SALEH ALI SAEED

IMPROVEMENT THE BATTERY LIFE OF
WIRELESS SENSOR

NEU
2017

IMPROVEMENT THE BATTERY LIFE OF WIRELESS SENSOR NETWORKS

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES OF NEAR EAST UNIVERSITY

By

SALEH ALI SAEED

In Partial Fulfilment of the Requirements for the degree of Master of Science in Software Engineering

NICOSIA, 2017

IMPROVEMENT THE BATTERY LIFE OF WIRELESS SENSOR NETWORKS

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES OF NEAR EAST UNIVERSITY

By

SALEH ALI SAEED

In Partial Fulfilment of the Requirements for the degree of Master of Science in Software Engineering

NICOSIA, 2017

SALEH ALI SAEED: IMPROVEMENT THE BATTERY LIFE OF WIRELESS SENSOR NETWORKS

Approval of Director of Graduate School of Applied Science

Prof. Dr. Nadire CAVUS

We certify this thesis is satisfactory for the award of the degree of Masters of Science in Software Engineering

Examining Committee in Charge:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name:

Signature:

Date:

To my parents…

ACKNOWLEDGEMENTS

This Thesis have been cover with the help, support and patience of my Supervisor, my friends, and my deepest gratitude goes to Assist. Prof. Dr. Yoney Kirsal Ever, for her constant encouragement and guidance.

I would like to thanks Assist. Prof. Dr. Boran Sekeroglu who has been very helpful through the duration of my thesis.

Above all my unlimited thanks would be dedicated to my dearest family for their loyalty and their great confidence in me.

At the end, I would like to thanks to all my friends I can’t mention them all but I would like to thank them from all of my heart for their valuable help and support since I was in my early study until now.

ABSTRACT

Wireless sensor networks have been widely employed in a wide range of applications. As powered by small batteries, the wireless sensor node has constrained lifetime. Wireless consists of special distributed sensor nodes to monitor the physical environment conditions such as; Temperature, motion of object, etc. These wireless nodes operate on the battery power and limited life of the battery is a fundamental problem of wireless sensor network. In contrary, changing the battery in wireless sensor network is a difficult task and recharging is almost impossible in real scenario. To resolve this issue there are several algorithms how to maximize the network lifetime have been proposed. However, all these schemes has been limitation and drawback. In this dissertation, we study the network lifetime elongation problem in wireless sensor network lifetime. Specially how to maximize the wireless sensor node lifetime and the sensing quality. In this thesis, we introduce the new technique of wireless sensor network by adopting the sleep and awake function for each of the cluster in the wireless sensor network. Hence, sensor node transfer low data information to base station, and processing nodes will wake up for short period of time to accomplish the task and back again to sleep after finishing process. If node did not receive any data it means battery consumption will be zero and remain same.

Keywords: Wireless sensor network; Battery lifetime; cluster based wireless sensor network; Optimum solution of WSN; Sleep and Active nodes

ÖZET

Kablosuz algılayıcı ağları geniş bir uygulama yelpazesinde yaygın olarak kullanılmaktadır. Küçük piller tarafından desteklendiği üzere, kablosuz sensör düğümü ömrü sınırlandırmıştır. Kablosuz, fiziksel çevre koşullarını izlemek için özel dağılan sensör düğümlerinden oluşur; Sıcaklık, cisim hareketi vb. Bu kablosuz düğümler pil gücü ile çalışır ve pilin sınırlı ömrü, kablosuz sensör ağı için temel bir problemdir. Aksine, kablosuz algılayıcı şebekesinde pilin değiştirilmesi zor bir iştir ve gerçek senaryoda şarj işlemi neredeyse imkansızdır. Bu sorunu gidermek için, ağ ömrünü en üst düzeye çıkarmanın çeşitli algoritmaları önerilmiştir. Bununla birlikte, tüm bu planlar bir sınırlama ve dezavantaj olmuştur. Bu tezde, kablosuz sensör ağı ömrü boyunca ağ ömrü uzama problemi incelenmektedir. Özellikle kablosuz sensör düğüm ömrünü ve algılama kalitesini en üst düzeye çıkarmak için. Bu tezde, kablosuz sensör ağı içindeki kümenin her biri için uyku ve uyanık fonksiyonu benimseyerek yeni kablosuz sensör ağı tekniğini tanıtmaktayız. Dolayısıyla sensör düğümü, düşük veri bilgisini baz istasyonuna aktarır ve işlem düğümleri, görevin tamamlanması için kısa bir süre uyanır ve bitiş işleminden sonra tekrar uyku moduna geçirilir. Düğüm herhangi bir veri almadıysa, pil tüketiminin sıfır olacağı ve aynı kaldığı anlamına gelir.

Anahtar Kelimeler: Kablosuz algılayıcı ağı; Pil ömrü; Küme tabanlı kablosuz sensör ağı; WSN'nin optimum çözümü; Uyku ve Aktif düğümler

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

Wireless senor networks are gaining popularity day by day in wide area of different applications. WSN (wireless sensor Network) consists of special distributed sensor nodes to monitor the physical environment conditions such as; temperature, motion of object, etc. they forward the data to a base station (BS). The wireless sensor nodes operate on battery power in physical environment wherein a limited life of battery is fundamental problem for sensor networks. In contrast, changing the batteries in physical distribution network is a complicated task and recharging of sensor networks almost impossible during operation. Therefore, vendors or system developers make changes in the basic WSN architecture to reduce the energy consumption especially of the nodes in order make the network and overall system application more energy efficient. Therefore, IEEE 802.15.4 standard was introduced for low data-rate application which needs to last for longer duration by consuming relatively less energy.

One of the most challenging topics for researchers in wireless sensor node communication is energy efficiency. The lifetime of wireless node depends on available energy sources and its overall energy consumption. Moreover, increasing the capacity of batteries is impossible due to small size requirement of the nodes. To resolve the batteries problem, various approaches like energy conservation (Tassiulas, 2000; Rus, 2001). ambient energy (Kansal and Srivastava, 2004). incremental deployment and battery replacement (Zhang, 2009). has been proposed. However, all these schemes have some limitation or drawback. Energy conservation schemes can only slow down energy consumption but not compensate energy depletion (Li, 2013). Harvesting environmental energy, such as solar, wind and vibration is subject to their availability which often uncontrollable and hardly manage for required size of node. Incremental deployment approach may not be applicable as it not environmentally friendly because deserted sensor nodes can pollute the environment. The battery or node can be replaceable in one scenarios that human can access it or robots that can locate and replace it. Recently new emerging wireless sensor node charging technology provides an alternative technique which address to constraint problem of sensor nodes (http://www.powercastco.com.) Unlike energy harvesting technology, sensor nodes together with more mature and cheap mobile robots can create controllable and perpetual energy source. Comparing with battery replacement, mobile robots carrying a wireless charger while mobile robots moves along trajectory and transfer energy to sensor nodes. Nevertheless, a small contribution has been reported of the wireless charging technology and implementing a wireless charging system for sensor network. Proposed technology has significantly simulated results recharging the batteries of sensor network.

In spite of above technologies, (Li, 2013). introduce a new system design such a system that consists of a mobile (1) wireless power charger; second is a network of sensor nodes associated with wireless power receiver; and third is energy station that is responsible for monitoring the energy status of sensor nodes, deciding power charging sequences to be performed by mobile charger. They evaluate their system feasibility and performance in small-scale wireless sensor networks. Additionally, they have conducted wide scale simulations to study the performance of the proposed system in the large scale networks.

Our objective is to analyze the current optimal propose systems of wireless sensor nodes and consideration of sensor batteries' lifetime issues. However, is a problem in maintaining the battery when the wireless network sensor nodes under operation. We will analyze the schemes in order to reduce the consumption of sensor node battery during operation. Our approach is to introduce a new technique in which wireless sensor nodes consume less power in long time by adding simple *sleep* and a*wake* function on wireless sensor network nodes. When wireless node transfer low data rate information to base station, processing node will wake up for a short period of time to accomplish the processing task and end of process will again lead the node into sleep mode in term of saving battery life time. Hence, each node associates temporarily work load in a few minutes per hour. Unlikely, above changes may applicable with some limitations such as sensor nodes busy most of the time in an hour, system can't go in idle state in this situation above method don't give significant results. In order to real time application, we have to find somehow approach wherein battery can be recharge or replace when nodes are the part of wireless sensor network.

1.1 Contribution

The contribution of thesis is to develop a mechanism which consume less power in long time by adding simple *sleep* and a*wake* function on wireless sensor network nodes. The node will active only when they send data or receive data otherwise they will in sleeping mode. Hence, a very short time only node need to awake and remaining time the node will sleeping mode where energy consumption will be zero.

Remaining thesis is organized as following way; second chapter describes related work of sensor network and battery lifetime, third chapter addresses to a propose idea and mathematical calculations. Fourth shows simulation results. Chapter five represents the conclusion of this thesis.

CHAPTER 2
RELATED WORK

As we know the sensor network consist of sensor nodes and mainly contain small batteries. After deployment sensor nodes is mostly difficult to access by users and thus replacement of power source is not easy task (ganathan, 2005). Hence, we have some limitation of wireless sensor networks application in this section of thesis, briefly described the main methods used to optimize the lifetime of wireless Sensor Network. We can analyze these methods in three ways such as improve energy capacity, reducing energy consumption and harvesting environment energy. Electrochemical energy which store in battery/cells, are used to provide power to wireless sensor nodes in network today (Kim and Sang, 2001). Researcher proposed several approaches to improve the energy density of the batteries. Sim et al. introduced a micro power source using a micro direct methanol fuel cell to enhance the energy (Kim and Sang, 2001). However, the size of battery is decreased in a significant manner while electronic circuits have decreased by orders of magnitude. This scheme is applicable for micro heat engines and has very high energy density of hydrocarbon fuels but it is not suitable for Wireless Sensor Network because the output power is too high and it's difficult to control such as turned off and on (Zhang, 2000).

There is another alternative way of potential energy sources for WSN. For one bright day, the light on the earth surface can produce power density of roughly 100mW/cm^2. Zhao et al. exploit that solar cells can achieve 24.4% efficiency (Roundy and Rabaey, 2003; Zhao, 1993). detailed diagram is shown in figure 2.1.

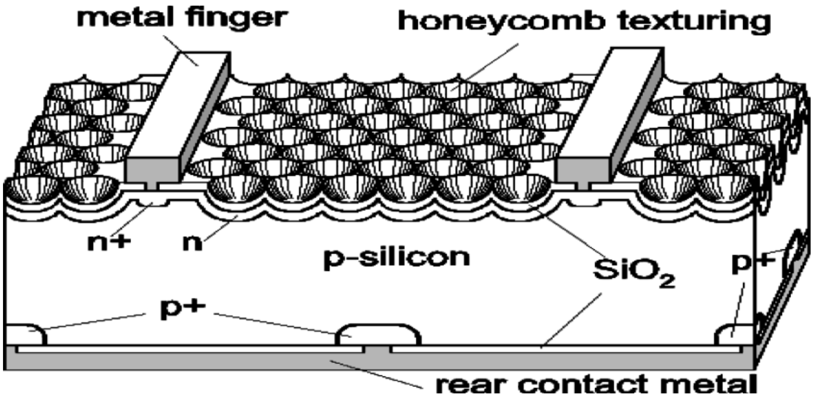


Figure 2.1: Efficient multicrystalline silicon solar cells with honeycomb surface texturing (Zhao, 1993)

However, the dim light or at the absence of light, the energy density of solar is not sufficient to meet a need. Stordeur and stark et al. introduced a low power thermoelectric generator to achieve electrical energy from thermal energy (Stordeur and Stark, 1997). It is micro self-sufficient system but the problem is that it is hard to get greater than 10 centigrade thermal gradients in a volume of 1 cm^3(Roundy and Rabaey, 2003). In consequence, the improving in battery capacity and sensor communications can decrease the energy consumption of the wireless sensor network but the lifetime of the sensor network still limited. Regarding to harvesting environment energy, its continuous work is field to environment.

Power of wireless sensor nodes network is affected by two main factors; one is amount of data manipulation and second is distance between communications. However, some research focus on minimizing the amount of data collection effectively such as the sensor protocol for information via negotiation which check the redundant data and COUGAR on between nodes to eliminate it (lman, 1999; Ghrke, 2002). It uses declare the queries to abstract the processing from network layer and utilize them into network data aggregation to preserve energy. The purpose of the literature review is to discuss the energy efficient communication protocols that focus on shortening the communication distance and manipulation of data.

LEACH (Rakasan and Balakrishnan, 2000). is the first energy efficient protocol for creating clusters that minimize the transmission routes and amount of data. A little number of sensor nodes in network in LEACH are divided into clusters and each node are selected as head in a self-organizing manner. These head nodes collect the data from the each node in the cluster and combine them into one data group. After, the combination of data, they sent from the cluster to the Base Station (BS). However, the node which is selected as head of cluster rapidly expired because of spending energy on data transmission. W. R Heinzelman et al (Lindsay and Raghavendra, 2002). proposed an effective way by utilizing random rotation of cluster nodes to prevent the cluster heads from quickly termination and the transmission cost is eight times compared with direct transmission.

Raghavendra et al. (Lindsay and Raghavendra, 2002). exploit further improvement in LEACH by utilizing of each sensor node only communicate with adjacent neighbors and in each round only assigned node transmit to base station. This approach could be used as distributed resource consumption in sensor nodes network. PEGASIS created a chain to measure the node which should transmit or receive data from which node. Hence, the nodes will not be visited again in the chain and distance to their neighbor is increased. Additionally, the chain should be again reconstructed when any node terminates. In this technique, they gather the data after constructing the chain in each round and each node receives data from a neighboring node. Neighbor data is aggregated with its own and transmits one message to Base Station. Nevertheless, the PEGASIS outperformed LEACH by three times in simulation results. Moreover, it is assumed that both schemes are performed at least 100m from the base station and the network size was only 50X50 m. unfortunate, these schemes not feasible in recent wireless sensor network systems such as HEMS and FEMS that based on ZigBee devices. Since the distance from sensor nodes to BS is too far as network size is very small.

