected* with high probability, it is also *k-connected* with high probability [14, 15].

Fuzzy control has been proved to be very effective to deal with complex nonlinear and time-varying systems, such as [16] which applies Takagi Sugeno (TS) fuzzy logic for

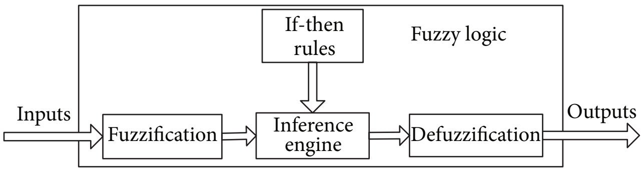


Figure 1: Fuzzy-logic control system.

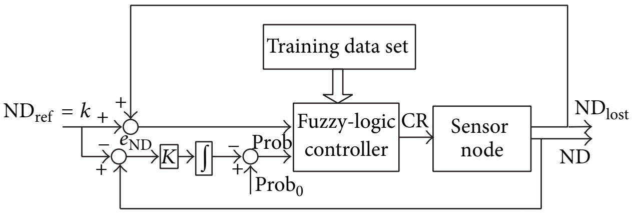


Figure 2: Learning-based Fuzzy-logic Topology Control System (LFTC).

electromagnetic suspension systems. Moreover, the fuzzy control can integrate with other control techniques; for example, [17] applies the fuzzy logic system to the sliding mode control and [18, 19] integrate the fuzzy control with the proportional integral derivative (PID) technique. Lyapunov theory is a common approach being used to analyze the stability of fuzzy controller [20, 21]. Some literatures try to obtain the optimal controller parameters and analyze the fuzzy controller's performance. Reference [22] analyzes the output feedback controller such that the closed-loop discrete time TS fuzzy systems with time-varying delays are asymptotically stable. Similarly, [23] calculates the asymptotic stability conditions for the state feedback fuzzy controller. In this paper, the fuzzy control is applied to address the challenges in WSNs and the numerical simulations are carried out to evaluate the proposed approaches.

The fuzzy linguistical input variables can be residual energy, concentration, and centrality [24]; remaining battery power, number of neighbors, distance from cluster centroid, network traffic [25]; residual energy and local distance [26]; distance to base station and residual energy [27]; energy and traffic load [28]; remaining energy, distance to base station and node density [29]; consistency, completeness, QoI from the lower level [30]. The fuzzy outputs could be the communication range, the competition range to be a cluster head, or the probability to be a cluster head. In this paper, the first proposal takes the node degree and the probability to have that node degree as inputs and the communication range as fuzzy controller output; in the second approach, we take the residual energy and the node degree error as inputs and the incremental of communication range as output.

3. WSNs Topology Control Using Fuzzy Logic

This paper focuses on how to design the fuzzy controller. Figure 1 shows a typical fuzzy-logic control system. The "fuzzification" transforms a crisp input variable (e.g., node degree) into a linguistic variable, for example, high, medium, and low; the "inference engine" maps the linguistic inputs onto linguistic outputs based on "if-then" rules; "defuzzification" converts the linguistic outputs of inference engine to crisp variables. Both linguistic input and output are represented by the membership functions.

Sometimes, a dynamical and distributed network is able to be characterized by mathematical forms, but it may happen that there are no ways to describe a network mathematically. For the former case, the fuzzy-logic membership functions and rules can be learnt from those mathematic descriptions; for the latter case, the fuzzy-logic controller could be established by a heuristic approach. In this paper, we leverage both approaches to construct fuzzy-logic controller to control the communication range of each node, with the aim that the network maintains desired node degree and improves the network fault-tolerant capability. Besides, the location information of sensor nodes is not needed in our fuzzy controllers, since it is not always available for sensor nodes.

In Section 3.1, we propose a learning-based fuzzy-logic controller based on neuroadaptive learning technique; in Section 3.2, a fuzzy-logic controller based on heuristic rules is proposed. They are both localized controllers, because all information the fuzzy-logic controllers needed can be obtained from the node itself and its one-hop neighbor(s). Throughout this paper, we use following abbreviations.

(i) ND: node degree or number of neighbors a node has.

(ii) ND_{ref}: reference ND or desired ND. In this paper, sometimes we also use k to represent ND_{ref}.

(iii) ND_{lost}: number of lost neighbor(s).

(iv) e_{ND}: node degree error $e_{ND} = ND - ND_{ref}$.

(v) CR: communication range.

(vi) E: node residual energy.

(vii) BS: base station.

3.1. LFTC: Learning-Based Fuzzy-Logic Topology Control. Our proposed Learning-based Fuzzy-logic Topology Control (LFTC) is a localized controller, because all parameters can be obtained locally by the node itself or one-hop neighbor(s).

3.1.1. Learning-Based Fuzzy-Logic Control System Design. Figure 2 shows the control system design of LFTC. It is a PI fuzzy-logic controller. Provided a training dataset, the fuzzy controller is obtained through neuroadaptive learning technique. On the one hand, each node can detect the number of lost neighbor(s) in the network. The fuzzy inputs are $ND_{ref} + ND_{lost}$ and the probability Prob that a node has $ND_{ref} + ND_{lost}$. On the other hand, Prob is controlled by an integral controller. The parameter K is set to 0.02 if $ND \geq ND_{ref}$ and 0.05 if $ND < ND_{ref}$. $Prob_0$ is 0.8.

3.1.2. Training Dataset. If sensor nodes in WSNs are randomly and uniformly distributed, its node degree distribution is a Poisson distribution [31]. The probability of a node having N neighbors is given by (1); therefore the probability that ND is bigger than k is (2), where r is the communication range and ρ is the node density which is defined as total nodes n in WSNs divided by the area of the deployment field A; namely, $\rho = n/A$. However, in practice the node degree distribution in WSNs possibly is non-Poisson distribution. For instance, based on a realistic radio channel fading model, [32] shows that the degree distribution in WSNs is approximated by a binomial distribution if the average node degree is low (e.g., less than 18). So, we can use (3) to represent the probability that ND is bigger than k, where $p(r)$ represents the probability of two nodes having a link at distance r, which is given in [32]. In addition, a mobility model of sensor nodes can be included in $p(r)$ [33]. It must be recalled that $\mathrm{ND}_{\mathrm{ref}} = k$ in this paper:

$$P_1(\mathrm{ND} = N) = \frac{\left(\rho\pi r^2\right)^N}{N!} e^{-\rho\pi r^2}, \tag{1}$$

$$P_1(\mathrm{ND} \geq k) = f_1(r,k) = 1 - \sum_{N=0}^{k-1} \frac{\left(\rho\pi r^2\right)^N}{N!} e^{-\rho\pi r^2}, \tag{2}$$

$$\begin{aligned} P_2(\mathrm{ND} \geq k) &= f_2(r,k) \\ &= 1 - \sum_{N=0}^{k-1} \binom{n-1}{N} p(r)^N (1 - p(r))^{n-1-N}. \end{aligned} \tag{3}$$

