A Tree-Based Power Saving Routing Protocol for Wireless Sensor Networks

Iman ALMomani, *MIEEE*, Maha Saadeh, Mousa AL-Akhras, *MIEEE*, and Hamzeh AL Jawawdeh

***Abstract*—**Wireless Sensor Networks (WSNs) are a recent promising technology used to facilitate the provision of many civilian, military, and industrial services. Many challenges hinder the effective use of WSNs to support different applications, such as the resource limitations of sensor devices and the finite battery power. This paper proposes a new routing protocol that considers sensors power limitation and prolongs the network's lifetime by avoiding excessive messages between nodes. This protocol is based on Tree Routing (TR). It routes the data over the shortest path using parent-child links in accompany with neighbors' links. Also, it solves the problem of node's failure. The proposed protocol is analyzed and compared with other tree-based routing protocols.

***Keywords*—**Power Consumption, Shortest Path, Tree-Based Routing, Wireless Sensor Network, WSN.

I. INTRODUCTION

WIRELESS Sensor Networks (WSNs) are infrastructure-based networks that consist of small sensors scattered in the sensing environment and one or more sink node(s). Sensors are used to collect environmental data and send them to the sink [1]. They are embedded with a microprocessor and a wireless transceiver along with the sensory unit providing data processing and communication capabilities in addition to the sensing facility [2]. Fig. 1 shows the structure of the sensor device. The sink node is used to process sensed data and connect the sensor network to the Internet. Sink node is usually a powerful device that is connected to a power supply.

WSNs are envisaged to become a very significant enabling technology in many sectors as they are widely used in many civilian and military as well as industrial applications. The significant interest in WSNs comes from its successful use in many applications such as environment monitoring, disaster relief, emergency rescue operations, biomedical and healthcare applications and others [1], [3]. Some of these applications are considered sensitive in which the data should be private and confidential. Other applications need high reliability and bounded delivery time. To support these applications it is important to design and implement resource-efficient routing protocols that will transmit the data while considering nodes resource limitations and attempt to give the best power saving to prolong the network's life time and increase its availability.

Several challenges hinder the effective use of WSNs, one challenge is the resource constraints of sensor devices such as limited computational capabilities, limited storage, and short radio range. The design and implementation of WSNs protocols and operations should consider theses resource limitations. Another important challenge in WSNs is the finite power amount; sensors usually run using a battery with finite power rather than a mains-power supply. Any protocol proposed for WSN should consider this limited power and attempt to prolong the network's lifetime by reducing the amount of power consumption.

There are many factors affecting the network energy consumption such as nodes distance, number of sent and received messages between nodes, message size, number of intermediate nodes between source and destination, and the required level of local data processing. For routing protocols, the challenge is to route the message using a suitable path and at the same time to prolong the network's life time by avoiding excessive message exchange between nodes, thus, reducing the overall consumed power.

In this paper a new tree-based routing protocol is proposed. The goal of this protocol is to prolong the network's lifetime by considering sensors power limitation and avoiding excessive messages between nodes. A tree will be constructed between network nodes and a new addressing scheme will be used during the tree construction to assign logical addresses to the network nodes. Each node should maintain neighbors' information such as neighbor's logical address, MAC address, and power in its neighbors table.

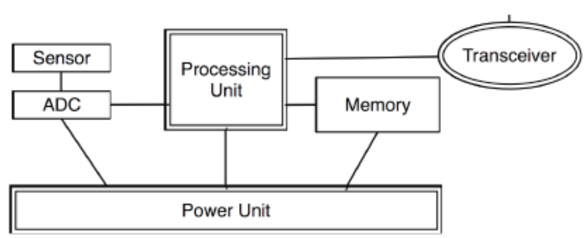


Figure 1: Sensor Device

Manuscript received March 11, 2011: Revised version received April 4, 2011.

Iman ALMomani is an Assistant Professor in Wireless Networks and Security with the Computer Science Department, King Abdullah II School for Information Technology (KASIT), The University of Jordan, P.O. Box 13835, Amman, 11942, Jordan (phone: +962-6-5355-000 extension 22570; fax: +962-6-3500-233; e-mail: i.momani@ju.edu.jo).

Maha Saadeh is a M.Sc. student with the Computer Science Department, King Abdullah II School for Information Technology (KASIT), The University of Jordan, (e-mail: saadeh.maha@yahoo.com).

Mousa AL-Akhras is an Assistant Professor in Artificial Neural Networks & Communications with the Computer Information Systems Department, King Abdullah II School for Information Technology (KASIT), The University of Jordan, (e-mail: mousa.akhras@ju.edu.jo).

Hamzeh AL Jawawdeh is a M.Sc. student with the Computer Science Department, King Abdullah II School for Information Technology (KASIT), The University of Jordan, (e-mail: hamzeh@jawawdeh.com).

The proposed protocol consists of many stages. Firstly, it constructs a logical tree between network nodes, assigns addresses to each one and builds the neighbor tables. Secondly, it exploits neighbor links to transmit messages considering intermediate nodes energy during transmission. Finally, it solves the consequences of node failure and new nodes entrance.

The rest of the paper is organized as follows: section 2 summarizes some of the related works. Section 3 discusses the proposed routing protocol. The analysis and comparison are presented in section 4. Finally, section 5 concludes the paper and presents possibilities for future work.