Moreover, (Olariu and Stojmenovic, 2006). described the uneven energy hole depletion problem in sensor node such as rapid terminating of the sensor node in LEACH and

PEGASIS. They divide network area into concentric coronas from the sink, they made trial to find the specific radius to maximize the network life-time.

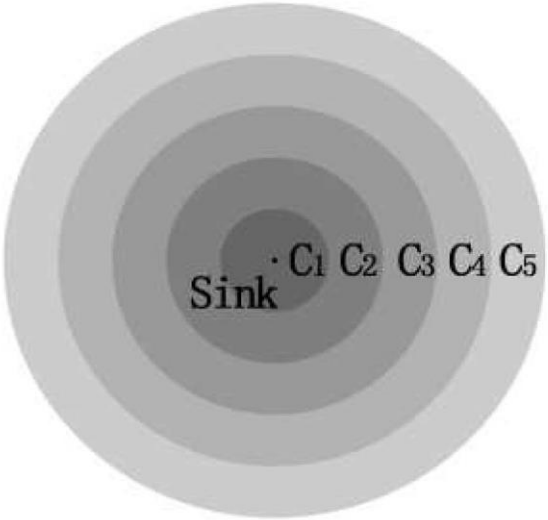


Figure 2.2: Circular area consisting of live coronas (Olariu and Stojmenovic, 2006)

Their idea is more effective as compared to previous approaches and they considered the network topology as a circular and sink node is the center of this circle. Nevertheless, author did not address to wireless communication technology and transmission distance like ZigBee. (Nakamoto and Hussin, 2013). Proposed an effective method, besides considering available transmission distance and he also attempt to enhance the energy efficiency of the particular node which part of this network. Additionally, he proposed a model to measure the location of sensor nodes in order to access the communication distance. Sensor Protocol for Information Negotiation (SPIN) (Chandrakasan, 2002). also improves the energy consumption while data is transmitted only needed neighbor nodes which contain meta-data to reduce redundant information; basic diagram is shown in figure 2.3.

Figure 2.3: Time line showing LEACH operation. Adaptive clusters are made the set up phases (Chandrakasan, 2002)

Scheme, distributed energy efficient clustering(Younis, 2004). is proposed which is based on LEACH by using residual energy. The cluster head is selected on the base of residual energy and node degree and the join of nodes in cluster is to minimize the communication cost. This approach proposed multiple power level of sensor nodes that operates in multiple hop systems using effective power to transmission in communication within the cluster.

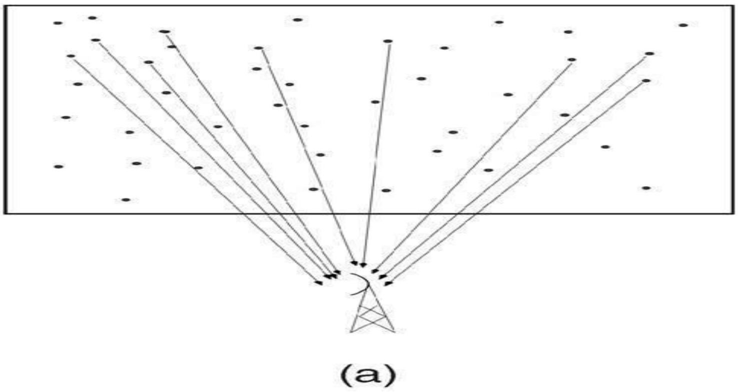
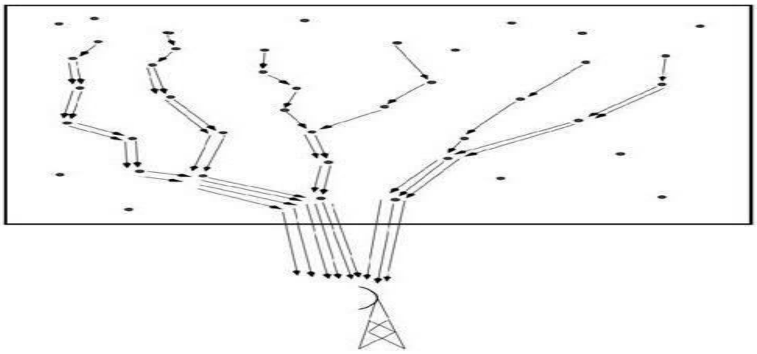
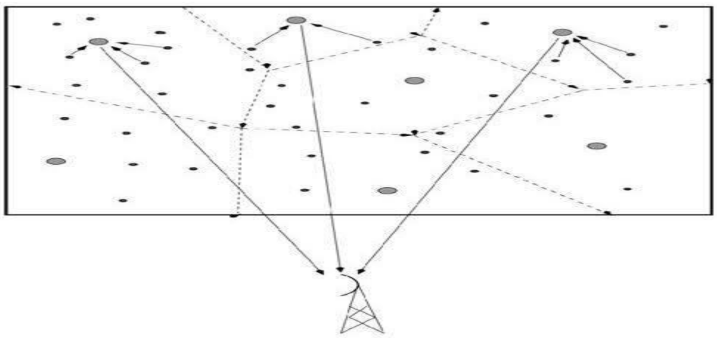


Figure 2.4a: Shows single hop without clustering where each of the head connected to the central control of the Wireless Sensor Network.



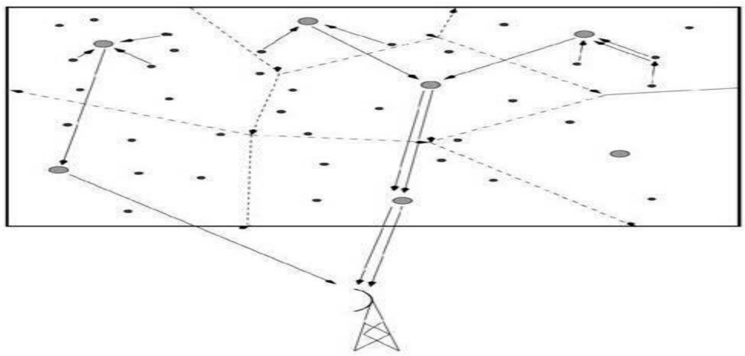
(b)

Figure 2.4b: Shows multiple hop without clustering where information is following on the head nodes in the network



(c)

Figure 2.4c: Single hop with clustering, each node sends information to the hop in the cluster where each cluster contains single hope



(d)

Figure 2.4d: Clustering with Multi-hop connected to the main base station.

Energy efficient clustering scheme (Chengfa, 2006). which focuses on single hop wireless sensor networks and cluster head is elected with more residual energy in independent way in local radio communication. Hence, in this research we will discuss an effective distance based method to balance the load among the cluster heads to enhance the node life time.

In load balance scheme (Zhang, 2011). weight can be measured by distance between the head and the member energy to improve the cluster member choice. The scheme utilized optimization threshold value for load balancing. It also creates balance cluster by load balancing to prolong network lifetime.

On the other hand, a multi hop clustering schemes (Israr, 2006). for load balancing in wireless sensor network also capable on homogenous WSN. It builds on layers approach for intra and inter -cluster communication. Similarly, clustering and load balance in Hybrid sensor network (Ming, 2006). also has proposed an algorithm which suitable on positioning of mobile cluster heads and balance traffic loads in WSN. Hence, it is stated that the location of cluster head can affect the network life time in significant manner by moving the better location in the network.

In (Meenakshi, 2012), they proposed new way to improve the lifetime by utilizing the selection of two cluster heads and hierarchical routing. The basic structure of clustering is shown in figure 2.5. In this scheme, they address two cluster heads in the data routing method from node to base station to save battery life in wireless sensor nodes

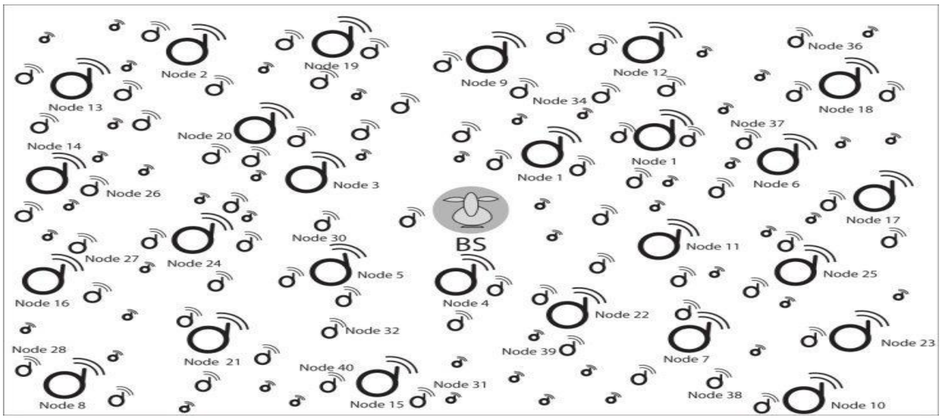


Figure 2.5: Figure Describes the Cluster Formation, CH Selection, Communication Information Form node to BS and Energy Maintenance in Sensor Node (B. Meenakshi, 2012)

A new scheme is introduced in (Jabari, 2008). where they proposed a novel cluster-base routing protocol ELCH (Extending Life-time of Cluster Head). The purpose of proposed algorithm is hierarchical routing and self-configuration properties. The clusters in the network are equally distributed and for the selection of head in a particular cluster, neighbor node will cost the nodes in order to elect suitable cluster heads. However, most of them using random selection for electing the head nodes in a cluster. Fault et al. (Munaga, 2008). proposed clustering (FTTC) algorithm in which cluster head is selected on the base of traffic and rotates cluster head periodic manner. They adopt data aggregation to reduce the packet size therefore wireless communication cost is decreased by reducing the data packets. Hence, lifetime is extended by reducing the energy consumption in the network.

On Life Time of Sensor Network (OLTSN) (Chen, 2005). a comprehensive algorithm is discussed to maintain the lifetime of wireless sensor networks. They propose medium access protocol which introduced both channel state information and residual energy information of a single node. They try to maximize the minimum energy across the wireless network during each data collection.

The above schemes are focusing on the battery optimization and reduce the communication burden on nodes. Now we want to review the wireless power transfer schemes which are introduced in recent years.

2.1 Wireless Power Transfer

In this part of review, we introduce the research on vehicles and wireless power transfer in working WSNs.

A number of researchers focused on supplying power to the aerial vehicles from ground to increase their flight time. A microwave powered helicopter was introduced in 1964 to fly 60feets above a transmitting antenna (William, 1984). Achtelik et al [32] design a quadcopter structure which burst the micro aerial vehicle survival record with LASER beam, model is shown in figure 1. It based on infrared laser system and they utilized complex optics to direct the laser beam to optimized solar cell equipped on the quadcopter. In this solar cell set, the laser beam transfer electric energy. According to Achtelik et al. experiment, this one kg quadcopter can do unlimited flight time.

In contrast, the micro Unmanned Aerial Vehicle (UAV) scheme which gives novel solution to charge the sensor nodes in the network. Hence, our interest is to find out optimized prolonging lifetime of sensor network by using UAV.

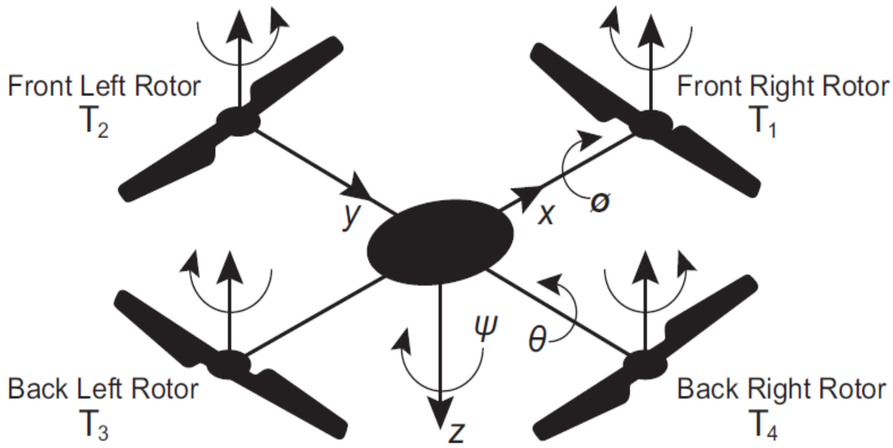


Figure 2.6: Quadcopter Structure for UAV Scheme. Router 1 and 3 spin in one direction while Router 2 and 4 will spin in the Inverse Direction

The UAV is high flexible in movement which are more beneficial for mobile power station for wireless devices that are not easily accessible by regular energy sources. Griffin and Detweiler et al. create a UAV based wireless power transmission system which built on magnetic resonant power transfer (Griffin and Detweiler, 2012). Kurs et al (Kurs, 2007). demonstrate a efficient medium range distance nonradiative power transfer system. Additionally, it has low interference with any neighbor objects and can work around and through objects area. Hence, it is beneficial to use for charging to remote sensor which are not easily accessible with regular charging sources.

2.2 Charging Algorithms

In this section, we described the current work on charging strategies in wireless sensor network. Due the reason of different condition, such as energy consumption of each sensor node, number of chargers, there are several scenarios.

Pen et al (Peng, 2010). described the problem in a scenario where the nodes periodically send message of energy information to the head and combined report consist of energy information about the k shortest lifetime nodes. They equationte the problem and sketch of the NP=completeness proof, reduction from TSP problem. They proposed two algorithm time complexity are superpolynomial. The main idea was to test each permutation of the charging sequence. They construct the proof of concept prototype of the system and used simulation to analyze the proposed algorithms. We study that their proof is not accurate because the TSP problem requires that each node to be visited only one time while the transformed problem requires that each node to be visited at least once.

Singh et al (Recently, 2016). represents the battery issues and focus on the energy loss problem where it occurs mostly on the edges of wireless sensor network when most of the packets drops during transmission. Thus, they introduced the concept of super node which can be deployed at edges of the network or inside to the network. For the Cluster head election the both techniques; clustering technique and direct communication are employed to minimize the energy consumption loss. Their scheme improved form of E-zone model.

Tong et al (Tong, 2017) first investigate the problem of energy consumption in wireless sensor network, aiming to maintain the energy consumption among the sensor nodes during the data propagation process. By using hop by hop transmission model for data toward the sink node, address the poor energy balancing over the network. They applied slice based energy model and divide the problem into slice and intra slice energy balancing problems. They approach is based on the probability based strategy, called as inter-slice mixed transmission protocol and an intra-slice forward technique. Their contribution achieved significant energy balancing and reduce the loss of energy in term of decrease network delay compared with the hop by hop transmission and cluster based protocol.