Equations (2) and (3) can be used to generate training dataset. The learning process can be performed on this training dataset afterwards. Define set $k = \{k_1, k_2, \ldots, k_s\}$, $r = \{r_1, r_2, \ldots, r_t\}$, and $p = \{p_1, p_2, \ldots, p_w\}$, where $w = s \cdot t$. Given node density ρ, according to (2), we have

$$p_{i \cdot j} = f_1\left(r_j, k_i\right). \tag{4}$$

Define a $w \times 3$ matrix $\mathbf{T}_{w\times 3}$, three elements at row m are formed by (5)

$$\mathbf{T}(m) = \left[k_i, p_{i\cdot j}, r_j\right], \tag{5}$$

where i, j are both integer, $0 < i \leq t$, $0 < j \leq s$, and $m = i \cdot j$. Similarly, the training data for binomial distribution can be obtained from (3).

The matrix $\mathbf{T}$ is used as training dataset where the fuzzy-logic controller can be learnt from. The learning technique employed in this paper is the adaptive neurofuzzy training provided by Matlab ANFIS tool. Depending on the network deployment, the node degree k could range from 1 to tens in order to obtain a wide range training data. Regarding variable r, it could range from several meters to hundreds meters, depending on real sensor devices.

The benefit of this approach is that there is no need to design the membership functions and if-then rules; instead the membership functions and rules are learnt from the training dataset.

3.2. RFTC: Rules-Based Fuzzy-Logic Topology Control

3.2.1. Rules-Based Fuzzy-Logic Topology Control System Design. In this section, we propose another fuzzy logic controller, called Rules-based Fuzzy-logic Topology Control (RFTC), because we need to design the if-then rules. Unlike LFTC, RFTC is shown in Figure 3(a). Here, the fuzzy controller of RFTC is not automatically generated from the training dataset, and the fuzzy rules and membership functions are generated by the heuristic approaches or experiences instead. The input parameters are different as well. The input variable probability Prob is replaced by residual energy E, and e_{ND} becomes input. The output is the CR incremental, $\Delta\mathrm{CR}$. CR_0 is the initial value of a sensor node, which is random for each node. This paper leverages "Mamdani" type fuzzy inference system.

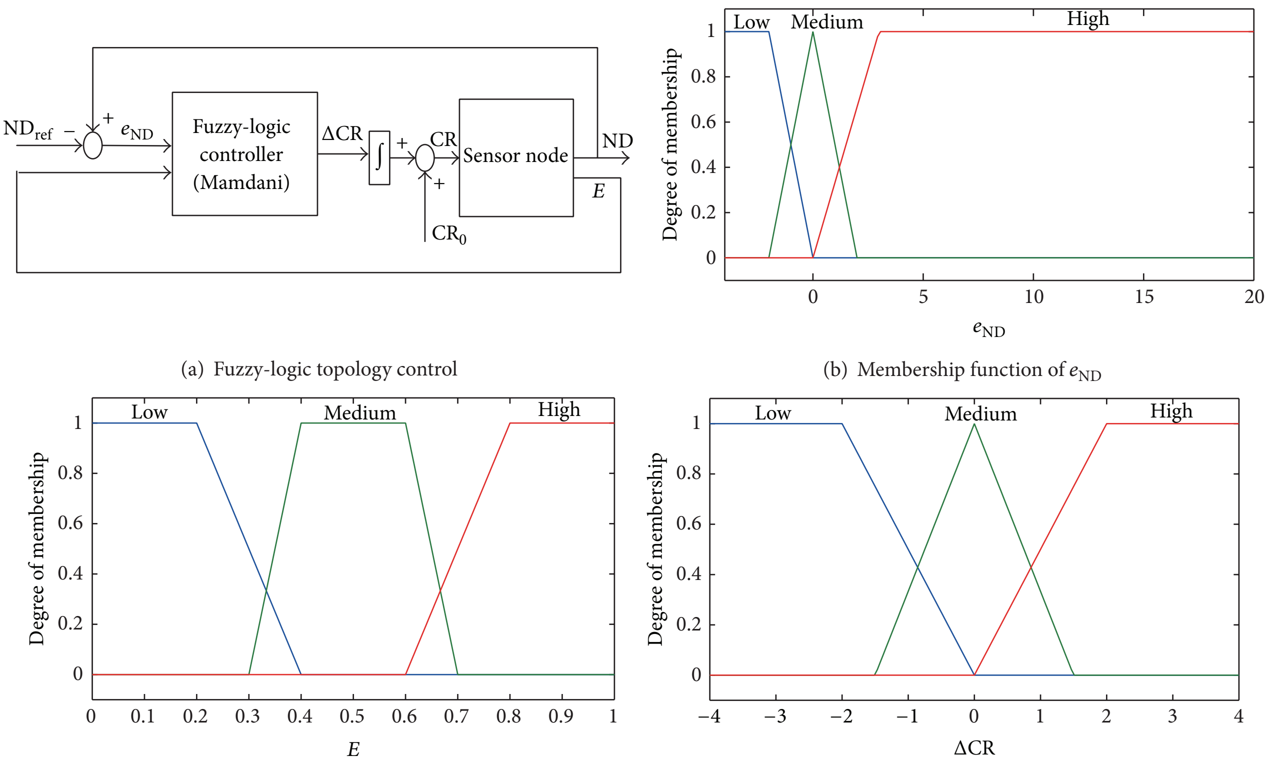
3.2.2. Membership Functions and If-Then Rules. In this paper, for each input and output, there are three fuzzy sets: high, medium, and low. Their membership functions are shown in Figures 3(b), 3(c), and 3(d). Intuitively, if e_{ND} is high and E is high, $\Delta\mathrm{CR}$ should be low; if e_{ND} is low, no matter what E is, $\Delta\mathrm{CR}$ should be High, because maintaining the network connectivity is the top priority. The details of if-then rules are shown in Table 1.

The design of membership functions and if-then rules is heuristic. The change of membership functions and if-then rules has significant impact on the performance. It is necessary to tune the membership shapes and positions and change the rules according to different network deployment strategies and network models.

