A PROPOSED FUZZY METHOD TO DETECT FAULTY NODES IN WIRELESS SENSOR NETWORK CLUSTERING STRUCTURE

S.J. Dastgheib  1  H. Oulia 1  H. Gholamishiri 1  A. Ebrahimi 2

1. Young Researchers Club, Shiraz Branch, Islamic Azad University, Shiraz, Iran
dastqeib@gmail.com, hamed.oulia@gmail.com, gholami7@gmail.com
2. Shiraz Branch, Islamic Azad University, Shiraz, Iran, aisoudaebrahimi@yahoo.com

Abstract- Wireless sensor networks (WSN) consist of hundreds or thousands tiny nodes called sensor that work together and are connected to each other in order to do some special tasks. Because of limited nodes energy supply, clustering structure is introduced to reduce energy consumption in WSN. Due to specific features of WSN, nodes are prone to failure so it is very important to detect faulty nodes in clustering structures. We use fuzzy logic, the network available data and cluster member nodes to propose a comprehensive method for detecting faulty nodes in the clustering. It will be demonstrated that the proposed method is marvelously more efficient than the existing methods in the literature.

Keywords: Wireless Sensor Networks, Fault Tolerance, Fuzzy Logic, Clustering.

I. INTRODUCTION

Wireless sensor network (WSN) consists of hundreds or thousands tiny sensor nodes called node, which connected to each other to do some special tasks. Due to limited energy supply, some techniques are needed to reduce energy consumption of the nodes. Generally, to reduce energy consumption in the network, just some of the nodes send the data to the sink. This structure is called clustering and the node, which is connected to the sink, is called cluster head. The other nodes that send their data to nearest cluster head called cluster member. Cluster head can be perform some tasks such as compression on the data sent by other nodes. Consequently, small amounts of data transmitted to the sink and energy consumption is reduced [1, 2].

Due to specific features of WSN, nodes are prone to failure. The faulty nodes even may prevent the benefit of the network, so it is very important to detect them. There have been plenty of research works on detecting faulty nodes [3-6]. However, rarely comprehensive and efficient methods have been introduced to detect faulty nodes in clustering structure [7-9]. Between These methods, we will examine the FATP [10] and voting [11] methods since they are robust and able to detect all faulty nodes. However, they have some fundamental weaknesses (which will be referred to in section V), such as high energy consumption and delay.

We present the proposed method in two fields; cluster member and cluster head fault detection. Fuzzy logic, which is briefly introduced in section II, is used to detect the faulty cluster members. Fuzzy logic is used in the proposed method in order to reduce computational complexity, delay, and energy consumption, improve accuracy and performance. Some of the areas it has been applied to are cluster-head election [12, 13], security [14, 15], data aggregation [16], routing [17, 18], MAC protocols [19], and QoS [20, 21]. However, not much work has been done on using fuzzy logic for detecting faulty nodes.

In this paper, a distributed, scalable, energy efficient, load balanced, fast and accurate approach is proposed to detect faulty nodes in the cluster structure. We will prove that proposed method amazingly is more efficient than the existing approaches. The rest of the paper is organized as follows. In Section II, an introduction to fuzzy logic is given. Sections III and IV are devoted to illustration of the proposed faulty member detection and faulty head detection methods, respectively. In Section V, the proposed scheme is compared to other existing methods. Finally, Section VI concludes the paper.

II. FUZZY LOGIC

Before discussing the proposed method to detect faulty nodes, it is necessary to do an overview on fuzzy logic. Fuzzy Logic (FL) is defined as the logic of human thought, which is much less rigid than the calculations computers generally perform. Fuzzy Logic offers several unique features that make it a particularly good alternative for many control problems. It is inherently robust since it does not require precise, noise-free inputs and can be programmed to fail safely [22-26].

The output control is a smooth control function despite a wide range of input variations. Since, the FL controller processes user defined rules governing the target control system, it can be modified and tweaked easily to improve or drastically alter system performance. Fuzzy Logic deals with the analysis of information by using fuzzy sets, each of which may represent a linguistic...
The range of real values over which the set is mapped, called domain and the membership function describes fuzzy sets. A membership function assigns a truth (crisp) value between 0 and 1 to each point in the fuzzy set’s domain. Depending upon the shape of the membership function, various types of fuzzy sets can be used such as triangular, beta, PI, Gaussian, sigmoid, etc. The trapezoidal and triangular membership functions are suitable for real-time operation because they do not complexity computations and are having enough accuracy [30, 31] so we use triangular and trapezoidal membership functions in the proposed method.