II. Related Work

Many tree-based routing protocols have been proposed for WSNs. These protocols can be classified according to the route discovery and network structure.

According to route discovery, a protocol can be reactive, proactive, or a hybrid between them. Reactive protocols find a route on demand by flooding the network with route request packets. The advantage of this type of protocols is that it does not have to keep routing information to all nodes. The main disadvantages are high latency time in route finding and overhead of excessive flooding [4]. A well-known example of this category is Adhoc On demand Distance Vector (AODV) protocol [5]. AODV attempts to find the shortest possible route. When a node requires a route, it initiates a route discovery procedure and broadcast route request (RREQ) messages. When receiving RREQ, a node creates and sends a route reply (RREP) message back to the originator node if either it has a valid route entry to the destination or it is the destination itself. Each node maintains route entries with forward and backward next hop information along with their expiry time.

Proactive protocols, on the other hand, maintain fresh lists of destinations and their routes by distributing routing tables throughout the network. The tables are updated either periodically or when changes are recognized. The main disadvantage is the overhead of maintaining and updating the routing tables. Hybrid protocols use a combination of the above two ideas [4].

According to network structure, a routing protocol can be either flat or hierarchical. In hierarchical routing, network nodes are connected to form a specific structure such as a cluster or a tree. In [6] both clusters and trees are constructed between network nodes. First, nodes are grouped into clusters and a cluster head is elected for each group, then a tree routing is used for inter-cluster communication. In [7] the authors proposed a cluster-based protocol that considers different factors to choose the cluster head and to avoid creating redundant cluster heads within a small geographical range. Other hierarchical based routing protocols are proposed in [8]-[10].

One protocol proposed for WSNs is the Tree Routing (TR) which is supported by IEEE 802.15.4 [11]. It is suited for small memory, low power and low complexity networks with lightweight nodes. This protocol aims to eliminate the overhead of path searching and updating, therefore, it reduces extensive messages that are exchanged between network nodes. Two parameters are used by the TR protocol to control the tree construction. These parameters are the maximum number of children a node can have, and the maximum depth of the tree. As the number of children increases/decreases the depth will decrease/increase. An address scheme is used to assign logical network addresses to the network nodes.

Although this protocol works well, it suffers from two drawbacks. First, message transmission depends on source depth; the deeper the node, the longer the path. Second, it suffers from node/link failure that may cause nodes isolation.

To overcome the drawbacks of TR, many protocols have been proposed to enhance TR performance. The authors in [4] proposed a Plus Tree (PT) routing protocol that utilizes the neighbors' links in order to find the shortest path to sink. To transmit a message, PT first constructs the parent-child links and then each node broadcast its ID to construct the neighbors' tables. Although PT finds the shortest path and solves link failure problem, it does not consider energy consumption in its solutions. In [5] the authors proposed Enhance Tree Routing (ETR) protocol for ZigBee networks. In this protocol each node should maintain information about its neighbors in a neighbors table. A structured address assignment scheme is used to assign addresses to tree nodes. Then the relations between nodes' addresses are exploited to find a shorter path to sink. Another protocol for ZigBee networks is ImpTR as proposed in [12]. However, both ETR and ImpTR do not take network energy into consideration.

In this paper we propose a new tree-based routing protocol that will consider both number of intermediate nodes to the destination and residual energy at each node.

III. PROPOSED ROUTING PROTOCOL

The proposed tree-based routing protocol aims to enhance the tree routing and attempts to prolong the network's life time. The proposed protocol consists of different stages. Firstly, it constructs a logical tree between network nodes. During the construction each node gets an address (ID) and constructs the neighbors table. Secondly, it exploits neighbor links to transmit messages taking into consideration intermediate nodes energy and depth while transmitting. Finally, the tree, or part of it, is reconstructed due to node's failure and new nodes entrance. In the next subsections, the protocol stages will be discussed in details. Different control messages are defined for this protocol. These messages are listed in Table I with their descriptions.

A. *Network Model*

The network model is described as follows:

1) Sensors are scattered in the network field without isolation.
2) All sensor nodes have same capabilities, same transmission ranges and limited power resources.

3) Symmetric model is assumed. That means if sensor A is located within the transmission range of sensor B, then B is also located within A's transmission range.
4) All sensors sense data and transmit it to the sink for processing.
5) The sink node assumed to have unconstrained resources.
6) All sensor nodes are located in fixed places without mobility.

B. Proposed Protocol stages

The proposed protocol consists of four stages: logical tree construction, message transmission, new node entrance, and node failure. This subsection will discuss these stages in details:

Tree Construction: A sink-rooted tree is constructed between network nodes before sending a message to the sink. A new addressing scheme is used in this stage to assign a logical ID for each node. Each node uses the ID to calculate its depth and its neighbors' depth. When a node receives an *Engagement-Acceptance* message it calculates its own ID as *parentID || offeredID* where *offredID* is represented by m digits representing the number of digits required to represent C_{max} nodes where Cmax is the maximum number of children. For example if $C_{max} < 10$ then we need only one digit to represent the offered ID 0…9, but for large networks if $C_{max} < 100$ then we need 2 digits 00…99, and so on. Using this addressing scheme, each node will be able to know the depth for a particular ID.