4. Performance Evaluation

In this section, we evaluate our two fuzzy-logic approaches with other three localized algorithms by using Matlab. There are many topology control algorithms/protocols proposed in the literature. Some of the state-of-the-art topology control algorithms are Local Tree-based Reliable Topology (LTRT) [6], Local Minimum Spanning Tree (LMST) [34], and Fault-tolerant Local Spanning Subgraph (FLSS) [35], which are similar to each other as they are based on the spanning tree algorithm. In this paper, we choose LTRT as the representative algorithm in this category. On the other hand, the list-based topology control [7] is selected to represent the algorithm that does not rely on constructing the spanning tree but utilizes the neighbors' information. In addition, we also compare them with the one without any control approaches or algorithms, which means that the CR is not changed during the simulation. It is called NONE algorithm in this paper.

4.1. LTRT: Local Tree-Based Reliable Topology. Local Tree-based Reliable Topology (LTRT) [6] is a localized algorithm. Basically, it is a variant of spanning tree algorithm. When conducting spanning tree algorithm k times, the resultant network is a k-edge-connected if the original network is at least k-edge-connected network. More specifically, it repeatedly processes the network k times as follows: given a s-edge-connected network $G(V, E)$, where V is the set of nodes, E is the set of links, and $s \geq k$. First one of its spanning tree $T(V, \widehat{E}_1)$ is calculated by a localized algorithm, then all



(a) Fuzzy-logic topology control

(b) Membership function of e_{ND}

(c) Membership function of E

(d) Membership function of ΔCR

FIGURE 3: Rules-based Fuzzy-logic Topology Control (RFTC).

links in $\widehat{E}_1$ from E are removed and the resulting network is denoted as $G(V, E - \widehat{E}_1)$. Next time, the same process is performed on $G(V, E - \widehat{E}_1)$. Repeating this process k times, the resultant network will be $G(V, E - \widehat{E}_1 - \widehat{E}_2 - \cdots - \widehat{E}_k)$. The final k-edge-connectivity network is formed by combining all trees together; that is, $G(V, \widehat{E}_1 + \widehat{E}_2 + \cdots + \widehat{E}_k)$. The final CR of each node is selected from the maximum CR that connects to its neighbors in $G(V, \widehat{E}_1 + \widehat{E}_2 + \cdots + \widehat{E}_k)$.

The LTRT requires that the original network is at least a k-edge-connected network, and it requires the location information of its neighbors. LTRT needs that each node runs at its maximum CR before it starts running the algorithm. LTRT has been compared with Cone-Based distributed Topology Control CBTC(α) [36] and Fault-tolerant Local Spanning Subgraph ($FLSS_k$) [35]. $FLSS_k$ is a near optimal algorithm with high complexity. The simulation results of LTRT show that LTRT achieves comparable performance as that of $FLSS_k$, but at a much lower cost.

4.2. List-Based Topology Control. List-based topology control [7] is a cooperative algorithm. It is called list based because the change of CR relies on the list of its neighbors. Each node does not change its CR (increases or decreases) until its neighbors require its CR to be changed. In other words, each node is able to ask for its neighbors to change their CR when it needs them to do so. If a node has more neighbors than it needs, it will request the closer neighbors to change their CR, and other neighbors will remain their CR. For instance, if node u wants its ND to be 4, it broadcasts a request message. All nodes within its CR will receive the request, and they change their CR to reach node u. If there are more than 4 nodes that can reach u, only 4 nodes closer to u finally increase their CR, and other nodes will not modify their CR.

TABLE 1: Fuzzy-logic if-then rules.

e_{ND}	E		
	High	Medium	Low
High	Low	Medium	Medium
Medium	Medium	Medium	Low
Low	High	High	High

The list-based topology control is a localized algorithm, but it needs the location information of its neighbors as well, because the length of CR needed is calculated according to their location information.

4.3. Network Model and Configurations. Before starting the simulation, we first introduce the network model and configurations.

(i) Training dataset can be obtained according to different network models. In this paper, we employ the disk model, which means that CR is modeled as a disk with radius r. A link exists between two distinct nodes only when they are both in each other's CR; thus, all links are undirected. All nodes are randomly and uniformly deployed in a $100 \times 100\,\mathrm{m}^2$ field; therefore only (2), rather than (3), is used in the fuzzy-logic leaning process.

(ii) All nodes in the field are stationary after the deployment.

(iii) Each node is capable of adjusting its CR ranging from 10 m to 30 m. In addition, the initial CR of each node is a random value chosen from $[10, 30]$ m in order to simulate heterogeneous WSNs in terms of CR.

(iv) There is a special node in the network called base station (BS) located at the center of the field.

(v) Each node transmits sensor data to BS periodically. Each node updates its CR according to different control approaches or algorithms after transmitting 800000 bits packages. It is called one "round" simulation. Note that 800000 bits packages are not necessary to be transmitted at one time. They can be fragmented into many small packages.

(vi) The routing algorithm is the shortest distance to BS. The simulation is terminated when BS no longer receives packages.

(vii) The energy dissipation model is the same as [27]. Equations (6) and (7) represent power consumed when a node transmits/receives a L bits package to/from another node at distance d. Constant $E_{\mathrm{elec}} = 50\,\mathrm{nJ/bit}$ and $\epsilon_{\mathrm{amp}} = 100\,\mathrm{pJ/bit/m}^2$ are related to the circuit and antenna design of sensor nodes. Each node is charged with 1 J energy at the beginning of simulation. Nodes stop sending or receiving packages when there is no battery left:

$$E_{\mathrm{tx}} = L \times E_{\mathrm{elec}} + L \times \epsilon_{\mathrm{amp}} \times d^2, \qquad (6)$$

$$E_{\mathrm{rcv}} = L \times E_{\mathrm{elec}}. \qquad (7)$$

(viii) Apart from running out of battery, in order to simulate random attacks or damage by malicious people or nodes, a configurable parameter called failure probability is introduced in the simulation. Each node experiences identical failure probability at each round.

4.4. Simulations and Discussions. Simulations are divided into two parts. In the first part, we only consider the effect of the topology control approaches or algorithms on the initial network topology. In other words, we only observe the topology changes after all nodes are deployed without any data transmission in the network. Therefore, some configurations in Section 4.3 are not applicable to this simulation part, such as the energy dissipation model, the routing algorithm, and the failure model. In the second part, we simulate the networks with packages being sent at each node. The main differences between two parts are in the second simulation part, the energy of each node will be decreasing, and some of the nodes may run out of battery during the simulations. Particularly, the relay nodes deplete energy faster. As a result, the links between nodes are dynamic. Because E is one of the inputs for RFTC, the energy status has an impact on the controller output. Besides, the failure probability also influences the connections among nodes. Therefore, the second part simulation is a more dynamic scenario.

