J. Appl. Environ. Biol. Sci., 4(9S)24-35, 2014

© 2014, TextRoad Publication

ISSN: 2090-4274
Journal of Applied Environmental
and Biological Sciences
www.textroad.com

An Energy Efficient and Fault Tolerant Mobile Wireless Sensor Network Model for Military Applications

Ilker Korkmaz[1], Ali Kanat[1], Onur Unver[2], Berkay Simsir[1], Halil Gultekin[1]

[1]Department of Computer Engineering, Izmir University of Economics, Izmir, Turkey
[2]Department of Software Engineering, Izmir University of Economics, Izmir, Turkey

Received: September 12, 2014
Accepted: November 23, 2014

ABSTRACT

Wireless Sensor Network (WSN), a self organized group of distributed sensor nodes, is a promising technology that can be used in many different areas, e.g., environmental monitoring, health, object tracking, and military applications. This paper presents an energy-efficient mobile sensor network model to be used in military applications. The proposed approach is designed to be fault tolerant regarding the case that mobile nodes may be disconnected and reconnected in the network anytime. Real case military scenarios are considered in simulations such as informing about soldiers' medical condition unattended by using cardiac rhythm and body temperature sensors. The paper also gives the TinyOS simulation results in terms of energy consumption and lifetime metrics.

KEYWORDS: Wireless Sensor Network (WSN), Clustered Topology, IEEE 802.15.4, Finite State Machine (FSM), Tiny OS.

1 INTRODUCTION

The subject of Wireless Sensor Networks (WSNs) is an emerging research area that many workshops and conferences have dealt with. Moreover, WSN is a promising field that can be used in many different areas and can be a solution to many different problems. This paper gives an energy efficient and fault tolerant mobile WSN model to be used in military applications where soldiers are simulated as mobile nodes. We aimed to build a WSN system that runs unattended even if a soldier is injured or dead as the considered network does not require any user interaction. By using cardiac rhythm and body temperature sensors embedded on nodes, base station is going to be informed about mobile soldiers' medical condition.

IEEE 802.15.4 MAC protocol standard [1] is selected to be simulated in the system. IEEE 802.15.4 is suitable for WSNs because of its low data rate and low power consumption [2, 3]. As to simulate an appropriate infrastructure for real case groups of mobile soldiers, a clustered topology is designed to overcome challenges that derive from the structure of WSN. The main reasons of choosing a clustering approach are the aim to reduce the volume of inter-node communication and the desire to create a scalable network. The system is implemented in and simulated via TinyOS platform, which is a popular operating system used for WSNs.

This investigation mainly aims the use of the proposed WSN model by Special Forces of militaries. Militaries may face many difficulties because of environmental conditions. Without the necessity of user interaction, a self-organized and self-operating network is required. Regarding the whole integrated system, we believe that WSN is the most suitable platform in such a case.

2 BACKGROUND

There exist some similar applications specifically in military or in the field of health care. However, our project includes the requirements of both military and health care applications. An example military application is VigilNet [4] which is a WSN designed to detect enemy capabilities and positions of hostile targets. Main aim of VigilNet is to alert allies, enemies’ mobile vehicles or soldiers in hostile region.

Moreover, a game scenario that gives us some clues about the challenges we may face is implemented in [5]. The corresponding pursuit-evasion game (PEG) application is deployed in the environment where the game is played and cooperates with the pursuers’ team. The application consists of many interesting research problems in the areas of tracking, control design, security, and robustness. For a PEG, the sensor network must be capable of multiple-vehicle tracking which can distinguish pursuers from evaders. The

---

*** Corresponding Author:** Ilker Korkmaz, Department of Computer Engineering, Izmir University of Economics, Izmir, Turkey,

network requires a dynamic routing ability to deliver information to pursuers in a convenient time. Since the game will be played in a distributed manner, non-central sensing, control, and actuation need to be taken care during controller design. To prevent the evader's team from intercepting sensitive information, the network must provide additional security features. Finally, the control algorithms should be aware of that a sensor node can fail.

2.1 Network Topologies Used In Wsn Applications

Networks differ as hierarchical and non-hierarchical ones. Hierarchical networks use robust backbones to increase efficiency. Since in hierarchical networks, inexpensive network nodes communicate with their backbones, creating a network by limiting number of expensive backbones will be cheaper. Therefore, simple nodes can be mobile. On the other hand, in non-hierarchical networks, all nodes are equal. By creating identical cells and keep one node active in the cell, connectivity can be managed with limited power [6]. It will be preferred a hierarchical network because it is wanted that backbones communicate with the base station in order to provide energy efficiency. By switching backbones according to an algorithm, energy efficiency is also considered since the energy consumption in the system is expected to be the main problem.

Several network topologies which are commonly used in WSN applications are analyzed below.

Star topology. Each member node in a star topology is connected to the sink or base station in one hop distance. Detection of faulty devices is very simple in that kind of hierarchy. Since all nodes are directly connected to sink, it can easily be determined which sensor node has a connection problem. An advantage of star topology is that setting up the network is simple. Management and error detection are simple and can be handled and recovered quickly. A disadvantage of star topology is that providing a survivable topology is difficult when nodes are mobile. It could not offer a reliable communication in case of a single point of failure since there is no any alternate path for nodes [7].