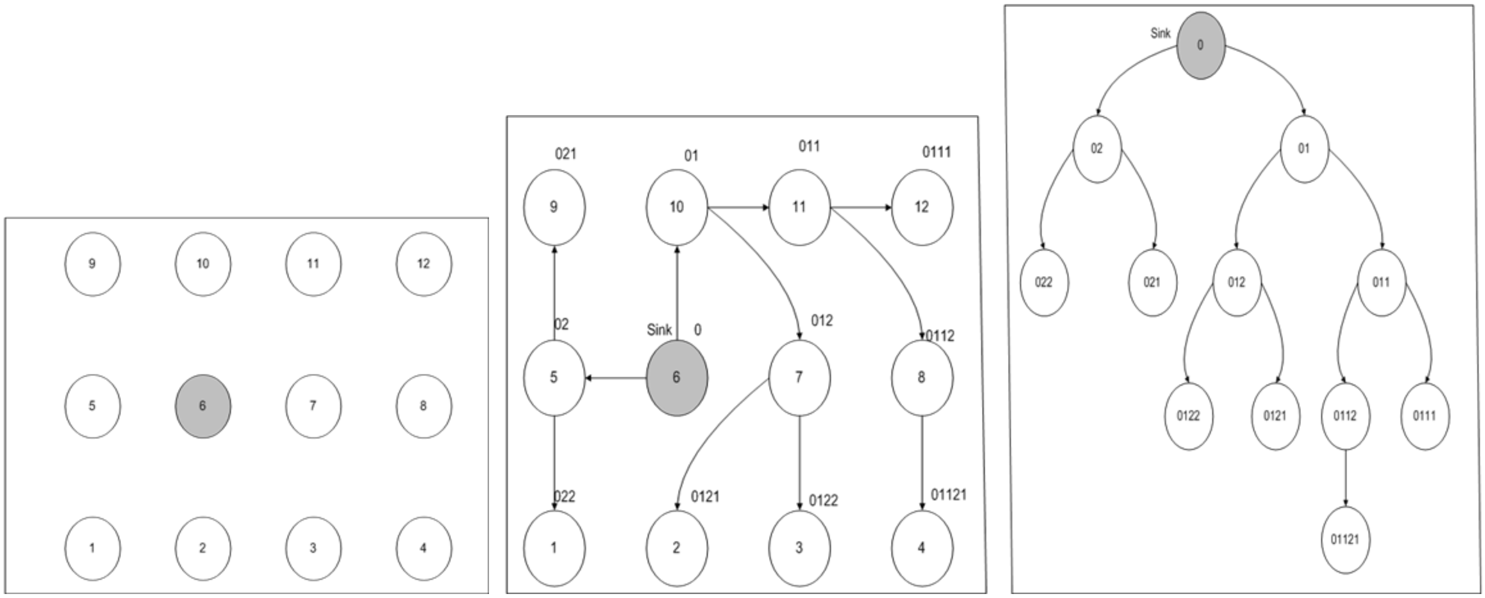
To construct the tree we assume that all nodes, initially, do not have IDs and they have full energy. Parents can have at most C_{max} children, and at the end of this stage each node should have an ID from which it can calculate its depth. Fig. 2 shows an example of logical tree construction.

The sink node will start the tree construction by broadcasting a *Ready* message to its neighbors, this message contains the sink ID (sink ID = initial ID), and sink energy. Each node receives this message will store the sink information in its neighbors table and after a short period, it will send an *Engagement* message to the sink. The *Engagement* message has two purposes: request for a parent, and request for an ID. For each *Engagement* message, the sink node will reply by sending *Engagement-Acceptance* message only if its children are less than C_{max}.

When a node gets the *OfferedID* it will calculate its ID and broadcast a *Ready* message allowing its neighbors to send *Engagement* messages. Note that the Engagement messages are sent after a short period during which they can receive *Ready* messages from other nodes. This waiting will force the node to be associated with the best possible node among others (the node that has the maximum amount of power), thus, keeping the balance between nodes. This process will continue until all nodes get IDs and no more *Ready* messages are sent. The message flow of this stage is illustrated in Fig. 3.

Table I: Proposed Protocol Control Messages.

Message Type	When To Send	Actions By Receivers	Message Structure
Ready	Is sent when: 1. The node gets an ID, so it is used to broadcast the ID and tell other nodes that it is ready to accept children. 2. The node receives a *NewNode* or *RequestParent* messages.	1. Store the information in the neighbors table. 2. If receiver node does not have ID, it will send *Engagement* message to the node with the maximum power.	1- Node ID 2- Node power
UnReady	Used only to broadcast the ID.	Store ID in the neighbor table.	1- Node ID 2- Node power
Engagement	Sent when receiving a *Ready* message and used as request for ID and request for parent.	May send *Engagement-Acceptance* message if its children are less than C_{max}.	
Engagement-Acceptance	Sent as a reply to an *Engagement* message.	1- Refresh neighbors table. 2- Calculate the ID. 3- Send *Ready* message	1- Node ID 2- Node power 3- Offered ID
NewNode	Sent when a new node wants to join the network.	Send either *Ready* message or *UnReady* message.	
RequestParent	Sent when a node cannot reach its parent.	Send either *Ready* message or *UnReady* message.	
Inform	Sent to tell the neighbors that the node will go down.	Reconstruct the tree according to their relation with the dead node.	Node ID
ChangeID	Sent when any node change its ID due to some failure to tell other nodes to modify the ID in their tables.	1- Modify the ID in neighbors table. 2- If the receiver is one of sender's children, it will update its own ID and send *ChangeID* message.	1- Node ID 2- Node power



(a) (b) (c)

Figure 2: An Example of Tree Construction. (a) Original Node Distribution, (b) The Constructed Tree Using the Proposed Protocol, (c) The Tree Logical View.

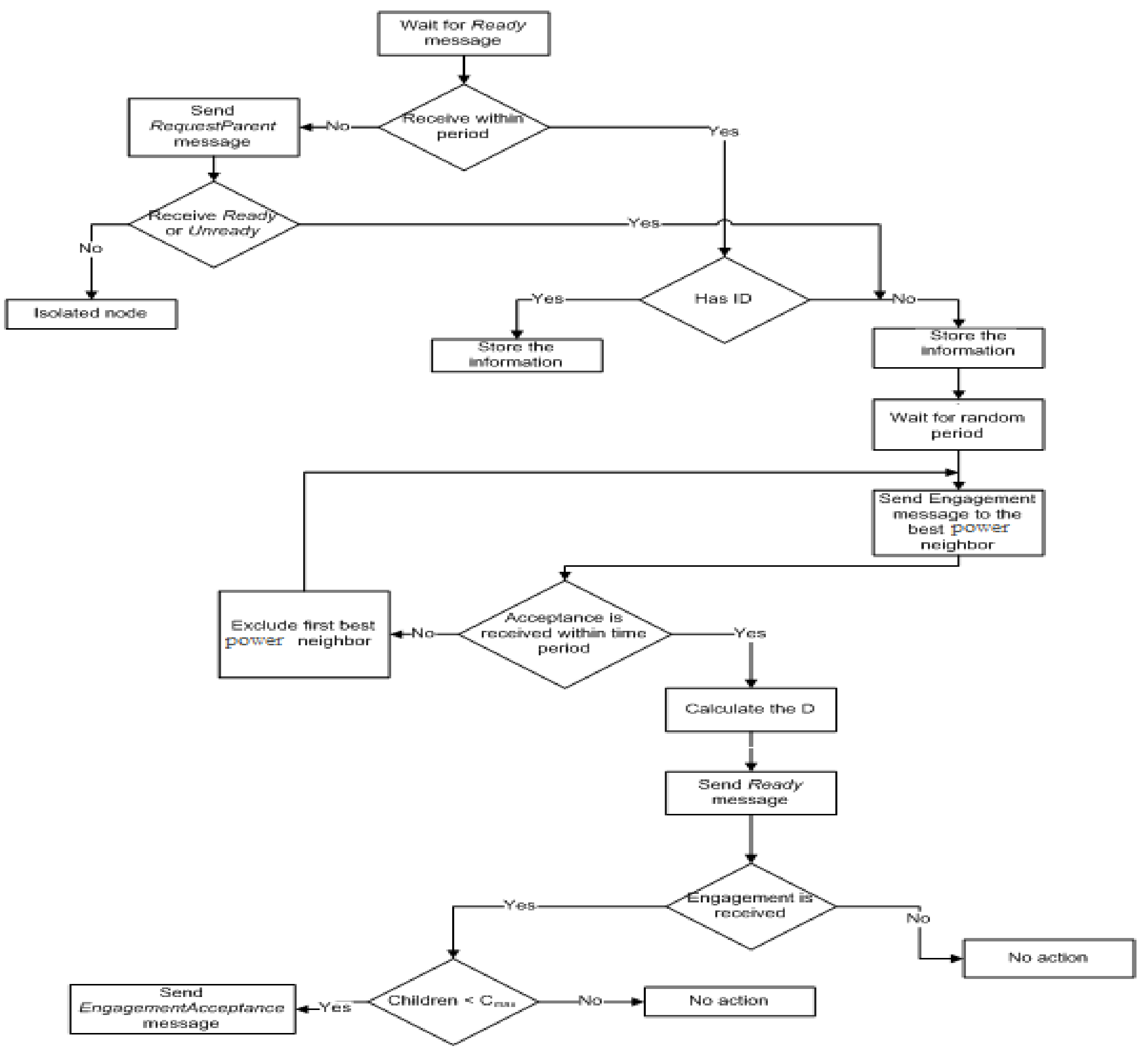


Figure 3: Message Flow during Tree Construction Stage.

Message Transmission: As stated earlier, the tree should be rooted at the sink node and all other nodes will send data to the sink node. The message should be forwarded over the best path. To choose the next hop, the sender will consider both neighbors depth and power. The neighbor that has the minimum depth and a power larger than a specific threshold will be chosen. If all smaller depth neighbors have critical energy then the sender will send the data through the parent. In this way the load is balanced between nodes instead of overloading the parent node as in TR or the less depth neighbor node as in [4], [5], [12].

New Node Engagement: When any node wants to join the network, it should broadcast a *NewNode* message. Then all neighbors will reply by either *Ready* message if number of children < C_{max}, or *UnReady* message if number of children = C_{max}. The new node will store all information in its neighbors table and will be associated with the parent that has the maximum power. Then it will broadcast a *Ready* message. The message flow during new node engagement stage is illustrated in Fig. 4.

Tree Reconstruction: If a node's energy reaches a specific threshold, it should inform its parent, children, and neighbors that it will go down by broadcasting its ID in an *Inform* message, so that they can take an action and prepare themselves to reconstruct the tree. The message flow for tree reconstruction is shown in Fig. 5.

Each node has a relation with the dead node should take an action. There are three possible cases; the first one when the dead node is a parent. In this case each child has to find a new parent. Each child broadcasts a *RequestParent* message and only neighbors with children less than C_{max} will reply by a *Ready* message, other nodes send *UnReady* message. The child then chooses the node that has the maximum power as a new parent. Then it will broadcast *ChangeID* message to its neighbors to update the ID in their neighbors' tables. If any neighbor is a child for this node it will change its ID and broadcast *ChangeID* message. This process continues until all affected nodes modify their IDs.

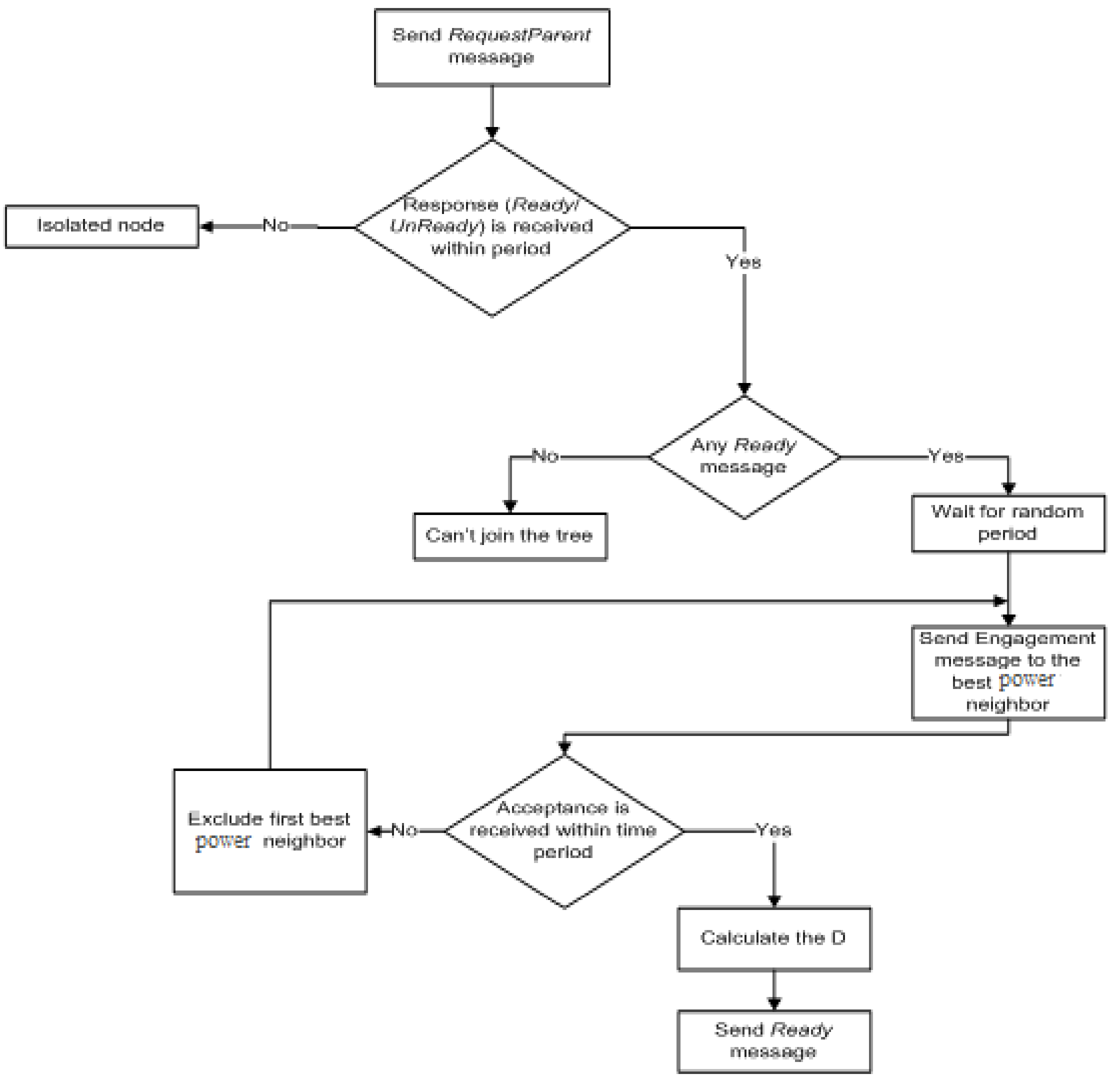


Figure 4: Message Flow during New Node Engagement stage.