The fuzzified values are processed by the inference engine, which consists of a rule base and various methods for inferring the rules. The fuzzy system used in the inference engine of the expert system is the Mamdani fuzzy system. The Mamdani fuzzy system is a simple rule-base method that does not require complicated calculations and which can employ the IF…THEN… rules to control systems [29]. All the rules in the rule-base are processed in a parallel manner by the fuzzy inference engine. The defuzzifier performs defuzzification on the fuzzy solution space. That is, it finds a single crisp output value from the solution fuzzy space. Some techniques are introduced for defuzzification like Center of Area (COA), mean of maximum and etc. COA is most suitable technique for WSN so we use this technique for defuzzification [30]. The crisp value adopting the COA defuzzification method was obtained by Equation (1).

$$\text{Crisp Output}(\alpha) = \frac{\int_{\text{Low}}^{\text{High}} \mu_a(x) dx}{\int_{\text{Low}}^{\text{High}} dx}$$

where $\alpha$ is the crisp value for the $z$ output and $\mu_a(x)$ is the aggregated output membership function [2-8].

### III. FAULTY CLUSTER MEMBER DETECTION

We in this section propose a new method to detect faulty cluster members. In our proposed method, cluster head detects failure in its cluster members using the fact that sensed values of nodes in a cluster are very similar. We use a fuzzy system whose components are shown in Figure 1 to obtain the failure amount of a node.

![Figure 1. The fuzzy system components to detect faulty cluster member](image)

If this failure value exceeds from a predefined threshold value, the node is considered as a faulty node; otherwise, the node continues to work but the effect of its data on network decisions is reduced according to the node failure value. This failure value for further computations returns back to the input of the fuzzy system. As soon as data received from a node, cluster head performs the fuzzy computations to calculate the amount of node failure. This operation can be executed always or at certain times.

As you can see, in our proposed method that all operations are done by normal in-network data, no additional data is needed. Since we assume that fuzzy engine is used, and since the parallel processing delay is negligible, the complexity of this method is about $O(1)$. The fuzzy membership functions and some of the existing rules, which are used in the Mamdani inference system, are shown in Figure 2 and Table 1.

![Figure 2. Fuzzy membership functions](image)

<table>
<thead>
<tr>
<th>The node failure rate</th>
<th>Node, sensed value</th>
<th>Cluster head sensed value</th>
<th>The node failure rate (Output)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Very Low</td>
</tr>
<tr>
<td>Medium</td>
<td>Low</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td>Low</td>
<td>High</td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>Low</td>
<td>High</td>
<td>High</td>
<td>Very Low</td>
</tr>
<tr>
<td>Medium</td>
<td>High</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>High</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>High</td>
<td>High</td>
<td>Low</td>
<td>Very High</td>
</tr>
</tbody>
</table>

We will compare the proposed method to detect faulty cluster members by some existing approaches in section V and will show the proposed method is more efficient than the other existing methods. It is very important to be sure about the accuracy of the cluster head. For this propose, we present a new method that is explained in the following section.

### IV. FAULTY CLUSTER HEAD DETECTION

We propose an efficient approach to detect faulty cluster head using cluster members. As soon as cluster formation, cluster head sets an identification number,
ranging from 1 to N, to each cluster member where N is the number of the cluster members. The cluster head is then checked at specified times by three of its cluster members. These three cluster members are chosen like the following sequence:

\[
\begin{align*}
1, & \quad 2, \quad 3; \\
2, & \quad 3, \quad 1; \\
1, & \quad 2, \quad 3
\end{align*}
\]

The energy consumption for the proposed method and FATP methods when they detect faulty cluster head according to Heinzelman et al [31] radio model. For simplification calculations, we consider following assumptions:

- Each cluster has \(N+1\) nodes; The distances between nodes (denoted as \(d')\) are identical; The distances between cluster heads (denoted as \(d\)) are identical. In the proposed method, the cluster head is checked after 10 packets sent to the sink; The average of steps between cluster head and sink, the size of each data packet and the number of bits sent by the cluster head are respectively \(m, k\) and \(k(N+1)/2\); and \(k=8, m=3, d'=2, d=10\) and the probability of detecting failure by one or both of checking nodes are \(1/4\).

The energy consumption for the proposed method and FATP to detect faulty cluster head in each time is calculated according to Equations (2) and (3).

\[
\text{Proposed Method} = \left(\frac{k}{2}\right)E_{\text{elec}} + \left(\frac{k}{2}\right)d^2E_{\text{amp}} + kE_{\text{elec}} + \frac{1}{4}\left(\frac{k}{2}\right)E_{\text{elec}} + \left(\frac{k}{2}\right)d^2E_{\text{amp}} + \frac{1}{4}\left(\frac{k}{2}\right)E_{\text{elec}} + kE_{\text{elec}}
\]

\[
\text{FATP} = \frac{k}{2}\sum_{i=1}^{N} E_{\text{elec}} + \frac{k}{2}\sum_{i=1}^{N} d^2E_{\text{amp}}
\]

Figure 3 compares energy consumption in the proposed and FATP methods when they detect faulty cluster head according to Heinzelman et al [31] radio model. Figure 3 compares energy consumption in the proposed and FATP methods when they detect faulty cluster head according to Heinzelman et al [31] radio model. For simplification calculations, we consider following assumptions:

- Each cluster has \(N+1\) nodes; The distances between nodes (denoted as \(d')\) are identical; The distances between cluster heads (denoted as \(d\)) are identical. In the proposed method, the cluster head is checked after 10 packets sent to the sink; The average of steps between cluster head and sink, the size of each data packet and the number of bits sent by the cluster head are respectively \(m, k\) and \(k(N+1)/2\); and \(k=8, m=3, d'=2, d=10\) and the probability of detecting failure by one or both of checking nodes are \(1/4\).