Clustered topology. In a wireless sensor network that uses clustered topology, the sensor units are grouped into disjoint set clusters. The cluster head is responsible for coordination among the cluster nodes and gathering of their data and transmission of the aggregated data to the sink directly or via multi-hop transmission.

To conserve energy, clustered WSNs offer three major advantages over non–clustered WSNs. Clustered WSNs are capable of reducing the volume of inter-node communication by localizing data transmission within the formed clusters and more importantly by decreasing the overall number of transmissions to the sink. Secondly, clustered WSNs are capable of extending the nodes' sleep times by allowing cluster heads to coordinate and optimize the activities of other cluster members through some form of TDMA based scheduling [8]. One more advantage is that clustered systems localize the route set-up within the cluster and reduce the size of the routing table stored at the individual sensor node [9].

Clustered tree topology. Clustered tree topology model is one of the tree-based logical topologies of WSN, where nodes are organized randomly. It describes a cluster that means there is no order in node relations. Nodes at the lowest depth are reduced function devices such as sensors, controllers and actuators. These nodes are connected to powerful nodes, called full function devices. These full function devices are able to perform network routing functions and are connected to the personal area network (PAN) coordinator. A clustered tree network may include a different number of star networks connected with their central nodes which has direct access to single PAN coordinator [10]. In such a topology star networks can be seen as the clusters, where cluster leaders can be seen as the nodes on the backbone path of the spanning tree that can be constructed starting from a sink node.

2.2 Routing Protocols

Routing in wireless sensor networks is more different than fixed networks. Since there are more variables in wireless networks, sensor nodes may not work properly and routing must be changed based on dead/failed nodes. Routing mechanisms used in WSNs are made to adapt to such kind of situations. Routing protocols that are specified to be used in WSNs are listed in seven main categories in [11], namely location-based, hierarchical, data-centric, mobility-based, multipath-based, heterogeneity-based and quality of service (QoS) based protocols.

Location-based protocols [11] are used when sensor nodes are identified by their locations. Most protocols calculate the distance between two certain nodes in order to predict energy level so that the nodes can be managed to save their energy. Some location-aware routing protocols [11, 12] update the routes by using node coordinates, gathered from Global Positioning System (GPS) [13].

Data-centric protocols [11, 14] are different from other address-centric protocols. In data centric protocols, node sensors transmit their data to the sink. In address-centric protocols, node sensors' proper data is sent to the sink independently. In data-centric protocols when an initiator sensor transmits its raw data to the sink, some intermediate data can be added by other sensors which are between the initiator and the sink. That is not only a kind of gathering event but much more an aggregation mechanism. An advantage of using such information passing protocols is energy saving since less communication is needed between source sensors and the sink.

Hierarchical protocols [11, 15] are based on clustered topologies. Clustering is used to construct an energy preserving communication protocol by setting hierarchy in the communication. Every cluster has a unique node called as cluster head, and this node manages and coordinates its cluster. Then, inter-cluster and intra-cluster communication are done as two different hierarchical messaging.

Mobility-based protocols [11] have two main challenges due to the mobile nodes in the network. One is that there may not be an end-to-end data delivery guarantee as the network may be disconnected. The other issue is the overhead on energy consumption as the nodes move around. Mobility-based protocols intend to optimize the trade-off between providing a connected network and offering an energy efficient schema.

Regarding multipath-based protocols [11], the messages may use different paths between the source and target. Data transfer between sink and an ordinary node in the network, where the network is seen as a huge tree, has two ways of transmitting node's data to sink, namely single-path routing and multipath routing. When each sensor transmits its data to the sink by using only the shortest path, it can be referred as single-path routing. Multipath routing [11, 12] refers that each sensor calculates its n different paths to sink and balances its load evenly through these paths.

Heterogeneity sensor network [11, 16] structure involves mainly two different types of node devices regarding their resources. Resource powerful sensors have less or even no energy limitation, whereas the resource constrained battery-powered sensors have limited working life. To increase its lifetime, a sensor should use its remaining battery efficiently by reducing data transmission and computation.

QoS based protocols [11, 17] aims to optimize energy consumption in routing layer regarding the predetermined threshold quality values for the required services. Fault tolerance, energy consumption, delay, reliability and some other required services such as security may be taken care as the QoS metrics on routing in WSNs. All these QoS metrics need to be considered overall in a sort of trade-off comparison to prevent much energy consumption.

3 CHALLENGES

There are several challenges for nodes that flow the data monitored from environment to sink, such as limited memory, computational capacities and battery power. In addition, our proposed system deals with fault tolerance, scalability, production cost, power consumption and security issues that are explained in details in [18].

Fault tolerance. Some sensor nodes may be damaged due to physical conditions and lack of power. The system should be designed considering those environmental circumstances so that overall network should still be working if any of its nodes dies. On the other hand, a part of the network may be disconnected due to transmission range problems of the nodes or their mobile behaviors if supported. In such cases, the system should recover itself and keep on running with its available members for its common objective.

Scalability. The productivity of data flow may increase when the number of nodes in the network is raised. However, if the total number of sensor nodes increase dramatically, the system may be overloaded and disrupted. In order to specify a system as scalable, its operation should still run without concerning the number of nodes in the network.