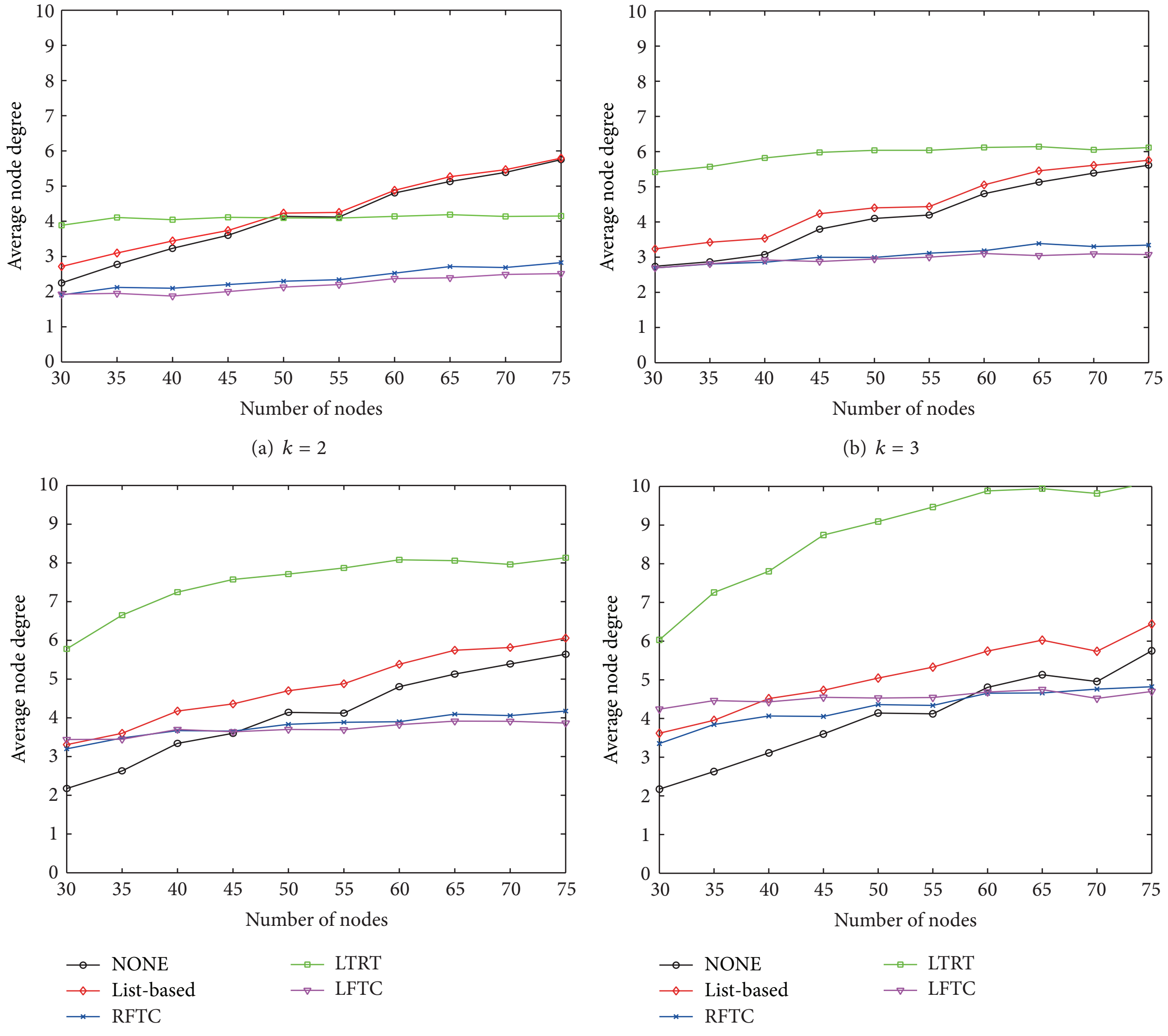
For each part, the deployment area is fixed but the number of nodes deployed varies from 30 to 75 nodes to change the node density. Since the deployment is random, 50 different networks are generated for every algorithm with the same configurations (e.g., same number of nodes and failure probability). Obtained results are the average of 50 networks.

4.4.1. Topology Control on Initial Network. Figures 4(a), 4(b), and 4(c) show the average node degree, which is calculated by the sum of the node degrees of all nodes divided by the number of nodes in a network. As mentioned in Section 4.3, the link between two nodes is undirected. A node connected by a directed link is not counted as a neighbor. We observe that our proposed two approaches, LFTC and RFTC, are able to trace the reference $k = 2, 3, 4$ as the number of nodes deployed in the field increases. But Figure 4(d) shows that the network is unable to trace $k = 5$ when the node number is less than 60. Because the maximum CR is limited, it is less likely that each node has node degree at least 5 if the network density is not high enough. On the contrary, other algorithms are unable to trace the desired k. LTRT has the highest ND, because it is the most aggressive algorithm, which needs each node to run at maximum CR before it starts running LTRT. Higher average ND is good for the network connectivity but also introduces higher signal inferences.

It is worth noting that LFTC and RFTC are very close to each other in Figures 4(a), 4(b), and 4(c), but in Figure 4(d) LFTC demonstrates better performance at tracing the desired k than RFTC when the nodes deployed are less than 60. LFTC manifests better adaptivity than RFTC in response to the network dynamics. As we mentioned in Section 3.2, RFTC has to tune the control parameters to adapt the network dynamics, but the control parameters are the same for RFTC in the simulations.