The energy consumption for the proposed method and FATP to detect faulty cluster head in each time is calculated according to Equations (2) and (3).

\[
\text{Proposed Method} = \left(\frac{k}{2}\right)E_{\text{elec}} + \left(\frac{k}{2}\right)d^2E_{\text{amp}} + kE_{\text{elec}} + \frac{1}{4}\left(\frac{k}{2}\right)E_{\text{elec}} + \left(\frac{k}{2}\right)d^2E_{\text{amp}} + \frac{1}{4}\left(\frac{k}{2}\right)E_{\text{elec}} + kE_{\text{elec}}
\]

\[
\text{FATP} = \frac{k}{2}\sum_{i=1}^{N} E_{\text{elec}} + \frac{k}{2}\sum_{i=1}^{N} d^2E_{\text{amp}}
\]
FATP = \frac{mk \left( \frac{N+1}{2} \right) E_{el} + \frac{mk}{2} d^2 E_{amp} + k \left( \frac{N+1}{2} \right)}{E_{elec} + d^2 E_{amp} + k \left( \frac{N+1}{2} \right)} (3)

- According to assumptions

The energy consumption for the proposed method and FATP to detect faulty cluster head in each time are equal to 1,856,800pj and 22,000,000pj, respectively. According to what was discussed, FATP is not a good approach for faulty node detection especially in cluster head fault detection field. We summarize the properties of voting, FATP and the proposed method in the following table. You can see that compared to the existing methods, the proposed method is scalable, energy efficient, accurate and very fast. Therefore, it is very suitable for WSNs and more efficient than other approaches.

<table>
<thead>
<tr>
<th>The proposed method</th>
<th>FATP</th>
<th>Voting</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Low</td>
<td>Very High</td>
<td>Low</td>
<td>Delay</td>
</tr>
<tr>
<td>Very High</td>
<td>Medium</td>
<td>High</td>
<td>Accuracy</td>
</tr>
<tr>
<td>Low</td>
<td>Very High</td>
<td>High</td>
<td>Energy Consumption</td>
</tr>
<tr>
<td>Very High</td>
<td>Low</td>
<td>Very Low</td>
<td>Lifetime</td>
</tr>
</tbody>
</table>

VI. CONCLUSIONS

The main goal of this study was to propose an effective approach to detect faulty nodes in WSN clustering structure. The proposed method in cluster member fault detection field provides an optimal approach with minimum delay, computational complexity and energy consumption using fuzzy logic and available data in network. In addition, the proposed method in cluster head fault detection could provide an efficient approach using available data in network and cluster members. It was demonstrated that proposed method is an optimized approach in comparison with existing methods.

REFERENCES

American, Institute of Electrical and Electronics Engineers (IEEE), pp. 436-441, USA, Jun. 2006.

BIographies

Seyyed Jalaleddin Dastgheib was born in Shiraz, Iran, in 1986. He received the B.Sc. degree in Computer Software Engineering from University of Tabriz, Tabriz, Iran, in 2008. He also received the M.Sc. degree in Computer Hardware Engineering from Dezful Branch, Islamic Azad University, Dezful, Iran, in 2011. His research interests include computer arithmetic, wireless sensor network, fuzzy systems, and fault tolerant systems.

Hamed Oulia was born in Kazeroon, Iran, in 1987. He received the B.Sc. degree in Computer Software Engineering from Sepahan University, Isfahan, Iran, in 2009. He is currently M.Sc. student in Computer Hardware Engineering from Dezful Branch, Islamic Azad University, Dezful, Iran, from 2010 to now. His research interests include computer arithmetic, wireless sensor network, Network on chip, fuzzy systems, and fault tolerant systems.

Hamid Gholamishiri was born in Fasa, Iran, in 1989. He is currently B.Sc. student in the field of Industrial Management in Shiraz Branch, Islamic Azad University, Shiraz, Iran. He is also a member of young researchers club of the university. His research interests include fuzzy screening management and quality function deployment.

Aisouda Ebrahim was born in Gachsaran, Iran in 1982. She received the B.A. degree in English Translation from Shiraz Branch, Islamic Azad University, Shiraz, Iran in 2011. Her research interests include editing and translation of scientific papers and translating