Production cost. Number of sensor nodes in a general environmental monitoring scenario may reach to thousands or millions. On the other side, regarding the military applications, a less number of sensor nodes, i.e., 10 to 20, are used, especially for Special Forces. Therefore, node cost is usually not an issue for such special force applications.

Power consumption. Using battery power effectively is significant for system life-time. Self configuration and self organizing mechanisms are required in order to provide unattended operation. Nodes operate with limited power resources and limited ability to recharge [19]. Besides, when the sensor nodes which are disposed onto soldiers move away each other, the power consumption may be increased.

Security. Information needs to be secured to protect any private data used in network. The encryption, authentication and integrity mechanisms are necessary to reach a secure system.

4 SYSTEM DESIGN

Finite State Machine (FSM) usage is common in WSN applications. Therefore, we used FSM during the design process. The FSM diagram of the proposed system model is given in Figure 1.

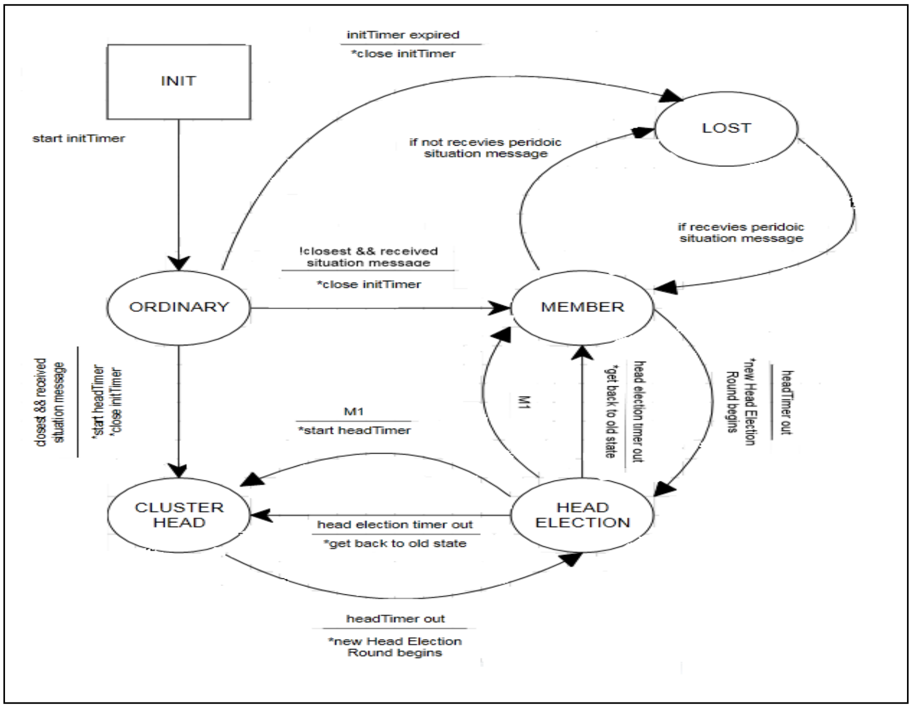

Figure 1. FSM diagram of the proposed system.

As shown in Figure 1, the nodes in our proposed model can be in any of the six states that are, INIT, ORDINARY, LOST, CLUSTERHEAD, MEMBER, HEADELECTION. To see all the states, messages and the relations between them, FSM table in Figure 2 can also be examined. Nodes change their states according to the messages they receive and act according to those messages. On the other side, nodes have other four attributions that are dead, healthy, injured, disabled. These results are decided by the algorithm running in base station according to cardiac rhythm and body temperature information sent by a node.

Nodes start initTimer when they pass to ORDINARY state. In ORDINARY state if a node do not become a MEMBER or CLUSTERHEAD during a predetermined period of time, that node will pass to LOST state and close initTimer. By such a timer use, we aim to prevent synchronizing problems. First in ORDINARY state, each node sends a situation message that includes the coordinate of the node and its distance to base station and each node updates its neighboring table based on the received messages. Initially, the node closest to base station in distance becomes CLUSTERHEAD and fires headTimer that works for choosing heads periodically. Other nodes which are not the closest become MEMBER. If a member of a cluster do not receive periodic situation message of cluster head, that member will no longer be considered as connected to the cluster and so it will pass to LOST state. It will remain in LOST state until it receives a periodic situation message from any of the cluster heads. Since the nodes are mobile and the topology always changes, the nodes not only switch between LOST and MEMBER but also switch between different clusters. So, the proposed approach provides a fault tolerant network as to offer to solve the connectivity problem by the nature of mobility of the nodes.

STATE	**start initTimer**	**InitTim r expire**	**Received situation message && closest**	**Received situation message && !(closest)**	**If receives situation message**	**If not receives situation message**	**receive (M1) from all its neighbours && highest weight**	**receive (M1) from all its neighbours && !(highest weight)**	**HeadTimer out**	**Head election timer out && (old state == CLUSTER HEAD)**	**Head election timer out && !(old state == CLUSTER HEAD)**
INIT	ORDINARY/-	-	-	-	-	-	-	-	-	-	-
ORDINAI Y	-	LOST/ close InitTimer	CLUSTER HEAD / start HeadTimer, close InitTimer	MEMBER/ close InitTimer	-	-	-	-	-	-	-
CLUSTER HEAD	-	-	-	-	-	-	-	-	HEAD ELECTION/ new round begins	-	-
HEAD ELECTION	-	-	-	-	-	-	CLUSTER HEAD/ start HeadTimer	MEMBER/-	-	CLUSTER HEAD	MEMBER
MEMBER	-	-	-	-	-	LOST/-	-	-	HEAD ELECTION/ new round begins	-	-
LOST	-	-	-	Member/-	-	-	-	-	-	-	-

Figure 2. FSM table.

