Macrothink Institute™

Network Protocols and Algorithms
ISSN 1943-3581
2012, Vol. 4, No. 3

Chain Based Fault Tolerant Routing Protocols

Samia A. Ali
Department of Electrical Engineering
Assiut University, Assiut, 71516, Egypt
Tel: +2088-241-1040 E-mail: samya.hassan@eng.au.edu.eg

Shreen K. Refaay
Department of Electrical Engineering
Assiut University, Assiut, 71516, Egypt
Tel: +2088-241-1040 E-mail: Shreen.khalef@hotmail.com

Received: July 1, 2012 Accepted: September 3, 2012 Published: September 29, 2012

DOI: 10.5296/npa.v4i3.2026 URL: http://dx.doi.org/10.5296/ npa.v4i3.2026

Abstract

Sensor networks have emerged as a promising tool for monitoring (and possibly actuating) the physical world; utilizing self-organizing networks of battery-powered wireless sensors that can sense, process, and communicate. In wireless sensor networks (WSNs), energy is a critical resource; hence power efficient routing protocols is necessary for data transmission in order to extend the network lifetime. Recently number of efficient chain based protocols has been proposed for WSNs routing. These routing protocols have achieved lowest consumed energy and delay. However, fault-tolerance was not considered in these protocols. Since node failures are inevitable in WSNs due to the harsh deployment environment, nodes mobility, etc. Therefore, fault tolerance is a must for successful routing protocols. In this paper, fault tolerance is incorporated for chain based routing protocols. More specifically, two techniques for fault detection and fault recovery for chain based routing protocols are proposed. The two techniques employ the same strategy for fault detection. However, the recovery strategy is different for the two techniques. The first technique overcomes the fault by having every predecessor node to a failed node instead of sending its data to the failed node forward it to the successor node of the failed node. The second technique gets around the fault by choosing a backup node for the faulty from the neighboring chain closest the sink which satisfies minimum energy consumption. The fault detection phase of the two proposed protocols may be applied at each data round or at varying intervals as dedicated by the application and the environment in which the WSN is deployed. The simulation results indicate that the two proposed protocols achieved fault tolerance efficiently (energy and time wise) for a single node failure at each chain.

Keywords: Chains, Clustering, Fault tolerant, Routing Protocols, Wireless Sensor Network.

1. Introduction

Wireless sensor networks (WSNs) have received significant attention in recent years due to their potential applications in military sensing, disaster management, traffic surveillance,

health care, environment monitoring, building structures monitoring, etc. [1]. A WSN is a self-organized network that consists of a large number of sensor nodes; each node has limited resources in terms of CPU power, size of memory, and storage capacity. Moreover, this type of network encounters power constraints because sensor nodes need a battery to operate properly [2]. WSNs are deployed in harsh physical environment where it is impossible to charge or replace the batteries of these sensor nodes. Also the sensor nodes can fail due to the hazardous environment deployed in or their mobility or drop in their energy level. Therefore, it is essential to design communication network protocols for those WSNs which incorporate fault tolerance of some kind and efficiently utilize the energy sources of the sensor nodes. Many routing protocols have been proposed in the literature to prolong the lifetime of the sensor nodes. LEACH (low energy adaptive clustering hierarchy) [3][4][5] has been the first hierarchical cluster-based routing protocol for WSN. The use of LEACH protocol achieved an eight times improvement over the direct transmission protocol. Its main disadvantage is due to heavy usage of cluster heads "typically cluster heads die at an early stage" [6]. The cluster head nodes carry heavier traffic loads, therefore, these nodes would deplete their energy faster, leading to what is known as energy holes or sometimes called the hot spot problem [7].

PEGASIS (Power-Efficient gathering in Sensor Information Systems) has been introduced in [8][9][10].It is a near optimal protocol for high rate data gathering applications in WSNs. The key idea of the PEGASIS protocol is the formation of a chain among the sensor nodes so that each node will receive from and transmit to a close neighbor. Gathered data moves from node to node, get fused, and eventually a designated node transmits it to the Base Station (BS).

CCM (Chain-Cluster based mixed routing) protocol [11] makes full use of the advantages of LEACH and PEGASIS, and provides improved performance over both of them. CCM protocol mainly divides a WSN into a number of chains and runs in two phases. In the first phase, sensor nodes in each chain transmit data to their own chain head nodes in parallel, using an improved chain routing protocol. In the second phase, all chain head nodes grouped as a cluster in a self-organized manner, where they transmit fused data to a voted cluster head using the cluster based routing.

An efficient CCBRP routing protocol for WSNs has been proposed in [12] to achieve both minimum energy consumption and minimum delay. The CCBRP protocol mainly divided a WSN into a number of chains (Greedy algorithm is used to form each chain as in PEGSIS protocol) and executed in two phases. In the first phase, sensor nodes in each chain transmit gathered data to their chain leader nodes in parallel. In the second phase, all chain leader nodes form a chain, (also, using Greedy algorithm) choose randomly a leader node, and all chain leader nodes send their data to the randomly chosen leader node. Then this chosen leader node fuses the data and forwards it to the BS.

One characteristic of wireless sensor networks is its high sensor failure probability either due to out of power or physical failure. In addition, it is very difficult if not impossible to replace the failed sensors. After some sensors fail, there will be some holes in the sensor network which block the routing. Thus, the routing protocols for wireless sensor networks should provide some degree of fault tolerance. In other words, the routing protocols should be able to by-pass such hole and prevent its enlargement. Therefore, fault tolerance is necessary for proper functioning of WSNs. In all the above mention protocols, fault-tolerance was not considered, so if a node fails due to mobility or drop in its energy or hazardous environment the data packets of this node as well as all the nodes in the chain preceding this failed node

will be lost can never be recovered.

The paper is organized as follows. Review for previous works related to fault tolerant routing protocols for wireless sensor networks is given in Section 2. The two proposed chain based fault tolerant for single fault per chain are presented in Section3. The proposed protocols complexity is presented in section 4. The simulation results for the two proposed chain based fault tolerant protocols for routing in WSNs and popular chain based routing protocols without providing fault tolerance are reported in Section 5. The conclusion of this research is given in Section 6.

2. Related Work

WSNs consist of tiny sensor nodes that are deploy in harsh environment, the sensor node has limited energy and it is very difficult to recharge so the node can be faulty due to loss of power or physical failure. Thus Fault tolerance is very important if not essential in WSNs. In This section we present a brief review of prior studies related to fault tolerant routing protocols.

Recently, a Fault Tolerant Trajectory Clustering (FTTC) for selecting cluster heads in WSNs [13] has been introduced. The FTTC protocol selects the cluster heads based on traffic which is changed periodically. Up to our knowledge, it is mainly the first technique in the literature for selecting the cluster heads to alleviate the hot spot problem and hence prolonging the networks lifetime [13].

In [14] a fault-tolerant clustering protocol for WSNs which is a run-time recovery mechanism based on consensus of healthy gateways is presented to detect and handle faults in one faulty gateway. It is a two-phased protocol; detection and recovery mechanism. Detection is achieved through Status updates message sent from a gateway to inform all the gateways whereabouts of the rest of the clusters in the system. When a gateway “A” does not receive an update message from another gateway “B”, “A” considers “B” to be faulty. A gate way should not be considered completely failed until all the gateways in the network are unable to communicate with it. Once the gateways reach a consensus about the presence of a fault, the next step is to identify the type of faults and allocate other sensors to replace the failed gateway node. The status message is parsed to extract the identity of sensors that cannot communicate with the replacing gateway due to range faults in the gateways. Furthermore, when a gateway is identified as completely failed all the data packets of the sensors in its cluster are recovered.

Hong Min et al. [15] have proposed a smart checkpointing scheme for improving the reliability and reducing the recovery latency of clustering routing protocols. Mainly in this protocol, a cluster head sends routing and the collected data information to backup nodes, which periodically updates the state of its cluster head. If a cluster head is in transient fault, then one of the backup nodes detects the cluster head failure and the backup node takes on the role of its cluster head. Using checkpointing, clusters can quickly recover from a transient fault of cluster heads by omitting re-election of the faulty cluster head and preventing loss of the collected information [15].

In this paper, we propose two chain based fault tolerant routing protocols to detect faulty sensor nodes and recover data supposed to be sent through those detected faulty sensor nodes to the BS in clustering WSNs. More specifically, two techniques are proposed for the detection of faulty sensor nodes and the recovery from the detected faults; one for both single chain and chain-chain WSNs while the other only for chain-chain WSNs. The two proposed

techniques employ the same strategy for fault detection; every node sends a notifying message to its next neighboring node in its chain, if it is alive it replies with a ready message else the notifying node deduces that its next neighboring node is faulty. However, the recovery strategy is different for the two proposed techniques. The first technique (for single chain and chain-chain WSNs) overcomes the fault by having every sensor node neighboring to a failed sensor node sends its data packets to the successor sensor node of the failed node in its chain instead of the failed sensor node itself. The second technique (only for chain-chain WSNs) gets around the fault by choosing the node which satisfies minimum energy consumption from the closest neighboring chain to replace the faulty node. The performed simulation illustrates that the two proposed techniques provided fault tolerance for a single node failure in each chain at a very small cost energy and time wise.

3. The Proposed Fault-Tolerant Protocols

In this section, we present the details of the two proposed chain based fault tolerant protocols for single faulty node in each chain of clustering WSNs. The two proposed techniques provide fault tolerance in two phases; detection and recovery.

3.1 Detection Phase

In the detection phase of the chain based fault tolerant protocols; each sensor node in every chain of the clustered WSN sends a NOTIFY message to its successor neighbor sensor node in its chain. If the notified successor sensor node is alive it replies with a READY message, else the notifying node deduces that its successor neighbor sensor node in its chain is Faulty. Fig.1 illustrates a clustering WSN with fifty nodes; divided into five chains each chain contains ten nodes. Every node in each of the five chains except the faulty ones sends two messages and receives two. It sends a NOTIFY message to its successor node and READY message to its predecessor in its chain. The two received messages are a NOTIFY message from its predecessor node and a READY message from its successor node in its chain. It is clear from Fig.1that faulty sensor nodes do not send NOTIFY messages but receive NOTIFY messages, however, they never receive or send READY messages.