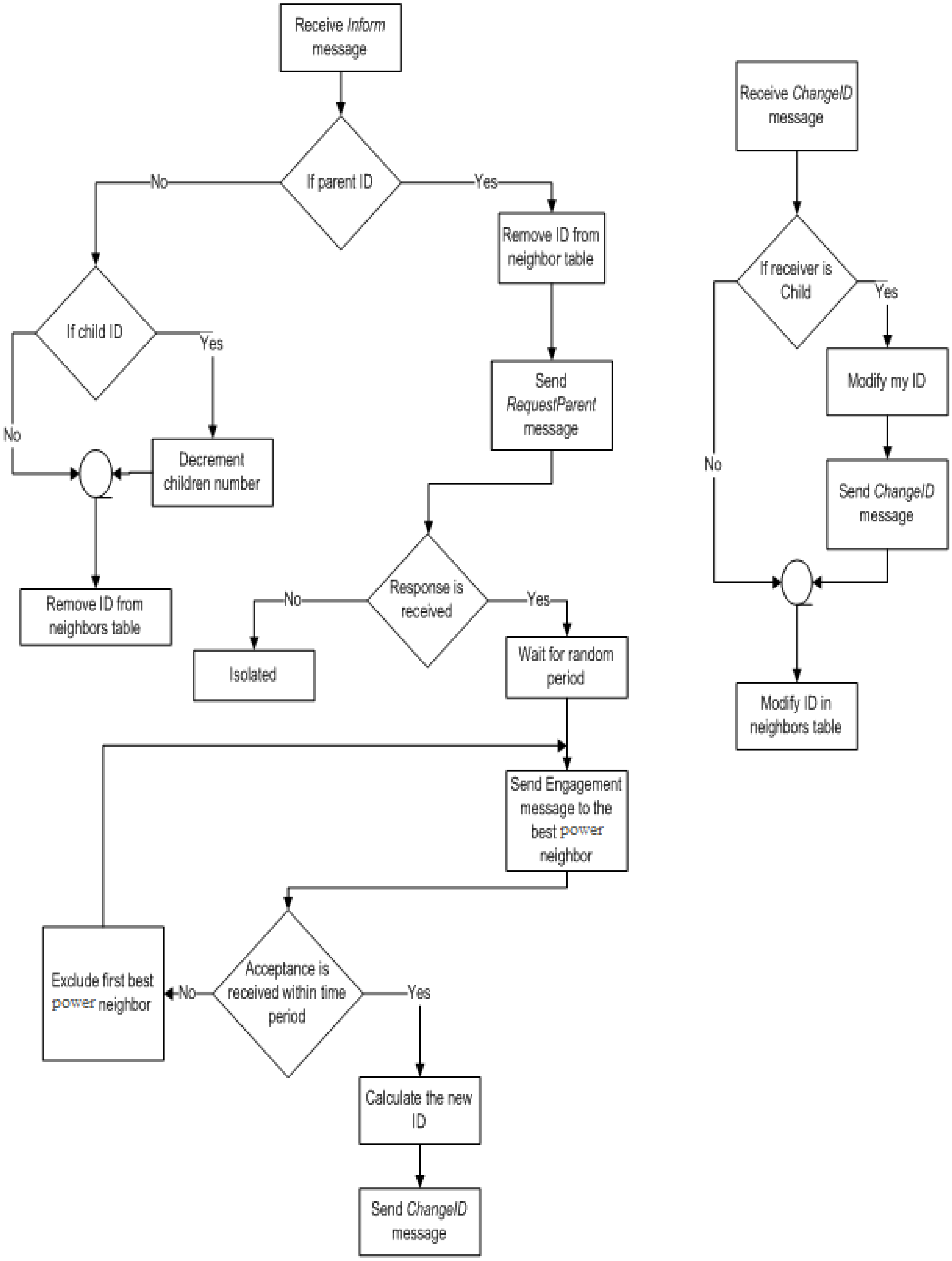


Figure 5: Message flow During Tree Reconstruction Stage.

The second case is when the dead node is a child. In this case parent should remove the node from its neighbors table and decrement the number of children. The last case is when the dead node is a neighbor then neighbors will remove it from their neighbors' tables.

In some cases, a node goes down before informing other nodes that it ran out of power. In this case when a neighbor node, whether a child or a parent, discovers this absence, it should broadcast the dead node's ID in an *Inform* message and then each node will take an action as discussed above.

IV. ANALYSIS AND COMPARISON

In this section the proposed protocol is compared with PT protocol [4] in terms of number of sent and received messages during tree construction and the consumed power due to messages exchange.

A. Number of Sent and Received Control Messages

This subsection shows the maximum number of messages that are exchanged between network nodes during the tree construction in both PT and the proposed protocol. Table II illustrates the comparison between the two protocols.

Proposed protocol: Three types of messages are used during tree construction. The first one is *Ready* message where each node sends one *Ready* message to broadcast its ID. For N nodes the total is N *Ready* messages. Note that each node will receive at most $Ne(n_i)$ *Ready* messages from its neighbors, where $Ne(n_i)$ is the neighbors for node i. This number of received messages can be characterized as $\sum_{i=1}^{N} Ne(n_i)$.

The second message is *Engagement* message where in the worst case each node will send an *Engagement* message to every received *Ready* message. Each node expects to receive an *Engagement* message from its neighbors. The maximum *Engagement* messages that will be received at each node = $Ne(n_i)$, the total number of received messages by all nodes is characterized as $\sum_{i=1}^{N} Ne(n_i)$.

Table II: Comparison between the Proposed and PT Protocol in Terms of Required Control Messages for Tree Construction.

Plus-Tree (PT) Protocol			**Proposed Protocol**		
Message	**Number of Sent Messages**	**Number of Received Messages**	**Message**	**Number of Sent Messages**	**Number of Received Messages**
Association	Each node sends one, the total is N messages	Each node receives $Ne(n_i)$ the total = $\sum_{i=1}^{N} Ne(n_i)$	***Ready***	Each node sends one, the total is N messages.	Each node receives $Ne(n_i)$ the total = $\sum_{i=1}^{N} Ne(n_i)$
Association-Reply	Each node will send *Association-Reply* to every *Association*, the total = $\sum_{i=1}^{N} Ne(n_i)$	Each node expects to receive Replies from its neighbors. the total = $\sum_{i=1}^{N} Ne(n_i)$	***Engagement***	Each node will send *Engagement* to every received *Ready* message, the total = $\sum_{i=1}^{N} Ne(n_i)$	Each node expects to receive an *Engagement* from its neighbors. the total = $\sum_{i=1}^{N} Ne(n_i)$
ID	N ID messages	Only one ID message contains the ID	***Engagement-Acceptance***	N *Engagement-Acceptance*	Only one *Engagement-Acceptance* message with the *OfferedID*
HelloNeighbor	N messages	Each node receives $Ne(n_i)$, the total = $\sum_{i=1}^{N} Ne(n_i)$			
ReplyHello-Neighbor	Each node will send *ReplyHello-Neighbor* to every *HelloNeighbor* message the total = $\sum_{i=1}^{N} Ne(n_i)$	Each node receives a *ReplyHello-Neighbor* from its neighbors. the total = $\sum_{i=1}^{N} Ne(n_i)$			

The third message is the *Engagement-Acceptance* message where each node will send at most C_{max} *Engagement-Acceptance* messages since it can have at most C_{max} children. The total is N messages since each node can get only one *OfferedID*. On the other hand, each node will receive only one *Engagement-Acceptance* message with the *OfferedID*.

PT Protocol: In this protocol the root (parent) sends an *Association* message to its neighbors, therefore neighbors can be attached to this parent by sending an *Association-Reply* message. The parent checks if it can accept the child then it responds by sending a message containing the logical ID. For N nodes the total is N *Association* messages and $\sum_{i=1}^{N} Ne(n_i)$ *Association Reply* messages. As each node may have only one ID, so there will be only N parents' ID responses. When the tree is constructed, each node will broadcast its ID and collect its neighbors' IDs to construct the neighbors table.

B. The Consumed Power

Sensor power is affected by local processing and communication operations. Since communication operations consume more power than data processing, sensors lose most of their power during sending and receiving of messages [2]. According to [13], the node requires $E_{Tx}(k,d)$ to send k bits message to destination at distance d, and $E_{Rx}(k)$ to receive k bits message. $E_{Tx}(k,d)$ and $E_{Rx}(k)$ are defined as:

$$E_{Tx}(k,d) = E_{Tx-elec}(k) + E_{Tx-amp}(k,d) = E_{elec} * k + \varepsilon_{amp} * k * d^2 \quad (1)$$

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

Where $E_{elec} = 50\ nJ/bit$, and $\varepsilon_{amp} = 100\ pJ/bit/m^2$.

Utilizing Eq.1 and Eq.2 the maximum power that will be consumed during the tree construction can be calculated. Table III illustrates the comparison between the proposed protocol and the PT protocol. As illustrated in the table, the proposed protocol requires less number of messages to construct the tree, consequently, less consumed power.

V. CONCLUSIONS AND FUTURE WORK

In this paper a tree-based routing protocol for WSNs that considers both shortest path and energy balance between nodes is proposed. The proposed protocol consists of different stages; sink-rooted tree construction, messages transmission, and node failure problem solving. The protocol is compared with other tree-based protocols such as Plus Tree (PT) routing protocol. The results showed that the new protocol is more energy-efficient than the PT protocol. As a future work we will implement this routing protocol to consider more sophisticated scenarios and compare it with other related protocols.

Table III: Comparison between the Proposed Protocol and PT Protocol.

	Plus Tree (PT)	Proposed Protocol
Sent Messages (SM)	$3N + 2\sum_{i=1}^{N} Ne(n_i)$	$2N + \sum_{i=1}^{N} Ne(n_i)$
Received Messages (RM)	$N + 4\sum_{i=1}^{N} Ne(n_i)$	$N + 2\sum_{i=1}^{N} Ne(n_i)$
Consumed Power	$E_{Tx}(k,d)$ * # SM + $E_{Rx}(k)$ * # RM	$E_{Tx}(k,d)$ * # SM + $E_{Rx}(k)$ * # RM

REFERENCES

[1] J. Al-Karaki, and A. Kamal, "Routing Techniques in Wireless Sensor Networks: A Survey," *IEEE Wireless Communications*, vol.11, no.6, pp. 6-28, December 2004.

[2] A. Jamalipour, and J. Zheng. *Wireless Sensor Networks: A Networking Perspective*, Wiley-IEEE Press, 2009.

[3] R. Biradar, V. Patil, S. Sawant, and R. Mudholkar, "Classification and Comparison of Routing Protocols in Wireless Sensor Networks," *Special Issue on Ubiquitous Computing Security Systems, UbiCC Journal*, vol.4 , pp.704-711, 2009.

[4] Y. Park, and E. Jung, "Plus-Tree: A Routing Protocol for Wireless Sensor Networks," *Lecture Notes in Computer Science LNCS*, vol. 4413, pp.638–646, 2007.

[5] W. Qiu, E.Skafidas, and P. Hao, "Enhanced Tree Routing for Wireless Sensor Networks," *Ad Hoc Networks*, vol.7, no.3, pp.638–650, May 2009.

[6] I. Almomani, and E. Almashakbeh, "A Power-Efficient Secure Routing Protocol for Wireless Sensor Networks," *WSEAS TRANSACTIONS on COMPUTERS*, vol.9, no.9 , pp.1042-1052, September 2010.

[7] T. Kang, J. Yun, H. Lee, I. Lee, H. Kim, Byunghwa LEE, Byeongjik, Lee, K. Han, "A Clustering Method for Energy Efficient Routing in Wireless Sensor Networks," *Proceedings of the 6th WSEAS Int. Conf. on Electronics, Hardware, Wireless and Optical Communications*, Corfu Island, Greece, 16-19 February, 2007, pp. 133-138.

[8] E. Saad, M. Awadalla, and r. Darwish, "Adaptive Energy-Aware Gathering Strategy for Wireless Sensor Networks," *International Journal Of Computers*, vol.2, no.2 , pp.148-157, 2008.

[9] E. Saad, M. Awadalla, and R. Darwish, "Energy Aware Distributed Clustering Scheme for Wireless Sensor Networks," *International Journal Of Computers*, vol.2, no.3 , 201-209, 2008.

[10] M. Zhang, C. Gong, Y. Feng, and C. Li, "Energy Predicted Shortest Routing Tree Algorithm in Wireless Sensor Networks," *Advances in Computation and Intelligence, Lecture Notes in Computer Science, LNCS*, vol.5370, pp.117-124, 2008.

[11] IEEE, *ZigBee Specification Version 1.0, ZigBee Alliance*. IEEE, 2005.

[12] M. Al-Harbawi, M. Rasid, and N. Noordin, "Improved Tree Routing (ImpTR) Protocol for ZigBee Network," *International Journal of Computer Science and Network Security*, vol.9, no.10, pp.146–152, October 2009.

[13] W. Heinzelman, A. Sinha, A. Wang, and A. Chandrakasan, "Energy-Scalable Algorithms and Protocols for Wireless Microsensor Networks," *International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, vol.6, Istanbul, Turkey, 2000, pp.3722-3725.

Iman M. AL-Momani obtained her B.Sc. degree in computer science from the United Arab Emirates University, UAE, in 2000, her M.Sc. degree in computer science from the University of Jordan, Amman, Jordan in 2002. Then she worked as teaching assistant at the University of Jordan. She earned her Ph.D. degree in 2007 from De Montfort University, Leicester, UK. Her Ph.D. specialization is wireless networks and security.

She works as an Assistant Professor in the Computer Science Department, King Abdullah II School for Information Technology at the University of Jordan since 2007. Her research interests include wireless mobile ad hoc networks (WMANETs), Wireless Sensor Networks (WSNs), and security issues in wireless networks.

Dr. ALMomani is a member of IEEE. She is in the organizing and technical committees for a number of local and international conferences. Also, she serves as a reviewer in a number of local and International Journals.

Maha K. Saadeh obtained her B.Sc. degree in computer science from the University of Jordan, Amman, Jordan in 2009. Currently she is pursuing her M.Sc. degree from the same university. Her research interests are: Wireless networks, network security, image processing, and robotics.

Mousa T. AL-Akhras obtained his B.Sc. and M.Sc. degrees in computer science from the University of Jordan, Amman, Jordan in 2000 and 2003, respectively. Then he worked as teaching assistant at the University of Jordan. He earned his Ph.D. degree in 2007 from De Montfort University, Leicester, UK. His Ph.D. specialization is artificial neural networks & communications.

He works as an Assistant Professor in the Computer Information Systems Department, King Abdullah II School for Information Technology at the University of Jordan since 2007. His research interests include problems in the area of artificial intelligence and particularly artificial neural networks. His research interests include Voice over IP, multimedia communication, robotics, genetic algorithm, fuzzy logic, and statistics. He is also interested in the area of electronic learning (e-learning), and mobile learning (m-learning).

Dr. AL-Akhras is a member of IEEE and he was elected as a secretary for general activities of the IEEE executive committee, Jordan Section, region 8 for the years 2010-2011. He was also elected as a vice-chair for Computational Intelligence/Computer Joint Societies Chapter, Jordan section for the years 2010-2011. He is also a member of IEEE CIS & RAS Societies. He is in the organizing and technical committees for a number of local and international conferences. Also, he serves as a reviewer and a member of the editorial board in a number of local and International Journals. He is a member of the Jordan Society for Scientific Research (JSSR), he also serves as a judge in the national and Arabic robotic contest (First Lego League).

Hamzeh AL Jawawdeh obtained his B.Sc. degree in computer science from the University of Jordan, Amman, Jordan in 2005. Currently he is pursuing his M.Sc. degree from the same university.

He works as Project Manager at Independent Financial Indexes (www.almostaqellah.com). His research interests include wireless sensor networks, routing protocols, B-trees algorithms, and genetic algorithms.