When headTimer is out, cluster head or members will pass to HEADELECTION state and cluster head starts head election timer. We considered implementing head election timer in order to prevent synchronization problems of nodes messaging. In HEADELECTION state, nodes will send their remaining battery level to their neighbors. If head election timer is not out, a node will decide to be the cluster head according to the following: first, the node should receive remaining battery level message, M1 in Figure 1, from all its neighbors, then the node should have the highest remaining battery level. Otherwise, the corresponding node will be a member. However, if head election is not completed during a predetermined period of election time, all nodes in that cluster will return to their previous states.

In addition, when a disconnected or lost node attempts to reenter to system, it is at the moment in LOST state. As a positive side-effect, this design can also be used same for the newcomer nodes of the system, if any. Any possible urgent newcomers are to be considered as in LOST state first, since they are not connected to any cluster. Consequently, when an unconnected node receives a situation message from a cluster head it becomes directly a member of that cluster.

With the help of this FSM design, in overall, we expect to share out the responsibility of being cluster head. Cluster head node changes regarding the remaining battery levels of the nodes in the cluster, which will lead us to build a system in which energy consumption is equally distributed. So, the overall lifetime of the network will also be increased.

4.1 Extension Modules: There are extra modules that extend the system to be used for a military application that takes into account the health care of the soldiers. The proposed extension modules support the fault tolerant and energy efficient properties of the model. On the other hand, we still work on mobility of system. Due to mobility, nodes may be disconnected from cluster so that the system is fault tolerant and energy consumption may be too high. Therefore, the algorithms running within the two main extension modules, module#1 and module#2 are considered based on the real case scenarios depicted in different schemes in Figure 3.a, Figure 3.b and Figure 3.c.

Module 1. The soldier cannot make any ability to communicate with team, but he can communicate with base station. The base station is going to send messages to team about direction of missing soldier, his location and soldier's state. Therefore, a person who is from team may help to find the missing soldier as regards the messages that was sent from base station.

Module 1.a. If the soldier is injured or damaged due to physical conditions, the team will get the message of injured soldier who would need any help. Figure 3.a explains that the injured soldier's state and his location will be sent to base station so that anyone who is a team member is going to reach the location of injured soldier. Therefore, the team can help the soldier within the shortest time.

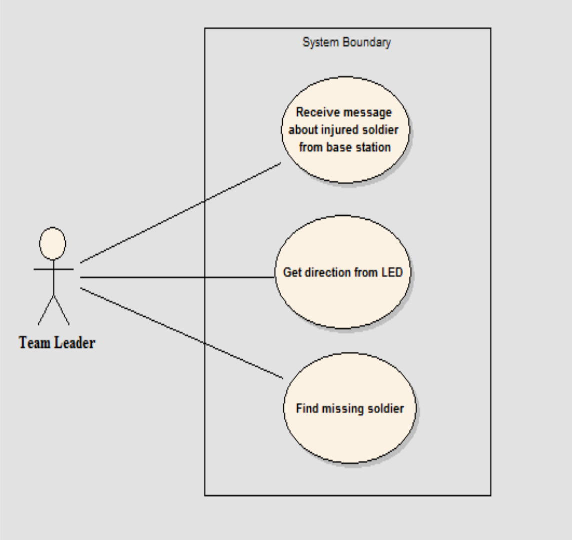

Figure 3.a. Module 1.a.

Module 1.b. Figure 3.b illustrates that if a soldier is healthy, but he lost his team. In that case, the soldier is going to get a message of team's location and he will try to find out his team and team mates by using given location and direction.

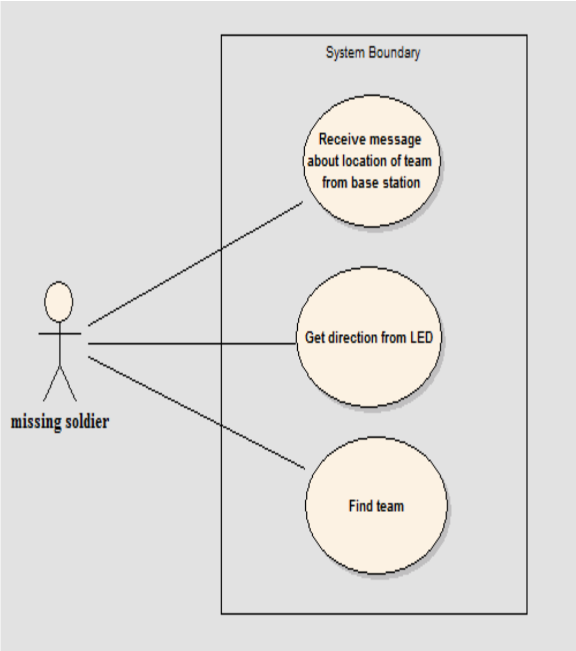

Figure 3.b. Module 1.b.