As shown in (6), energy consumption is propositional to the squared distance between a transmitter and a receiver. Similar to the way of evaluating LTRT [6], we use average CR ($\mathrm{CR}_{\mathrm{avg}}$), average maximum CR ($\mathrm{CR}_{\mathrm{max}}$), and Energy Expended Ratio (EER) (defined as $\mathrm{EER} = 100 \times (\mathrm{CR}_{\mathrm{avg}}/\mathrm{CR}_{\mathrm{max}})\%$) to evaluate network energy consumption. Figures 5, 6, and 7 illustrate the average CR and average maximum CR, and its EER when $k = 3$, respectively. The $\mathrm{CR}_{\mathrm{avg}}$ of LFTC, RFTC, and LTRT decreases as the number of nodes increases, but the network running list-base algorithm and NONE, the $\mathrm{CR}_{\mathrm{avg}}$ does not change to a great extent. They are expected results because

(1) for LFTC, RFTC, and LTRT, if there are more nodes deployed, it indicates that the node density becomes higher. Therefore, the lower CR can obtain desired k neighbors. In other words, energy consumption



(a) $k = 2$

(b) $k = 3$

(c) $k = 4$

(d) $k = 5$

Figure 4: Average node degree with different k.

is higher when the network is sparse, while energy consumption is lower when the network is dense,

(2) for the algorithm NONE, because the CR does not change and the initial CR randomly chosen from [10, 30] m; therefore its $\mathrm{CR}_{\mathrm{avg}}$ is always about 20 m,

(3) for the list-base algorithm, a node can ask for other nodes, which are not its neighbors but within its CR, to increase their CR, but the maximum CR between any two nodes is limited by maximum CR between them. For instance, nodes u and v have communication ranges of 15 m and 20 m, respectively. The node v may need u to increase CR to reach v; however, the maximum between u and v is impossible bigger than maximum between them, that is, 20 m. In other words, the node CR can increase but the incremental is limited.

As far as the average energy is concerned, we conclude that the energy consumption of LFTC and RFTC is always lower than LTRT regardless of the node density, and LFTC and RFTC outperform list-based and NONE after the number of nodes higher than 40. LFTC is slightly better than RFTC. As far as the most power consumption node is taken into account, as shown in Figure 6, LFTC has the lowest maximum CR. It indicates that the most power consuming node running LFTC in a network has the lowest power consumption than running other algorithms. EER in Figure 7 shows the same trend as Figure 5. We expect that EER is low. RFTC maintains lowest EER than other algorithms. LFTC is higher than RFTC due to the maximums being lower than RFTC, as shown in Figure 6.

In short, the simulations performed in this section only focus on the network connectivity and corresponding energy

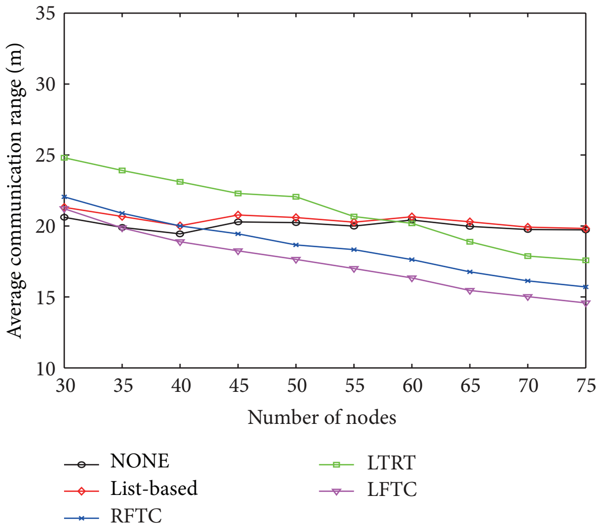


Figure 5: Average communication range ($k = 3$).

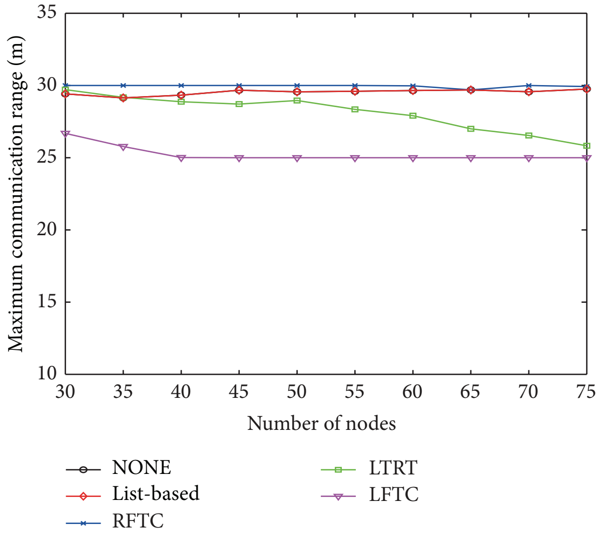


Figure 6: Maximum communication range ($k = 3$).

consumption. The simulation results imply that our two proposals (LFTC and RFTC) are able to maintain the desired node degree, which perfectly shows the effectiveness of the feedback control loops, while the resulting node degree are higher than expected when using other conventional methods. On the other hand, from the energy consumption point of view, our proposals manifest lower energy consumption on the resulting networks than other algorithms.

4.4.2. Topology Control on Network with Random Failures. In this section, we evaluate the network performance when the nodes send and receive sensor data periodically. Moreover, we consider not only nodes running out of battery, but also nodes that are damaged on purpose. The node damage is modeled by introducing random failures at each round. In the simulation, there are 60 nodes deployed in the field. Since each node leaves the network randomly at each round, $\mathrm{ND}_{\mathrm{avg}}$ is calculated in a different way. For any node i in the network, let ND at round j be denoted as $\mathrm{ND}(i, j)$, and let $\mathrm{round}(i)$ be the number of rounds before node i runs out of battery or is attacked. Average ND of node i, denoted as $\mathrm{ND}_{\mathrm{avg}}(i)$, is calculated by the sum of ND at each round divided by the number of rounds for node i, as shown in (8). The $\mathrm{ND}_{\mathrm{avg}}$ of a specific network is the average of $\mathrm{ND}_{\mathrm{avg}}(i)$ for all nodes, as shown in (9), where n is the total number of nodes deployed:

$$\mathrm{ND}_{\mathrm{avg}}(i) = \frac{\sum_{j=1}^{\mathrm{round}(i)} \mathrm{ND}(i, j)}{\mathrm{round}(i)}, \tag{8}$$

$$\mathrm{ND}_{\mathrm{avg}} = \frac{\sum_{i=1}^{n} \mathrm{ND}_{\mathrm{avg}}(i)}{n}. \tag{9}$$

Figure 8 shows the $\mathrm{ND}_{\mathrm{avg}}$ with failure probabilities 0%, 4%, 8%, and 12%. First of all, $\mathrm{ND}_{\mathrm{avg}}$ decreases for all algorithms, because the failure probability increases. Second, $\mathrm{ND}_{\mathrm{avg}}$ of LFTC and RFTC decreases slower than others (especially when failure probabilities are 4% and 8%), implying that our proposed approaches are able to effectively resist $\mathrm{ND}_{\mathrm{avg}}$ (or, network connectivity) decreasing as the failure probability increases. RFTC outperforms LFTC, but both worse than LTRT. Third, $\mathrm{ND}_{\mathrm{avg}}$ of list-base and NONE is very close, while LTRT still has the highest $\mathrm{ND}_{\mathrm{avg}}$.