CHAPTER 3
METHODOLOGY

3.1 Background

Due to the progress in wireless technology wireless sensor networks have been wieldy employed in different applications, including environmental and structural monitoring, surveillance, medical diagnostics, and manufacturing process flow. Most of wireless sensor network applications are designed to operate for long periods of time without human intervention, despite depending on batteries or energy harvesting for energy resources (Jamal, 2004). So that several researches focused on how to extend the network lifetime through the efficient use of energy by applying different proposed methods.

Wireless sensor networks (WSN) contain numerous number of sensor node, each node has its battery of sole energy source. The batteries of these sensor nodes cannot be recharged, therefore wireless sensor network devices have limited battery energy to complete processing and communication task. So, one of the main challenges issues in wireless sensor networks domain is how to design a low energy consuming network.

Recently different methods have been introduced in order to analyze and improve battery life duration (Kerasiotis, 2010). Various techniques have been employed to manipulate battery lifetime and energy consumption. To maximize the operational lifetime of powered systems like device nodes, it's necessary to take into account the advantage of battery characteristics to have friendly better battery.

In some systems, there are four important sources for energy consumption:

energy that necessary for keeping communication radios on, transmission and reception of control packets energy, energy for keeping sensors on, and actual data transmission and reception energy. Normally the total energy required for actual data transmission and reception is relatively small in these systems, because events occur so seldom. On the other hand, sense events required energy is mostly constant and cannot be controlled.

Thus, the energy required for keeping the communication system on is consume the most of the energy, so it's necessary to be controlled in order to extend the network lifetime.

Sleep-wake scheduling is one of the effective mechanisms to extend the lifetime of energy in sensor networks. Nodes are put to sleep when there are no events, so that the energy consumption of the sensor nodes can be significantly reduced.

Asleep-wake scheduling technique is utilized in order to extend the battery lifetime and optimize power consumption. When nodes are not in use sleep is activated, and wake up status activated when the node receive transmitted data from the neighborhood node. When necessary to reduce the delays "any cast" based packet forwarding schemes is developed, where each node opportunistically forwards a packet to the first neighboring node that wakes up among multiple candidate nodes.

In the literature several types of sleep-wake scheduling protocols have been proposed to prolong the battery lifetime. In (Tseng and Hsieh, 2003; Elson and Estrin, 2002). the researchers proposed a synchronized sleep-wake scheduling protocols where sensor nodes periodically or a periodically exchange synchronization information with neighboring nodes. While authors in (Shih, 2001; Nosovic and Todd, 2000). introduced on-demand sleep-wake scheduling protocols where nodes circuitry are turn off mostly and secondary low-powered receiver turn on to be able to listen for wake-up calls from neighboring nodes when there is a need for relaying packets. But the problem of these on demand sleep-wake scheduling it increase the cost of sensor motes due to the additional receiver.

In this work, we are intended to develop a new method that based on sleep-wake scheduling function to reduce the energy consumed by wireless sensor networks.

In this introduced approach there are two main control functions for each duty cycle which are wake function and sleep function. During wake function status, sensor nodes perform communication with neighbor nodes. While during sleep function status, all communications in sensor nodes suspended so as to save the energy.

To reduce the energy consumption sensor nodes put to sleep or wake periodically, sensor nodes are put to sleep when there are no events; this significantly reduced the energy consumed by the sensor nodes. The introduced method proposes asynchronous sleep–wake scheduling. In this method, node wakes up independently of neighboring nodes in order to save energy. On other hand the process of switching on and off is also result in energy consumption therefore it is no recommended to put the node to sleep very frequently. There are different techniques have been used to model this problem, like charging switching cost whenever the node turn on

3.2 The System Architecture

In the first stage the wireless sensor network is build, and then sensor nodes are set to sleep, when data transmitted from neighbor nodes wakeup status will activated. Energy table is establish to record the consumed energy rates, and then based on this table sleep and wake function of sensor nodes arranged. During the building of the network, sensor nodes distributed within the range based on the center of the circle and the radius R.

After sensor nodes distributed in that area randomly, *A* will process the action of delaminating and the levels are separated. The energy consumption is performed by using below equations and parameters:

$$ETX\,(L,\,d) = L * Eelec + L * \_amp * d\ 2 \qquad (1)$$

$$ERX\,(L) = L * Eelec \qquad (2)$$

Equation 1 represents the energy consumed by sensor nodes for data transmission. While equation 2 represents the energy consumed by sensor nodes when they receive data. The bit of packet load represented parameter of *L*, where *Eelec* means the required power of sensor node in a data transmission. Data transmission process lead to e enlarge the whole wireless power, which also result in the increasing of $L * {}^{\varepsilon}amp * d^{r}$. The parameter *d* represents the distance between sensor nodes, and the parameter ${}^{\varepsilon}amp$ represent the required power for enlarging the wireless power.

In order to minimize energy consumption firstly all nodes must set to sleep and then the probability of each node to enter sleep status the density of the entire wireless sensor networks can be calculated according to equation (3):

$$\lambda = N/A \tag{3}$$

where λ represent the distributive density, N means the number of all sensor nodes, and A is the area of the entire wireless sensor networks that computed by $A = \pi R^2$

The density can be computed by equation 4 as follow:

$$\lambda_i = N_i ,/A_i, \; i = 1,2,3\ldots.n \tag{4}$$

Where Ni is the number of active sensor nodes in each level, Ai refer to the area in each level where $A_i = (2_i - 1) - r^2$ (r is the radius in each level). λ_i can also be calculated by

Equation 5 as shown below:

$$\lambda_i = (1 \_ Ps_i)\, \lambda , \; i = 1, 2, 3\ldots n \tag{5}$$

Psi represent level I sleep probability , 1 _ Psi is the probability of being wake for sensor nodes in level i, equation 5 represents the density in level i for wake sensor nodes, and it can also present the density of the whole wireless sensor networks multiply by the active probability.

Thus equation 4 and equation 5, can be combined to compute λ_i as show in equation 6 below:

$$(1 - Ps_i)\, \lambda = N_i /A_i, \; i = 1, 2, 3\ldots. n \tag{6}$$

The number of active or wake sensor nodes in level i can be estimated by equation 7:

$$N_i = (1 - Ps_i)\, \lambda A_i, \; i = 1, 2, 3\ldots n \tag{7}$$

, equation 4 and equation 7 can be combined to result in equation 8:

$$(1 \_ Ps_i)A_i = \lambda_i A_i, \; i = 1, 2, 3,.. \text{n} \quad (8)$$

$$Psi = 1 - (\lambda_i / \lambda) \quad (9)$$

Equation 9 give the probability of sleeping of sensor nodes in each level.

When the probability of sleeping of each level is calculated, we can determine which sensor node should sleep and which sensor node should be active or wake as shown by equation 10:

$$N_i * Ps_i = S_i \quad (10)$$

Where N_i is the number of all sensor nodes in the level i, S_i represent the number of the sleep sensor nodes. When we find the number sleep sensor nodes S_i we select the sleep sensor node in this level randomly, then the sleep schedule process start, and the sensor nodes that prepare to sleep will be put to sleep according to their residual power and the energy table.

3.3 Simulation Process

The simulation is conducted by setting up some necessary fixed groups', accordingly the active proportion of residual power in each stage is fixed. The sleep and active or wake proportion of each stage is considered to be vary, and the optimal combination is made by specific measurement. The process is performed as follows:

For first sage, the sleep and active proportion is set at 50% and the residual power is set up to 80-90 (%). So, we only make the sleep and active proportion of the 90-100 (%) residual power vary. The sleep proportion and active proportion is 10% and 90% respectively as shown in Table 3.1.

Table 3.1: Starting power level of the battery 1

E ratio (%)	Sleep (%)	Active (%)
90-100	10	90
80-89	50	50
70-79	50	50
60-69	50	50
50-59	50	50
40-49	50	50
30-39	50	50
20-29	50	50
10-19	50	50

From the second stage, the proportion is set to 10% and 90% into 20% and 80%, as indicated in Table 3.2.

Table 3.2: Starting power level of the battery 2

E ratio (%)	Sleep (%)	Active (%)
90-100	20	80
80-89	50	50
70-79	50	50
60-69	50	50
50-59	50	50
40-49	50	50
30-39	50	50
20-29	50	50
10-19	50	50

Hence we continue fixing the proportion for other stages and set the sleep and active proportion to 90- 100 (%) residual power as in Table 3.3.

Table 3.3: Starting power level of the battery 3

E ratio (%)	Sleep (%)	Active (%)
90-100	20	80
80-89	50	50
70-79	50	50
60-69	50	50
50-59	50	50
40-49	50	50
30-39	50	50
20-29	50	50
10-19	50	50

In this simulation the optimal sleep and active or wake proportion is considered to the stage of 90-100 (5). But the problem we have to take in our account is that the sleep and active proportion for sensor nodes is not often 50% and 50% in the actual wireless sensor networks (take the fixed group (10% and 90%) for example, as shown in Table 3.4.a, b, c).

Table 3.4a: Starting power level of the battery 4

E ratio (%)	Sleep (%)	Active (%)
90-100	10	90
80-89	10	90
70-79	10	90
60-69	10	90
50-59	10	90
40-49	10	90
30-39	10	90
20-29	10	90
10-19	10	90

In this simulation study as shown in Figure 3.1 – 3.9, we introduced a method to build the future power table, , sensor nodes followed by their own status and the remaining amount of electric power based on the table to do sleep with the scheduled action mechanism.

Table 3.4b: Starting power level of the battery 5

E ratio (%)	Sleep (%)	Active (%)
90-100	20	80
80-89	10	90
70-79	10	90
60-69	10	90
50-59	10	90
40-49	10	90
30-39	10	90
20-29	10	90
10-19	10	90

Table 3.4c: Starting power level of the battery 6

E ratio (%)	Sleep (%)	Active (%)
90-100	30	70
80-89	10	90
70-79	10	90
60-69	10	90
50-59	10	90
40-49	10	90
30-39	10	90
20-29	10	90
10-19	10	90

It can be noticed that if there are more remaining power situation, the proportion of sleep become relatively short, so it is clear that sensor node is measured better in the electric case, and the active cycle is longer to operate more with the sensing ability, increasing the whole wireless sensor network performance.

Averaging the ratio of the fixed group, lead to the best proportion of sleep and wake as shown in Table 3.5.

Table 3.5: Optimal Power Combination

E ratio (%)	Sleep (%)	Active (%)
90-100	17	83
80-89	25	75
70-79	32	68
60-69	43	57
50-59	55	45
40-49	58	42
30-39	70	30
20-29	90	10
10-19	90	10