Module 2. Figure 3.c shows the case that the soldier has communication capability with neither base station nor his team. Base station will send the predicted direction and last location of missing soldier to the nearest team. Thus, the nearest team tries to find missing soldier.

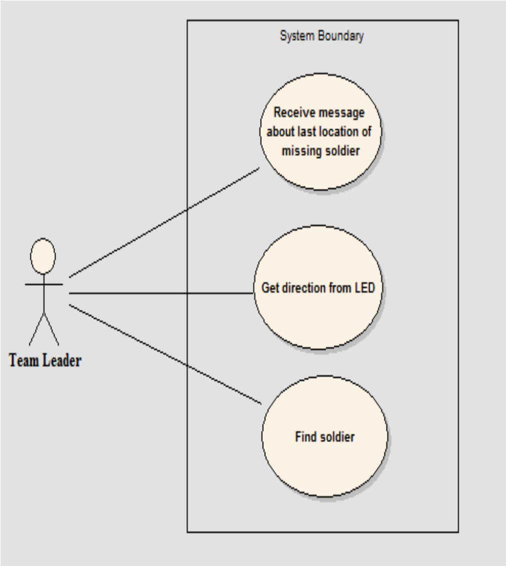

Figure 3.c. Module 2.

4.2 Healthcare Procedure

Within the requirement analysis phase of the system, we have interviewed with specialists. Based on the recommendations of doctors for a person's medical condition, the health care procedure, given in Figure 4, has been implemented for a soldier to decide on a result among four values; dead, healthy, injured, disabled. After obtaining the information about a soldier's medical situation, the sink node runs this algorithm to give a decision on the corresponding soldier's medical condition.

The algorithm in Figure 4 is mainly based on the conditional structures regarding the heart rate (number of heart beats per minute) and the body temperature extracted from the medical data message. Initially, we assumed that each soldier has no any medical issues, all of them are healthy. Actually, there are two significant cases that are body temperature and heart rate. These cases analyze the critical points. Priority gives the soldier's medical condition. When the body temperature is greater than or equal to 36.5 °C and less than or equal to 37.5 °C, the heart rate is going to be checked whether it is between 60 and 100. If this condition is satisfied, the soldier's state is healthy and s/he is alive. Besides, when the body temperature is between 32.2 °C and 35.5 °C, the heart rate infers that the soldier's hearth rate increases dramatically. The soldier may be injured and the priority may be supposed as medium.

On the other hand, if body temperature is between 28 °C and 32 °C, it could be inferred that this soldier is exactly injured and his priority seems high. Finally, if the soldier's body temperature is less than 28 °C, it could be decided that the soldier is in coma and will die absolutely. The priority of such a soldier is low due to preserve the remaining resources of team for the rest of the soldiers.

```
// Assume that all soldiers are healthy initially,
// they have not any hormonal/biological sickness.
bodyTemp = body temperature of soldiers
heartRate = heart rate of soldiers
pri = priority of soldiers' medical condition
if (bodyTemp>=36.5 && bodyTemp<=37.5)
    if(heartRate>=60 && heartRate<=100)
            state = healty&alive
else if (bodyTemp>=32.2 && bodyTemp<=35.5)
    //heartRate increases dramatically.
    state = may injured
    priorityToSave = medium
else if(bodyTemp>=28 && bodyTemp<=32)
    state = exactly injured
    priorityToSave = high
else if(bodyTemp<28)
    state = coma & die
    priorityToSave = low
```

Figure 4. Pseudocode of healthcare procedure.

4.3 Simulation Results

The proposed network is implemented using nesC programming language on TinyOS platform [20].The considered topology of the proposed sensor network involving 15 sample nodes is designed in TinyOS as illustrated in Figure 5. In Figure 5, it is shown that the network is composed of four clusters named as A, B, C, and D. The nodes in Figure 5 are clustered according to the proposed FSM. Each cluster has a cluster head and a few cluster members. Members in a cluster will only communicate with their cluster head. On the other hand, a head of a cluster will also communicate with base station. We assume that base station is one hop distance away from the cluster heads. Therefore, cluster heads have ability to flow information that they sense and received. We aimed to flow information through other clusters, in case of having no communication with base station. Consequently, if a cluster head is disabled to a direct communication with base station due to mobility or any environmental circumstances, it will flow information to base station through other nodes that are able to communicate with base station in one hop distance.

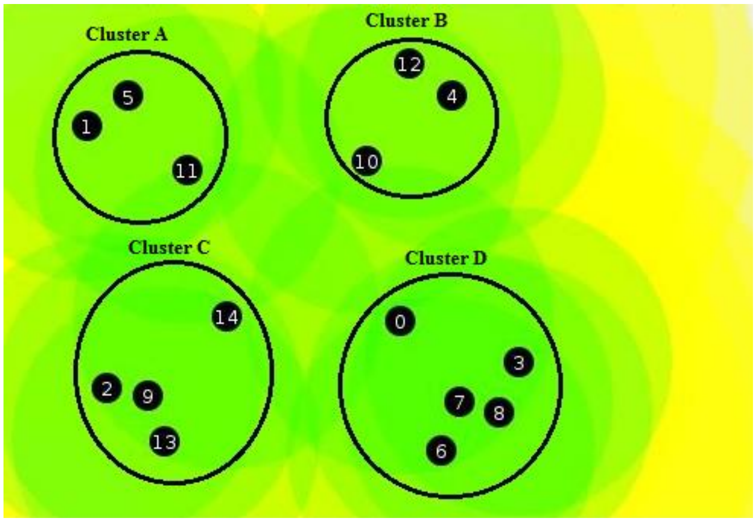

Figure 5. Topology.