In this section, the simulations are performed on the more realistic scenarios, where nodes transmitting, receiving, and routing packages, meanwhile considering that nodes are experiencing running out of battery and malicious attacks. The results indicate that the control approaches in our proposals are still valid. The performance degradation is slower than other algorithms in the case that random failures occur.

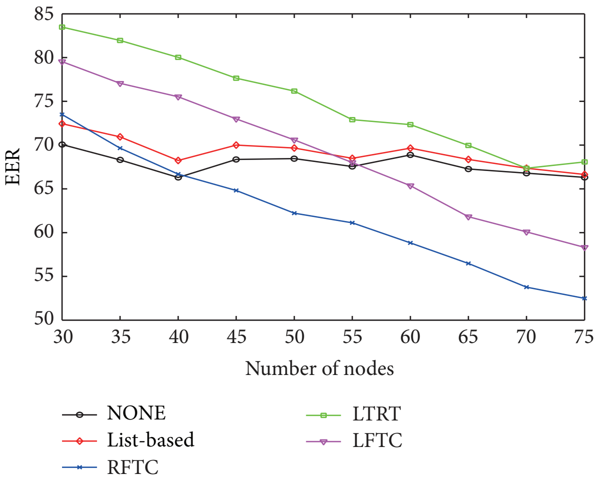


Figure 7: Energy expended ratio (EER) ($k = 3$).

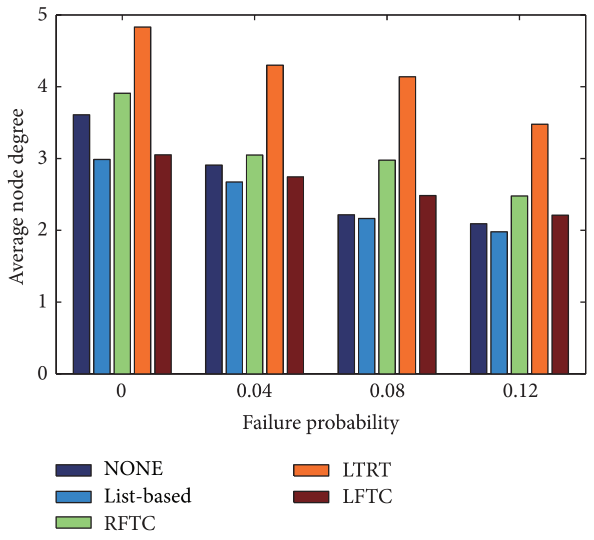


Figure 8: Average node degree for different failure probabilities ($k = 3$, $n = 60$).

5. Potential Applications

In this section, we discuss some potential applications, including fault-tolerant network topology design, sensor nodes power management, and routing protocol design.

5.1. Fault-Tolerant Network Topology Design. By applying the control approaches proposed in this paper, the WSNs are able to tolerate the desired amount of node failure(s). Literature [14] proved that the network will be asymptotical *k-connected* with high probability, provided that the node degree of each node is at least bigger than $0.5193 \log n$, where n is the total number of nodes in the network. If a network is *k-connected*, then it means that the network can tolerate at most $k-1$ node failure(s), and the remaining network is still connected. For example, by configuring the desired node degree to be 6 in our control systems, our control approaches will automatically achieve this node degree through adaptively adjusting the communication range of each node, and the network will be able to keep connected as long as the number of node failures is no more than 5.

5.2. Sensor Nodes Power Management. In a real system, the communication range usually cannot be directly changed; instead the power transmission is the parameter that can be modified. The higher power transmission is, the longer communication range is, and also the higher energy consumption will be. The energy is one of the significant resources that WSNs nodes need to preserve. In the RFTC proposed in this paper, the energy is one of the controller inputs. Therefore, by appropriately defining the membership functions and if-then rules, the tradeoff between network connectivity and energy-efficiency can be fulfilled.

5.3. Routing Protocol Design. As the energy consumption is one of the most critical resources for battery-powered sensor nodes, an important goal in WSNs design is to maximize the lifetime of the network by choosing a routing path which consumes lower energy (such as Energy-Balanced Routing Protocol (EBRP) [37]), or by using energy efficient distributed algorithms (e.g., [38]). However, the routing algorithms/protocols usually do not change the communication range or power transmission itself, so our proposals possibly can be integrated with those energy-aware algorithms/protocols. For instance, EBRP forces packets to move toward the sink through the dense energy area, so our control approaches could be employed to balance the distribution of energy density in order to further extend the lifetime of the network.

6. Conclusions and Future Works

In order to improve the network connectivity of WSNs in the presence of node failures, this paper proposed two localized and energy-efficient approaches, called LFTC and RFTC, based on the fuzzy logic. As for LFTC, the fuzzy-logic controller is obtained through the training dataset, while the fuzzy controller is based on the heuristic if-then rules and membership functions for RFTC. However, both approaches can achieve almost the same goals. LFTC and RFTC both have strengths and weaknesses. The main benefit of LFTC is that it relies on a mathematical model, so there is no need to adjust fuzzy controller parameters, but the mathematical model may be not available or accurate enough for realistic sensor nodes deployment. In contrast, the control parameters of RFTC do not depend on the mathematical model, but the parameters have to be tuned according to specific node deployment to achieve best performance. It could be not an easy task, or even not feasible.

Compared LFTC and RFTC with other three algorithms NONE, LTRT, and list based by extensive simulations, our two proposals can achieve desired node degree and save more energy. Furthermore, in the case that random node failures exist, such as nodes running out of battery or suffering intentional attacks, LFTC and RFTC show their capability to resist node failures by adaptively adjusting the communication range. In our simulations, we employ the disk model which is an ideal wireless channel. Nonetheless, our approaches can be extended to more realistic models as long as the node degree is known, for example, binomial distribution [32]. The only difference is the way to obtain the training dataset.

In this paper, all nodes in the network are stationary once deployed, but it is possible that the nodes in WSNs can be relocated from one place to another. In addition, it is desirable to deploy our fuzzy controllers in real sensor nodes running in a harsh environment. The mobility and implementation in real sensor nodes will be considered in our future works.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the European project “Design, Monitoring, and Operation of Adaptive Networked Embedded Systems” (DEMANES) (Project code ARTEMIS-JU: 295372) and “Ministerio de Industria, Energía y Turismo” of Spain (Project code ART-010000-2012-2).