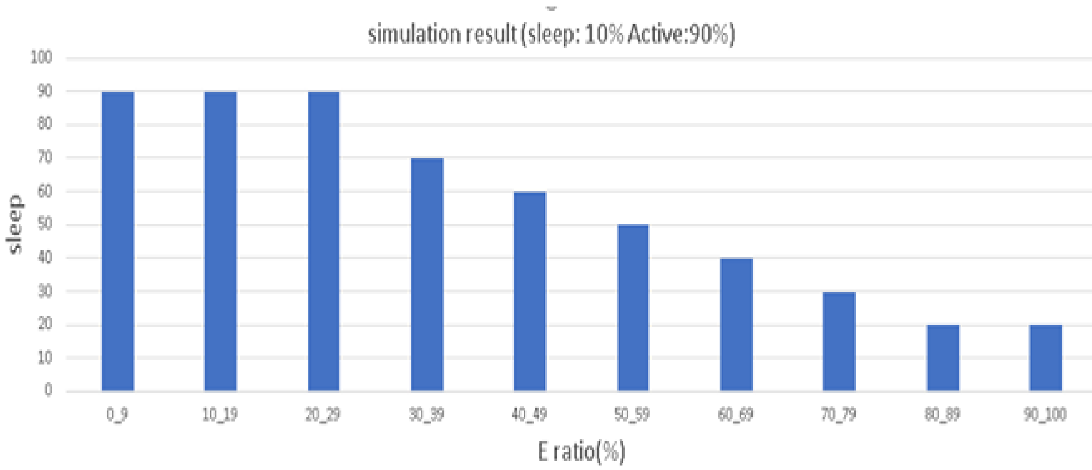


Figure 3.1: Simulation Result at Sleeping:10% and Active:90%

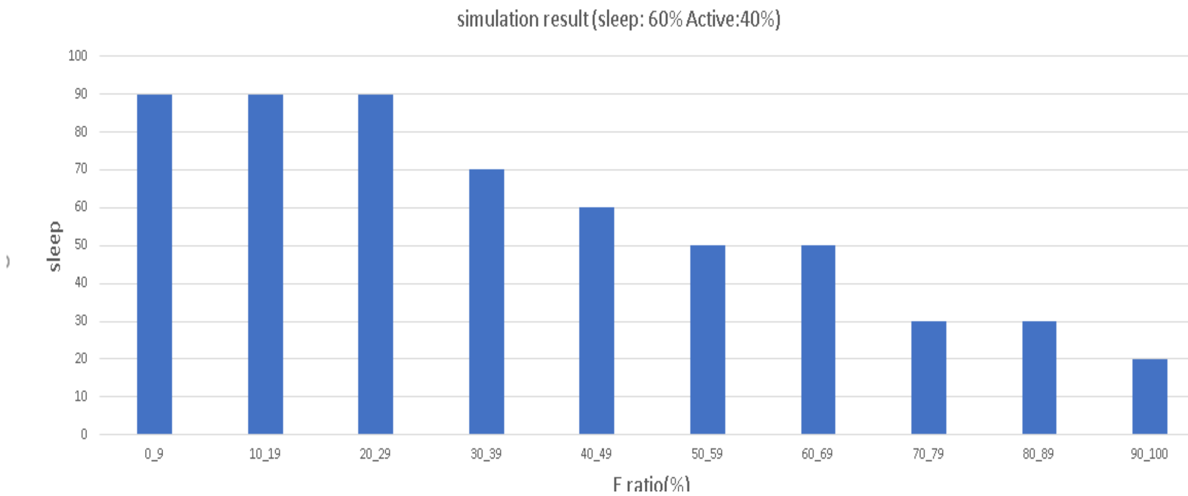


Figure 3.2: Simulation Result at Sleeping:60% and Active:40%

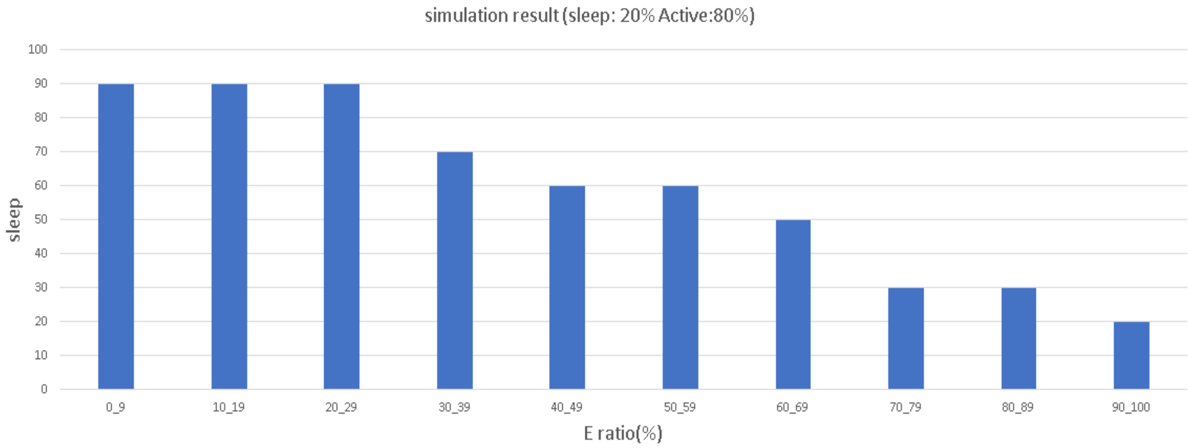


Figure 3.3: Simulation result at Sleeping:20% and Active:80%

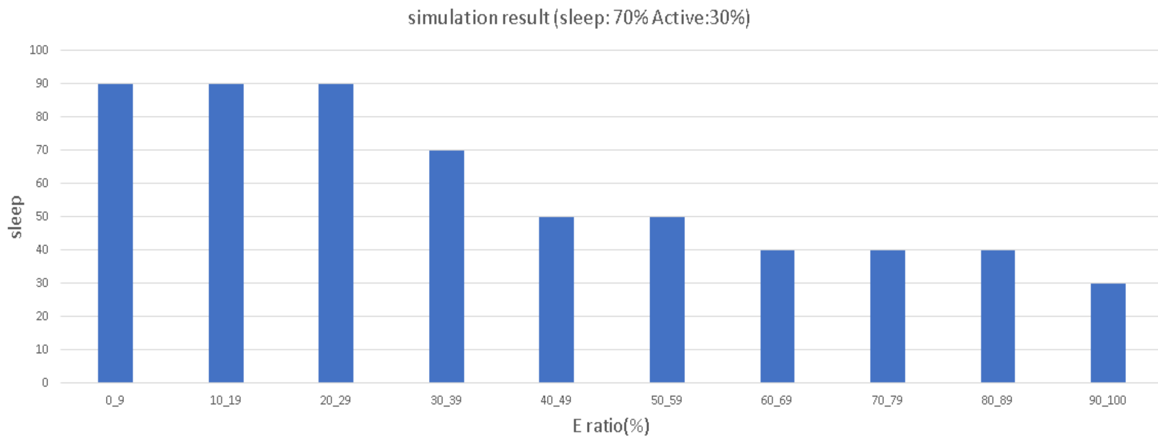


Figure 3.4: Simulation result at Sleeping: 70% and Active:30%

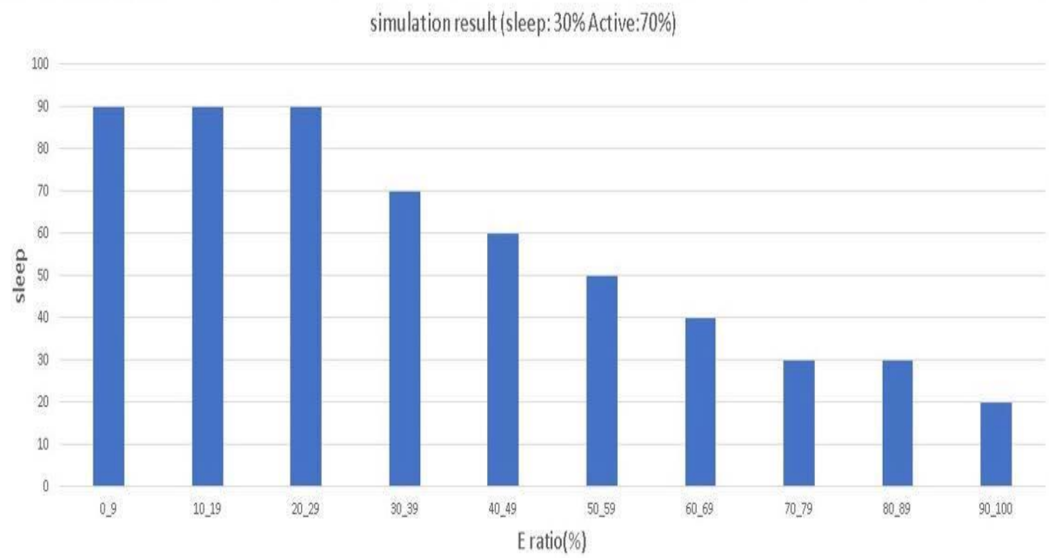


Figure 3.5: Simulation result at Sleeping: 30% and Active: 70%

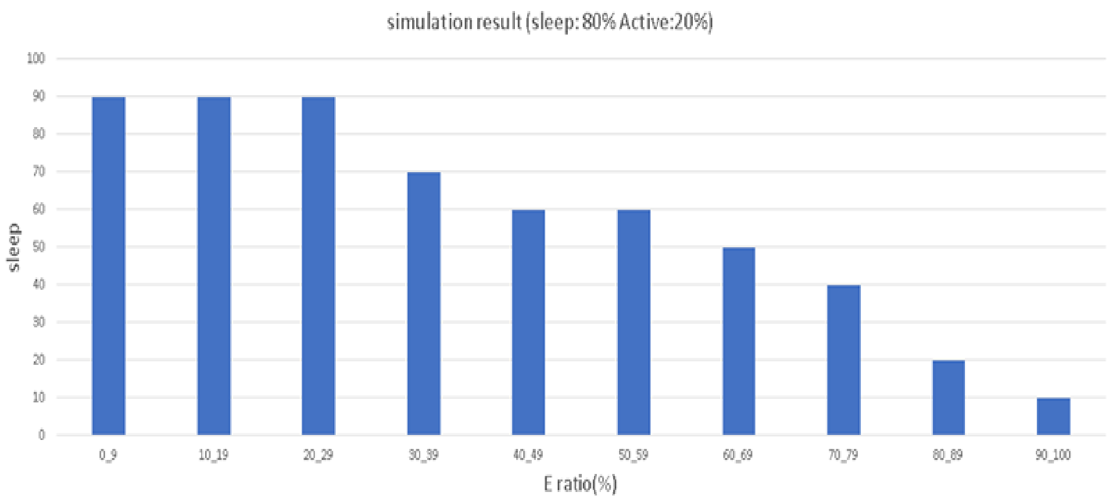


Figure 3.6: Simulation result at Sleeping: 80% and Active:20%

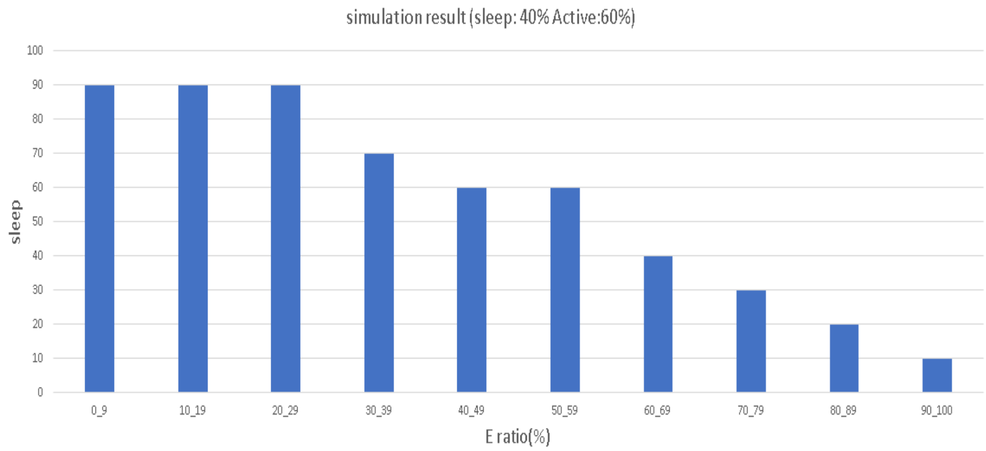


Figure 3.7: Simulation result at Sleeping:40% and Active:60%

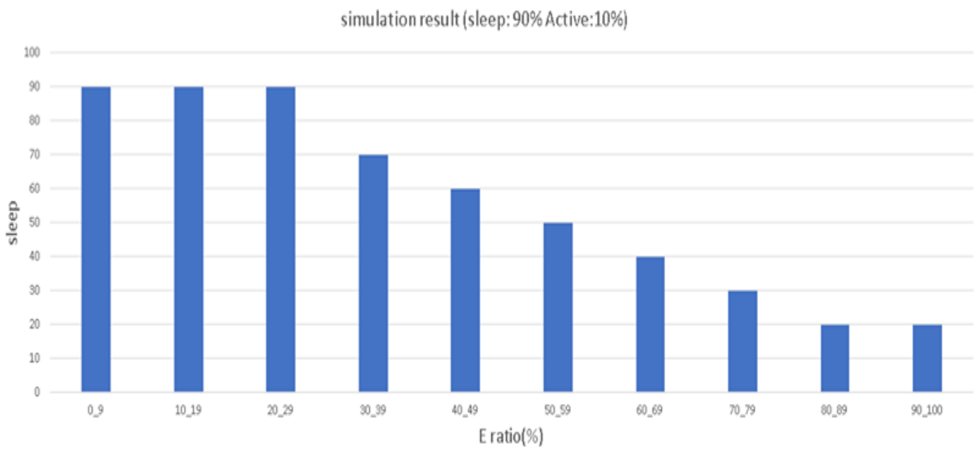


Figure 3.8: Simulation result at Sleeping:90% and Active:10%

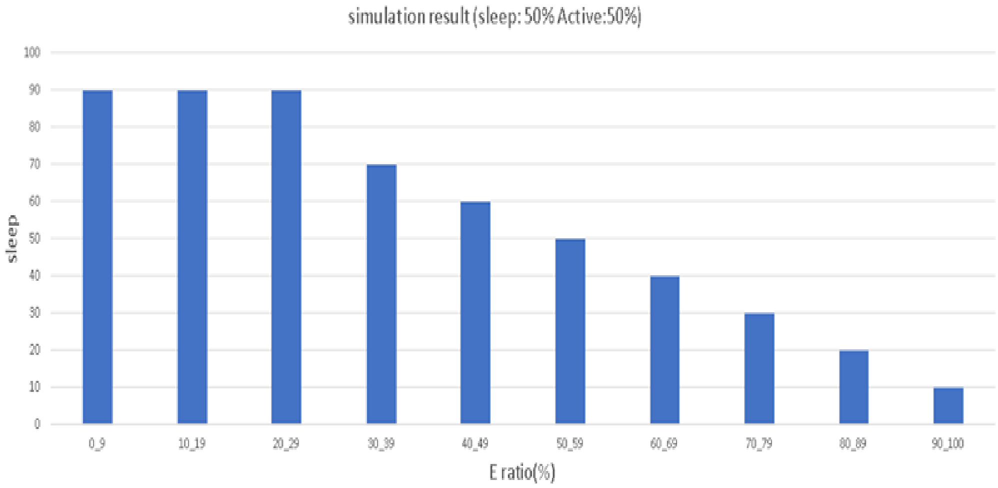


Figure 3.9: Simulation result at Sleeping: 50% and Active: 50%

To specify which node will enter into sleep mode the probability of sleep status for each level is calculated, the nodes will be selected to get into sleep status randomly. Each node is set the sleeping scheduling according to the power list and the steps of the below flowchart:

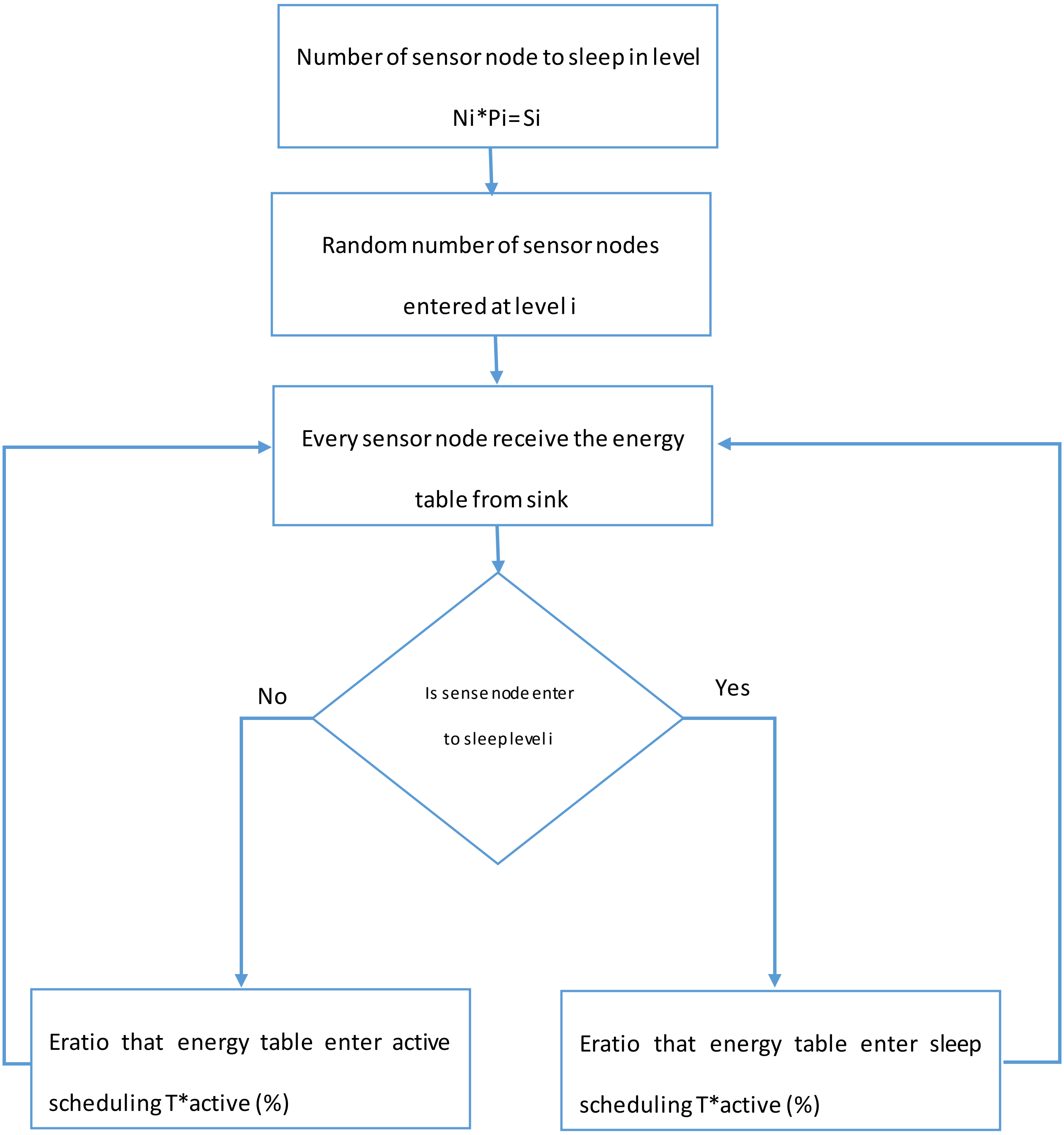


Figure 3.10: flowchart of proposed algorithm

CHAPTER 4
SIMULATION RESULTS AND ANALYSIS

4.1 Analysis

In this proposed method for sleep and wake function the control mechanism is based on dynamic scheduling method. The sleep probability for each level is calculated by the density. The nodes that are away from the sink lead in the increase of the sleep probability so it decreases the forward frequency of the nodes near to the sink. So by this way, the nodes closer to the sink can share the consumption of energy and preserve the energy.

The sleep and wake is very important for the wireless sensor networks.

If the node is set up to active status for long period of time, a lot of energy will be waste. In contrast, if the nodes set up to sleep status for long duration, the transmission will fall in delay problem. In this paper, we design a optimal sleeping control mechanism to avoid both of the situations.

4.2 Simulation Results

Matlab2016a is used for simulation of proposed system as it shown in appendix 1 and appendix 2. The environment of wireless sensor networks used in the operation of simulation is as follows:

- Environment area: 25 m * 25 m
- Sensor nodes: distribute 300 pcs randomly
- Packet load: 40000 bits
- Initial power: 2J
- Sensor node sensing power: 5 * 10-8 J
- Transmission range: 2 m
- Duty Cycle T: 20 time slots.

The average of the remaining energy of sensor node is illustrated as shown in Figure 4.1 and Matlab code is shown in Appendix 3.

Normal method is not considering the sleep wake function, so we can notice that the energy consumption is unstable. For the random method sensor nodes are set to sleep for a short duration, this cause the power consumption to be rapidly in random sleep method. In contrast, the sensor node that sleep for long time may have too much remaining energy and this will stop the network from operating. Since the remaining energy of nodes is too high it is easier to obtain the remaining energy of whole network.

In the proposed method sensor node load is balanced by using energy table in order to control the sleep and active time.

The comparison of the sensor lifetime is illustrated as in Figure 4.2. In normal method the nodes that keep in wake status for long time usually dead, while in random method, the energy is exhausted for the nodes that sleep for short duration.

In our proposed method, the probability is computed by using the distributive density before the sleep function is performed. This produce a proper throughput for the entire wireless sensor networks which balance the load of the whole network, and also reduce the reply frequency for the sensor nodes that more closer to sink.

The simulation results proved that the proposed method is effectively save the energy of sensor nodes and prolong the network lifetime. By controlling the sleep and active time based on remaining energy of sensor nodes we can save more power for sensor nodes so the life time of the entire wireless sensor networks can be extended.

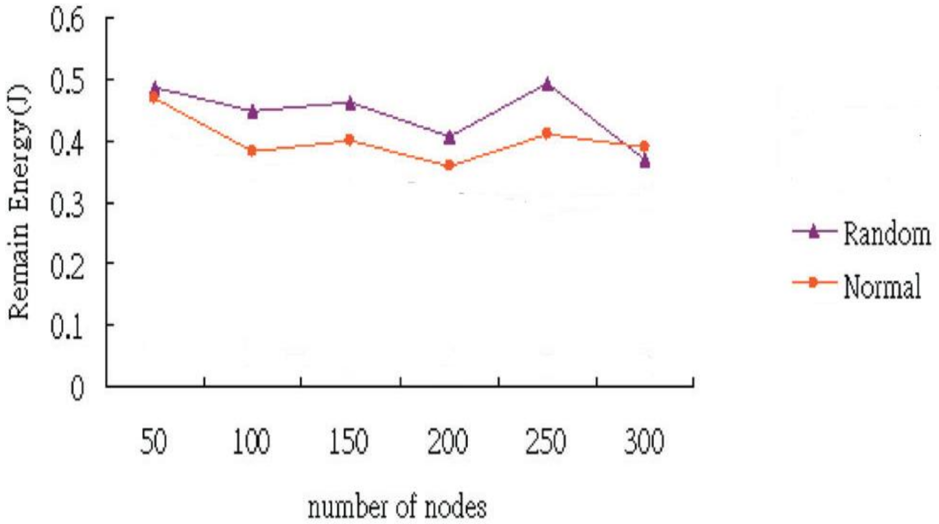


Figure 4.1: Comparison of the Remain Energy among Sensor Nodes.

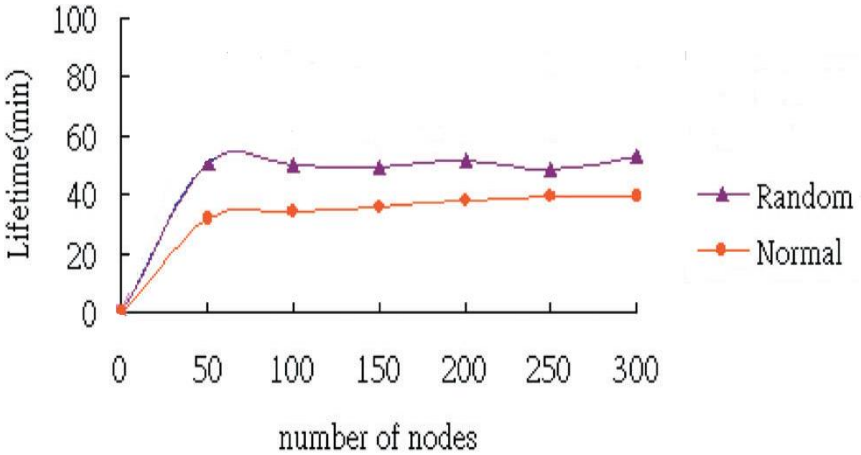


Figure 4.2: Comparison of the Remain lifetime of Sensor Nodes

CHAPTER 5
CONCLUSION & FUTURE WORK

Wireless sensor networks have been widely employed in a broad range of applications. Wireless sensor networks (WSNs) is a collection of small sized sensor nodes, which form a distributed sensing network that used data propagation technique to collect information in the network of the sensor nodes in the physical environment. The sensor nodes have limited energy resources and depend on batteries to supply energy, so it is important to efficiently manage this power. We conclude that the wireless communication the time energy consumption taken for the packet to be sent and received is the main factor. The remaining energy decreases with increase the number of days after deployment of the network nodes. However, the several algorithms adopted to maximize the life of wireless sensor network nodes but still it is subject of interest.

The sensor node has four basic components: sensing unit, processing unit, radio unit, and power unit. The basic role of sensor networks is to gather data from a remote terrain periodically, where each node senses the environment continually and sends back the data to the base station (BS) for further analysis; the BS is usually located considerably far from the target area.

The basic problem of wireless sensor network lifetime is the limited energy of the deployed sensor nodes. Because sensor nodes usually contain limited energy of power source, this mechanism takes into account the issue of energy efficiency management. So that in this paper, we design sleep and wake up mechanism to extend the sensor wireless battery lifetime. By managing the sleep and wake time according to the remaining energy of sensor nodes. The introduced method will save more energy for sensor nodes that obviously extend the life time of the entire wireless sensor networks. the simulation results has shown that the proposed mechanism could effectively reserve the energy of sensor nodes and prolong the network lifetime. By controlling the sleep and active time based on remaining energy of sensor nodes we can save more power for sensor nodes so the life time of the entire wireless sensor networks can be extended.

Future Work

The future work is that two Wireless Sensors exchange operation between Wi-Fi and Sensor operates only the first sensor and the second device is running Wi-Fi only when receiving the sensor data operates Wi-Fi was sent data to the device that is running Wi-Fi only to be sent Data to the next node, knowing that it is switching between the devices by a time counter at the time of the operation of Wi-Fi and the operation of the sensor, for example, operates Wi-Fi and stop the sensor in the first device for one hour the second device will work. Hence, exchange between the two devices.

REFERENCES

Achtelik, C., Stumpf, J., Gurdan, D., & Doth, K. (2011). Design of a flexible high performance quadcopter platform breaking the mav endurance record with laser power beaming. In *Intelligent robots and systems (iros), 2011 ieee/rsj international conference on* (pp. 5166-5172). IEEE.

Karaki, N., & Kamal, A. (2004). Routing techniques in wireless sensor networks: a survey. *IEEE wireless communications*, *11*(6), 6-28.

Anisi, H., Abdul-Salaam, G., Idris, I., Wahab, A., & Ahmedy, I. (2017). Energy harvesting and battery power based routing in wireless sensor networks. *Wireless Networks*, *23*(1), 249-266.

Brown, W. (1984). The history of power transmission by radio waves. *IEEE Transactions on microwave theory and techniques*, *32*(9), 1230-1242.

Chang, H., & Tassiulas, L. (2000). Energy conserving routing in wireless ad-hoc networks. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE* (Vol. 1, pp. 22-31). IEEE.

Chen, Y., & Zhao, Q. (2005). On the lifetime of wireless sensor networks. *IEEE Communications letters*, *9*(11), 976-978.

Elson, J., Girod, L., & Estrin, D. (2002). Fine-grained network time synchronization using reference broadcasts. *ACM SIGOPS Operating Systems Review*, *36*(SI), 147-163.

Griffin, B., & Detweiler, C. (2012). Resonant wireless power transfer to ground sensors from a UAV. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on* (pp. 2660-2665). IEEE.

Heinzelman, B., Chandrakasan, P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on wireless communications*, *1*(4), 660-670.

Heinzelman, R., Chandrakasan, A., & Balakrishnan, H. (2000). Energy-efficient communication protocol for wireless microsensor networks. In *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on* (pp. 10-pp). IEEE.

Heinzelman, R., Kulik, J., & Balakrishnan, H. (1999). Adaptive protocols for information dissemination in wireless sensor networks. In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking* (pp. 174-185). ACM.

Israr, N., & Awan, I. (2006). Multi-hop clustering algo. For load balancing in WSN. *International Journal of SIMULATION*, *8*(1).

Kansal, A., Potter, D., & Srivastava, M. (2004). Performance aware tasking for environmentally powered sensor networks. *ACM SIGMETRICS Performance Evaluation Review*, *32*(1), 223-234.

Kerasiotis, F., Prayati, A., Antonopoulos, C., Koulamas, C., & Papadopoulos, G. (2010). Battery lifetime prediction model for a wsn platform. In *Sensor Technologies and Applications (SENSORCOMM), 2010 Fourth International Conference on* (pp. 525-530). IEEE.

Kurs, A., Karalis, A., Moffatt, R., Joannopoulos, D., Fisher, P., & Soljačić, M. (2007). Wireless power transfer via strongly coupled magnetic resonances. *science*, *317*(5834), 83-86.

Li, Q., Aslam, J., & Rus, D. (2001). Online power-aware routing in wireless ad-hoc networks. In *Proceedings of the 7th annual international conference on Mobile computing and networking* (pp. 97-107). ACM.

Li, Z. (2013). Improving sensor network lifetime with wireless charging technology.

Lindsey, S., & Raghavendra, C. S. (2002). PEGASIS: Power-efficient gathering in sensor information systems. In *Aerospace conference proceedings, 2002. IEEE* (Vol. 3, pp. 3-3). IEEE.

Liu, T., Gu, T., Jin, N., & Zhu, Y. (2017). A Mixed Transmission Strategy to Achieve Energy Balancing in Wireless Sensor Networks. *IEEE Transactions on Wireless Communications*, *16*(4), 2111-2122.

Lotf, J., Bonab, N., & Khorsandi, S. (2008). A novel cluster-based routing protocol with extending lifetime for wireless sensor networks. In *Wireless and Optical Communications Networks, 2008. WOCN'08. 5th IFIP International Conference on* (pp. 1-5). IEEE.

Ma, M., & Yang, Y. (2006). Clustering and load balancing in hybrid sensor networks with mobile cluster heads. In *Proceedings of the 3rd international conference on Quality of service in heterogeneous wired/wireless networks* (p. 16). ACM.

Meenakshi, B., & Anandhakumar, P. (2012). Lifetime extension of wireless sensor network by selecting two cluster heads and hierarchical routing. In *Proceedings of the International Conference on Advances in Computing, Communications and Informatics* (pp. 1254-1260). ACM.

Mehra, A., Zhang, X., Ayón, A., Waitz, I., Schmidt, M., & Spadaccini, C. (2000). A six-wafer combustion system for a silicon micro gas turbine engine. *Journal of Microelectromechanical systems*, *9*(4), 517-527.

Munaga, H., Murthy, R., & Venkateswarlu, N. (2011). A Fault Tolerant Trajectory Clustering (FTTC) for selecting cluster heads inWireless Sensor Networks. *arXiv preprint arXiv:1108.0747*.

Muruganathan, D., Ma, C., Bhasin, R., & Fapojuwo, O. (2005). A centralized energy-efficient routing protocol for wireless sensor networks. *IEEE Communications Magazine*, *43*(3), S8-13.

Nosovic, M., & Todd, T. (2000). Low power rendezvous and RFID wakeup for embedded wireless networks. In *In Annual IEEE Computer Communications Workshop*.

Olariu, S., & Stojmenovic, I. (2006). Design guidelines for maximizing lifetime and avoiding energy holes in sensor networks with uniform distribution and uniform reporting. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings* (pp. 1-12). IEEE.

Peng, Y., Li, Z., Zhang, W., & Qiao, D. (2010). Prolonging sensor network lifetime through wireless charging. In *Real-time systems symposium (RTSS), 2010 IEEE 31st* (pp. 129-139). IEEE.

Powercast, "Online link," http://www.powercastco.com.

Roundy, S., Steingart, D., Frechette, L., Wright, P., & Rabaey, J. (2004). Power sources for wireless sensor networks. *Wireless sensor networks*, 1-17.

Roundy, S., Wright, K., & Rabaey, J. (2003). A study of low level vibrations as a power source for wireless sensor nodes. *Computer communications*, *26*(11), 1131-1144.

Shih, E., Cho, H., Ickes, N., Min, R., Sinha, A., Wang, A., & Chandrakasan, A. (2001). Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks. In *Proceedings of the 7th annual international conference on Mobile computing and networking* (pp. 272-287). ACM.

Sim, Y., Kim, G., & Yang, S. (2001). Fabrication of micro power source (mps) using a micro direct methanol fuel cell (/spl mu/dmfc) for the medical application. In *Micro Electro Mechanical Systems, 2001. MEMS 2001. The 14th IEEE International Conference on* (pp. 341-344). IEEE.

Singh, S., Panchal, M., & Jain, R. (2016). Fuzzy Logic Based Energy Efficient Network Lifetime Optimization in Wireless Sensor Network. In *Micro-Electronics and Telecommunication Engineering (ICMETE), 2016 International Conference on* (pp. 493-498). IEEE.

Stordeur, M., & Stark, I. (1997). Low power thermoelectric generator-self-sufficient energy supply for micro systems. In *Thermoelectrics, 1997. Proceedings ICT'97. XVI International Conference on* (pp. 575-577). IEEE.

Tong, B., Li, Z., Wang, G., & Zhang, W. (2009). On-demand node reclamation and replacement for guaranteed area coverage in long-lived sensor networks. *Quality of service in heterogeneous networks*, 148-166.

Tong, B., Wang, G., Zhang, W., & Wang, C. (2011). Node reclamation and replacement for long-lived sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, *22*(9), 1550-1563.

Tseng, C., Hsu, S., & Hsieh, T. (2003). Power-saving protocols for IEEE 802.11-based multi-hop ad hoc networks. *Computer Networks*, *43*(3), 317-337.

Yang, S., Dai, F., Cardei, M., Wu, J., & Patterson, F. (2006). On connected multiple point coverage in wireless sensor networks. *International Journal of Wireless Information Networks*, *13*(4), 289-301.

Yao, Y., & Gehrke, J. (2002). The cougar approach to in-network query processing in sensor networks. *ACM Sigmod record*, *31*(3), 9-18.

Younis, O., & Fahmy, S. (2004). HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on mobile computing*, *3*(4), 366-379.

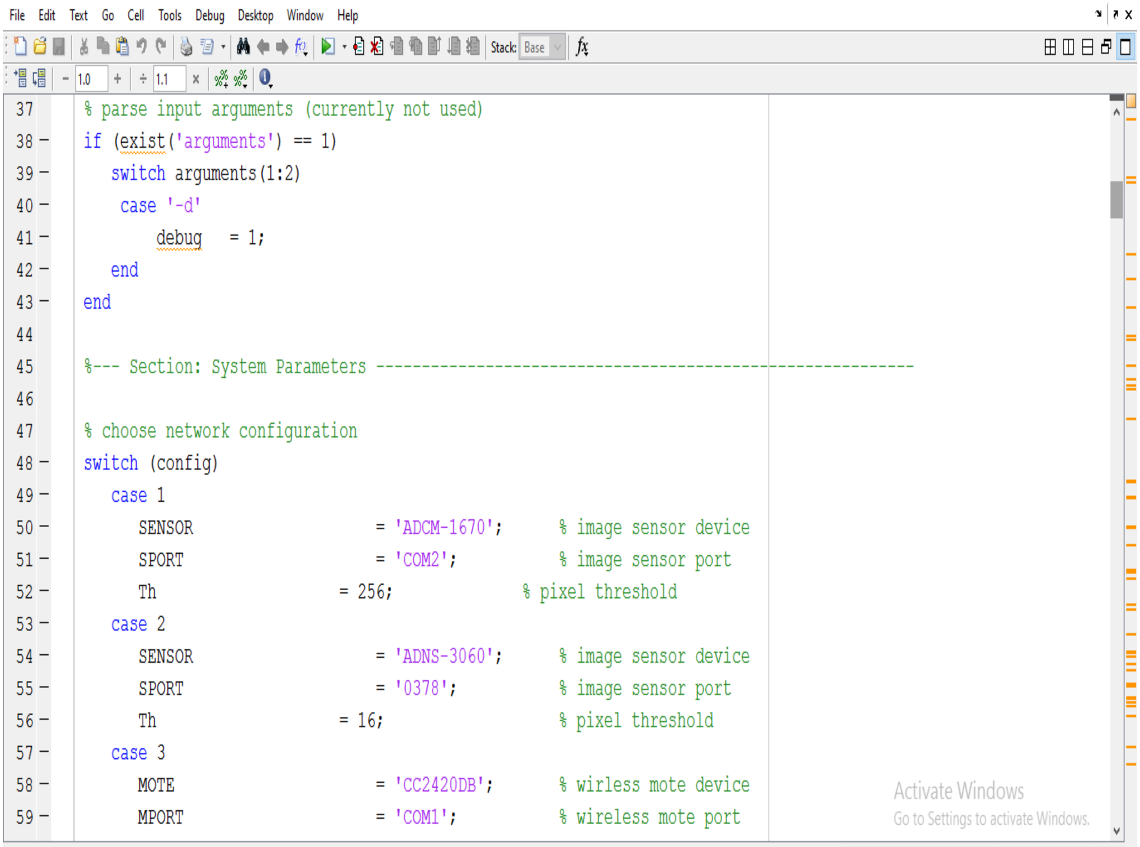
Zhang, H., Li, L., Yan, F., & Li, X. (2011). A load-balancing clustering algorithm of WSN for data gathering. In *Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011 2nd International Conference on* (pp. 915-918). IEEE.

Zhao, J., Wang, A., Green, M., & Ferrazza, F. (1998). 19.8% efficient "honeycomb" textured multicrystalline and 24.4% monocrystalline silicon solar cells. *Applied Physics Letters*, *73*(14), 1991-1993.

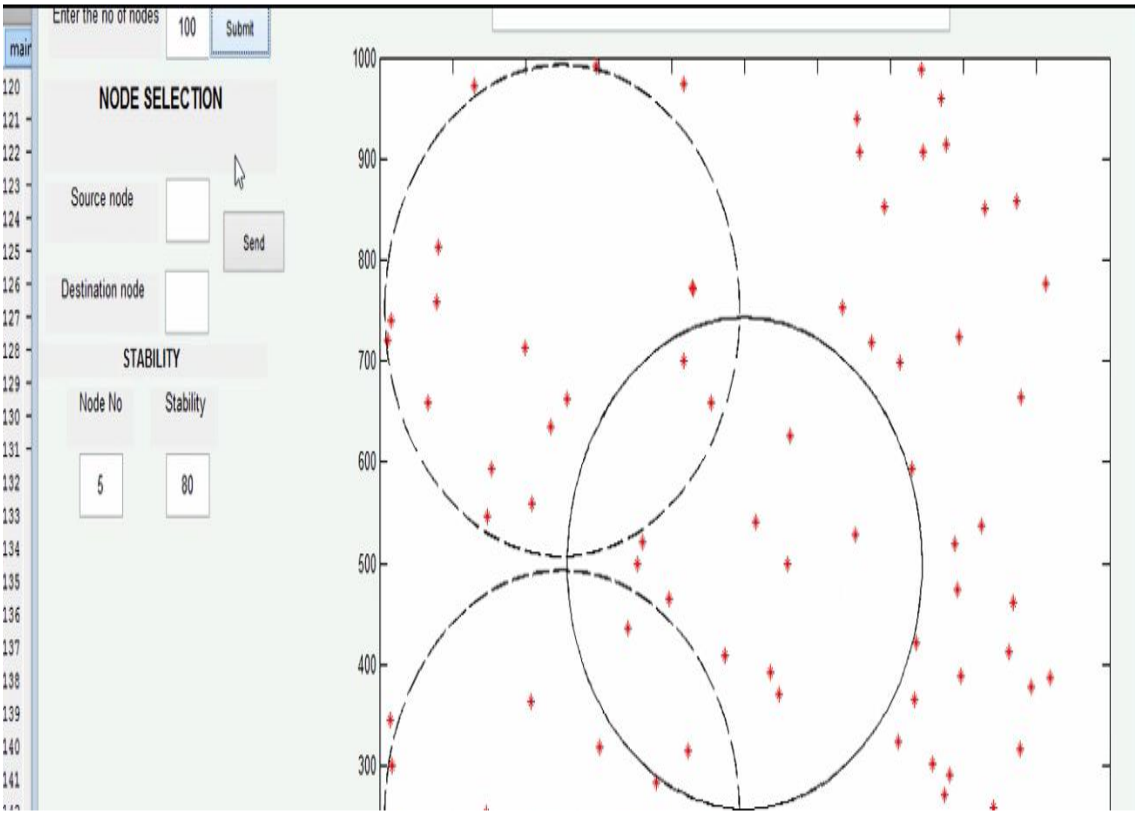
Zhao, Q., Nakamoto, Y., & Hussin, Z. (2013). Energy-efficient protocol for extending battery life in wireless sensor networks. In *Distributed Computing Systems Workshops (ICDCSW), 2013 IEEE 33rd International Conference on* (pp. 268-273). IEEE.

APPENDICES

APPENDIX 1
INTERFACE OF MATLABR2016A



APPENDIX 2
SIMULATION OF PROPOSED SCHEME



APPENDIX 3
MATLAB CODE OF IMPROVING THE BATTERY LIFE TIME IN WIRELESS SENSOR NETWORK

```
function tour2=ApplyAction(tour1,action)

   i=action(1);
   j=action(2);
   a=action(3);

   tour2=tour1;

   switch a
      case 1
         % Swap
         tour2([i j])=tour2([j i]);

      case 2
         % Reversion
         if i<j
            tour2(i:j)=tour2(j:-1:i);
         else
            tour2(j:i)=tour2(i:-1:j);
         end

      case 3
         % Insertion
         if i<j
            tour2=[tour2(1:i) tour2(j) tour2(i+1:j-1) tour2(j+1:end)];
         else
            tour2=[tour2(1:j-1) tour2(j+1:i) tour2(j) tour2(i+1:end)];
         end

   end

end

function ActionList=CreateTSPActionList(n)

   nSwap=n*(n-1)/2;
   nReversion=n*(n-1)/2;
   nInsertion=n*(n-1);

   nAction=nSwap+nReversion+nInsertion;

   ActionList=cell(nAction,1);

   c=0;

   for i=1:n-1
      for j=i+1:n
         c=c+1;
         ActionList{c}=[i j 1];
      end
```

```
    end

    for i=1:n-1
        for j=i+1:n
            if abs(i-j)>2
                c=c+1;
                ActionList{c}=[i j 2];
            end
        end
    end

    for i=1:n
        for j=1:n
            if all(j~=[i-1 i i+1 i+2])
                c=c+1;
                ActionList{c}=[i j 3];
            end
        end
    end

    ActionList=ActionList(1:c);

function h=PlotTour(model,tour)

    x1=model.x;
    y1=model.y;

    if ~isempty(tour)
        tour=[tour tour(1)];
    end
    x2=x1(tour);
    y2=y1(tour);

    h={0,0};

    h{1}=plot(x2,y2,'b');
    hold on;
    h{2}=plot(x1,y1,'r.','MarkerSize',15);

    axis equal;

end

%%
% *Simulate the basic processes of UWSN in Matlab...*


%% *Basic Operation*

%%
% *Remove specified figure:*
%%
% _Deletes the current figure or the specified figure(s)._

close all
```

```
%%
% *Remove items from workspace, freeing up systemmemory:*
%%
% _Removes all variables from the current workspace, releasing them from_
% _systemmemory._

clear all

%%
% *Clear Command Window:*
%%
% _Clears all input and output from the Command Window display, giving you_
% _a "cleanscreen"._

clc

%%
% *You can choose number of nodes:*
%%
% _The UWSN is built of "nodes" from a few to several hundreds or even_
% _thousands, where each node is connected to one (or sometimes several)_
% _sensors._

n =50;

%%
% *You can choose length of the network:*

w = 2*n;

%%
% *You can choose width of the network:*

h = 2*n;

%%
% *The net contains the database of the UWSN networks:*
%%
% _In the form of Matlab matrixes with the node's X,Y coordinates._

net = [1:n;rand([1,n])*w;rand([1,n])*h];
net1 = net;

%%
% *You can choose radio range in meters:*

R = n/1.5;


%% *Create figure of the "Base Network":*
%%
% _Loads a selected network model from the net and displays its layout_
% _into the figure._
```

```
figure(1),plot(net(2,:),net(3,:),'ko','MarkerSize',5,'MarkerFaceColor','k');
title('Base Network');
xlabel('\it x \rm [m] \rightarrow');
ylabel('\it y \rm [m] \rightarrow');
hold on;

for i = 1:numel(net(1,:))

    for j = 1:numel(net(1,:))
        X1 = net(2,i);
        Y1 = net(3,i);
        X2 = net(2,j);
        Y2 = net(3,j);
        xSide = abs(X2-X1);
        ySide = abs(Y2-Y1);
        d = sqrt(xSide^2+ySide^2);

        if (d<R)&&(i~=j)
            vertice1 = [X1,X2];
            vertice2 = [Y1,Y2];
            plot(vertice1,vertice2,'-.b','LineWidth',0.1);
            hold on;
        end

    end

end

v = net(1,:)';
s = int2str(v);
text(net(2,:)+1,net(3,:)+1,s,'FontSize',8,'VerticalAlignment','Baseline');


%% *Create figure of "Distance to Zero":*
%%
% _Optimization UWSNs localization using an algorithm that calculate the_
% _distance of each nodes to Zero._

for i = 1:numel(net(1,:))
    X1 = 0;
    Y1 = 0;
    X2 = net(2,i);
    Y2 = net(3,i);
    xSide = abs(X2-X1);
    ySide = abs(Y2-Y1);
    d(1,i) = sqrt(xSide^2+ySide^2);
end

net(4,:) = d(1,:);
[p,q] = sort(net(4,:));
net = net(:,q);
net(1,:) = 1:n;

figure(2),plot(net(2,:),net(3,:),'r.','MarkerSize',15);
title('Distance to Zero');
```

```
xlabel('\it x \rm [m] \rightarrow')
ylabel('\it y \rm [m] \rightarrow')
hold on;

for i = 1:numel(net(1,:))-1

    X1 = net(2,i);
    Y1 = net(3,i);
    X2 = net(2,i+1);
    Y2 = net(3,i+1);
    xSide = abs(X2-X1);
    ySide = abs(Y2-Y1);
    d = sqrt(xSide^2+ySide^2);

    vertice1 = [X1,X2];
    vertice2 = [Y1,Y2];
    plot(vertice1,vertice2,'b');
    hold on;

end

v = net(1,:)';
s = int2str(v);
text(net(2,:)+1,net(3,:)+1,s,'FontSize',8,'VerticalAlignment','Baseline');


%% *Create figure of "Distance to previous nodes":*
%%
% _Optimization UWSNs localization using an algorithm that calculate_
% _distance of each nodes to previous nodes._

X1 = 0;
Y1 = 0;
not = [];

for i = 1:numel(net(1,:))

    d = [];

    for j = 1:numel(net(1,:))

        X2 = net(2,j);
        Y2 = net(3,j);
        xSide = abs(X2-X1);
        ySide = abs(Y2-Y1);

        if(sqrt(xSide^2+ySide^2)~=0)
            d(1,j) = sqrt(xSide^2+ySide^2);
        end

    end

    min = d(1,1);
    minj = 1;
    for j = 1:numel(net(1,:))
```

```
        if(min>d(1,j))
            min = d(1,j);
            minj = j;
        end

    end

    not(:,i) = net(:,minj);
    net(2,minj) = inf;
    net(3,minj) = inf;
    X1 = not(2,i);
    Y1 = not(3,i);

end

not = [1:n;not(2,:);not(3,:)];

figure(3),plot(not(2,:),not(3,:),'r.','MarkerSize',15);
title('Distance to previous nodes');
xlabel('\it x \rm [m] \rightarrow')
ylabel('\it y \rm [m] \rightarrow')
hold on;

for i = 1:numel(not(1,:))-1

    X1 = not(2,i);
    Y1 = not(3,i);
    X2 = not(2,i+1);
    Y2 = not(3,i+1);
    xSide = abs(X2-X1);
    ySide = abs(Y2-Y1);
    d = sqrt(xSide^2+ySide^2);

    vertice1 = [X1,X2];
    vertice2 = [Y1,Y2];
    plot(vertice1,vertice2,'b');
    hold on;

end

v = not(1,:)';
s = int2str(v);
text(not(2,:)+1,not(3,:)+1,s,'FontSize',8,'VerticalAlignment','Baseline');


%% *Create figure of "Tabu Search":*
%%
% _Optimization UWSNs localization using Tabu Search (TS) algorithm._
%%
% *Inputs Definition:*

pos = net1';
pos(:,1) = [];
```

```
x = pos(:,1);
y = pos(:,2);

n = numel(x);
D = zeros(n,n);

for i = 1:n-1
    for j = i+1:n
        D(i,j) = norm([x(i) y(i)]-[x(j) y(j)]);
        D(j,i) = D(i,j);
    end
end

model.n = n;
model.x = x;
model.y = y;
model.D = D;

CostFunction = @(tour) TourLength(tour,model.D);     % cost function

nVar = model.n;                    % number of unknown variables
VarSize = [1 nVar];                 % unknown variables matrix size

%%
% *TS Parameters:*

MaxIt = n;

Actions = CreateTSPActionList(nVar);

nActions = numel(Actions);

TL0 = round(0.5*nActions);

%%
% *Initialization:*

TL = zeros(size(Actions));

Sol.Position = randperm(nVar);
Sol.Cost = CostFunction(Sol.Position);

BestSol = Sol;

BestCost = zeros(MaxIt,1);

%%
% *Solution Plot:*

OnlinePlot = true;

if OnlinePlot
    figure(4),hPlots = PlotTour(model,BestSol.Position);
    title('Tabu Search (TS)');
    pause(0.001);
```

```
end

%%
% *TS Main Loop:*

for it = 1:MaxIt

    BestNewSol.Position = [];
    BestNewSol.Cost = inf;

    BestAction = 0;

    for k = 1:nActions
        NewSol.Position = ApplyAction(Sol.Position,Actions{k});
        NewSol.Cost = CostFunction(NewSol.Position);

        % Aspiration Criterion
        if TL(k)>0  && NewSol.Cost<BestSol.Cost
            TL(k) = 0;
        end

        if TL(k)==0
            if NewSol.Cost<BestNewSol.Cost
                BestNewSol = NewSol;
                BestAction = k;
            end
        end
    end

    TL = max(TL-1,0);

    TL(BestAction) = TL0;

    Sol = BestNewSol;

    if Sol.Cost<BestSol.Cost
        BestSol = Sol;
    end

    if OnlinePlot
        UpdatePlot(hPlots,model,BestSol.Position);
        pause(0.001);
    end

    BestCost(it) = BestSol.Cost;
    disp(['Iteration ' num2str(it) ': Best Cost = ' num2str(BestCost(it))]);
end

%%
% *Results:*

net = BestSol.Position;

for i = 1:numel(net1(1,:))
```

```
    for j = 1:numel(net1(1,:))

        if net(1,i)==net1(1,j)
            net(2,i) = net1(2,j);
            net(3,i) = net1(3,j);
        end

    end

end


for i = 1:numel(net(1,:))
    X1 = 0;
    Y1 = 0;
    X2 = net(2,i);
    Y2 = net(3,i);
    xSide = abs(X2-X1);
    ySide = abs(Y2-Y1);
    d(1,i) = sqrt(xSide^2+ySide^2);
end

net(4,:) = d(1,:);

[p,q] = sort(net(4,:));

z = q(1);

net2 = circshift(net,[0,numel(net(1,:))+1-z]);
net = net2;
net(1,:) = 1:n;

figure(5),plot(net(2,:),net(3,:),'r.','MarkerSize',15);
title('Tabu Search (TS)');
xlabel('\it x \rm [m] \rightarrow')
ylabel('\it y \rm [m] \rightarrow')
hold on;

for i = 1:numel(net(1,:))-1

    X1 = net(2,i);
    Y1 = net(3,i);
    X2 = net(2,i+1);
    Y2 = net(3,i+1);
    xSide = abs(X2-X1);
    ySide = abs(Y2-Y1);
    d = sqrt(xSide^2+ySide^2);

    vertice1 = [X1,X2];
    vertice2 = [Y1,Y2];
    plot(vertice1,vertice2,'b');
    hold on;

end
```

```
v = net(1,:)';
s = int2str(v);
text(net(2,:)+1,net(3,:)+1,s,'FontSize',8,'VerticalAlignment','Baseline');


%% *The Degree of each node:*
%%
% _The degree of each node is the number of connection of each node by_
% _other nodes._

Degree=[];

for i = 1:numel(net(1,:))

    Degree(i)=0;

    for j = 1:numel(net(1,:))
        X1 = net(2,i);
        Y1 = net(3,i);
        X2 = net(2,j);
        Y2 = net(3,j);
        xSide = abs(X2-X1);
        ySide = abs(Y2-Y1);
        d = sqrt(xSide^2+ySide^2);

        if (d<R)&&(i~=j)
           Degree(i)= Degree(i)+1;
         end

    end

end

%% *Create figure of "Fuzzy Inference System":*
%%
% _Optimization UWSNs localization using Fuzzy Inference System (FIS)._

fisName = 'Optimization';
fisType = 'mamdani';
input = 2;
output = 1;
andMethod = 'min';
orMethod = 'max';
impMethod = 'min';
aggMethod = 'max';
defuzzMethod = 'centroid';

a = newfis(fisName,fisType,andMethod,orMethod,...
impMethod,aggMethod,defuzzMethod);

a = addvar(a,'input','Distance',[0 n]);
a = addmf(a,'input',1,'low','gaussmf',[n/5 0]);
a = addmf(a,'input',1,'medium','gaussmf',[n/5 n/2]);
a = addmf(a,'input',1,'high','gaussmf',[n/5 n]);
```

```
mD = max(Degree);
a = addvar(a,'input','Degree',[0  mD]);
a = addmf(a,'input',2,'low','trimf',[0   mD/6  mD/3]);
a = addmf(a,'input',2,'medium','trimf',[mD/3   mD/2  mD*2/3]);
a = addmf(a,'input',2,'high','trimf',[mD*2/3   mD*2.5/3  mD]);

a = addvar(a,'output','Priority',[0  n]);
a = addmf(a,'output',1,'First','gaussmf',[n/20   n/10]);
a = addmf(a,'output',1,'Second','gaussmf',[n/5  n/2]);
a = addmf(a,'output',1,'Third','gaussmf',[n/20   n-n/10]);

ruleList=[
 1 1 1 1 1
 1 2 1 1 1
 1 3 2 1 1
 2 1 1 1 1
 2 2 2 1 1
 2 3 3 1 1
 3 1 2 1 1
 3 2 3 1 1
 3 3 3 1 1];
a = addrule(a,ruleList);

writefis(a,'Optimization');

Inputs = [net(1,:)'  Degree(1,:)'];
Fuzzy  = readfis('Optimization');
Evaluation = evalfis(Inputs,Fuzzy);
Outputs = [net(1,:)'  net(2,:)'  net(3,:)'  Evaluation];
[p,q] = sort(Outputs(:,4));
Outputs = Outputs(q,:);
Outputs(:,1) = 1:n;
Outputs = Outputs';

figure(6),plot(Outputs(2,:),Outputs(3,:),'r.','MarkerSize',15);
title('Fuzzy   Inference System (FIS)');
xlabel('\it  x \rm [m] \rightarrow')
ylabel('\it  y \rm [m] \rightarrow')
hold on;

for i = 1:numel(net(1,:))

    for j = 1:numel(net(1,:))
        X1 = net(2,i);
        Y1  = net(3,i);
        X2 = net(2,j);
        Y2  = net(3,j);
        xSide  = abs(X2-X1);
        ySide = abs(Y2-Y1);
        d = sqrt(xSide^2+ySide^2);

        if (d<R)&&(i~=j)
            vertice1 = [X1,X2];
            vertice2 = [Y1,Y2];
            plot(vertice1,vertice2,'-.b','LineWidth',0.1);
```

```
            hold on;
        end

    end

end

v = Outputs(1,:)';
s = int2str(v);
text(Outputs(2,:)+1,Outputs(3,:)+1,s,'FontSize',8,'VerticalAlignment','Baseline');

figure(7),plotfis(Fuzzy);
title('Fuzzy  Inference System');

figure(8),plotmf(Fuzzy,'input',1);
title('Memberships  Functions of Input1');

figure(9),plotmf(Fuzzy,'input',2);
title('Memberships  Functions of Input2');

figure(10),plotmf(Fuzzy,'output',1);
title('Memberships  Functions of Output');

ruleview(Fuzzy);

surfview(Fuzzy);
%%
% *Simulate  the basic processes of UWSN in Matlab...*


%% *Basic Operation*

%%
% *Remove specified figure:*
%%
% _Deletes the current figure or the specified figure(s)._

close all

%%
% *Remove items from workspace, freeing up system memory:*
%%
% _Removes all variables from the current workspace, releasing them from_
% _system memory._

clear all

%%
% *Clear Command  Window:*
%%
% _Clears all input and output from the Command  Window display, giving you_
% _a "clean screen"._

clc
```

```
%%
% *You can choose number of nodes:*
%%
% _The UWSN is built of "nodes" from a few to several hundreds or even_
% _thousands, where each node is connected to one (or sometimes several)_
% _sensors._

n =50;

%%
% *You can choose length of the network:*

w = 2*n;

%%
% *You can choose width of the network:*

h = 2*n;

%%
% *The net contains the database of the UWSN networks:*
%%
% _In the form of Matlab matrixes with the node's X,Y coordinates._

net = [1:n;rand([1,n])*w;rand([1,n])*h];
net1 = net;

%%
% *You can choose radio range in meters:*

R = n/1.5;


%% *Create figure graphics object1:*
%%
% _Loads a selected network model from the net and displays its layout_
% _into the figure._

subplot(231),plot(net(2,:),net(3,:),'ko','MarkerSize',5,'MarkerFaceColor','k');
title('Base Network');
xlabel('\it x \rm [m] \rightarrow');
ylabel('\it y \rm [m] \rightarrow');
hold on;

for i = 1:numel(net(1,:))

    for j = 1:numel(net(1,:))
        X1 = net(2,i);
        Y1 = net(3,i);
        X2 = net(2,j);
        Y2 = net(3,j);
        xSide = abs(X2-X1);
        ySide = abs(Y2-Y1);
        d = sqrt(xSide^2+ySide^2);
```

```
        DD(:,i)=d;

        if (d<R)&&(i~=j)
            vertice1 = [X1,X2];
            vertice2 = [Y1,Y2];
            plot(vertice1,vertice2,'-.b','LineWidth',0.1);
            hold on;
        end

    end

end

v = net(1,:)';
s = int2str(v);
text(net(2,:)+1,net(3,:)+1,s,'FontSize',8,'VerticalAlignment','Baseline');

Cost1=sum(DD);

%% *Create figure graphics object2:*
%%
% _Optimization UWSNs localization using an algorithm that calculate the_
% _distance of each nodes to Zero._

for i = 1:numel(net(1,:))
    X1 = 0;
    Y1 = 0;
    X2 = net(2,i);
    Y2 = net(3,i);
    xSide = abs(X2-X1);
    ySide = abs(Y2-Y1);
    d(1,i) = sqrt(xSide^2+ySide^2);
end

net(4,:) = d(1,:);
[p,q] = sort(net(4,:));
net = net(:,q);
net(1,:) = 1:n;

subplot(232),plot(net(2,:),net(3,:),'r.','MarkerSize',15);
title('Distance to Zero');
xlabel('\it x \rm [m] \rightarrow')
ylabel('\it y \rm [m] \rightarrow')
hold on;

for i = 1:numel(net(1,:))-1

    X1 = net(2,i);
    Y1 = net(3,i);
    X2 = net(2,i+1);
    Y2 = net(3,i+1);
    xSide = abs(X2-X1);
    ySide = abs(Y2-Y1);
    d = sqrt(xSide^2+ySide^2);
```

```
    DD(:,i)=d;

    vertice1 = [X1,X2];
    vertice2 = [Y1,Y2];
    plot(vertice1,vertice2,'b');
    hold on;

end

v = net(1,:)';
s = int2str(v);
text(net(2,:)+1,net(3,:)+1,s,'FontSize',8,'VerticalAlignment','Baseline');

Cost2=sum(DD);

%% *Create figure graphics object3:*
%%
% _Optimization UWSNs localization using an algorithm that calculate_
% _distance of each nodes to previous nodes._

X1 = 0;
Y1 = 0;
not = [];

for i = 1:numel(net(1,:))

    d = [];

    for j = 1:numel(net(1,:))

        X2 = net(2,j);
        Y2 = net(3,j);
        xSide = abs(X2-X1);
        ySide = abs(Y2-Y1);

        if(sqrt(xSide^2+ySide^2)~=0)
            d(1,j) = sqrt(xSide^2+ySide^2);
        end

    end

    min = d(1,1);
    minj = 1;
    for j = 1:numel(net(1,:))

        if(min>d(1,j))
            min = d(1,j);
            minj = j;
        end

    end

    not(:,i) = net(:,minj);
    net(2,minj) = inf;
    net(3,minj) = inf;
```

```
    X1 = not(2,i);
    Y1 = not(3,i);

end

not = [1:n;not(2,:);not(3,:)];

subplot(233),plot(not(2,:),not(3,:),'r.','MarkerSize',15);
title('Distance to previous nodes');
xlabel('\it x \rm [m] \rightarrow')
ylabel('\it y \rm [m] \rightarrow')
hold on;

for i = 1:numel(not(1,:))-1

    X1 = not(2,i);
    Y1 = not(3,i);
    X2 = not(2,i+1);
    Y2 = not(3,i+1);
    xSide = abs(X2-X1);
    ySide = abs(Y2-Y1);
    d = sqrt(xSide^2+ySide^2);

    DD(:,i)=d;

    vertice1 = [X1,X2];
    vertice2 = [Y1,Y2];
    plot(vertice1,vertice2,'b');
    hold on;

end

v = not(1,:)';
s = int2str(v);
text(not(2,:)+1,not(3,:)+1,s,'FontSize',8,'VerticalAlignment','Baseline');

Cost3=sum(DD);

%% *Create figure graphics object4,5:*
%%
% _Optimization UWSNs localization using Tabu search (TS) algorithm._
%%
% *Inputs Definition:*

pos = net1';
pos(:,1) = [];

x = pos(:,1);
y = pos(:,2);

n = numel(x);
D = zeros(n,n);

for i = 1:n-1
    for j = i+1:n
```

```
        D(i,j) = norm([x(i)  y(i)]-[x(j)  y(j)]);
        D(j,i) = D(i,j);
    end
end

model.n = n;
model.x = x;
model.y = y;
model.D = D;

CostFunction = @(tour) TourLength(tour,model.D);    % cost function

nVar = model.n;                 % number of unknown variables
VarSize = [1 nVar];             % unknown variables matrix size

%%
% *TS Parameters:*

MaxIt = n;

Actions = CreateTSPActionList(nVar);

nActions = numel(Actions);

TL0 = round(0.5*nActions);

%%
% *Initialization:*

TL = zeros(size(Actions));

Sol.Position = randperm(nVar);
Sol.Cost = CostFunction(Sol.Position);

BestSol = Sol;

BestCost = zeros(MaxIt,1);

%%
% *Solution Plot:*

OnlinePlot = true;

if OnlinePlot
    subplot(234),hPlots = PlotTour(model,BestSol.Position);
    title('Tabu Search (TS)');
    pause(0.001);
end

%%
% *TS Main Loop:*

for it = 1:MaxIt

    BestNewSol.Position = [];
```

```
    BestNewSol.Cost = inf;

    BestAction = 0;

    for k = 1:nActions
        NewSol.Position = ApplyAction(Sol.Position,Actions{k});
        NewSol.Cost = CostFunction(NewSol.Position);

        % Aspiration Criterion
        if TL(k)>0  && NewSol.Cost<BestSol.Cost
            TL(k) = 0;
        end

        if TL(k)==0
            if NewSol.Cost<BestNewSol.Cost
                BestNewSol = NewSol;
                BestAction = k;
            end
        end
    end

    TL = max(TL-1,0);

    TL(BestAction) = TL0;

    Sol = BestNewSol;

    if Sol.Cost<BestSol.Cost
        BestSol = Sol;
    end

    if OnlinePlot
        UpdatePlot(hPlots,model,BestSol.Position);
        pause(0.001);
    end

    BestCost(it) = BestSol.Cost;
end

%%
% *Results:*

net = BestSol.Position;

for i = 1:numel(net1(1,:))

    for j = 1:numel(net1(1,:))

        if net(1,i)==net1(1,j)
            net(2,i) = net1(2,j);
            net(3,i) = net1(3,j);
        end

    end
```

```
end


for i = 1:numel(net(1,:))
    X1 = 0;
    Y1 = 0;
    X2 = net(2,i);
    Y2 = net(3,i);
    xSide = abs(X2-X1);
    ySide = abs(Y2-Y1);
    d(1,i) = sqrt(xSide^2+ySide^2);
end

net(4,:) = d(1,:);

[p,q] = sort(net(4,:));

z = q(1);

net2 = circshift(net,[0,numel(net(1,:))+1-z]);
net = net2;
net(1,:) = 1:n;

subplot(235),plot(net(2,:),net(3,:),'r.','MarkerSize',15);
title('Tabu Search (TS)');
xlabel('\it x \rm [m] \rightarrow')
ylabel('\it y \rm [m] \rightarrow')
Cost3=Cost3+100;
hold on;

for i = 1:numel(net(1,:))-1

    X1 = net(2,i);
    Y1 = net(3,i);
    X2 = net(2,i+1);
    Y2 = net(3,i+1);
    xSide = abs(X2-X1);
    ySide = abs(Y2-Y1);
    d = sqrt(xSide^2+ySide^2);

    vertice1 = [X1,X2];
    vertice2 = [Y1,Y2];
    plot(vertice1,vertice2,'b');
    hold on;

end

v = net(1,:)';
s = int2str(v);
text(net(2,:)+1,net(3,:)+1,s,'FontSize',8,'VerticalAlignment','Baseline');


%% *The Degree of each node:*
%%
% _The degree of each node is the number of connection of each node by_
```

```
% _other nodes._

Degree=[];

for i = 1:numel(net(1,:))

    Degree(i)=0;

    for j = 1:numel(net(1,:))
        X1 = net(2,i);
        Y1 = net(3,i);
        X2 = net(2,j);
        Y2 = net(3,j);
        xSide = abs(X2-X1);
        ySide = abs(Y2-Y1);
        d = sqrt(xSide^2+ySide^2);

        if (d<R)&&(i~=j)
            Degree(i)= Degree(i)+1;
        end

    end

end

%% *Create figure graphics object6:*
%%
% _Optimization UWSNs localization using Fuzzy Inference System (FIS)._

fisName = 'Optimization';
fisType = 'mamdani';
input = 2;
output = 1;
andMethod = 'min';
orMethod = 'max';
impMethod = 'min';
aggMethod = 'max';
defuzzMethod = 'centroid';

a = newfis(fisName,fisType,andMethod,orMethod,...
    impMethod,aggMethod,defuzzMethod);

a = addvar(a,'input','Distance',[0 n]);
a = addmf(a,'input',1,'low','gaussmf',[n/5 0]);
a = addmf(a,'input',1,'medium','gaussmf',[n/5 n/2]);
a = addmf(a,'input',1,'high','gaussmf',[n/5 n]);

mD = max(Degree);
a = addvar(a,'input','Degree',[0 mD]);
a = addmf(a,'input',2,'low','trimf',[0 mD/6 mD/3]);
a = addmf(a,'input',2,'medium','trimf',[mD/3 mD/2 mD*2/3]);
a = addmf(a,'input',2,'high','trimf',[mD*2/3 mD*2.5/3 mD]);

a = addvar(a,'output','Priority',[0 n]);
a = addmf(a,'output',1,'First','gaussmf',[n/20 n/10]);
```

```
a = addmf(a,'output',1,'Second','gaussmf',[n/5  n/2]);
a = addmf(a,'output',1,'Third','gaussmf',[n/20  n-n/10]);

ruleList=[
   1 1 1 1 1
   1 2 1 1 1
   1 3 2 1 1
   2 1 1 1 1
   2 2 2 1 1
   2 3 3 1 1
   3 1 2 1 1
   3 2 3 1 1
   3 3 3 1 1];
a = addrule(a,ruleList);

writefis(a,'Optimization');

Inputs = [net(1,:)'  Degree(1,:)'];
Fuzzy  = readfis('Optimization');
Evaluation = evalfis(Inputs,Fuzzy);
Outputs = [net(1,:)'  net(2,:)'  net(3,:)'  Evaluation];
[p,q] = sort(Outputs(:,4));
Outputs = Outputs(q,:);
Outputs(:,1) = 1:n;
Outputs = Outputs';

subplot(236),plot(Outputs(2,:),Outputs(3,:),'r.','MarkerSize',15);
title('Fuzzy  Inference System (FIS)');
xlabel('\it  x \rm [m] \rightarrow')
ylabel('\it  y \rm [m] \rightarrow')
hold on;

for i = 1:numel(net(1,:))

   for j = 1:numel(net(1,:))
      X1 = net(2,i);
      Y1  = net(3,i);
      X2 = net(2,j);
      Y2  = net(3,j);
      xSide  = abs(X2-X1);
      ySide = abs(Y2-Y1);
      d = sqrt(xSide^2+ySide^2);

      if (d<R)&&(i~=j)
         vertice1 = [X1,X2];
         vertice2 = [Y1,Y2];
         plot(vertice1,vertice2,'-.b','LineWidth',0.1);
         hold on;
      end

   end

end

v = Outputs(1,:)';
```

```
s = int2str(v);
text(Outputs(2,:)+1,Outputs(3,:)+1,s,'FontSize',8,'VerticalAlignment','Baseline');

figure
subplot(221),plotfis(Fuzzy);
title('Fuzzy Inference System');

subplot(222),plotmf(Fuzzy,'input',1);
title('Memberships Functions of Input1');

subplot(223),plotmf(Fuzzy,'input',2);
title('Memberships Functions of Input2');

subplot(224),plotmf(Fuzzy,'output',1);
title('Memberships Functions of Output');

ruleview(Fuzzy);

surfview(Fuzzy);

Cost4=BestCost';

%% *Create figure graphics object7:*
%%
% _Cost of each type of optimization methods._

TotalCost=[Cost1,Cost2,Cost3,Cost4];

disp(['Cost of Default Network:                        ' num2str(Cost1)]);
disp(['Cost of Distance of Each Nodes to Zero:           ' num2str(Cost2)]);
disp(['Cost of Distance of Each Nodes to Previous Nodes: ' num2str(Cost3)]);
disp(['Cost of Fuzzy Inference System:                  ' num2str(BestCost(n))]);

figure
plot(TotalCost,'-.r','LineWidth',2)

xlabel('Type of Optimization')
ylabel('Distance (m)')
title('Cost of Optimization Methods')

annotation('textbox',...
    [0.5 0.6 0.3 0.3],...
    'VerticalAlignment','middle',...
    'String',{...
    ['\fontsize{20}\oplus \fontsize{10}Cost of Default Network = ',num2str(Cost1)],...
    ['\fontsize{20}\oslash \fontsize{10}Cost of Distance of Each Nodes to Zero = ',num2str(Cost2)],...
    ['\fontsize{20}\otimes \fontsize{10}Cost of Distance of Each Nodes to Previous Nodes =
',num2str(Cost3)],...
    ['\fontsize{20}\copyright \fontsize{10}Cost of Fuzzy Inference System = ',num2str(BestCost(n))]},...
    'LineStyle',':',...
    'LineWidth',2,...
    'FitBoxToText','on',...
    'BackgroundColor',[1 1 1]);

text(1,Cost1,'\fontsize{20}\color{black}\oplus',...
```

```
    'HorizontalAlignment','center')

text(2,Cost2,'\fontsize{20}\color{black}\oslash',...
    'HorizontalAlignment','center')

text(3,Cost3,'\fontsize{20}\color{black}\otimes',...
    'HorizontalAlignment','center')

text(n+3,BestCost(n),'\fontsize{20}\color{black}\copyright',...
    'HorizontalAlignment','center')

function L=TourLength(tour,D)

    n=numel(tour);

    tour=[tour tour(1)];

    L=0;
    for i=1:n
        L=L+D(tour(i),tour(i+1));
    end

end

function UpdatePlot(h,model,tour)

    if ~isempty(tour)
        tour=[tour tour(1)];
    end
    x2=model.x(tour);
    y2=model.y(tour);

    set(h{1},'XDataSource','x2');
    set(h{1},'YDataSource','y2');

    refreshdata(h(1),'caller');

end
```