Every node has a specific range for transmitting data. Green circles in Figure 5 shows these ranges. Therefore, a node can only transmit data to other nodes that are in range. In addition, yellow circles in Figure 5 show unconnected region. The extension modules described in Section 4.1 are problematic real-case scenarios that may be occurred in any fault of the network. Either the case that a soldier move in a different direction than his/her neighbors or the case that a cluster as a whole team move in a different path than their neighbor clusters, there may exist a disconnection in any part of the network due to unconscious mobility. The extension modules are run to recover the regarding part of the graph and to reconnect the network. For that reason, the modules make the system much more fault-tolerant. So far, we have not implemented the extension modules in the system; main reason to give details about those modules is our intension to propose the model as a whole. The project implementation is still in progress.

A different scenario to demonstrate disconnection of a node where the system runs on the same topology given in Figure 5 is drawn in Figure 6. Figure 6 simply points out the basic connectivity problem in the system. The green circular territory around the alone node 3 represents its transmission range, whereas the yellow parts beyond its antenna range represent the area out of communication.

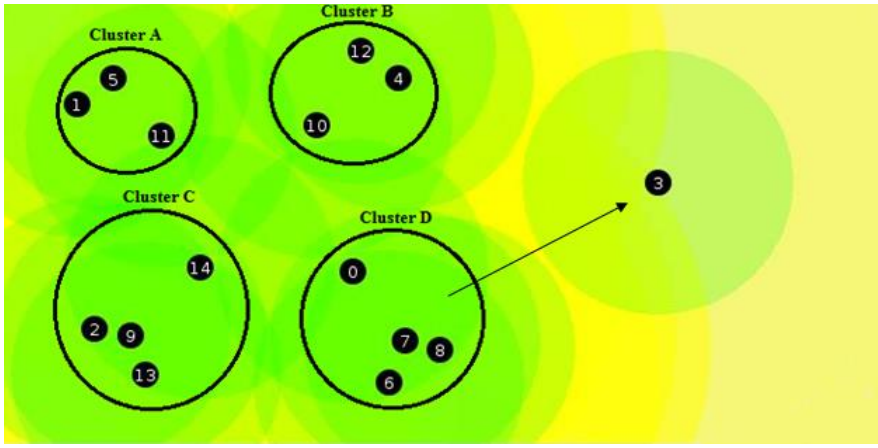

Figure 6. A scenario for a disconnected node.

It is depicted in Figure 6 that node 3 is assumed to move away from its cluster due to its random mobility. At a convenient far away point, node 3 will get out of connection at any time. When a node goes out of range of its cluster, it is no longer a member of its cluster and it directly switches to a disconnected status (the state LOST in FSM table in Figure 2). When a node goes out of range of any cluster in the network, the node is disconnected from the network. Therefore, the network as a graph will not be connected fully after that moment.

As described with the extension modules proposed, there are two main different scenarios for a node to attempt to recover from its disconnected status. The first is that the disconnected node (soldier) itself will attempt to join the nearest different cluster if any is available in communication range. If not, the node will attempt to transmit its location to base station. The other alternative is the case that the corresponding node (soldier) cannot construct a communication with the base station. In such a case, when the periodic situation information is not received about the node, the base station will transmit to the nearest cluster head (team leader) the last stored location of the corresponding disconnected node together with a predicted direction based on the last movement history. Thus, the nearest team will move to that direction and try to find out the disconnected node to make it join the cluster.

The simulation results given in this paper do not involve any disconnection scenario. All nodes are assumed to be able to connect the base station in one hop and the network is assumed to be always connected. The aim of the simulations conducted so far is to show the energy efficiency of the proposed clustering approach together with the proposed FSM model. We have measured the energy consumption of a full-time connected scenario running on the topology given in Figure 5. The corresponding results are depicted in Figure 7.a and Figure 7.b. In the simulations, it is assumed that the nodes consume approximately 16 mA for transmitting or receiving single radio message [21].

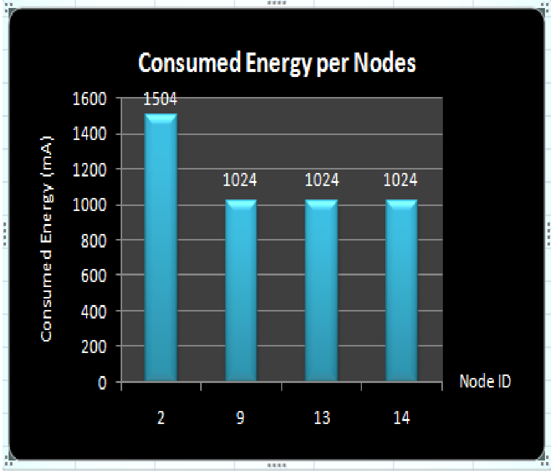

Figure 7.a. Constant cluster head.