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[2] A. A. Aziz, Y. A. Sekercioglu, P. Fitzpatrick, and M. Ivanovich, “A survey on distributed topology control techniques for extending the lifetime of battery powered wireless sensor networks,” *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 121–144, 2013.

[3] C. Zhu, C. Zheng, L. Shu, and G. Han, “A survey on coverage and connectivity issues in wireless sensor networks,” *Journal of Network and Computer Applications*, vol. 35, no. 2, pp. 619–632, 2012.

[4] M. Younis, I. F. Senturk, K. Akkaya, S. Lee, and F. Senel, “Topology management techniques for tolerating node failures in wireless sensor networks: a survey,” *Computer Networks*, vol. 58, no. 15, pp. 254–283, 2014.

[5] R. V. Kulkarni, A. Förster, and G. K. Venayagamoorthy, “Computational intelligence in wireless sensor networks: a survey,” *IEEE Communications Surveys & Tutorials*, vol. 13, no. 1, pp. 68–96, 2011.

[6] K. Miyao, H. Nakayama, N. Ansari, and N. Kato, “LTRT: an efficient and reliable topology control algorithm for ad-hoc networks,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 12, pp. 6050–6058, 2009.

[7] P. Costa, M. Cesana, S. Brambilla, and L. Casartelli, “A cooperative approach for topology control in wireless sensor networks,” *Pervasive and Mobile Computing*, vol. 5, no. 5, pp. 526–541, 2009.

[8] M. Abellanas, A. García, F. Hurtado, J. Tejel, and J. Urrutia, “Augmenting the connectivity of geometric graphs,” *Computational Geometry*, vol. 40, no. 3, pp. 220–230, 2008.

[9] M. Hajiaghayi, N. Immorlica, and V. S. Mirrokni, “Power optimization in fault-tolerant topology control algorithms for wireless multi-hop networks,” *IEEE/ACM Transactions on Networking*, vol. 15, no. 6, pp. 1345–1358, 2007.

[10] X. Han, X. Cao, E. L. Lloyd, and C.-C. Shen, “Fault-tolerant relay node placement in heterogeneous wireless sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 9, no. 5, pp. 643–656, 2010.

[11] A. Kashyap, S. Khuller, and M. Shayman, “Relay placement for higher order connectivity in wireless sensor networks,” in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM ’06)*, pp. 1–12, April 2006.

[12] B. Hao, J. Tang, and G. Xue, “Fault-tolerant relay node placement in wireless sensor networks: formulation and approximation,” in *Proceedings of the Workshop on High Perfomance Switching and Routing (HPSR ’04)*, pp. 246–250, April 2004.

[13] U. S. R. Murty and J. A. Bondy, *Graph Theory*, Springer, 2008.

[14] M. R. Brito, E. L. Chávez, A. J. Quiroz, and J. E. Yukich, “Connectivity of the mutual k-nearest-neighbor graph in clustering and outlier detection,” *Statistics & Probability Letters*, vol. 35, no. 1, pp. 33–42, 1997.

[15] Y. Huang, J. F. Martnez, J. Sendra, and L. López, “The influence of communication range on connectivity for resilient wireless sensor networks using a probabilistic approach,” *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 482727, 11 pages, 2013.

[16] X. Su, X. Yang, P. Shi, and L. Wu, “Fuzzy control of nonlinear electromagnetic suspension systems,” *Mechatronics*. In press.

[17] M. R. Soltanpour, B. Zolfaghari, M. Soltani, and M. H. Khooban, “Fuzzy sliding mode control design for a class of nonlinear systems with structured and unstructured uncertainties,” *International Journal of Innovative Computing, Information and Control*, vol. 9, no. 7, pp. 2713–2726, 2013.

[18] X.-G. Duan, H. Deng, and H.-X. Li, “A saturation-based tuning method for fuzzy PID controller,” *IEEE Transactions on Industrial Electronics*, vol. 60, no. 11, pp. 5177–5185, 2013.

[19] J. L. Meza, V. Santibáñez, R. Soto, and M. A. Llama, “Fuzzy self-tuning PID semiglobal regulator for robot manipulators,” *IEEE Transactions on Industrial Electronics*, vol. 59, no. 6, pp. 2709–2717, 2012.

[20] D. H. Lee, J.-B. Park, and Y.-H. Joo, “A new fuzzy lyapunov function for relaxed stability condition of continuous-time takagi-sugeno fuzzy systems,” *IEEE Transactions on Fuzzy Systems*, vol. 19, no. 4, pp. 785–791, 2011.

[21] K. Guelton, T.-M. Guerra, M. Bernal, T. Bouarar, and N. Manamanni, “Comments on fuzzy control systems design via fuzzy lyapunov functions,” *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 40, no. 3, pp. 970–972, 2010.

[22] X. Su, P. Shi, L. Wu, and Y.-D. Song, “A novel control design on discrete-time Takagisugeno fuzzy systems with time-varying delays,” *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 4, pp. 655–671, 2013.

[23] Y. Zhao, R. Zhang, and C. Shao, “Estimating delay bounds of time-delay systems wiht fuzzy model-based approach,” *International Journal of Innovative Computing, Information and Control*, vol. 9, no. 11, pp. 4343–4357, 2013.

[24] I. Gupta, D. Riordan, and S. Sampalli, “Cluster-head election using fuzzy logic for wireless sensor networks,” in *Proceedings of the 3rd Annual Communication Networks and Services Research Conference*, pp. 255–260, May 2005.

[25] H. Ando, E. Kulla, L. Barolli, A. Durresi, F. Xhafa, and A. Koyama, “A new fuzzy-based cluster-head selection system for WSNs,” in *Proceedings of the 5th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS ’11)*, pp. 432–437, July 2011.

[26] J.-M. Kim, S.-H. Park, Y.-J. Han, and T.-M. Chung, “CHEF: cluster head election mechanism using fuzzy logic in wireless sensor networks,” in *Proceedings of the 10th International Conference on Advanced Communication Technology*, pp. 654–659, February 2008.

[27] H. Bagci and A. Yazici, “An energy aware fuzzy approach to unequal clustering in wireless sensor networks,” *Applied Soft Computing*, vol. 13, no. 4, pp. 1741–1749, 2013.

[28] I. AlShawi, L. Yan, W. Pan, and B. Luo, “Lifetime enhancement in wireless sensor networks using fuzzy approach and a-star algorithm,” *IEEE Sensors Journal*, vol. 12, no. 10, pp. 3010–3018, 2012.