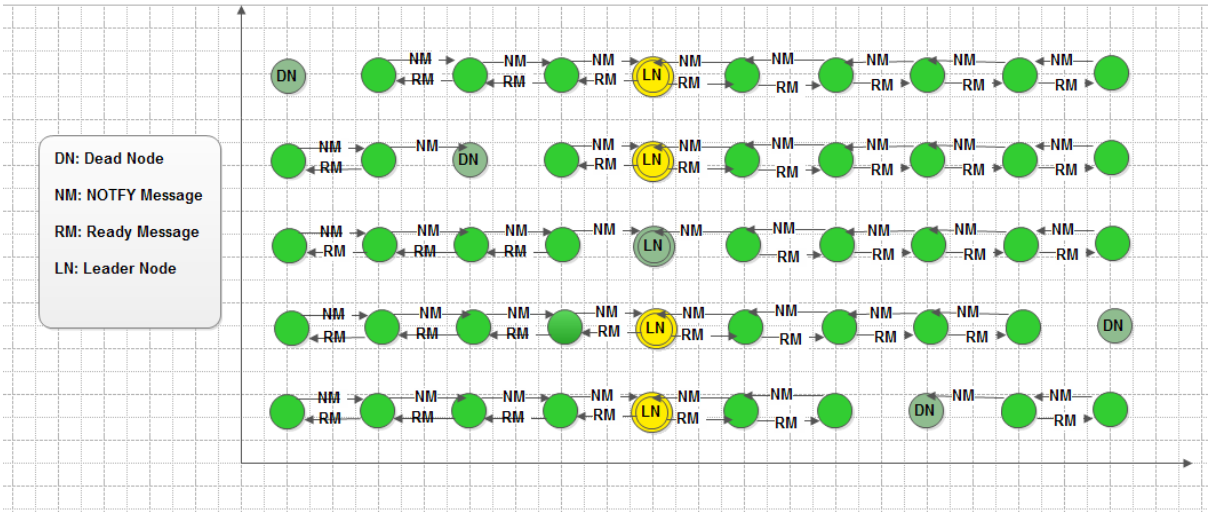


Figure 1. Transferred messages between sensor nodes for the fault detection phase of the proposed protocols.

3.1.1 Pseudo Code Of The Detection Phase

In this section pseudo codes of the detection phase for the two proposed chain based fault tolerant protocols are given. Let C be the number of chains in the WSN under consideration, Nc is the number of nodes in each chain, and LCI is the leader of Chain I.Figs.2 and 3 respectively present pseudo codes for the detection phase of the proposed chain based fault tolerant protocols for a single chain and chain-chain clustering WSNs.

```
Case LC:
    C (0): the first node of the chain
    For j=Nc-1:0
        C_I (j) sends a NOTIFY message To C_I (j-1)
        If (C_I (j-1) is alive)
        Then C_I (j-1) Send READY Message To C_I (j)
        Else C_I (j-1) is Faulty
        End if
    End For
    C (Nc-1): the Last node of the chain
    For j=0:Nc-1
        C(j) sends a NOTIFY message To C(j+1)
        If (C(j+1) is alive)
        Then C(j+1) Sends a READY Message To C(j)
        Else C(j+1) is Faulty
        End if
    End For
    C(1..Nc- 2): all the intermediate nodes of the chain
    For j=Nc-1: index of a leader node
        C(j) sends a NOTIFY message To C(j-1)
        If (C(j-1) is alive)
        Then C (j-1) Sends a READY Message To C(j)
        Else C (j-1) is Faulty
        End if
    End For
    For j=0: index of leader node
        C(j) sends a NOTIFY message To C(j+1)
        If (C(j+1) is alive)
        Then C(j+1) Sends a READY Message To C(j)
        Else C(j+1) is Faulty
        End if
    End For
End Case
End For
```

Figure 2. Pseudo code for the detection phase of a single chain fault tolerant protocol.

```
For I=0: C-1
Case LC_I:
    C_I (0): the first node of C_I
```

For j=Nc-1:0
C_I (j) sends a **NOTIFY** message To C_I (j-1)
If (C_I (j-1) is **alive**)
Then C_I(j-1) Sends a **READY** Message To C_I (j)
Else C_I (j-1) is **Faulty**
End if
End For
C_I(Nc-1): the Last node of C_I
For j=0:Nc-1
C_I(j) sends a **NOTIFY** message To C_I(j+1)
If (C_I(j+1) is **alive**)
Then C_I(j+1) sends a **READY** Message To C_I(j)
Else C_I(j+1) is **Faulty**
End if
End For
C_I(1..Nc- 2): all the intermediate nodes of C_I
For j=Nc-1: index of a leader node
C_I(j) sends a **NOTIFY** message To C_I(j-1)
If (C_I(j-1) is **alive**)
Then C_I (j-1) sends a **READY** Message To C_I(j)
Else C_I(j-1) is **Faulty**
End if
End For
For j=0:index of leader node
C_I(j) sends a **NOTIFY** message To C_I(j+1)
If (C_I(j+1) is **alive**)
Then C_I(j+1) sends a **READY** Message To C_I(j)
Else C_I(j+1) is **Faulty**
End if
End For
End Case
End For

Figure 3. Pseudo code for the detection phase of the proposed fault tolerant for Multi-chain WSNs.

3.2 Recovery Phase

The two versions for the recovery phase for the two proposed chain based fault tolerant routing protocols are given next. In order to recover the data packets that supposed to be send by the faulty sensor node we present two techniques to achieve the desired fault tolerance with minimum costs (energy and time delay). The first proposed recovery technique redirects the data packets that are supposed to go through the faulty sensor node on their way to the base station to bypass the faulty node to its successor node instead. This recovery technique is used for both single chain and chain-chain clustered WSNs. The second proposed recovery technique chooses a backup node for the faulty node from its neighboring chain closer to the base station which satisfies minimum energy consumption to redirect the data packets that are supposed to go through that faulty node.

3.2.1 Recovery of the First Proposed Protocol

There are three types of nodes according to their positions in the chain (terminal, intermediate, and leader) and hence they require different treatments by the proposed protocol. The first chain based fault tolerant routing protocol handles the recovery of the three different faulty nodes as follows:

1. The faulty node is a terminal node; this faulty sensor node will not be able to sense, collect or receive data. In other words the data that are supposed to be collected by this node will be lost and not included in the data packets sent to the base station.
2. Faulty node is an intermediate node; this is the easiest and most probable case. The predecessor node of the faulty node sends its data packets to the successor node of the faulty node.
3. Faulty node is a leader node; this is the worst case for node failure because without the proposed protocol, the data packets of the whole chain for which the faulty node is its leader will be lost. If the WSN under consideration has a single chain both the predecessor and the successor of the leader faulty node send their data packets directly to the base. However, if the WSN under consideration has multi-chains; the two nodes neighboring of the faulty leader node as well as the leader node of the previous chain send their data to the leader of the next neighboring chain (i.e., the leader that the faulty leader node is supposed to sends its data packets to).

In all the above three cases the base station is able to identify the faulty nodes in the WSN under consideration. The identity of each of the faulty nodes is detected by the base station since the data packets are stamped with the identities of the nodes originally collecting the data. For fast recovery of the data to be collected by the faulty nodes the base station sends messages to search for replacements in the neighborhoods of the faulty nodes to wake up nearby nodes or order a new deployment.

3.2.2 Recovery of the Second Proposed Protocol

The second recovery technique tolerates a single faulty node per chain in a multiple chained clustered WSNs. A backup node which satisfies minimum energy consumption is chosen from the neighboring chain closer to the base station to replace each of the faulty nodes. Similar to the first recovery protocol, there are three types of nodes (terminal, intermediate, and leader) each of which requires different treatment by the second proposed recovery protocol as follows:

1. The faulty node is a terminal node (the same treatment as in the first protocol); this faulty sensor node is not being able to sense or collect data and therefore no data is sent or received by this faulty terminal node. In other words the data that supposed to be collected by this faulty node will be lost and never send to the base station.
2. The faulty node is an intermediate node; the predecessor of the faulty node sends its data to a backup node from the neighboring chain closest to the base station which achieves minimum energy consumption, then the backup node sends the received data packets to the successor of the faulty node.
3. The faulty node is a leader node (this is the worst case scenario); the leader node that supposed to send its data packets to the faulty leader node in normal situations chooses either the predecessor or the successor of the faulty leader node; the one node which achieves minimum energy consumption and sends its data packets to it. Then both the predecessor and the successor of the faulty leader node send their data packets as well as

the data packets received from the leader node that supposed to send its data packets in the normal situations to the faulty leader node to the leader of the neighboring chain closest to the base station.

Similarly, as in the first recovery protocol the base station is capable of identifying the faulty nodes and searching for replacements by waking up nodes in the neighborhood of the faulty nodes or ordering new deployment.

3.3 Illustrative Example

To clarify the two proposed fault tolerant routing protocols; a network with 100 nodes, five chains is used. Fig. 4 illustrates all the possible types of nodes (terminal, intermediate, and leader) and their corresponding failures and how they are detected and tolerated using the two proposed chain based protocols. Accordingly there are five distinguishable cases:

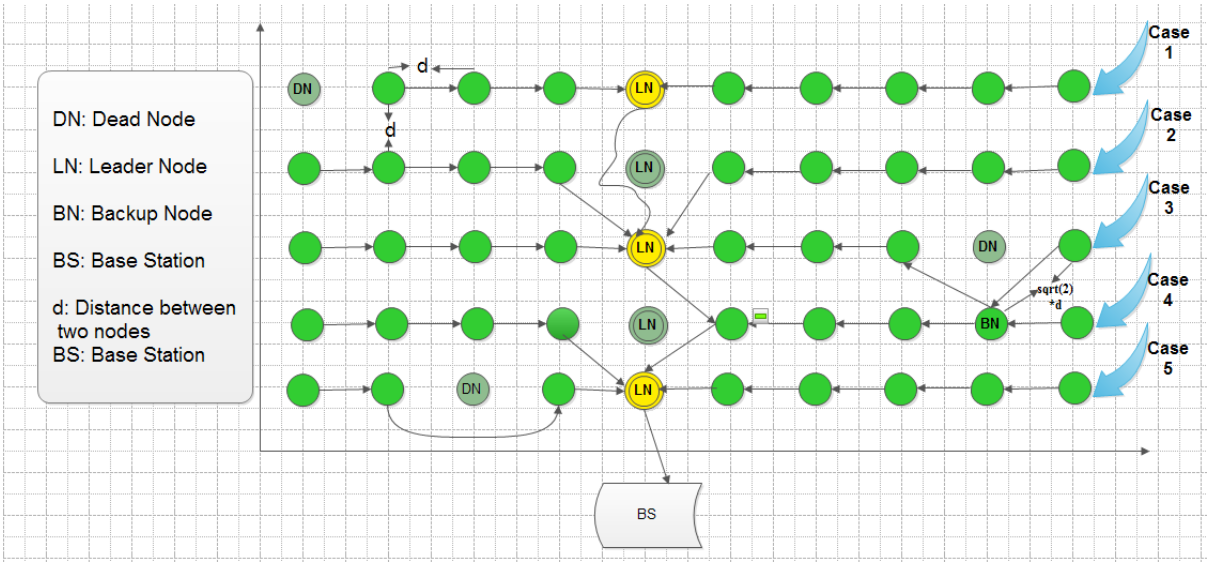


Figure 4. Data detection and recovery for all types of nodes (terminal, intermediate, and leader) failures using the two proposed fault tolerant protocols.

Case1: The faulty node is a terminal node (see Fig. 4 node 1of chain1). This case requires the same action from the two proposed fault tolerant protocols. No data is sensed and therefore no data sent to the neighbor node or the leader node of this chain. By the end of this round, the base station would receive all the WSN's collected data. Then it discovers that the data of the first node of chain 1 is missing and deduces that this node is faulty. Accordingly, the BS can replace this faulty node by assigning another node to take over.

Case2: The faulty node is a chain leader (see Fig.4 node 5 of chain 2). Therefore according to the first proposed fault tolerant protocol leader of the previous chain (node 5 of chain 1) sends its data to the leader of chain next to the chain of the faulty leader node (node 5 of chain 3). Also the predecessor and successor neighbors of the faulty leader send their data packets to the leader of the chain next to the chain of the faulty leader node as well.

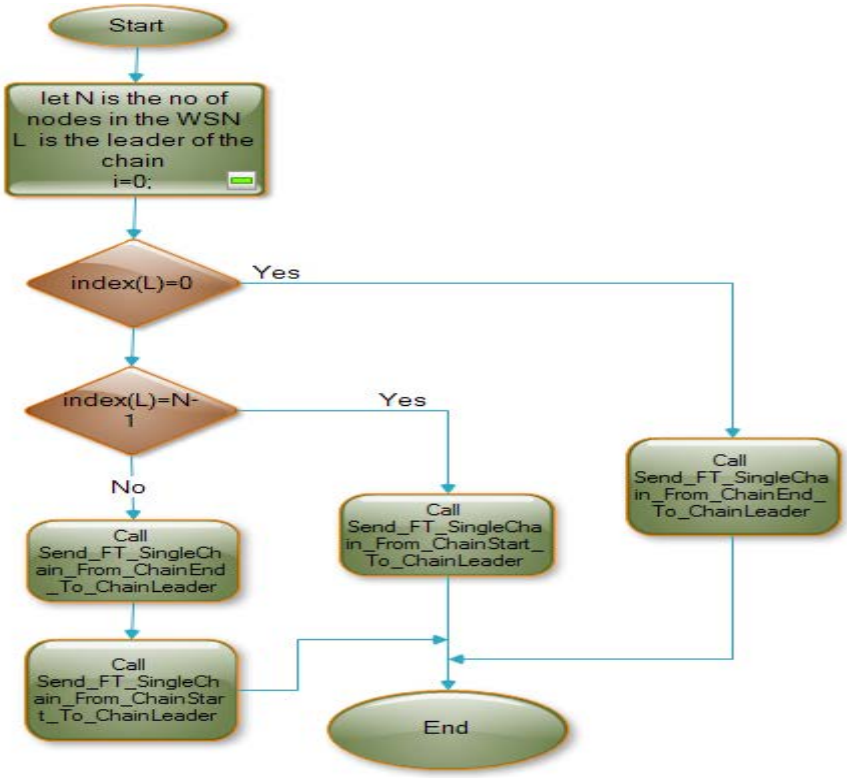
Case3: The faulty node is an intermediate node (see Fig. 4 node 9 of chain 3) however; its predecessor neighbor (node 10) does not have enough energy to send its data packets to the successor neighbor (node 8) of the faulty node. Thus according to the second proposed fault tolerant protocol the predecessor neighbor (node 10) of the faulty node sends its data

packets to a chosen backup node (node 9 of chain 4) which satisfies minimum energy consumption from the next neighboring chain to send on its behave its data packets to the successor (node 8 of chain 3) of the faulty node.

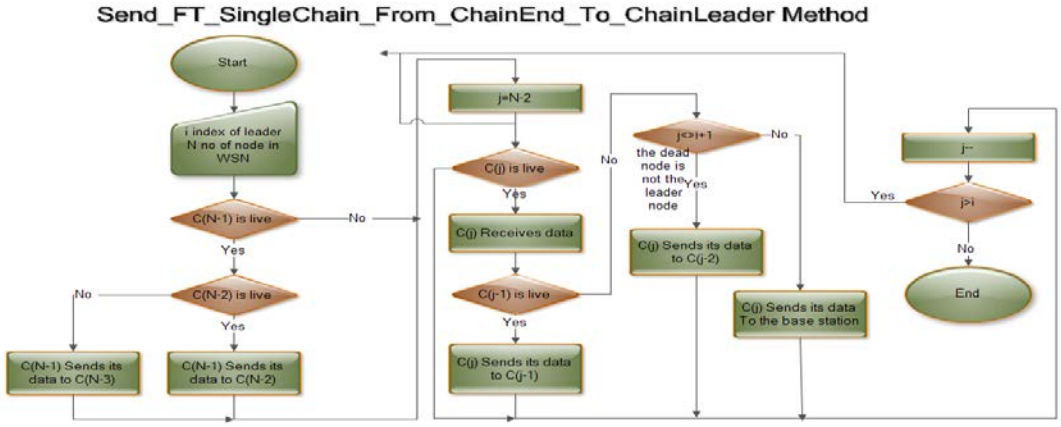
Case4: The faulty node is a leader node of chain (see Fig. 4 node 5 of chain 4). According to the second proposed fault tolerant protocol the leader of the previous chain (chain 3)sends its data packets to either node 4 or node 6 of chain5;the one which achieves minimum energy consumption. Then the chosen neighbor (node 6 of chain 4) of the faulty leader node fuses its data packets with the data packets of the leader of the previous chain to the leader node of the following chain (node 5 of chain 5). The other neighbor (node 5 of chain 4) of the faulty leader also sends its data to the leader of the following chain (chain 5).

Case5: The faulty node is an intermediate node (see Fig. 4 node 3 of chain 5). The predecessor neighbor (node 2) has enough energy. Therefore according to the first proposed protocol, node 2sends its data packets to the successor of the faulty node (node 4).

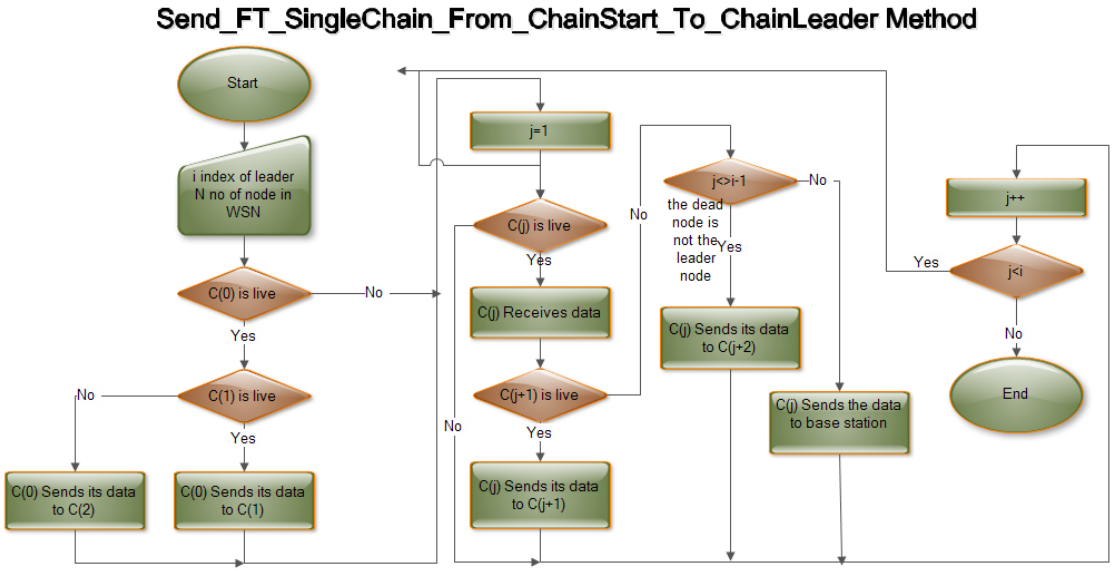
Figs. 5, 6, and 7 respectively present the recovery phase of the proposed fault tolerant routing protocols for clustering WSNs with a single chain, multi-chain when nodes of the WSNs have enough energy to bypass the faulty node to its successor (first protocol), and multi-chain when the WSNs nodes do not have enough energy to bypass the faulty node to its successor (second protocol).



a. Fault Tolerance Single Chain based protocols recovery phase

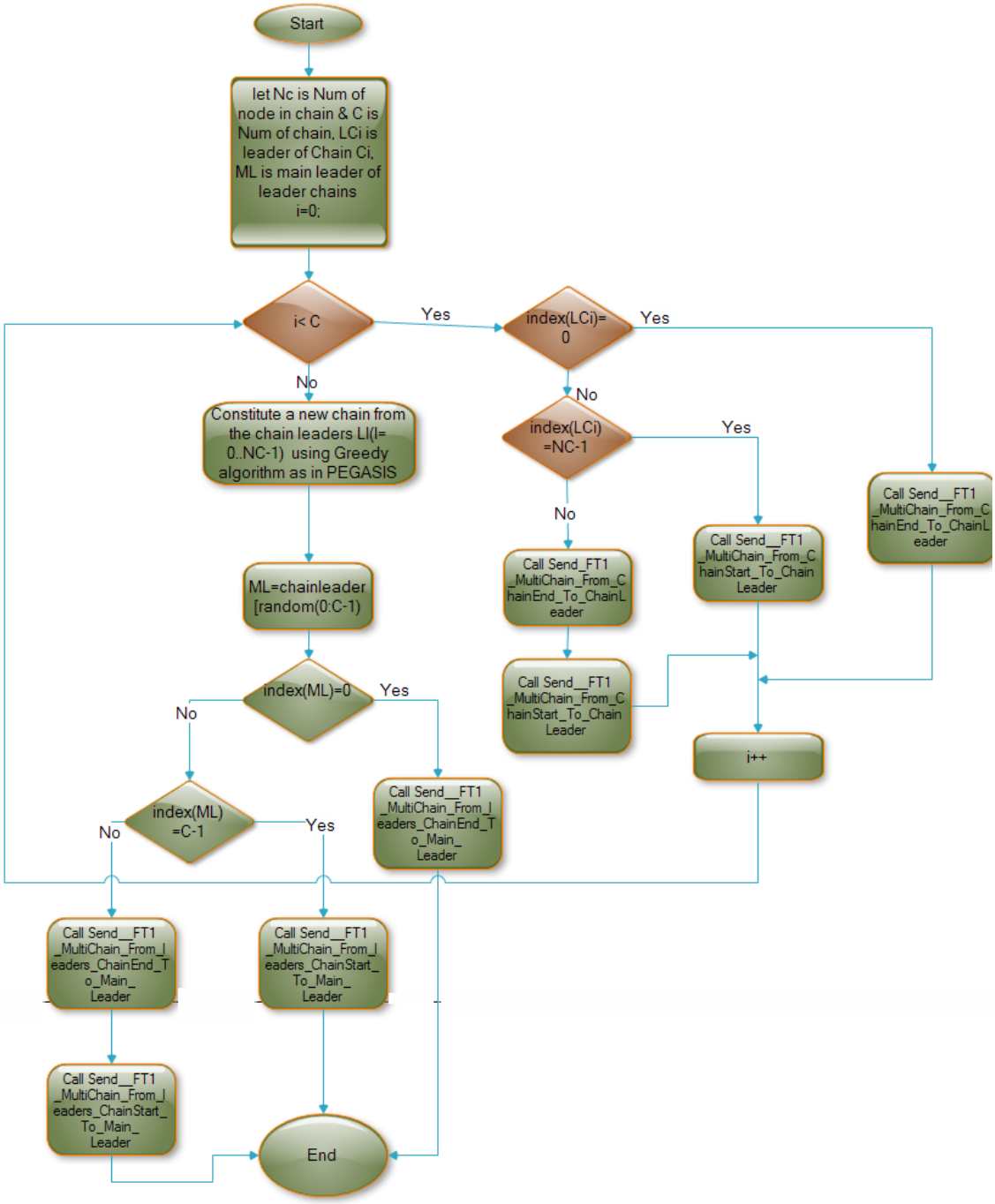


b. Send_FT_SingleChain_From_ChainEnd_To_ChainLeader method

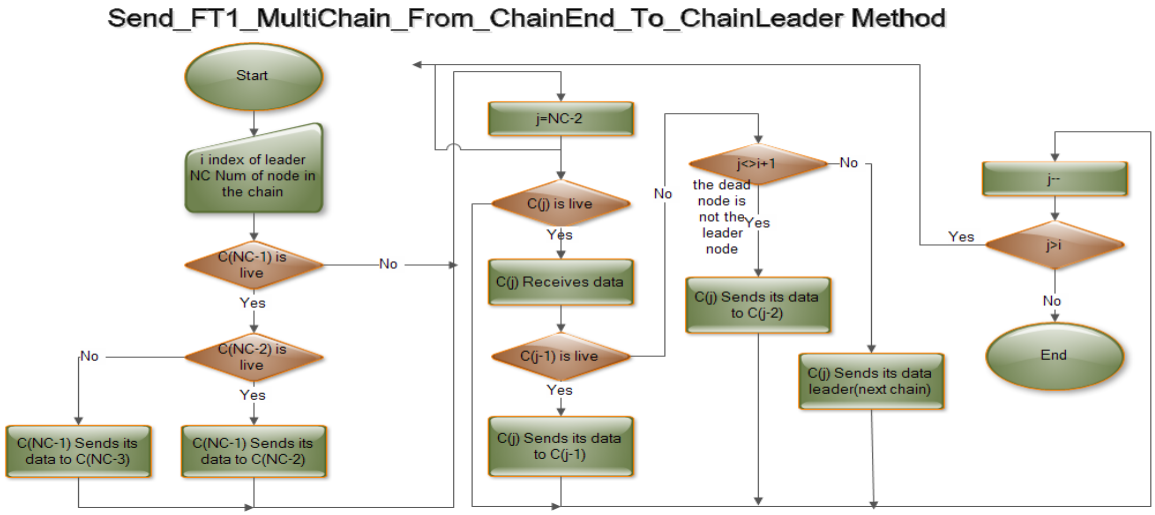


c. Send_FT_SingleChain_From_ChainStart_To_ChainLeader method

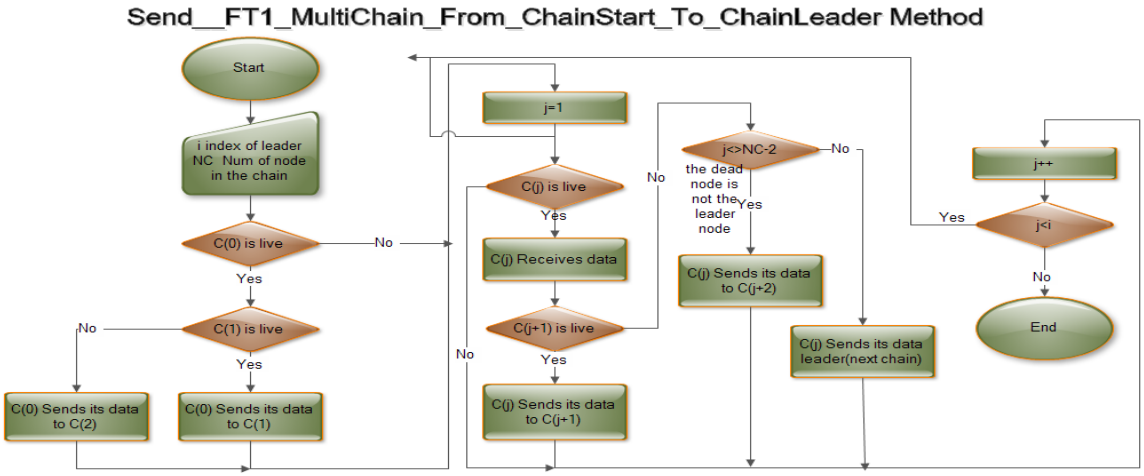
Figure 5. Flowchart for the recovery phase of a single chain fault tolerant routing protocol.



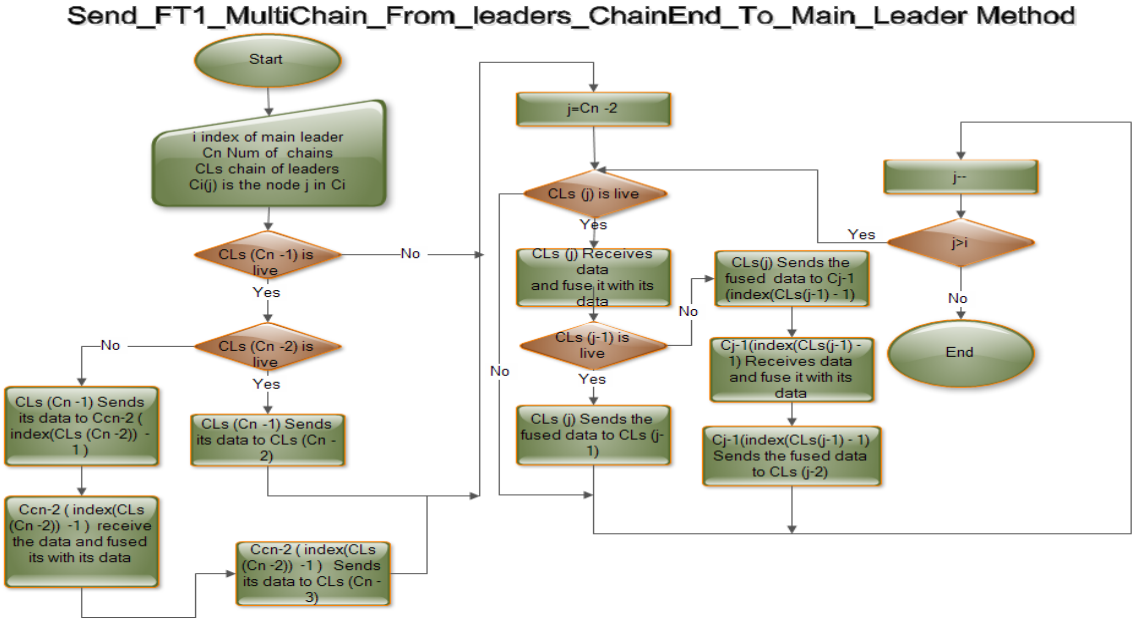
a. First fault tolerance multi chain based routing protocol recovery phase.



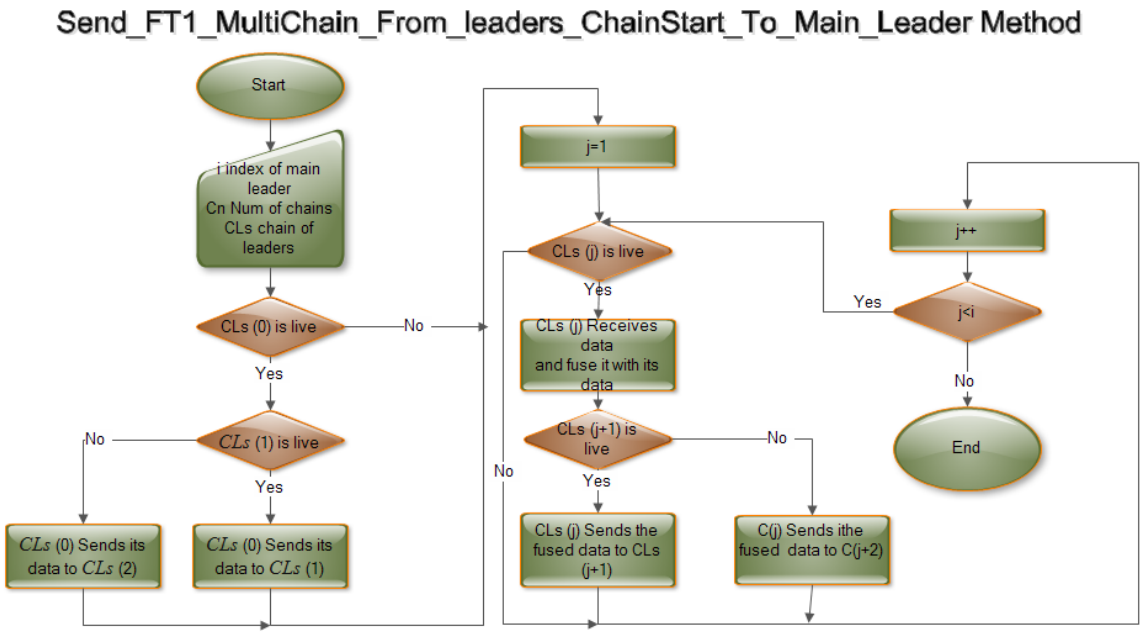
b. Send_FT1_MultiChain_From_ChainEnd_To_ChainLeader method



c. Send__FT1_MultiChain_From_ChainStart_To_ChainLeader method

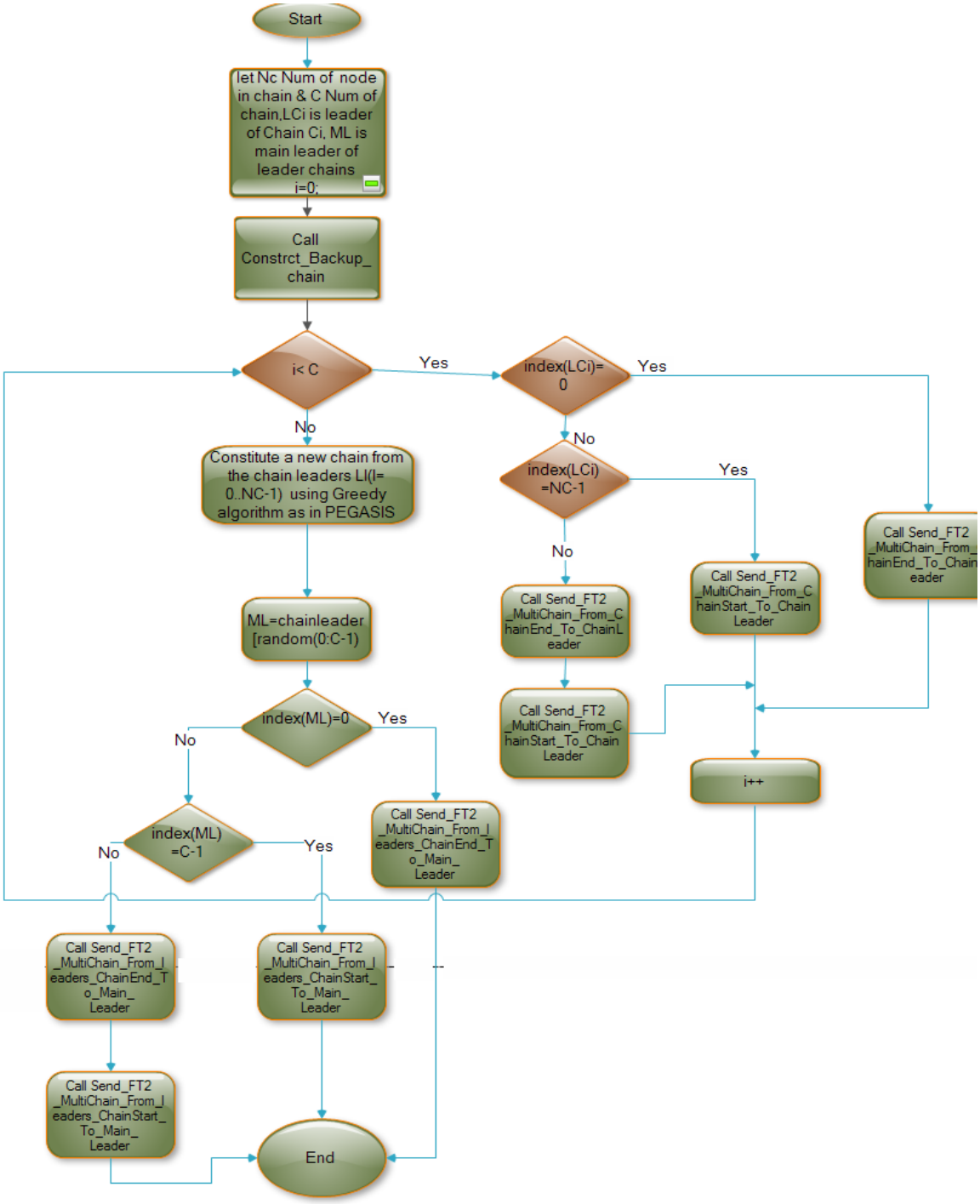


d. Send_FT1_MultiChain_From_leaders_ChainEnd_To_Main_Leader method

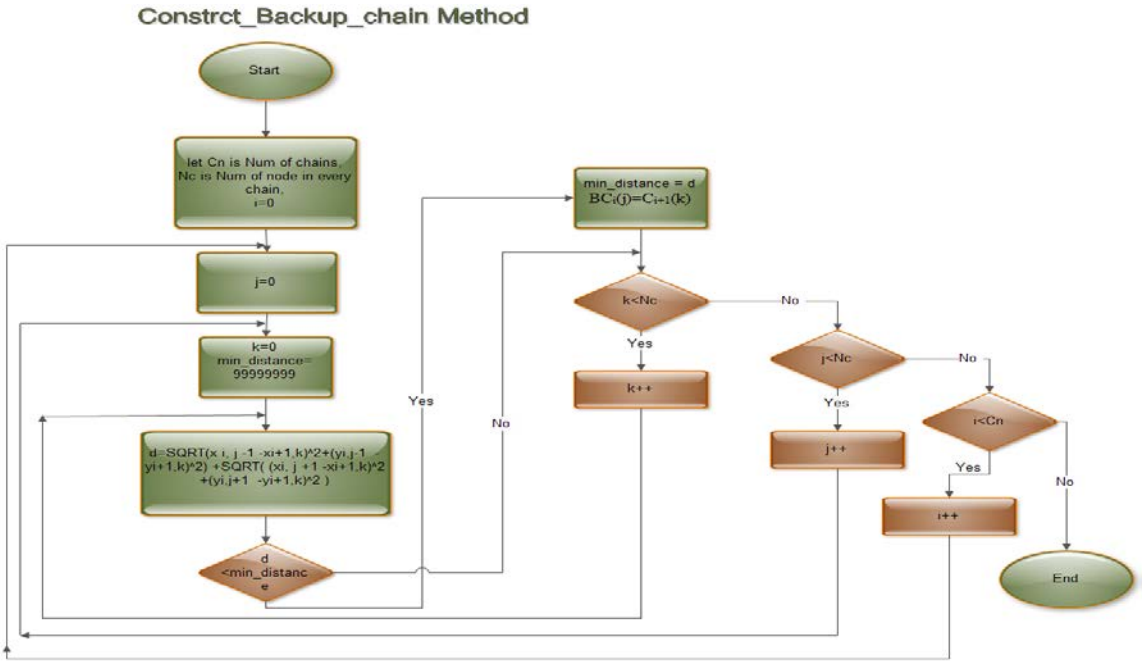


e. Send_FT1_MultiChain_From_leaders_ChainStart_To_Main_Leader method

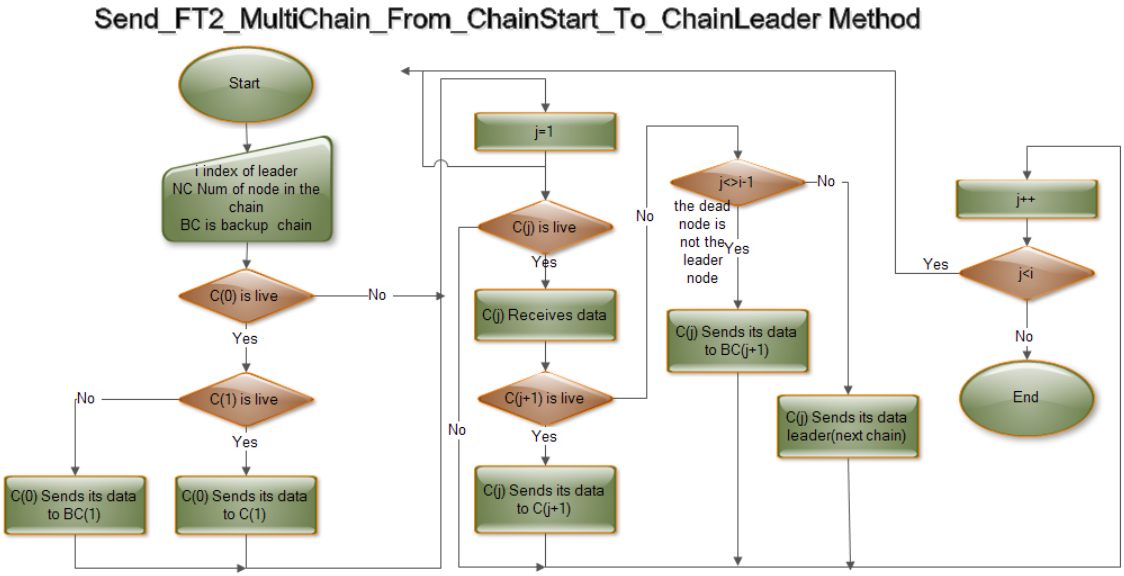
Figure 6. Flowchart of recovery phase of first fault tolerance multi chain based routing protocol.



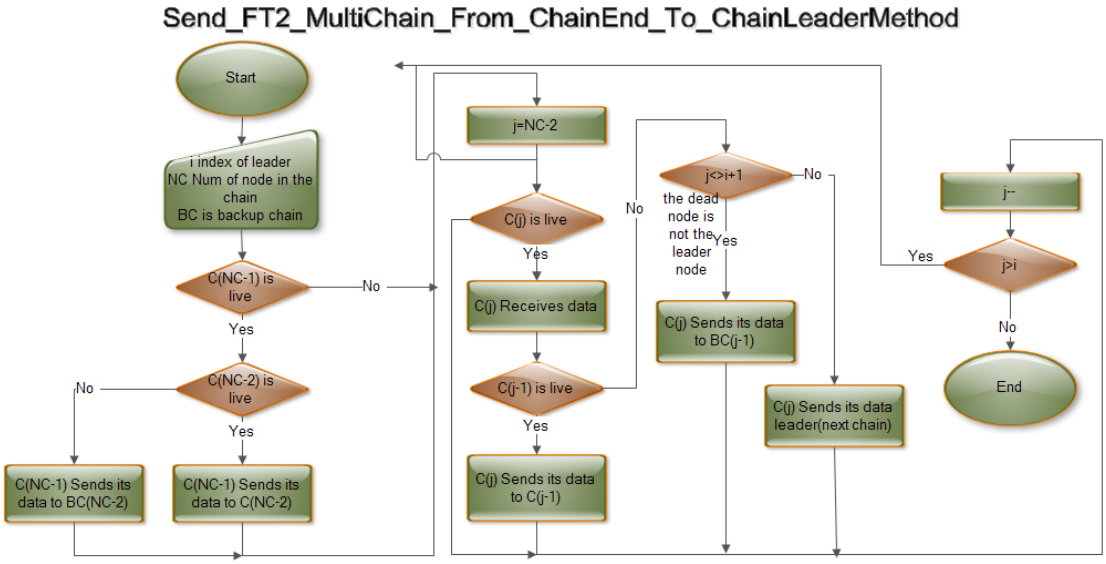
a. Second fault tolerance multi chain based routing protocol Recovery Algorithm



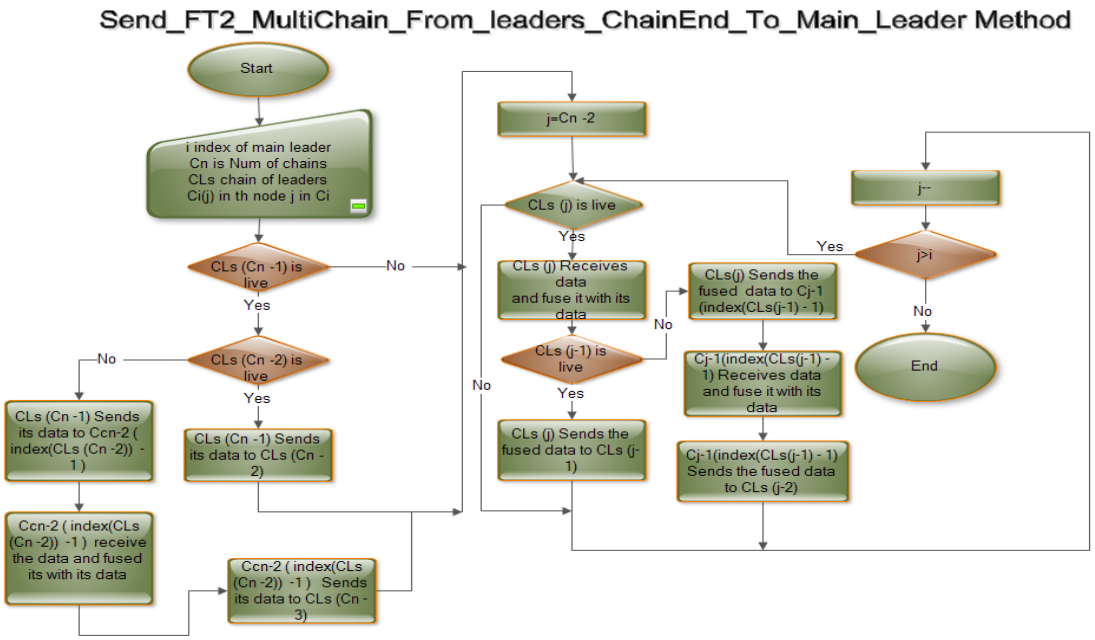
b. Construct_Backup_chain method



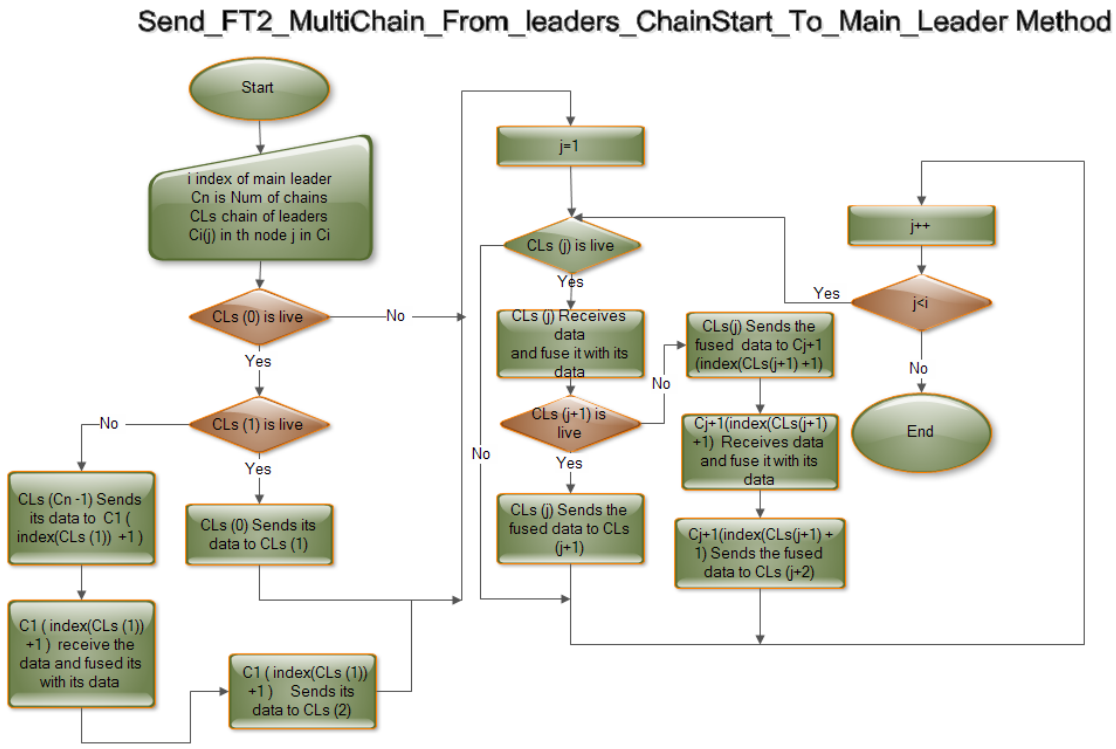
c. Send_FT2_MultiChain_From_ChainStart_To_ChainLeader method



d. Send_FT2_MultiChain_From_ChainEnd_To_ChainLeader method



e. Send_FT2_MultiChain_From_leaders_ChainEnd_To_Main_Leader method



f. Send_FT2_MultiChain_From_leaders_ChainStart_To_Main_Leader method

Figure 7. Flowchart for recovery phase Second fault tolerance multi-chain based routing protocol.

3.4 Fault Coverage

In literature, fault coverage is defined as the ability of the system to detect and recover from the occurrence of fault(s) during normal system's operation [11].A more specific definition for the fault coverage is the ratio of detected faults to total faults. For example, if the total number of faults is 200 butonly143 faults are detected the fault coverage is % 71.5.In this we have treated only single faulty node per chain and our goal is to show the percentage of multiple fault coverage achieved by it as well. The probability of r faulty nodes among Nc nodes in a chain with failure probability q is given by the binomial probability B(r: Nc, q).

$$B(r: Nc, q) = \binom{Nc}{r} q^r (1-q)^{Nc-r} \quad (1)$$

Thus the total number of faults due to r faulty nodes among the Nc nodes can be written as:

$$Nc \text{ choose } r = (Nc!) / (r!\,(Nc - r)!) \quad (2)$$

More specifically, for Nc=20 (the number of nodes in each chain) and q is the probability of an error in a node; the probability of faults due to two faulty nodes among the Nc nodes can be written as:

$$B(2:20, q) = \binom{20}{2} q^2 (1-q)^{18} \quad (3)$$

And the number of faults due to double node failure in a chain of twenty nodes is:

$$20 \text{ choose } 2 = (20!) / (2!(18)!) = 190. \quad (4)$$

The two proposed chain based fault tolerant routing for single node failure in each chain without any modification are capable of recovering from all double non-adjacent node

failures in a given chain. However, the number of adjacent nodes in a given chain of Nc nodes is Nc -1. Thus the percentage of fault coverage for double faults of the two proposed for single node failure in each chain without any modification is:

$$100\ ((Nc!) / (2!\ (Nc - 2)!)) – (Nc-1))/ (Nc!)/(2!\ (Nc - 2)!)). \quad (5)$$

For instance, for the case where Nc = 20 and two nodes failed in the same chain; there are 190 different combinations of two failed nodes in the same chain. Thus percentage of covered double faulty nodes in the same chain by the two proposed protocols is:

$$((190 -19) /190) = 90\%. \quad (6)$$

4. Complexity of the proposed protocols

For any protocol to achieve fault tolerance there is overhead time and expended energy involved. In this section we analyze the complexity of the two proposed fault tolerant routing protocols. The complexity of algorithms usually analyzed with respect to two aspects time and space. However, for WSNs routing protocols expended energy is very valuable in order to extend the network life time. Therefore, in our analysis we consider overhead time and expended energy for three scenarios; best case, average case, and worst case. The two proposed protocols adopt the same strategy for fault detection phase; therefore, the recovery phase is the determinate for the preference between the two proposed protocols. The complexity analysis is provided for best case, worst case and average case.

To ease the complexity analysis of the two proposed fault tolerant routing protocols some notation is given below:

N is the number of nodes in the WSN under consideration,
C is the number of chains in the WSN under consideration,
N_c is the number of nodes in each chain.
$E_{Tx\ (k,\ d)}$ is the energy required to transmit message of size K distance d,
K is the message size in bits, equals 2048bits for data,
d is the distance between two nodes,
$E_{Rx(k)}$ is the energy required to receive on a message,
E_{elec} is the electrical energy = 50 nJ/bit,
E_{amp}= 100 pJ/bit/m2 for the transmitter amplifier,
E_d is the energy expended for fault detection, and
E_t is total energy required for fault detection and fault recovering.

4.1 Overhead Time

The overhead time for the two proposed fault tolerant routing protocols is the sum of the extra time required by the proposed protocols to detect the faults and recovery data packets. Due to the data packets collection nature of the two proposed protocols there is no overhead time imposed by the recovery phase. Therefore, the overhead for the two proposed fault tolerant routing protocols is only due to the fault detection phase of the protocols. Since both protocols have identical fault detection phase and no overhead time is required by the recovery phase. Thus both the two proposed protocols incur the same overhead time. The overhead time of the detection phase is mainly the time required to transmit two messages; NOTIFY and READY messages, each has a size of five bytes (40 bits). It is assumed here as in [12] that the time required to transmit a data packet of k bits between any two neighboring nodes in a chain is one time unit. The two messages; NOTIFY and READY for fault

detection phase are sent from both sides of each chain sequential from node to its successor neighbor but simultaneously through all the chains of the WSN. Thus the overhead time for the fault detection phase and hence the overhead time for the two proposed fault tolerant routing protocols is given by:

$$(2* 40/(2048))*(Nc/2) \quad (7)$$

4.2 Expended Energy

The energy expended to detect a faulty node in the two proposed fault tolerant protocols does not depend on the node type. Therefore, there is a one to one corresponds between the number of messages sent and the energy expended. In the fault detection phase of the two proposed protocols; every node except terminal nodes sends two messages a NOTIFY message to its successor node and a READY message for its predecessor node. Terminal nodes send only NOTIFY messages to their successor nodes. So the total number of messages required for the fault detection phase of the two proposed fault tolerant protocols is: 2*C*(NC-2). Accordingly, the total amount of expended energy by the fault detection phase of the two proposed protocols is:

$$E_d = 2* (E_{Tx\,(k,\,d)} + E_{Rx(k)})*C*(NC - 2) \quad (8)$$

Where $E_{Tx\,(k,\,d)\ \text{and}}\ E_{Rx(k)}$)are as given in [6] to be:

$$E_{Tx\,(k,\,d)} = E_{elec}*k + E_{amp}* k* d\text{^}2 \text{ , and } E_{Rx(k)} = E_{elec}*k \quad (9)$$

4.2.1 First Proposed Protocol Expended Energy

The two proposed fault tolerant routing protocols adopt the same strategy for the detection phase. Thus the amount of expended energy for the detection phase of the first proposed fault tolerant routing protocol is as given in (8). The expended energy for the recovery phase for the first proposed fault tolerant routing protocol depends on the type of faulty node (terminal, intermediate or leader). There are three distinguishable cases (best, worst, and average).

1. Best Case

The minimum amount of expended energy for the two proposed protocols occurs when the faulty node is a terminal node. In this case, there is no expended energy for recover phase. Therefore, the total amount of expended energy is due to the detection phase of the protocol which is:

$$E_t = E_d + E_r = E_d = 2* (E_{Tx\,(k,\,d)} + E_{Rx(k)})*C*(NC - 2) \quad (10)$$

2. Worst Case

The worst case for the expended energy of the recovery phase for the first proposed protocol occurs when every other chain leader node is faulty. If all chain leader nodes are faulty, a new deployment is required the occurrence of this situation is very rare. The expended energy by the first proposed fault tolerant routing protocol for the recovery phase is:

$$E_r = (C/2) * (E_{Tx\,(k,\,2d)} + 2 * E_{Tx\,(k,\,sqr(2)*d)} - 4 * E_{Tx\,(k,\,d)}) \quad (11)$$

Thus total expended energy by the first proposed fault tolerant routing protocol is:

$$E_t = 2*(E_{Tx\,(k,\,d)}+E_{Rx(k)})*C*(NC-2) + (C/2) * (E_{Tx\,(k,\,2d)} + 2 * E_{Tx\,(k,\,sqr(2)*d)} - 4 * E_{Tx\,(k,\,d)}) \quad (12)$$

3. Average Case

The average case scenario for the expended energy for the recovery phase happens when one intermediate is faulty in one half of the WSN chains. In this case the amount of expended energy for recovery can be written as:

$$E_r = (C/2)*(E_{Tx\,(k,\,2d)} - 2 * E_{Tx\,(k,\,d)}) \quad (13)$$

Thus the total amount of expended energy for the first proposed fault tolerant protocol is the sum of the expended energy for the detection phase plus the expended energy to recover from an intermediate faulty node, which is:

$$E_t = E_d + E_r = 2* (E_{Tx\,(k,\,d)}+E_{Rx(k)})*C*(NC-2) + (C/2)*(E_{Tx\,(k,\,2d)} - 2 * E_{Tx\,(k,\,d)}) \quad (14)$$

4.2.2 Second Proposed Protocol Expended Energy

Similar to the first proposed fault tolerant routing protocol, three are three distinguishable cases (best, worst, and average) for the expended energy by the recovery phase of the second proposed protocol.

1. Best Case

The best case for the recovery phase of the second proposed fault tolerant is identical to the best case for the recovery phase of the first proposed fault tolerant routing protocol. Therefore, there is no expended energy for the recovery phase of the proposed protocol. Hence, the total expended energy for the second proposed protocol is only due to the detection phase of the protocol and given in (8).

2. Worst Case

The worst case for the recovery phase of the second proposed protocol happens when every other chain leader node is faulty in the WSN under consideration. If all chain leader nodes are faulty this situation necessitates a new deployment and this rarely happens. The expended energy for the recovery phase in this case can be written as:

$$E_r = (C/2) * (4* E_{Tx\,(k,\,sqr(2)*d)} - 4 * E_{Tx\,(k,\,d)}) \quad (15)$$

Thus the total amount of expended energy by the second protocol becomes:

$$E_t = E_d + E_r = 2* (E_{Tx\,(k,\,d)}+E_{Rx(k)})*C*(NC-2) + (C/2) * (4* E_{Tx\,(k,\,sqr(2)*d)} - 4 * E_{Tx\,(k,\,d)}) \quad (16)$$

3. Average Case

The average case for the recovery phase of the second proposed protocol occurs when one intermediate node in half of the chains of the WSN under consideration are faulty. The expended energy for recovery by the second proposed protocol is:

$$E_r = (C/2) * (E_{Tx\,(k,sqrt(2)*d))} - E_{Tx\,(k,\,d)}) \quad (17)$$

$$E_t = E_d + E_r = 2* (E_{Tx\,(k,\,d)}+E_{Rx(k)})*C*(NC-2) + (C/2) * (E_{Tx\,(k,sqrt(2)*d))} - E_{Tx\,(k,\,d)}) \quad (18)$$

5. Simulation Results

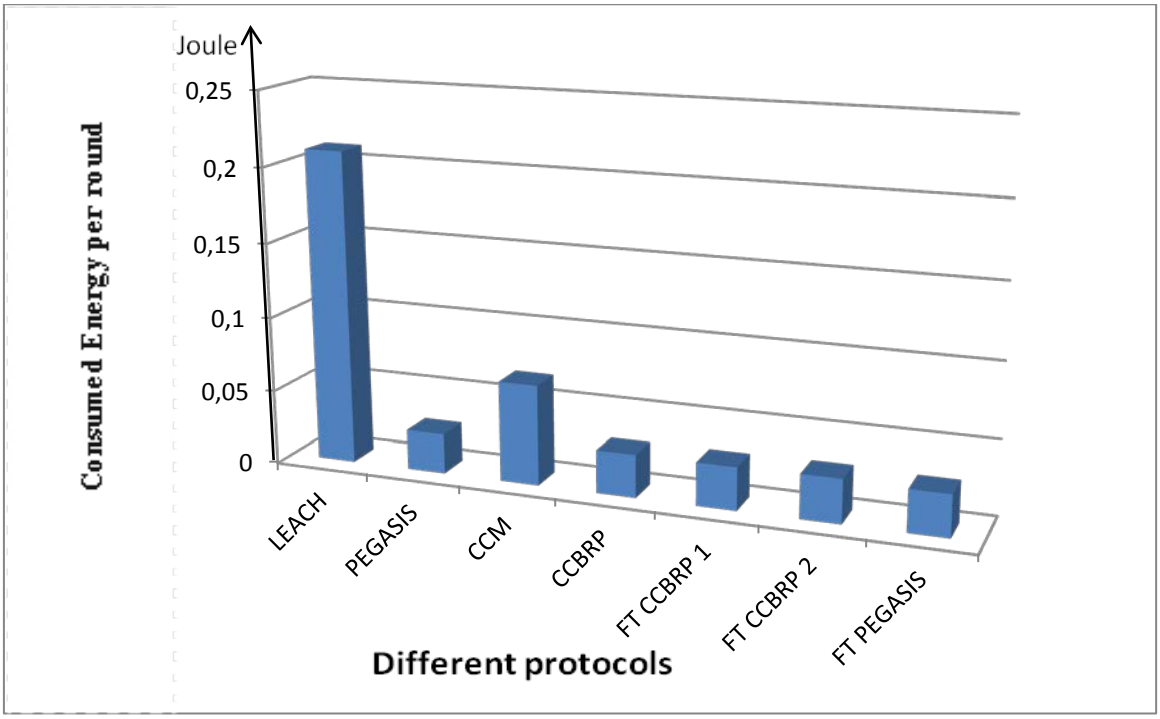
In this section, we provide experimental results to illustrate the applicability and efficiency of the two proposed fault tolerant routing protocols. More precisely, the first proposed fault tolerant routing protocol applied to both PEGASIS and CCBRP protocols. While the second proposed fault tolerant routing protocol due to its nature applied only to the CCBRP protocol. The simulation has been performed on a network consisted of one hundred sensor nodes, five chains each chain has twenty nodes, a coverage area of hundred by hundred and the base station is located at (50,300).Each sensor node is assumed to have an initial energy of one joule. We have randomly injected the simulation network with one fault per a chain.

The PEGASIS [12] protocol without applying the proposed fault tolerant consumed 0.0268J per round. Applying the proposed fault tolerant routing protocol on PEGASIS results in 0.001J consumed per round for fault detection, and 0.0004J for fault recovery. Thus the PEGSIS protocol consumed 0.0282J per round for both data transmission and fault tolerance. It is concluded from the simulation that our proposed protocol has expended 0.0014J to tolerate a single node failure. Hence, the percentage of the expended energy by our proposed protocol is 5.22% of the energy expended to transmit data with any fault tolerance. As for overhead time, the simulated network has 100 nodes without our proposed protocol data transmission consumed 100 time units. Our proposed protocol required 1.95 time units for detection and none for recovery. Therefore, the percentage of the overhead time by our proposed protocol for the PEGASIS protocol is 1.95%.

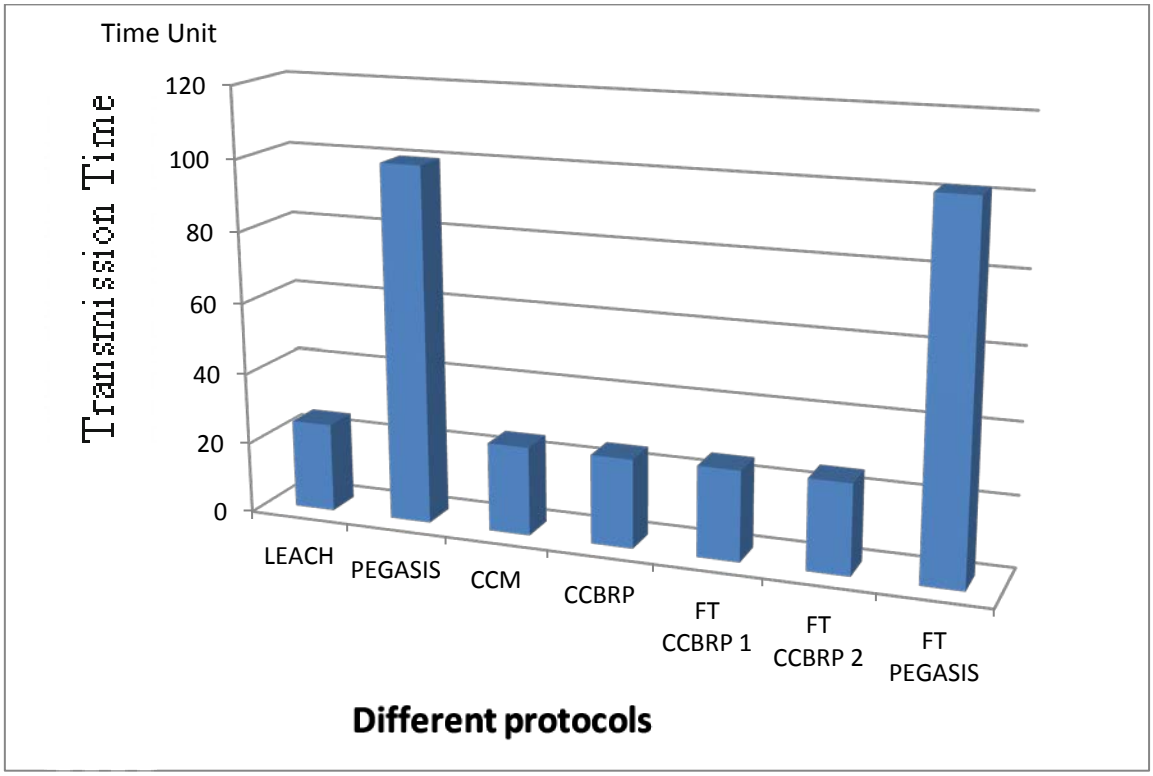
The simulation result for the CCBRP protocol without applying any of the two proposed fault tolerance protocols indicates that the energy consumed per round is 0.028636J. The detection phase of the two proposed protocols for the CCBRP protocol consumed 0.00008J per round. Applying the first proposed protocol on CCBRP protocol indicates that 0.000014J expended for recovery. Thus the percentage of the expended energy by the first proposed protocol for the CCBRP protocol for fault tolerance is 0.17%. The expended energy as a result of applying the second proposed fault tolerant protocol to the CCBRP protocol is 0.00088J. Therefore the percentage of the expended energy for fault tolerance by the second proposed protocol is 3.07%. The overhead time for applying either of the two proposed protocols to the CCBRP protocol is 0.39 time units. Thus the percentage of the overhead delay as a result of applying each of the two proposed fault tolerant protocol to the CCBRP protocol is 1.56%.

Table 1. The Time and Energy consumed by LEACH, PEGASIS, CCM, CCBRP, FT PEGASIS, FT CCBRP1, and FT CCBRP 2 protocols.

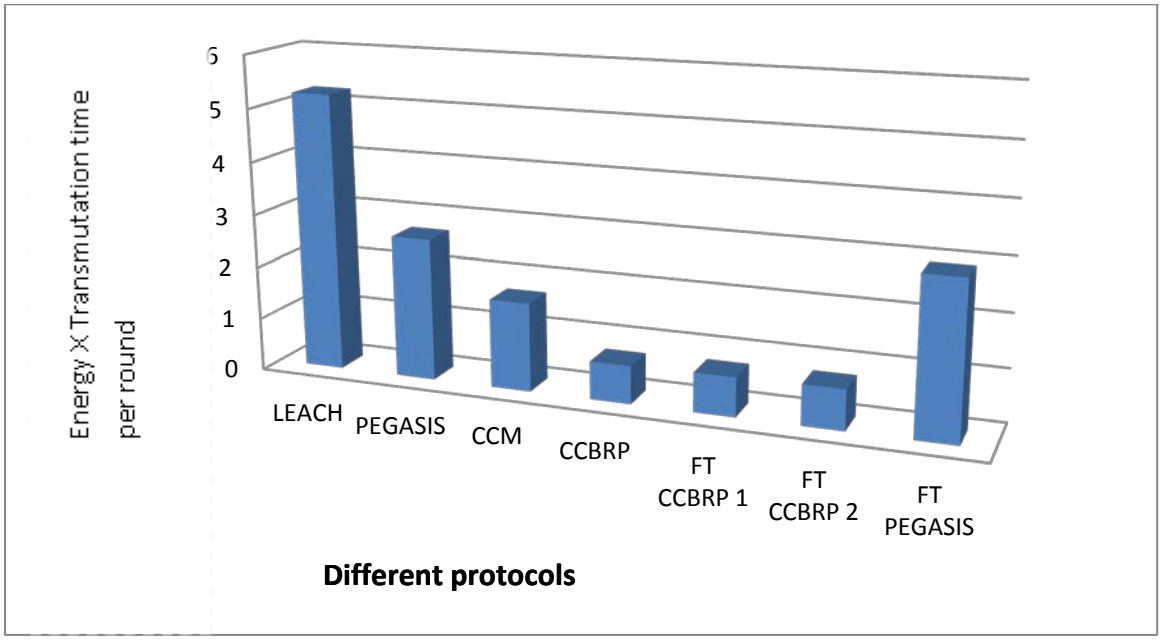
Routing Protocols	Consumed Energy per round in joule	Transmission time per round in time units	Energy × Transmission time
LEACH	0.21	25	5.25
PEGASIS	0.0268	100	2.68
CCM	0.067	25	1.68
CCBRP	0.028636	25	0.72
FT PEGASIS	0.028171	101.95	2.87
FT CCBRP 1	0.02873	25.39	0.73
FT CCBRP 2	0.02952	25.39	.75



a. Consumed energy per round



b. Transmutation delay



c. Energy × delay during a round.

Figure 8. Performance comparisons for LEACH, PEGASIS, CCM, CCBRP, FT CCBRP1, FT CCBRP2 and FT PEGASIS protocols.

In order to show the efficiency of the two proposed fault tolerant protocols a comparison is conducted (see Table 1 and Fig. 8) for chain clustered routing protocols with and without applying the two proposed protocols. It is clear from both Table 1 and Fig. 8 that PEGASIS, CCBRP, FT PEGASIS, FT CCBRP1, and FT CCBRP2 have the lowest expended energy. While the lowest expended energy achieved by CCM, CCBRP, FT CCBRP1, and FT CCBP2. Protocols CCBRP, FT CCBRP1, and FT CCBP2 have the best time energy product of all protocols. Also, the overhead for either of the two proposed fault tolerant for PEGASIS and CCBRP is very small compared with the result of the best protocols without providing fault tolerance. Moreover, the energy and time consumed by the two proposed fault tolerant for PEGASIS and CCBRPs for a single sensor failure per a chain is far less than similar protocols (LEACH, PEGASIS, and CCM) without any fault tolerance. Therefore, significant advantages can be gained using the either of the two proposed fault tolerant for chain clustered wireless sensor networks when node failures are expected.

6. Conclusion

In this paper we have presented two chain base fault tolerant protocols. The two protocols achieve fault tolerance with little extra energy and small delay over the same protocol without providing any fault tolerance. The two protocols employ the same strategy for fault detection; every node sends a NOTIFY message to its successor neighbor node, if the neighbor node is alive it replies with a READY message, otherwise, the node sending the NOTIFY message recognizes that its successor neighbor node is dead. However, the recovery strategy is different for the two protocols. The first protocol, FT_1, overcomes the faulty node by having every node predecessor to a failed node sends its data to the successor neighbor of the faulty node instead of the failed node itself. The second protocol, FT_2, gets around the faulty node by choosing a backup node which satisfies minimum energy consumption to replace it from a neighboring chain. The experimental results demonstrate that the energy consumption of either the two proposed fault tolerance protocols for a single faulty node per

chain is almost as same as for the PEGASIS and the CCBRP protocols without any fault tolerance. Moreover, the two proposed protocols for single node failure without any modification provide 90% fault coverage for double node failure in any chain. Therefore, significant advantages can be gained using the proposed protocol chain based clustered wireless sensor networks when node failures are expected. It worth noting that the fault detection and recovery phases for both the two proposed protocols do not have to be applied at each data round. It can be applied in varying intervals depending on the application and the environment in which the chain based clustered WSN is deployed.

References

[1] Akyildiz I., Weilian S., SankaraSubramaniam Y., Cayirci. E, "A survey on sensor networks", IEEE Communications Magazine, Vol. 40, Issue 8. Pp 102-114, August 2002. http://dx.doi.org/10.1109/MCOM.2002.1024422

[2] I.F. Akyildiz , W. Su, , Y. Sankarasubramaniam , E. Cayirci, Wireless sensor networks: a survey. Computer Networks Volume 38, Issue 4, 15 March 2002, Pages 393–422. http://dx.doi.org/10.1016/S1389-1286(01)00302-4

[3] Heinzelman, W., Chandrakasan, A., Balakrishnan, H., "Energy –Efficient Communication Protocol for Wireless Microsensor Networks", In: Proceeding of the 33rd Annual Hawaii International Conference on System Sciences (HICSS), Big Island, USA, Pp 3005–3014, 2000

[4] Shah, C. and Rabaey, M., "Energy aware routing for low energy ad hoc sensor networks", IEEE Wireless Communications and Networking Conference (WCNC2002.), March 18-21, 2002 , Orlando, Florida, USA, pp. 350-355. http://dx.doi.org/10.1109/WCNC.2002.993520

[5] Zhou, J. and De Roure, D., "Designing Energy-Aware Adaptive Routing for Wireless Sensor Networks", the 6th International Conference on ITS Telecommunications (ITST 2006), Chengdu, China, 21 - 23 June, 2006.

[6] Perillo M., Cheng Z., Heinzelman W., "On the problem of unbalanced load distribution in wireless sensor networks", Global Telecommunications Conference Workshops, 29 Nov.-3 Dec. 2004, Pp. 74 – 79. http://dx.doi.org/10.1109/GLOCOMW.2004.1417552

[7] AhnS., Kim D., "Proactive context-aware sensor networks" , In European Workshop on Wireless Sensor Networks (EWSN 2006), Zurich, Switzerland. February 13-15, 2006.

[8] Lindsey S., Raghavendra C., "PEGASIS: Power Efficient gathering in sensor information systems", In Proceedings of IEEE Aerospace Conference, Big Sky, Montana (USA), March 9-16, 2002. http://dx.doi.org/10.1109/AERO.2002.1035242

[9] Abbasia A., YounisM. ," A survey on clustering algorithms for wireless sensor networks", Computer Communications", Vol. 30, Issue 15, Pp. 2826-2841, October 2007

[10]Chang J., Tassiulas L., "Maximum lifetime routing in wireless sensor networks", IEEE/ACM Transactions on Networking, Vol. 12, Issue. 4, Pp 609-619, Aug. 2004. http://dx.doi.org/10.1109/TNET.2004.833122

[11] Tang F., You I., Guo S., Guo M., Ma, "A chain-cluster based routing algorithm for wireless sensor networks", In: journal of intelligent manufacturing, Volume 23, Number 4 (2012), Pp. 1305-1313, DOI: http://dx.doi.org/10.1007/s10845-010-0413-4.

[12] Ali S., Refaay S., "Chain-Chain Based Routing Protocol", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, Pp 1694-0814, May 2011

[13] Hazarath M., "A Fault Tolerant Trajectory Clustering (FTTC) for selecting cluster heads in wireless sensor networks", International Journal of Computational Intelligence Research (IJCIR), Vol. 6, Issue 3, Pp 359-372 , 2010

[14] Gupta G. and Younis M., “Fault-Tolerant Clustering of Wireless Sensor Networks”, Proc. IEEE Wireless Comm. and Networking Conf. Vol. 3, Pp 1579 – 1584, 2003. DOI: http://dx.doi.org/10.1109/WCNC.2003.1200622

[15] Min H., Jung J. , Kim B., Cho J., Heo J., Hong J. , “A Smart Checkpointing Scheme for Improving the Reliability of Clustering Routing Protocols”, Sensors, Vol. 10, Pp. 8938 – 8952, 2010. DOI: http://dx.doi.org/10.1109/10.3390/s101008938

[16] Dugan J., Trivedi K., "Coverage Modeling for Dependability Analysis of Fault-Tolerant Systems", IEEE Transactions on Computers, Pp. 775-87, June 1989. http://dx.doi.org/10.1109/10.1109/12.24286