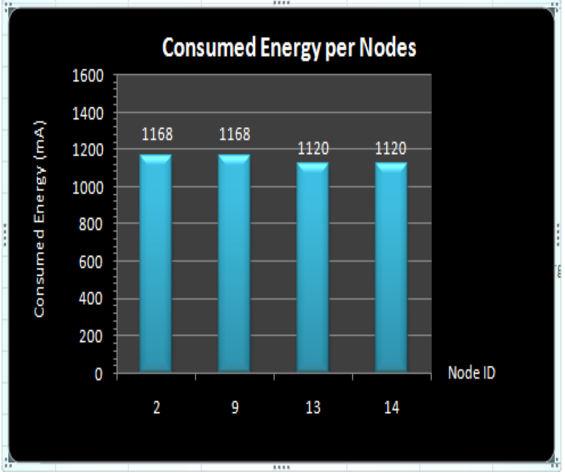

Figure 7.b. Dynamic cluster head.

Figure 7.a and Figure 7.b show the comparable results on energy consumptions for two different clustering scenarios. In Figure 7.a, the cluster head is elected once at the beginning and saves its state for all the simulation time. In other words, there is only one constant cluster head. The head does all the transmission to base station and consumes more energy than member nodes since the member nodes just send their data to their fixed head. Regarding this simulation scenario, the constant cluster head is node 2, seen as in Figure 7.a. However, in Figure 7.b, cluster head is decided according to our proposed FSM design. Namely, cluster head is decided regarding consumed energy; hence the head node is not fixed. The node which consumes less energy in total becomes cluster head. Therefore, the time period for all nodes to be the cluster head would be divided in a balanced manner, which leads to an almost equally energy consumption on each node. This also results an increase in overall network lifetime.

The simulations have been run for duration of 10 time periods for both scenarios of Figure 7.a and Figure 7.b. As shown in Figure 7.a, node 2 was always the fixed head in the first scenario and it consumed dramatically a huge energy compared to the other nodes. In the second scenario, the number of total periods in which the same node is being elected as head was tried to be much more balanced among all nodes in the same cluster using the proposed FSM. In Figure 7.b, since node 2 and node 9 were elected as head in more periods than node 13 and node 14, node 2 and node 9 consumed more energy than node 13 and node 14; however there has not been observed a distinct difference on energy consumption per nodes in any different simulation.

Figure 7.a and Figure 7.b compare the measured energy consumptions of the nodes in cluster C for two different simulations. The energy consumption amounts of the other nodes in other clusters were also measured balanced when using the proposed FSM. On the other side, the total energy consumption of each cluster is shown in Figure 8. Clusters A and B, which have same number of members, have same energy consumption amounts. Total energy consumption in cluster D is more than the other clusters indicated in Figure 8 and this result is consistent with that cluster D has more number of members than the others as shown in topology in Figure 5.

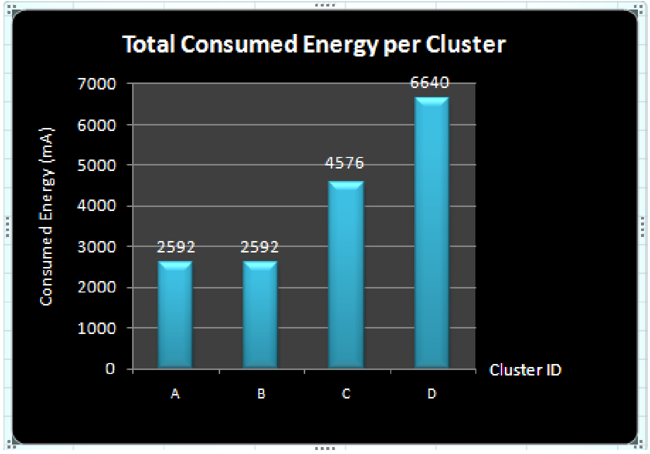

Figure 8. Energy consumption per cluster.

The systematic behind the energy calculation in Figure 8 can be summarized as following: Each node in a cluster with a number of N nodes initially broadcasts 1 message to update their neighboring table during initTimer interval. Hence, each node receives N-1different broadcasting messages. Therefore, number of corresponding messages either sent or received in initial duration is N for each of N nodes, that is N2 in total. Then, in headTimer interval, each of N-1 member nodes sends 1 message to cluster head. Cluster head receives N-1 messages from members and transmits these data to base station in 1 aggregated message. Consequently, number of related messages sent/received in headTimer duration is N regarding cluster head, 1 regarding each of N-1 members, that is 2N-1 in total. Lastly, in head election timer duration, each node in cluster broadcasts 1 message and receives N-1 messages from neighbors. After being elected, the new cluster head broadcasts 1 additional message to inform all neighbors about its status. Those N-1 members receive this 1 message. Therefore, number of messages sent/received is N+1 regarding the new head, N+1 regarding each of N-1 members, that is N2+N. When the simulation starts, initTimer interval expires once at the beginning, then headTimer and head election timer intervals expire periodically in each tour. Reminding that the simulation durations were chosen as 10 time periods/tours, the total number of messages sent/received in any cluster in Figure 8 with N nodes is N2+10[(2N-1)+(N2+N)], which can be simplified as [11N2+30N-10]. Considering the cluster C with N=4 nodes as shown in Figure 5 and using 16 mA for each send/receive operation, the total consumed energy per cluster C is calculated as 4576 mA, which is the corresponding measured value shown in Figure 8.

4 Conclusion

We proposed and built a self-configured, unattended, low cost WSN based system that involves both military and healthcare features. The overall system is considered to be fault tolerant to solve the connectivity problem due to mobile nodes. The model is also designed to be energy efficient to increase the lifetime of the nodes using a clustering approach with changing cluster heads.

The project implementation has been going on. As the main future work, it is planned to focus on disconnection-enabled scenarios and to evaluate the recovery success of the proposed system. Furthermore, the extension modules proposed will be implemented to offer a much more fault-tolerant system.

REFERENCES

1. IEEE Std 802.15.4TM-2003. IEEE Standard for Information Technology-Telecommunications and Information Exchange between Systems-Local and Metropolitan Area Networks-Specific Requirements-Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs).Fenton, 2003.
2. Gravogl. K, Haase. J and Grimm. C, "Choosing the best wireless protocol for typical applications", in Proceedings of the 24th International Conference on Architecture of Computing Systems, ARCS'2011, pp. 279-284, 2011.
3. Gascon. D, "IEEE 802.15.4 vs Zig Bee". An online white paper available at http://sensor-networks.org/index.php?page=0823123150 (Last accessed: April 2014).
4. Stankovic. AJ, Wood. DA and He. T, "Realistic Applications for Wireless Sensor Networks", in Theoritical Aspects of Distributed Computing in Sensor Networks (Eds. Nikoletseas. S and Rolim. JDP), pp. 835-863, 2011.
5. Schenato. L, Oh. S, Sastry. S and Bose. P, "Swarm Coordination for Pursuit Evasion Games using Sensor Networks", in Proceedings of the 2005 IEEE International Conference on Robotics and Automation, ICRA'2005, pp. 2493 – 2498, 2005.
6. NazrulAlam. SM and Haas. JZ. "Hierarchical and Non-hierarchical Three-Dimensional Underwater Wireless Sensor Networks" (unpublished, available at http://arxiv.org/ftp/arxiv/papers/1005/1005.3073.pdf. Last accessed: April 2014).
7. Srivastava. N, "Challenges of Next-Generation Wireless Sensor Networks and Its Impact on Society", Journal of Telecommunications, vol. 1, issue. 1, 2010pp. 128-133.
8. Vlajic. N and Xia. D, "Wireless Sensor Networks: To Cluster or Not To Cluster?", in WOWMOM '06 Proceedings of the 2006 International Symposium on World of Wireless, Mobile and Multimedia Networks, pp. 258-268, 2006.
9. Lee. K and Lee. H, "A Self-Organized and Smart-Adaptive Clustering and Routing Approach for Wireless Sensor Networks", International Journal of Distributed Sensor Networks, vol. 4, issue. 1, Article ID 156268, 2012.
10. Jurčík. P, Severino. R, Koubâa. A, Alves. M and Tovar. E, "Real-Time Communications over Cluster-Tree Sensor Networks with Mobile Sink Behaviour", in The 14th IEEE International Conference on Embedded Real - Time Computing Systems and Applications, 2008.
11. Singh. KS, Singh. PM and Singh. KD, "Routing Protocols in Wireless Sensor Networks – A Survey", in International Journal of Computer Science & Engineering Survey (IJCSES) Vol.1, No.2, 2010, pp. 57-79.

12. Boukerche. A, Turgut. B, Aydin. N, Ahmad. MZ, Bölöni. L and Turgut. D, “Routing protocols in Ad Hoc Networks: A Survey”, Computer Networks 55(13):3032-3080, 2011.
13. http://www.gps.gov/ (Last accessed: May 2014).
14. Stojmenovic. I and Olariu. S, “Data-centric protocols for wireless sensor networks”, Handbook of Sensor Networks, 2005, pp. 417-456.
15. Watteyne. T, Molinaro. A, Richichi. G and Dohler. M, “From MANET to IETF ROLL standardization: A paradigm shift in WSN routing protocols”, IEEE Communications Surveys & Tutorials, vol.13, no.4, 2011, pp. 688-707.
16. Mehran. A, Wysocki. T and Dutkiewicz. E, “A Review of Routing Protocols for Mobile Ad Hoc Networks”, Ad Hoc Networks 2. (1), 2004, pp. 1-22.
17. Villalba. LJG, Orozco. ALS, Cabrera. AT and Abbas. CJB, “Routing Protocols in Wireless Sensor Networks”, Sensors, vol. 9, no. 11, 2009, pp. 8399-8421.
18. Su. W, Akyildiz. FI, Sankarasubramaniam. Y and Cayirci. E, “A Survey on Wireless Sensor Networks”, IEEE Communications Magazine, vol. 40, no. 8, 2002, pp. 102-114.
19. Krishnamachari. B and Iyengar. S, “Distributed Bayesian Algorithms for Fault-Tolerant Event Region Detection in Wireless Sensor Networks”, IEEE Transactions on Computers, vol. 53, no. 3, 2004, pp. 241-250.
20. http://www.tinyos.net (Last accessed: May 2014).
21. Shnayder. V, Hempstead. M, Chen. B, Allen. WG and Welsh. M, “Simulating the Power Consumption of Large-Scale Sensor Network Applications”, SenSys'04 Proceedings of the 2nd international Conference on Embedded Networked Sensor Systems, 2004, pp. 188-200.