[29] S. Mao, C. Zhao, Z. Zhou, and Y. Ye, “An improved fuzzy unequal clustering algorithm for wireless sensor network,” in *Proceedings of the 6th International ICST Conference on Communications and Networking in China (CHINACOM ’11)*, pp. 245–250, August 2011.

[30] E. de Cristofaro, J.-M. Bohli, and D. Westhoff, "FAIR: fuzzy-based aggregation providing in-network resilience for real-time wireless sensor networks," in *Proceedings of the 2nd ACM Conference on Wireless Network Security (WiSec '09)*, pp. 253–260, March 2009.

[31] C. Bettstetter, "On the minimum node degree and connectivity of a wireless multihop network," in *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking & Computing*, pp. 80–91, June 2002.

[32] R. Hekmat and P. van Mieghem, "Degree distribution and hopcount in wireless ad-hoc networks," in *Proceedings of the 11th IEEE International Conference on Networks*, pp. 603–609, 2003.

[33] L. Guo, H. Xu, and K. Harfoush, "The node degree for wireless ad hoc networks in shadow fading environments," in *Proceedings of the 6th IEEE Conference on Industrial Electronics and Applications (ICIEA '11)*, pp. 815–820, June 2011.

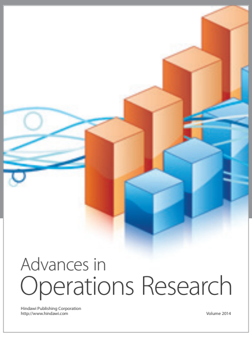
[34] N. Li, J. C. Hou, and L. Sha, "Design and analysis of an MST-based topology control algorithm," in *Proceedings of the INFOCOM IEEE 22nd Annual Joint Conference on the IEEE Computer and Communications Societies*, vol. 3, pp. 1702–1712, April 2003.

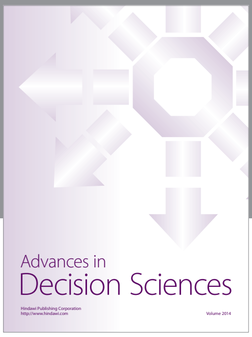
[35] N. Li and J. C. Hou, "Localized fault-tolerant topology control in wireless ad hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 4, pp. 307–320, 2006.

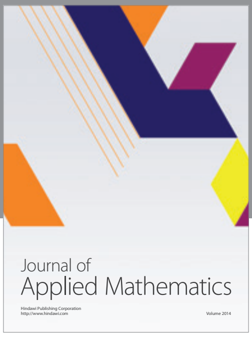
[36] L. Li, J. Y. Halpern, P. Bahl, Y.-M. Wang, and R. Wattenhofer, "Analysis of a cone-based distributed topology control algorithm for wireless multi-hop networks," in *Proceedings of the 20th Annual ACM Symposium on Principles of Distributed Computing*, pp. 264–273, August 2001.

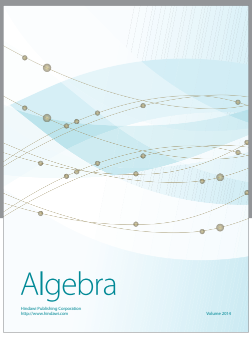
[37] F. Ren, J. Zhang, T. He, C. Lin, and S. K. D. Ren, "EBRP: energy-balanced routing protocol for data gathering in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 12, pp. 2108–2125, 2011.

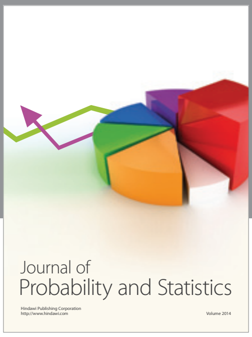
[38] A. Gagarin, S. Hussain, and L. T. Yang, "Distributed hierarchical search for balanced energy consumption routing spanning trees in wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 70, no. 9, pp. 975–982, 2010.


Advances in
Operations Research

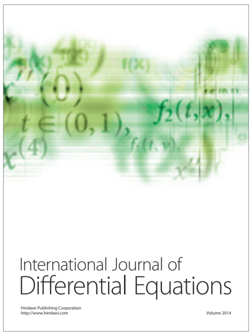

Advances in
Decision Sciences


Journal of
Applied Mathematics

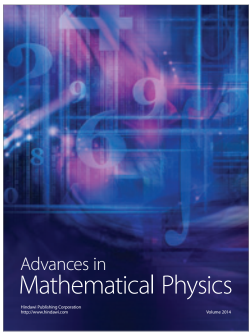

Algebra

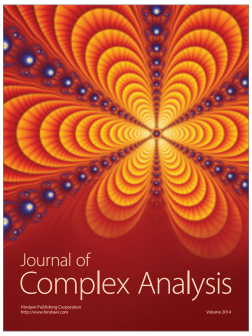

Journal of
Probability and Statistics

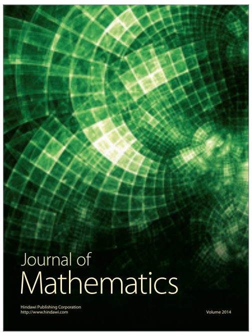
The Scientific
World Journal

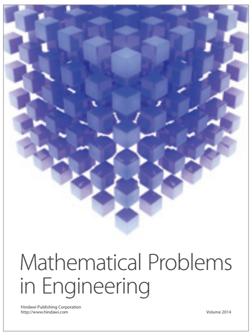

International Journal of
Differential Equations

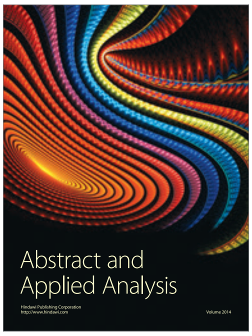
International Journal of
Combinatorics

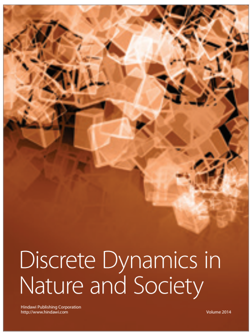

Advances in
Mathematical Physics

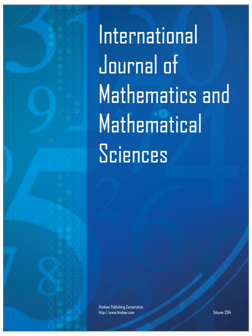

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

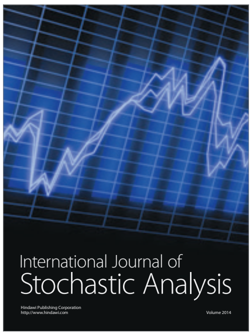

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization