A PROBABILISTIC DYNAMIC ENERGY MODEL FOR AD-HOC WIRELESS SENSORS NETWORK WITH VARYING TOPOLOGY

A Dissertation Presented

by

Amal Al-Husseini

to

The Graduate School of Engineering

In partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

In

Mechanical Engineering

Northeastern University

Boston, Massachusetts

May 2012

Abstract

In this dissertation we investigate the behavior of Wireless Sensor Networks (WSNs) from the degree distribution and evolution perspective. In specific, we focus on implementation of a scale-free degree distribution topology for energy efficient WSNs.

WSNs is an emerging technology that finds its applications in different areas such as environment monitoring, agricultural crop monitoring, forest fire monitoring, and hazardous chemical monitoring in war zones. This technology allows us to collect data without human presence or intervention. Energy conservation/efficiency is one of the major issues in prolonging the active life WSNs. Recently, many energy aware and fault tolerant topology control algorithms have been presented, but there is dearth of research focused on energy conservation/efficiency of WSNs.

Therefore, we study energy efficiency and fault-tolerance in WSNs from the degree distribution and evolution perspective. Self-organization observed in natural and biological systems has been directly linked to their degree distribution. It is widely known that scale-free distribution bestows robustness, fault-tolerance, and access efficiency to system. Fascinated by these properties, we propose two complex network theoretic self-organizing models for adaptive WSNs. In particular, we focus on adopting the Barabasi and Albert scale-free model to fit into the constraints and limitations of WSNs. We developed simulation models to conduct numerical experiments and network analysis. The main objective of studying these models is to find ways to reducing energy usage of each node and balancing the overall network energy disrupted by faulty communication among nodes.

The first model constructs the wireless sensor network relative to the degree (connectivity) and remaining energy of every individual node. We observed that it results in a scale-free network structure which has good fault tolerance properties in face of random node failures. The second model considers additional constraints on the maximum degree of each node as well as the energy consumption relative to degree changes. This gives more realistic results from a dynamical network perspective. It results in balanced network-wide energy consumption. The results show that networks constructed using the proposed approach have good properties for different centrality measures.

The outcomes of the presented research are beneficial to building WSN control models with greater self-organization properties which leads to optimal energy consumption.

Acknowledgements

Working with my advisors Abe Zeid and Sagar Kamarthi during these years has been a rich experience. With their guidance, I learned how to think critically about the research of others, and leverage this work to generate new ideas. I eventually became a better thinker and writer. They have been a continuous source of inspiration through the long years, and I cannot thank them enough.

I am gracious for the help of my entire committee - Ginestra Bianconi, gave me great insight and allowed me to deepen my investigations. I would like to thank Laszlo Barabasi for accepting being an honorary member on my committee, while handling his own very full schedule. I attended many of his talks in conferences and weekly lectures, and these have helped me uncover and understand far more in the realm of networks. I have taken part in a research project with Laszlo Barabasi, which helped me prepare for work related to my dissertation.

Beyond my dissertation committee, I would like to thank my parents, sisters and brothers who have been the reason for my success. Without my parents support, I would not see myself where I am standing now. They both had very different rules of motivation (sometimes very unlike my own), which kept pushing me closer towards completing my PhD. I owe a special thank you to my brother Mahmood who was good for listening to ideas and offering advice in software related work.

There are certainly many more people to thank. Vanessa Salome who has been there for me with all the ups and downs, Allyson McCann who was calling me doctor before I even started. Hillary Hall who taught me how to relax and unwind through yoga, and Aysun Sunnetci and Mehmet Erkan Ceyhan who both taught me how to have fun in the most challenging part of my

work. I would like to thank Mark Umina who was always there to help whenever I had a technical difficulty.

Finally I would like to thank the people in my lab. Jessica Chin, Selen Onel, Ilke Boyaci, Emre Tuncel, Arjun Duvvuru, Srinivasan Radhakrishnan, Gulam Moeen Uddin and Tony Sultornsanee. It has been a sincere pleasure to work with a group as nice as them.

Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Overview

Recent studies and advances in modeling networks, especially dynamical networks, have been of great interest across all fields of science. One of the most known dynamical systems is the wireless sensor network (WSN), where some primary concerns related to the structural and evolutionary properties remain unexplored. The structure and evolution of WSNs are very important issues to explore in order to deeply understand the complex behavior of the system, and achieve desired global properties such as self-tolerance and self-organization as an emergent behavior. WSNs have many requirements to fulfill as they are vulnerable and limited in resources, as well as they are required to operate in an environment where unpredicted and new events can occur.

As a result, control techniques that depend on programming each part of the system cannot always be efficient as new situations constantly arise. In addition, traditional control techniques are synchronized in time due to the fact that sensors were expensive and few so the amount of data collected were limited. With the advances of technology, sensors are now available in low prices and large quantities and this has led to a major shift from the limited amount of data to the large data collection. This shift brought issues regarding switching from the time

synchronized behavior (time-driven) to a more asynchronous one (data-driven) as the former consumes more power. However, this has been an open research argument as there is no evidence yet that one approach is better than the other. Therefore having a system environment where both can exist together will draw on advantages of each and make the most use of it.

The extent to which WSNs reached in complexity places them at a point where traditional control techniques cannot effectively provide the desired scale of functionality. This increased complexity is a result of the aforementioned technological advances that enabled to use a large amount of sensors and therefore increased the level of interactions between them. What gives the complexity issue more significance is the complex and dynamic environment where the WSN is required to operate in. The dynamics of the environment contribute to the dynamics of nodes and links of the sensor network leading to undesirable changes in the underlying topological structure. Therefore, simplifying the system's complexity contributes to decreasing its robustness.

This raises the need to shift from traditional control models and design new ones that incorporate: (1) having a system environment where both data-driven and time-driven modes of operation can coexist; and (2) achieving global properties such as adaptation, self-organization and self-tolerance as an emergent behavior to be at high priority as oppose to programming each part of the system. These two considerations support the use of complex

systems framework in the control design to be more attractive as it involves the aforementioned properties.

Beyond the mathematical models of WSNs, complex systems provide a strong theoretical framework by which we study how relationships between parts of a system give rise to its collective behavior and how the systems interacts with its environment. This dissertation aims to develop a complex systems driven control model for WSNs. A simulation model is constructed in order to model the WSN and observe the emerging behavior of the system and understand its complexity. Simulation can reveal the existence of some quantifiable trends in time which characterizes the system's desired behavior. This will enable us to extract such important measures based on degree distributions and evolution, and use towards network control.

1.2 Complex vs. Complicated Systems

Engineers, for many years, have been dealing with complicated systems composed of separable components that have well-defined functional specifications for each part as well as pre-determined interactions between them. For example, the number of transistors in microprocessors over the years has ranged from 2,250 to 3 billion. With the scale observed as the only difference, the design of such systems can be characterized by having well-understood principles.

On the other hand, complex systems involve uncertainty. Therefore, they must be adaptable in order to function in unknown complex environments. A complex adaptive system is composed of components that interact with each other and adapt to unforeseen contingencies (1) (2). For example, a swarm of autonomous robots, that can be used for the exploration of Mars. There is no centralized control, and the robots collaborate to perform tasks in unknown environment. Also, they must have a high degree of redundancy in order to compensate for failure. The robots need cameras, sensors, actuators, and computational elements to monitor the environment and communicate with neighboring robots. They need fuel cells to keep sensors running. From local observations, each robot must make decisions and act accordingly. Therefore, the emergent behavior of the swarm, with the new configuration is the result of individual actions of robots.

Similarly, the system of WSNs is also a complex adaptive system. The work in this dissertation studies the energy behavior of WSNs from a complex network and evolution perspective with particulate emphasis on the degree distribution of the network. Graph theory is used to characterize the WSN by analyzing the network centrality measures of nodes and links. The Barabasi and Albert Scale-free model (3) is used as a way to generate the WSN. Two energy evolution models are developed in order to generate energy efficient and fault tolerant WSNs. Therefore, we use emergent engineering as a framework for our approach.

1.3 Insight/Summary on WSNs

Wireless sensor networks (WSNs) are complex systems composed of spatially distributed sensor devices. These devices represent the nodes in the network. The number of nodes in a WSN can vary from several hundred to thousands depending on the application type. Usually the network consists of sensors and a base station. Sensors perform three main tasks which are: sensing, storing and transmitting information. Sensor nodes are able to communicate with the environment by sensing various physical parameters and communicate with each other to send the sensed information to a base station. The sensors collect data from the environment for different purposes such as detecting certain quantities, environmental monitoring and in general, data collection (4).

WSNs are very useful because they are composed of autonomous devices that can self-organize to a chosen topology without a central control. As a result, nodes are required to operate in the environment according to a control algorithm. Sensor devices have many limitations such as power (limited battery life) and range (sensing/ transmission radius). Also, they are vulnerable devices since they can be easily damaged. Therefore, new approaches with means to increase the network's lifetime are very important for WSNs research. The framework we consider depends on network science and complex systems theory to produce energy efficient sensor networks (5).

We assume that there is no outside control over the network so the system must handle new events and function without explicitly programming them into the sensor nodes. Sensor devices are limited in range and power, and are susceptible to damage ability (6). The vulnerability of sensor devices is a result of their limited power and ability to damage from the environment. Weather related conditions can cause damage to sensor devices or it can simply shift them to different areas. Such events result in changes in the underlying topology structure.

From a hardware perspective, nodes in a WSN fail due to battery outage or damage from the environment. Failures as a result of battery outage is not random as a sensor node battery life is known and it's possible to calculate its usage based on functionality and time.

Failure of nodes as a result of environmental damage is random as it is hard to predict. From a software perspective, nodes in a WSN can fail due to events such as: overload or congestion on a node. This is related to the efficiency of routing and scheduling algorithms to handle unpredicted environmental conditions such as high data rate for a specific area where sensors are deployed. In this context, regardless of the reason, sensor nodes are not damaged but are malfunctioned for a period of time. Failure of sensor nodes can either stay on a local level or propagate to a global level affecting the whole network. The extent to which the failure level reaches can be controlled if the network topology was structured such that failure on a local level does not carry consequent failure events that affect the global network. Therefore, we emphasize how the topological structure has great influence on increasing fault tolerance and system robustness. The main question that arises here is: which type of failure has a higher

probability of occurrence in a sensor network? The answer to this question depends on the kind of environment a sensor network is required to operate in. If it is harsh, the probability of random failure is high, otherwise it is low.

The main goal of the sensor network is to give global information from local data sensed by distributed nodes, so algorithms intended for WSNs cannot be centralized but must be distributed. To achieve that goal the network must be capable of the following (6) (7) (8):

First: sensor nodes must be deployed in the region in such a manner that maximizes the coverage of the area. Since sensor devices are limited in their sensing range, the deployment strategy must account for these limitations by insuring that the environment can be completely sensed by distributed nodes. This is usually handled by the deployment and coverage algorithms.

Second: sensor nodes must form a network that enables reliable communication between the sensor nodes. Reliable communication is concerned with the network's efficiency to deliver the desired information in terms of time and service level. In view of that, the topological structure must account for the distributed sensor devices limitations of transmission range to facilitate communication between them. Also, routing and scheduling algorithms must be efficient in order to minimize delay, avoid congestion and maximize throughput. Poor routing and scheduling could result in energy waist as a result of high traffic or overload on a node where it can be caused by some data-driven events (7). As a result, for routing and scheduling algorithms to be more efficient, they must take into consideration both the time-driven and data-driven events.

Third: the network must be robust so that it can tolerate failure of some nodes. Robustness of the network ensures that damage to some nodes will not result in disconnecting the network by which can cause situations where there is missing information, delay in signal or congestion.

Fourth: the network must have an energy efficient configuration to enable an extended life. The network must operate in a way that minimizes the power consumption of different nodes since each node has a limited power due to its battery life.

The above requirements and constraints can be addressed by using proper algorithms to control the topology in order to avoid structures that has long range links and minimize power consumption. Also, routing and scheduling techniques must be implemented to avoid the occurrence of some stressful conditions such as congestion as they contribute to power consumption. Therefore, those problems can be carried out by control mechanisms that are able to cope with such data-driven events through reconfiguration in terms of topological structure and functionality in order to increase robustness. In this context, complex systems approaches enable us to understand the complex behavior of the system and develop new models from the view of degree distribution and evolution.

1.4 Structure and Dynamics of Complex Systems

The extent to which networked systems reached in complexity rendered away traditional techniques from the ability to provide the desired scale of functionality. The complexity matter is a result of the large size these systems have reached and the great amount of interactions

between their entities. These systems are required to operate in a complex environment and therefore simplifying their complexity makes them less capable to survive in such environment (2).

Complex systems approach shows how relationships between parts of a system give rise to the global behavior of the system and how the system interacts with its environment. Usually when people think of any problem they consider it to be the parts or entities of a system overlooking the relationships between them. Conversely, the theory of complex systems pays attention to the relationships between the system parts i.e. connections or information flow and relates problems to them (2). This is demonstrated through significant results on increasing networks robustness, fault tolerance, efficiency and adaption.

These studies were attracted by global behavior properties such as self-organization and self-tolerance of natural and biological networks and derived ideas to model engineered systems afterwards. Biological systems in particular have a very robust and adaptive behavior as a result of the collective behavior of autonomous agents (9).

Complex systems field have been applied to a wide variety of science including, sociology, biology, transportation, etc. It has been shown that many real complex networks such as the internet or www share common topological properties with natural complex network structures. Most widely known properties are the small-world phenomenon and the scale-free

feature. These systems are characterized by being decentralized, self- organized, adaptive and robust (10).

1.5 Motivation

There is a great interest in self-organized and self-adaptable networking structures like the WSN. And the discoveries of scale-free and small world features in many real networks have attracted all kinds of disciplines in studying the properties of various networks. WSNs have many limitations and vulnerability issues which makes the control over the network very challenging. The ability of the WSN to provide the desired functionality level depends heavily on its capability to handle unforeseen environmental changes. This involves adapting the topological structure and functionality based on time-driven and data-driven events in order to cope with changes in environmental demands. Therefore, to build robust and adaptive WSNs we suggest that control schemes must integrate properties such as self-organization, self-optimization and self-repair through the systems' emergent behavior. The theory of complex systems along with studies of biological systems shows a remarkably robust behavior. This is recognized to be the result of interactions between autonomous agents of the system that adapt and learn from the environment. From a dynamical network perspective such as the WSN where complex behavior and unpredicted conditions occur, complex systems considerations represent a promising path to acquire since they involve the aforementioned properties (11).

1.6 Open Issues on Wireless Sensor Networks

Current research on WSNs involves many problems and open issues that are technically challenging or more fundamentally unresolved. We present some of these issues that falls under different areas of study (12; 13; 14):

1. Routing and scheduling: it has been shown that a hierarchical structure for WSN is advantageous by having cluster-heads. Cluster-heads enable sensors within the cluster region to conserve power by sending data to the cluster-head and avoid direct interaction with the base station. Additionally, clustering techniques vary and finding the best design of clusters for WSNs is still an open issue. As a result, having cluster heads requires different approaches to problems and control schemes such as deployment.
2. Power control: one of the schemes is to allow sensors to switch states between sleeping, relaying and sensing. Usually this is done by using global information about the network which is not practical for all the nodes in the network to compute. Therefore, finding solutions based on local information is considered a better approach.
3. Mathematical models: current research relies on mathematical models that are not too ideal. These models assume that signal is uniformly transmitted in all directions which in reality is not true. Sensor transmissions are affected by factors such as antennas and the geographical layout where nodes are deployed.

4. Topology control: most work concentrates on energy saving and assumes that signal inference is reduced automatically. However, energy saving and signal inference are two different problems that should be considered separately.

5. Heterogeneous structure: there is little research done on heterogeneous networks even though it is a more practical design for WSNs. This is due to the fact that node transmission will vary due to many factors, even if sensors where all identical. Additionally, all the work done on heterogeneous networks assumes that nodes can be accurately located where this is a whole new challenging area.

6. Mobile node location: aims to find solutions to accurately locate nodes in the network especially when nodes are mobile.

7. Other issues that are more fundamental are concerned with bridging the gap between communication, computing and control that use different approaches and modeling techniques. Computation and gathering data are asynchronous since sensor nodes are distributed in a manner that make them operate at different levels of spatial and temporal conditions. Data-driven mode of operation implies that unless something interesting happens in the environment, data is not collected and sensors do not participate. Communication and control operate in a synchronized timely structure and are based on time-driven modeling. Due to the limited recourses of wireless sensors one can assume that data-driven modeling is advantageous over time-driven since it consumes less power. Switching to a data-driven approach will not just require new sampling mechanisms but most likely new hardware technology embedded in sensors to cope with the nature of the data-driven data collection. However, defending that one approach is better than the other

remains an unresolved argument. Therefore, having a system where both modes can exist together, will supplement the advantages of each, and this is still as open problem that needs an in depth study.

1.7 Purpose and Scope

We aim in this dissertation to explore and investigate complex systems considerations in general and their models in particular to control Wireless Sensor Networks (WSNs). The major focus is on structure and evolution where related issues for WSNs remain unexplored. WSNs exhibit many constraints, limitations and vulnerability issues that make the network control problem very challenging. We intend to use the underlying concepts of complex systems theory to optimize network control based on time-driven and data-driven events. We plan to evaluate the performance of the network based on the degree distribution and evolution. The ultimate goal is to use the results of this work in order to provide new approaches for WSN control to enable building robust and adaptive networks that can ultimately confirm greater levels of Quality of Service (QoS) expectations. We investigate the system using dynamic network analysis. The outcomes of this research will offer a new theoretical direction to develop control solutions based on complex systems considerations. Further, our findings will supplement existing knowledge regarding the issue whether to switch from a time-driven to a data-driven event sampling by facilitating a system environment where both can coexist [1]. The current focus is on WSNs in the context of highly dynamical environments where unpredictable and new conditions constantly arise.

The increased complexity level of many systems made the control over the whole network unmanageable through traditional approaches. The environment of many systems can be categorized as dynamic and unpredictable. This classifies it as complex, and for any system to survive in a complex environment it must be complex as well. Satisfying the complexity matters require the system to pick the right choices to handle unknown environmental conditions. Picking the right choice depends on the system's ability to adapt to changes by reconfiguration in terms of structure and functionality in order to meet the requirements of the environment and exit that configuration with the same speed as the changes in the environmental demand.

The study of natural and biological systems demonstrates properties of self-organization and re-configurability leading to surprisingly adaptive, efficient and robust networks. Traditional algorithmic approaches depend on explicitly programming each part of the system, by which limit the behavior to known conditions. Conversely, complex systems approaches involve learning, adapting and evolution in order to achieve re-configurability of the system as a result of the collective behavior of its entities. This makes the system capable of handling the upcoming unknown events, and it is the way to achieve robustness.

Chapter 2 Control of Wireless Sensor Networks

Topology control is one of the main issues in WSNs since sensor devices are characterized by having limited resources of power and range, in addition to the nature of their distributed deployment. Control schemes must be designed in order to keep the rate of power consumption low while maintaining the desired requirements. These requirements are usually concerned with achieving: good coverage, reliable communication, minimize delay and increase resilience to failures. Common topology control approaches are related to adjusting power transmission or dynamically scheduling the sensor's cycle (7) (8) (15) (16) (17) (18) (19) (20).

2.1 Network Design Considerations

Design assumptions vary due to the different requirements needed for different applications. Below we present an overview of the assumptions considered for current control models (6) (21).

2.1.1 Network Structure. The structure can be either flat i.e. non- hierarchical or hierarchical. The first assumes that sensors have the same functionality. The second is usually based on introducing some powerful sensors as cluster-heads where they have more role than other sensors. Cluster-head nodes are usually equipped with better hardware capabilities.

2.1.2 Deployment Strategy. Sensors are most commonly distributed in the region either by random, regular or planned deployment.

2.1.3 Sensor's Characteristics.

Sensor devices are characterized by: sensing range, transmission range, time synchronization, damage ability, and location information.

2.1.4 Design Objectives.

Design objectives vary due to the different requirements needed for different applications. Moreover, different priorities are also considered when the optimization becomes a tradeoff between different objectives. These objectives are related to the Quality of Service (QoS) expectations. The following are the most common considerations for the design of WSNs:

a. Maximize life time

b. Maximize coverage

c. Maintain connectivity

d. Efficiency

e. Balance the energy consumption of each sensor node

f. State update: the sensor network must provide updates and early warnings about sensors conditions. This will aid to prevent failure or to allow for recovery strategies that can be controlled from the outside such as the deployment of additional sensors.

2.2 Power Control Schemes

In this section we briefly discuss relevant state-of-the art topology control schemes and discuss their limitations. Common approaches are related to either adjusting power transmission or dynamically scheduling the sensor's cycle (22) , (17), (23), (24) (25) (26).

2.2.1 Power Transmission Schemes

These approaches concentrate on saving the sensor device energy by changing the power transmission level based on a synchronized timely fashion.

1. Common Power (COMPOW) (27): is a power control mechanism applied for sensor nodes with a uniform transmission power. This technique minimizes the power through the network's connectivity and shows good performance when nodes are uniformly deployed. However, having common powers can result in a single relatively isolated node and cause all nodes in the network to have large power levels. So non-uniform deployment of nodes can result in more energy consumption, as some nodes will experience larges transmission rages.
2. Cluster Power (CLUSTERPOW) (27): is an improved version of the above, where it combines power control and route protocols. However for a large network size, this method can cause situations of heavy overhead and high routing update frequency. Therefore, CLUSTERPOW is best used for applications where the network size is limited. Since the frequency of change in topology happens at a higher rate than the frequency of update, lowering the update frequency will not improve performance. Yet, it is

possible to use the approach of reducing the frequency of updates for a large network size if the changes in topology are known to be low.

3. Link Information No Topology (LINT) and Local Information Link-State Topology (LILT) (28): are based on k-neighbors graph, i.e. the graph where every node is connected to its k closest neighbors. These algorithms dynamically adjust a node power so that the degree falls within a specified range. This technique can reduce power consumption, but will not guarantee connectivity. The network connectivity is not certain as the number of neighbors cannot be accurately estimated and silent neighbors cannot be detected.
4. Communications-Based Train Control (CBTC) (29) (30): is an algorithm that uses direction, where accurate direction of information is needed. The authors claim that the protocol provides a minimal direction-based distributed rule, to ensure that the whole network is connected while keeping the power usage of each node as small as possible. CBTC guarantees connectivity and fault tolerance but at a higher cost since sensors that can provide accurate direction are equipped with multiple antennas that are expensive.
5. Neighborhood graphs (31): Power control algorithms based on neighborhood graphs such as Directed Relative Neighbor- hood Graph (DRNG) and Directed Local Minimum Spanning Tree (DLMST), guarantees connectivity, and low average power transmission. However, they require accurate location information.

Power control schemes discussed above, do not take into account the wasted energy during a sensor's idle mode, or consider the redundancy of sensed data to be a factor. Therefore, state scheduling comes into picture to deal with these two issues.

2.2.2 State Scheduling Mechanisms

Approaches based on scheduling states consider the wasted energy due to a sensor idle time and redundancy in data. As mentioned the design of sensor networks can be flat or hierarchical and different state scheduling algorithms are developed for each type. Below we briefly discuss the relevant schemes (23) (16) (17) (15) (32) (33) (34):

2.2.2.1 Non-hierarchical networks

1. Randomized Independent Scheduling (RIS) (16): presents an algorithm which plans the scheduling process at the beginning of each cycle. With a given probability, sensors independently decide on whether to become active or go to sleep. RIS presents a control protocol that heavily depends on time synchronization which can affect the performance and therefore degrade the overall quality of desired results.
2. Maximization of Sensor Network Life (MSNL) (35): formulates the state scheduling problem as a network lifetime maximization problem with constraints on battery lifetime and sensing coverage. However, MSNL requires accurate location information, and neighborhood nodes will enter into the sleep mode simultaneously.
3. Lightweight Deployment-Aware (LDAS) (36): plans the sensors scheduling process based on partial redundancy. The algorithm guarantees coverage without any knowledge of location information. However, LDAS performance is limited to applications with uniform deployment.
4. Adaptive Self-Configuring Sensor Networks Topologies (ASCENT) (37): is an algorithm that creates the scheduling cycle by using a threshold value. That value sets the active sensor

and depending on the condition, a help message is sent to turn on more neighbors. The neighbors receiving the help message can be activated to transmit data. However, ASCENT does not guarantee connectivity, or balance the energy consumption.

5. Probing Environment and Adaptive Sensing (PEAS) (38): is used for applications where there is high density of sensors operating in harsh environmental conditions. Each sensor can switch between three modes of operation: SLEEPING, PROBING, or WORKING. Each node is given a timer and is initially set to the SLEEPING mode. When the timer expires the node enters into the PROBING mode. A PROBING node sends a message which can result in two conditions: first, a reply is received and the node reenters into SLEEPING mode; second, the node is set to the WORKING mode and continues in that mode until its energy depletes. PEAS advantage resides in its ability to ensure asymptotic connectivity. This implies that when the size of the sensor field approaches infinity, the probability that the WORKING sensors are connected approaches 1. The problem with PEAS is its inability to balance the energy consumption.

6. Probing Environment and Collaborating Adaptive Sleeping (PECAS) (38): presents an algorithm that is an extension of PEAS. The difference between them is that in PECAS when a node enters the WORKING mode, it doesn't necessary stay there till its energy depletes. A WORKING node informs its remaining working time in the reply to a PROBE message and eventually goes back to the SLEEPING mode when it expires. PECAS provides overall better QoS results compared to PEAS but results in less energy saving outcomes.

7. Coverage Configuration Protocol (CCP) (39) (40): the algorithm demonstrates the relationship between coverage and connectivity under two cases. The protocol formulates

the problem as to maximize the number of sleeping nodes, while maintaining both K-coverage and K-connectivity. The first case considers a transmission range value that is double the sensing range which demonstrates good results. In the second case the transmission range value falls below sensing range by a factor of two, and the network connectivity cannot be guaranteed. The authors combined CCP with SPAN where the network's original connectivity remains unchanged, yet it cannot configure the network to the designated connectivity.

2.2.2.2 Hierarchical networks

1. Low-Energy Adaptive Clustering Hierarchy (LEACH) (41): this control protocol depends on strict time synchronization. The distribution of cluster-heads under LEACH is not guaranteed to be a uniform. LEACH uses a single hop communication between cluster-heads while aggregation of nodes by which limits the scale of networks. It was shown that the time period of the setup phase is non-deterministic. Under some events such as collisions, longer durations for the setup time occurs leading to interruption in sensing services. Depending on the density of sensors, such cases might lead the protocol to become unstable during the set-up phase. As a result, LEACH may not provide a promising performance for applications where sensor networks require deployment in large regions.
2. Deng et al. proposed a Linear Distance-based Scheduling (LDS) (42): the algorithm considers only the sleep scheduling within clusters and assumes that the cluster architecture is available through some technique. LDS performance showed inability to provide uniform energy consumption, so Deng et al. proposed an enhanced LDS. The LDS scheme only

considers static clustering, basically implying that the selection of cluster-heads cannot change. The performance of LDS revealed the possibility of producing events of unbalanced sensing coverage which can lead to consequence undesired outcomes. This is explained as the further away a node is located from the cluster-head the less sensing coverage it has and vice versa. The consequence of such case can result in a non-uniform energy consumption within a cluster by which can lead to an unbalanced lifetime for nodes within the cluster-head region.

3. Geographical Adaptive Fidelity (GAF) (43): presents an algorithm where clustering is based on geographical locations. The algorithm creates the clustering architecture by dividing the sensing area into virtual square grids. Based on each node's location information, the matching grids are configured to position different node. Nodes in adjacent grids are considered to reside within the transmission range of each other. The ability to control the network under GAF depends on the sensor devices hardware capabilities as accurate geographic locations are required. GAF assumes that if nodes are close enough, a direct communication can be established. However, in real situations, this assumption is not valid due to the fact that close distance between any two nodes does not necessarily imply a one-hop i.e. direct communication between them. Moreover, the GAF cannot guarantee a balanced energy consumption result between sensor nodes.

4. Topology discovery (TopDisc) (44): Deb et al. proposed this algorithm based on the minimum dominating set (MDS) in the field of graph theory. Usually, topology control based on clustering techniques favors the methodology of choosing a network communication backbone with the fewest possible links with regards to reducing the costs of control and

maintenance. MDS and connected MDS (MCDS) present the best explanation for cluster in graph theory. It was shown that when a network is supplied with infinite energy, the energy consumption of MDS is the least. However, such clustering schemes offer a tradeoff between low energy consumption and computational complexity. MDS is NP-hard and MCDS is NP-complete, thus offering only an approximate solution. For high density deployment sensor networks, TopDisc can create hierarchical networks by quick formation of clusters and can arrange them in a tree structure. However, the resulting networks will exhibit many limitations as they lack flexibility and cost too much for repeated execution. Additionally, TopDisc do not take into consideration the remaining energy of nodes.

5. Hybrid Energy-Efficient Distributed Clustering (HEED) (45): introduced by Younis and Fahmy where they present the clustering algorithm by constant iteration without considering the scale of the network. The broad reach of HEED considers the lifetime, extendibility and load balance. Implementing HEED does not require any specifications for node deployment or capability, and does not require time synchronization. The weakness of HEED is related to its heavy dependency on non-synchronization which was acknowledged to impact the clustering quality. Moreover, the clustering process takes time to terminate introducing an extra factor that contributes to increase in energy consumption.

Chapter 3 The Science of Networks

3.1 The New Science of Networks

Network thinking means, focusing on relationships between parts rather than the parts themselves. This field of research shows how the complexity of a system largely arises from the complexity of the interactions among its parts.

Recently, there has been great interest in the structure of large complex networks across many fields of science. Certainly, many systems such as the internet and the airline rout maps are clear examples of the many natural, technological and cultural phenomena that can usefully be described as networks. The brain is biological network of neurons (nodes) and synapses (links). Social communities represent natural networks of people (nodes) and their relationships (links). For example, one can construct his/her Facebook social network as shown in figure 3.1. Another interesting way to visualize that same network is shown in figure 3.2.

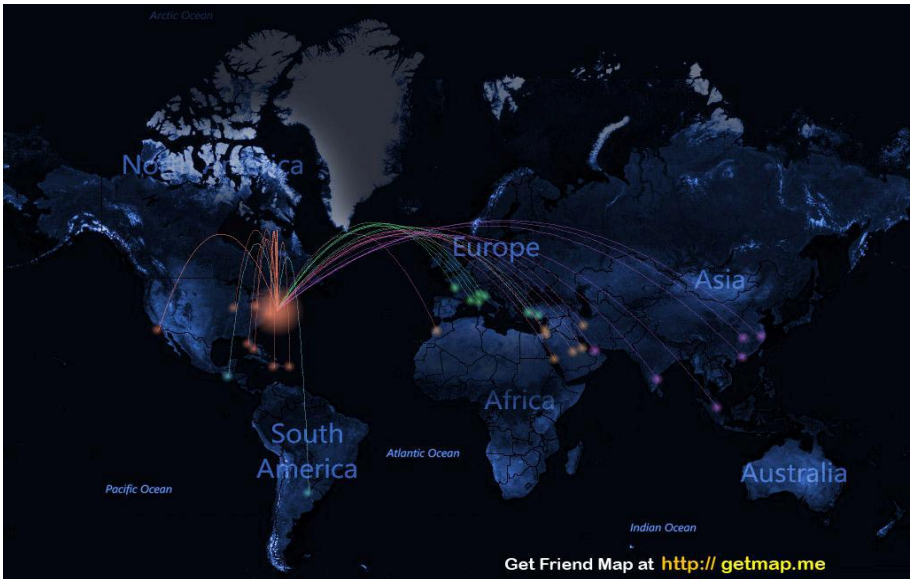

Figure 3. 1 The Facebook social geographical network

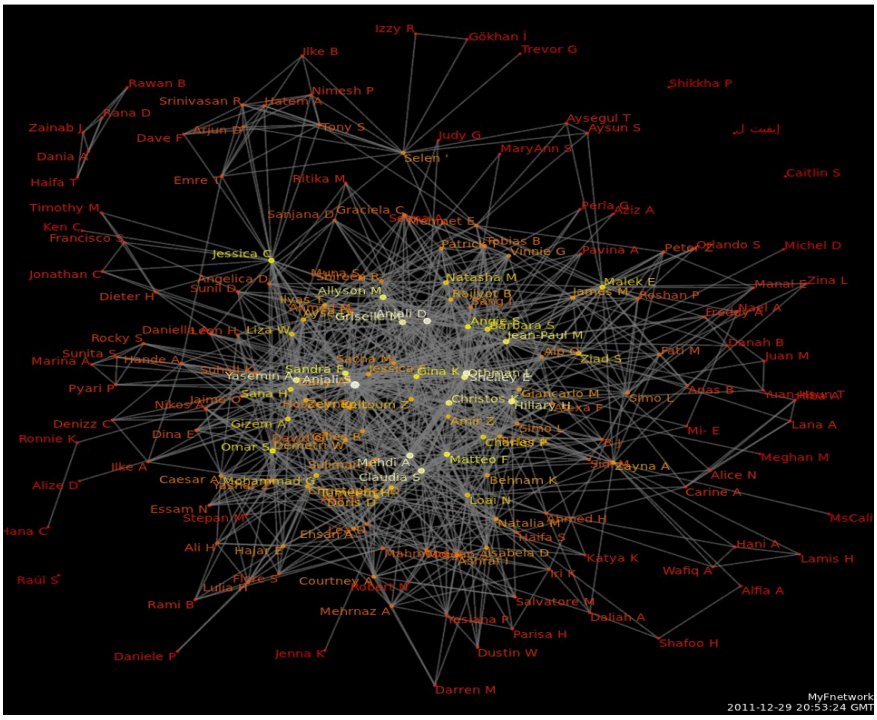

Figure 3. 2 The Facebook social network of nodes and links

Recently, there has been great interest in finding global principles which governs different kinds of networks. The reason behind this surge in interest is the influence of two famous papers published in the late 90s. The first published in Nature: “Collective Dynamics of Small World Networks” by Watts and Strogatz. The second one published in Science: “Emergence of Scaling in Random Networks” by Barabasi and Albert. More examples of small-world and scale-free properties in the real world network are constantly being discovered. Noticeably, natural, social and technological evolution favors such structure. This remarkable finding has been explained to be the reason behind the need to satisfy two demands: fast information transfer within a system, and the high cost to create and maintain reliable long distance connections.

Fascinated by the discoveries of global principles such as degree distribution, our contribution is to use these finding in order to build models for WSNs in the context of highly dynamic environment. The goal is to optimize energy consumption of each node, balance the energy consumption of the whole network and increase network redundancy through a scale-free degree distribution.

In short, scientific understanding of networked systems could offer great value not only to our understanding of natural and social systems, but also to our capability to engineer and effectively use complex networks, ranging from better web search and internet routing to controlling the reach of disease, the effectiveness of structured crime, and in our case the control of WSNs.

3.2 Small-World Networks

Watts and Strogats were the first to define the small-world network mathematically (46), and analyze the different kind of networks which have this property. The *average path length* was used as a measure to determine the extent or degree of a network's small-world property.

3.3 Scale-free Networks

Scale-free (SF)networks have a power-low degree distribution. They are known to have a "characteristic tale". The SF degree distribution shows the existence of few large degree nodes connecting many low degree nodes. The small-world (SW) phenomenon for any kind of network shows that two nodes chosen at random are connected by an average of six degrees of freedom. This illustrates the ability of the network to have on average short link lengths. These two features revealed properties of robustness, fault tolerance, efficiency and adaptation in the underlying networks. For example: the World Wide Web (WWW) is composed of Web pages as nodes and hyperlinks as links in the underlying network structure. The structure has a property where most of the nodes have a low degree, and very few nodes have high degree values which are known to be the 'hubs' in the network. This particular structure enables search engines such as Google to provide fast and reliable Web search capabilities. Google was the first to develop the idea of "PageRank" which is only feasible due to the aforementioned network structure, most widely known as scale-free degree distribution. Therefore, complex systems with a network structure of scale-free degree distributions facilitate the right environment for fast, reliable and efficient information structure of the resulting network.

3.4 Network Resilience

Networks structures having scale-free degree distributions have a very important property, which is its resilience to random deletion of nodes. This implies that when a large block of nodes along with their links are destroyed, basic network properties of heterogeneous degree distribution, short average path length and good clustering remain unchanged. This makes sense since most of the nodes in the network are of small degree so the random probability to hit a low degree node is higher than hitting a high degree node. However, if an attack hits a hub, devastating failure can occur.

3.5 The Impact of Complex Systems

We are interested in studying the structure and evolution properties of WSNs in order to find a node-link structure that would optimize energy performance and consumption in WSNs. The energy performance requirements of WSNs concentrate on minimizing the energy consumption per node and balance the energy consumption of all nodes in the network in order to extend the network lifetime. All this must be achieved without degrading the overall Quality of Service (QoS) expectations such as efficiency in terms of service level and time, fault tolerance and reliability. There is always a tradeoff between satisfying all the QoS requirements which usually are considered an application specific optimization problem. Therefore, by taking our framework of study, we hope to offer results which can be applied to control schemes in order to achieve more self-organization results which lead to more self-optimization properties of the network.

Traditional design methods of technology have been top-down approaches. They are based on systems engineering, decomposing the design into functional components, charactering the intended relationships among them, and verifying that the system is built and operated as intended. The performance is predictable and is capable to carry out predefined tasks in a bounded environment.

There is mounting evidence, however, that traditional approaches for systems engineering are reaching a point where they cannot effectively provide the desired level of functionality. Current methods are failing because of the large number of parts (components) and extensive software for each component in a system. Therefore, each of the designed components is a part of a larger system. This results in large number of interactions and changes that can lead to unexpected consequences. Therefore, the complexity of systems keeps on increasing. As a result, new approaches for engineering projects today involve the design of systems as a network of parts. Interactions between the parts of a system are nonlinear and can give rise to unpredictable responses. Networked approaches for systems engineering will allow having structures that can adapt in real-time to deal with unforeseen changes. In short, they enable adaptive and complex systems.

A case in point would be the WSN, a technology that is being considered for different applications such as environmental and health monitoring that enables continuous monitoring and data gathering. A system of wireless sensors must have the flexibility to operate in an unknown environment, collaborate in performing tasks, and adapt to unforeseen situations. The sensors are hardware devices designed to meet the requirements for different tasks and

conditions. There is no centralized control, so the system must have high degree of redundancy for fault-tolerance, and to compensate for failure of some of the individual sensors.

Emergent engineering is a convergence of complexity theory and science with such disciplines as communications, sensor technology, mechanics, computational intelligence, and control theory. The goal of emergent engineering is to produce robust complex systems, operating in uncertain environments, and capable of adaptation and change. From this perspective, this provides a good framework towards new solutions for WSN control.

3.6 Network Science Applications

Scientists are finding more and more examples of small-world and scale-free networks in the real world. Noticeably, natural, social and technological evolution favored such structure. This remarkable finding was explained to be the reason satisfying the need for fast information transfer within a system and the high cost to create and maintain reliable long distance connections.

Network science can be applied to many real-world networks. The best example to understand the underlying properties of small-world and scale-free networks is the brain. The brain have been mapped by neuroscientists and several groups and found evidence of small-world and scale-free network properties. The brain can be modeled as a network of nodes and links. Nodes correspond to neurons, and links represents synapses. Knowing that many neurons die and the brain doesn't get affected, resilience may be one of the reasons behind this. However, the hubs represent a different story. If a stroke hits a hub, devastating failure can occur.

Another reason is optimizing brain processing between local vs. global information. Scale-free networks have shown efficiency for information transfer (such as amount, time and energy) in its underlying network structure. On the contrary, if all nodes (corresponding to neurons) were fully connected, the brain would be using a massive amount of energy to send signals over the huge number of connections. Also, the small-world and scale-free network structures enable synchronization, where in the brain groups of neurons fire simultaneously. And this is important for efficient information transfer.

From this perspective, our contribution is to build a model derived from the universal principles such as degree distribution to optimize energy in WSNs. Initially the models are used in the context of random deployment of sensor nodes in a field were battery replacement is not practical. Therefore, the impact here is to produce energy efficient models for WSNs and measure the performance of the resulting network from the degree distribution and evolution perspective.

In the context of WSNs operating in harsh environments, the above mentioned properties represent desirable system properties which inspire us to use such ideas towards network control.

Although WSNs are complex systems, they have many constraints that contribute to the behavior of the system. The theory and models of complex systems give us a new insight on studying evolutionary algorithms while considering the constraints of the sensors network.

3.7 Models of Complex Systems

3.7.1 Random Network Erdös–Rényi (ER) Model

The random network model starts with N number of nodes and connects each pair of nodes with probability P. This process results in a network having a number of links (K) approximately equal to $pN(N-1)/2$. The node degrees follow a Poisson distribution which implies that a node degree is close to the average degree. The function clustering coefficient distribution ($C(K)$) is approximated by a horizontal line. The mean path length <L> is proportional to log N [12-18].

Characteristics of ER: the short mean path length shows the small-world property of random networks. The network shows a robust behavior to intentional attacks since all nodes of the network are equally important. However, the overall low clustering coefficient contributes to the network vulnerability from random failure.

3.7.2 The Scale-free Network Model

To explain the foundation of power-law degree distribution, Barabási and Albert (BA) [8, 9] proposed a scale-free (SF) model, by which a complex network can be generated with a power-law degree distribution in the form of $p(K) \sim k^{-3}$. The resulting scale-free network demonstrated great levels of robustness and efficiency even with a high failure rate. The BA model is based on the two rules of growth and preferential attachment, where the former is

responsible for the addition of new nodes and the latter governs the way new nodes attach to existing ones [12-25].

Characteristics of SF: The model shows a Power-law degree distribution with characteristics of random error tolerance and small-world property leading to robust and efficient networks. On the negative side, it characterizes a high failure vulnerability problem of intentional attacks.

3.8 Wireless Sensor Networks as Large Scale Networks

Prior to the recent rise of network science as a new field of study, different researchers were interested in studying network structures. For example, mathematicians studied abstract network structures in the field of graph theory. Psychologists such as Milgram (47) were interested in understanding human behavior in network structures. Airline executives studied networks representing airline route maps in order to find a node-link structure that would optimize profits given certain constraints.

Similar to the work of Airline executives, we are interested in studying the WSN in order to find a node-link structure that would optimize energy performance. The energy performance requirements of WSNs concentrates on minimizing the energy consumption per node and balance the energy consumption of all nodes in the network in order to extend the network lifetime. All this must be achieved without degrading the overall Quality of Service (QoS) expectations such as efficiency in terms of service level and time, fault tolerance and reliability. There is always a tradeoff between satisfying all the QoS requirements which usually is considered an application specific optimization problem. Therefore, by taking our framework of

study, we hope to offer results which can be applied to control schemes in order to achieve more self-organization results which lead to more self-optimization properties of the network.

Traditional design methods of technology have been top-down approaches. They are based on systems engineering, decomposing the design into functional components, charactering the intended relationships among them, and verifying that the system is built and operated as intended. The performance is predictable and capable to carry out predefined tasks in a bounded environment.

There is mounting evidence, however, that traditional approaches for systems engineering are reaching a point where they cannot effectively provide the desired level of functionality. Current methods are failing because of the large number of parts (components) and extensive software for each component in a system. Therefore, each of the designed components is a part of a larger system. This results in large number of interactions and changes that can lead to unexpected consequences. Therefore, the complexity of systems keeps on increasing. As a result, new approaches for engineering projects today involve the design of systems as a network of parts. Interactions between the parts of a system are nonlinear and can give rise to unpredictable responses. Networked approaches for systems engineering will allow having structures that can adapt in real-time to deal with unforeseen changes. In short, they enable adaptive and complex systems.

A case in point would be the WSN, a technology that is being considered for different applications such as environmental and health monitoring that enables continues monitoring and data gathering. A system of wireless sensors must have the flexibility to operate in an

unknown environment, collaborate in performing tasks, and adapt to unforeseen situations. The sensors are hardware devices designed to meet the requirements for different tasks and environments. There is no centralized control, so the system must have high degree of redundancy for fault-tolerance, and to compensate for failure of some of the individual sensors.

Emergent engineering is a convergence of complexity theory and science with such disciplines as communications, sensor technology, mechanics, computational intelligence, and control theory. The goal of emergent engineering is to produce robust complex systems, operating in uncertain environments, and capable of adaptation and change. From this perspective, this provides a good framework towards new solutions for WSN control.

Chapter 4| Modeling Wireless Sensors as Complex Systems

In this section we provide a review and critique of relevant work that has been done on WSNs from the theoretical view of complex systems. Most algorithms developed for WSNs based on the evolution and structure principles of complex systems concentrate on achieving scale-free networks. The major focus for this choice was to allow the network to increase its robustness and power efficiency, and minimize delay. These algorithms optimize some parameters and do not study the performance effect on other important parameter measures based on events of overload or congestion.

More importantly, most of them do not take the dynamics of the network into the time evolution of the algorithm such as changes in energy that affect the transmission and sensing range which can result from time-driven and data-driven events. For example, many algorithms that consider achieving a SF network topology rely heavily on the time-driven sampling without regards of data-driven sampling. Since WSNs are limited in energy and sometimes operate in harsh environments, control solutions that allow the network to operate in a synchronized and asynchronous fashion can offer increased QoS results.

The following sections, describe relevant control schemes for WSNs based on the evolutionary properties of complex systems theory. We briefly describe each method and provide our critique.

4.1 "Scale-free topology for large-scale wireless sensor networks" (48)

General description: the authors present a fault tolerant topology control algorithm (AWSF) for large-scale WSNs. The resulting network under this algorithm showed characteristics of a SF topology that is strongly connected and bi-directional. The performance was evaluated through simulation which showed qualities of: time efficiency, robustness and small average degree.

Drawbacks of AWSF: The algorithm does not account for the dynamical behavior of sensor devices. In particular, the affect of energy and data-driven events on the sensor node capabilities and vulnerability was not considered. The issues we consider that degrade the overall qualities of AWSF are listed below:

1. A static and equal uniform transmission range "r" was given for all sensor nodes. In real networks, this assumption is not accurate even if all sensors were given the same initial capabilities. Part of this is due to the different locations that sensor nodes occupy by which the geographic layout can interfere with the signal strength. The other part involves the energy depletion affect on decreasing the transmission range strength. Therefore, a better design would be to vary "r" by representing the range as a function that depends on the energy use and a random variable to account for geographical limitations.
2. The algorithm uses random weight sequence to generate the power-law distribution, where nodes with higher weights have a higher probability to connect to others. For a large network size this method can produce a SF topology with high degree of nodes

having higher weights. It is obvious that such topology will increase the risk of consuming energy for the hot connected nodes. In order to deal with this issue, the authors suggest that some redundancy mechanism can be used to backup such nodes. However, this cannot just simply be considered a solution due to: first, the fact that it must be tested before hand and prove it can work; second, the algorithm doesn't show how to reconnect to a redundant node in the case of the main node dies; third, maybe one redundant node is not sufficient due to the speed of death; fourth, the redundant node needs to know information about the nodes that it needs to connect with. This step might involve the update frequency and will most likely require time and increase in power consumption. Therefore, the possibility to employ the redundancy mechanism depends on factors such as the frequency of change in topology and the frequency of update.

3. The algorithm produces a scale-free topology under the constraint that "r" must be increased in some situations. This assumption was necessary to keep the number of edges low and maintain a low average path length with the cost of consuming more power. They compensate shorter average path length with more energy consumption for the longer range links. A better design would be to enable the algorithm self-optimize this according to desired conditions so that the edge length only increases when necessary, not for the purpose of producing a SF degree distribution.
4. The robustness of SF degree distribution relies under the fact that failure to low degree nodes do not result in disconnecting a huge area, increase delay or great loss of information. The proposed algorithm was tested to show the performance of the

network when random nodes where removed. The results revealed the existence of a huge connected component. However, this lacks the following:

a. Random failure in WSNs is the result of some environment conditions which can either damage nodes or reposition them. Usually this type of occurrence is considered random to low degree nodes since they occupy the majority of nodes and a higher failure probability is expected as oppose to the few high degree nodes. AWSF robustness was only measured in terms of the existence of a large connected component after random failure, but did not consider delay or missing information into account. However, a small chance of random failure to the hot connected nodes might cause the network to become disconnected.
b. A targeted failure in WSNs can happen if a large degree node was damaged due to some environmental conditions, but it is mainly concerned with the energy depletion of high degree nodes which are higher candidates to die faster than low degree nodes. The performance evaluation under simulation did not take into account when measuring its robustness failures as a result of energy outage.

4.2 "Evolution of wireless sensor networks" (49)

General description: a fault tolerant topology control algorithm is presented to produce a scale-free topology for cluster-head nodes. The selection of Cluster-heads was initially formed according to existing algorithms. The algorithm proceeds by connecting cluster-heads according

to the BA scale-free model in terms of growth and preferential attachments. Cluster-heads in WSNs are used to reduce the energy consumption of other nodes by making only cluster-heads communicate with the base station and other nodes only communicate within a cluster. The resulting network showed characteristics of SF networks in terms of robustness and small average path length.

Drawbacks: our perception about this method is that it will not offer very promising results since it considers the SF topological structure for cluster-heads. The reason is that SF topologies are appropriate when minorities of nodes are more important than the others, but cluster-heads are all equally important and protecting them is at high priority. It is more reasonable to apply a SF model within cluster-heads rather than between them. Cluster-heads must be connected according to a different process that aims to increase their robustness against random and targeted failures.

4.3 "The Reliability Performance of Wireless Sensor Networks Configured by Power-Law and other Forms of Stochastic Node Placement" (50)

General description: the algorithm is limited to applications where delay is permitted and volume of sensed data is low. The performance results revealed a SF nodal degree distribution with an exponent of ".1". Different conditions were considered to evaluate the performance which was done by varying values of the number of nodes and sensing range. The different variations had no affect on the exponent value of '.1'. The simulation was done by fixing the

data-driven conditions and looking at the network's time-driven behavior. Failure analysis was conducted by measuring the virtual sensing success (*vss*) which describes the event of successfully sensing and receiving a generated target point. Random failure was tested by measuring the *vss* after some portion of nodes were removed (*rb*) and comparing *vss* to *rb* to find relationships. For targeted failure due to battery exhaustions, the number of targeted points was increased to imply a more energy use of a sensor. *rb* and *vss* were compared to find relationships.

Drawbacks: The authors assume that if delay is permitted, the rate at which a sensor node senses data is low. However, this assumption is not accurate because delay and rate of sensing are different issues. If the rate of sensing is low enough, some targets can be missed. If sensing rate was not lowered so that data won't be missed but delay is permitted, then the sensed data can be sent at time intervals so that the information can reach the base station late. The only situation where their assumption can be valid is when targeted data (the phenomena to be sensed) does not change by time which implies that data cannot be missed and can be detected at later times. However, this is a very strict assumption because generally the environment is dynamic and targeted data points can change in terms of kind, level and location. Other issues involve the following:

1. The sensing, transmitting and receiving range were given constant values. These values must be varied according to geographical layout and remaining power.
2. Energy was only used by transmitting and receiving without considering sensing as a source of energy use. This could be a result of the initial assumption where the rate and volume of sensing is low so that energy used for sensing is low. However sensing,

transmitting and receiving consume the most amount of energy compared to computation.

3. The authors did not consider the effect of both failures happening at the same time, that is varying rb and the number of targets and evaluating the performance under this situation.
4. The performance results showed that the different value of lambda for the SF distribution had an effect on the performance by which was calculated to be close to ".1". The different values of the number of nodes and sensing range did not affect the value of lambda. We relate that to the fact that sensing was not considered as a source of energy use.

4.4 "A scale-free routing algorithm in WSN" (51)

General description: the algorithm begins by a sink node acquiring information about positions and energy of all nodes and breaks them into several layers. This step is done only at the beginning of the algorithm and the iterative nature of this algorithm is only within next steps.

Drawbacks: Energy and location of nodes are changing during the time evolution of the network and these layers will not remain the same. The algorithm lacks that dynamical behavior.

1. The algorithm does not clarify how layers are divided based on energy and locations. Cluster heads only connect to upper layer heads only. The reason behind this was not explained as well. A fusion function was used to detect data redundancy which was done by head-clusters or semi-cluster heads. Basically nodes within a cluster send the sensed information to these nodes and so that they only send the fusion results to the higher cluster unit. This method can be better implemented if:
    a. Data redundancy was node by the node sensing the information so that it only sends the fusion results as appose to the whole data to the head. Since data computation consumes less energy than transmitting data, nodes will better optimize their power if data fusion was done at that node. Redundancy can also occur when two nodes sense the same data, so a fusion function at the head should be done in order to detect redundancies of more than one single node. This procedure will help in detecting redundant nodes and schedule the sensor node power. Control schemes that use power scheduling can shut down one of these redundant nodes or lower their sensing range to balance power.
    b. Power scheduling can also be considered as redundant nodes can switch between ON and OFF to balance their energy use. This provides a power control scheme that is based on local information of each cluster. The computation complexity is lowered by using only local information and delay is eliminated. Optimality is achieved at the level of each cluster and by optimizing each cluster according to the data-driven conditions; the global optimality can be achieved. Since each cluster occupies a separate region of the sensing field, each region is

optimized so the global behavior of the networks is optimized as well. Cluster heads communicate with other clusters based on time and event driven mechanisms. If the load on a cluster is high due to the volume of sensed data in that region, another cluster can be chosen to route data to base station. We should not limit the commutation to certain clusters but choose according the best one at the current time.

4.5 "Energy and Coverage Aware Routing Algorithm in Self Organized Sensor Networks" (52)

General description: the routing algorithm divides the sensing region into virtual grids where sensors belong to different grids according to the transmission and sensing ranges. Sensors in each grid switch the role of being the cluster-head according to some mechanism. In each grid, only the cluster-head operates in an ON mode and the rest are turned OFF to conserve power. The advantage of this mechanism is that nodes switch states between ON (cluster-head rule) and OFF to balance energy usage.

Drawbacks: the disadvantage is that when only one sensor node is ON and the remaining and OFF, problems such as delay, congestion or missing information can occur. These problems can occur when the volume and rate of sensed data is high and only one node is operating. Basically, the routing algorithm depends on only time-driven sampling and does not consider

any data-driven conditions into the account. We suggest that the algorithm must optimize the number of active nodes in each grid by considering events such as the volume and rate at which data is sensed. This will also account for transmitting the data by more than one node. For instance, if targeted data is low enough so that one active node could possibly be efficient to sense data in that cluster grid, this might still result in congestion. Congestion on a node can occur if the channel incoming capacity is larger than outgoing capacity. Each cluster is not just in charge of sensing but also sending and receiving data from other units to transfer information to the base station. So each grid must also have enough sensor nodes to facilitate fast and reliable information transfer between grids.

Chapter 5 Characterize Wireless Sensor Networks as Complex System

Although WSNs are complex systems, they have many constraints that contribute to the overall behavior of the system. The theory and models of complex systems give us a new insight on studying evolutionary algorithms while considering the constraints of the sensor network.

5.1 Graph Theoretical Description

We present a graph description for WSNs based on graph theory and dynamic network measures (53) (54) (55). The graph characterizes the WSN in general meaning without forcing any control so there is a free behavior in order to graphically represent the WSN maximum capabilities and limitations. We explain the WSN graph measures and identify the properties that distinguish them from different networks.

G (N, E): is a graph with N and E representing nodes and edges respectively. N represent sensor devices, and E represent communication between nodes which is determined by the sensor node transmission range (signal). Transmission range is affected by power and geographical layout.

Degree: The *degree* of a node K represents its connectivity which is the number of links incident on a node. Generally, $K \in [0, N-1]$. A senor node degree is limited by the nodes lying within its transmission range; i.e. $K \in [0, Ns-1]$, where Ns is the number of nodes that are geographically present within the range. Ns is affected by the density distribution of sensor nodes. <K> is the *average degree* of the network which is usually calculated by <k> = $2E/N$.

Shortest path and mean path length: The *Link/path length* L is the number of edges involved to transfer data or pass through two nodes. It is a measure of network distance. $L \in [0, N-1]$. Shortest path length is the shortest link length between any two nodes. From the WSN view this can be done by formulating an optimization function to select the shortest longest path. Basically the path is selected to minimize nodes within a path and this implies that sensor nodes connected by longest links. The *average/mean link length* <L> is the average shorted path evaluated over the whole network.

Clustering Coefficient: The *clustering Coefficient* (*Cc*) describes the ability of the network to have alternative paths. The clustering coefficient basically shows that if nodes A and B are connected and B and C are connected, then A is most likely connected to C. The clustering coefficient is measured by: $Cci = 2ni/k(k-i)$, where ni: is the number of links between the neighbors of node i of degree k. $k(k-i)/2$ shows the possible number of connection between all neighbors of node i. This value is influenced by transmission range, the deployment strategy and how nodes are densely distributed. The *average clustering coefficient* <C>: characterizes the overall tendency of nodes to form clusters. <C> = sum *Cci/N*. $C(K)$ is a very

important measure that represents the *average clustering coefficient of nodes with degree K*. $C(K)$ is calculated for different values of K.

Degree distribution: The *degree distribution*, *P* (*k*), describes the probability of a node with degree K. *P*(*k*) is measured by counting the number of nodes N(*k*) with *k* = 1, 2… links and dividing by the total number of nodes. The degree distribution enables us to differentiate between different types of networks. For instance, the Poisson distribution of random networks describes a system without highly connected nodes, while a scale-free distribution describes the existence of high degree nodes and many low degree nodes. For WSNs this is influenced by the deployment strategy that can be random or planned and by the network structure that can be hierarchical or not. For example: a random deployment with a non-hierarchical structure might result in a Poisson distribution of node degree which implies most nodes have a degree = <K>. The sensors transmission range value will impact the value of <K>. Whereas a hierarchical structure with planned deployment might result in a SF distribution where few nodes have a high degree and the remaining nodes have a low degree. The deployment strategy has a great influence on the degree values and distribution.

The network analysis described above, shows characteristics of the sensor network in terms of capabilities and limitations. This gives us a clear vision of how the sensor network differs from other types of networks. This will allow us to explore the different models of complex system from the WSN perspective.

5.2 Topological Structure Models

The degree distribution is a very important property in the field of complex systems. This importance is given in accordance to the effect it has on the system behavior and the resulting characteristics. From the above analysis and based on our extensive research on network theory models, we investigate topological structures based on degree distributions such as random networks and scale-free (SF) networks [12, 14, 18, 22]. We analyze each type of network from the WSN point of view.

Erdős–Rényi (ER) (56) *model from a WSN perspective:* The intentional attack error tolerance property is a result of non-hierarchical structure which implies that all nodes are equally important demonstrated by the straight line clustering coefficient. However, the low clustering coefficient in random graphs represents a major problem to the sensor network as that will contribute to the sensor network vulnerability. Sensor nodes are very vulnerable to failure and it is at high priority to apply a mechanism that can increase the network resilience.

Therefore, the ER model is optimized if the sensor network is required to operate in an environment where harsh conditions are eliminated as that implies less random failure probability. When a hierarchical structure of sensor network is considered, the ER model provides an insight to increase fault tolerance to an intentional attack type of failure by introducing some nodes with the same importance as cluster-heads. The ER model shows that a

node degree value depends on the choice of p which implies that the node degree can be bounded. It shows characteristics of small-world and resilience to intentional attacks. Looking at these points from a WSN point of view shows an acceptable way to model the network to achieve the mentioned features while considering the following issues:

- The sensor node degree depends on the number of nodes within its range. The range value will influence the probability to establish connections and this will allow to account for geographical constraints of the WSN
- Varying the transmission range can be achieved in order to minimize power or increase efficiency for instance.
- The small-world property in a result of long edge lengths within a link path which for the sensor network implies large transmission rage and more power consumption.

The SF model from a WSN perspective: The robustness of SF networks depends on the existence of high degree nodes that represents a small percentage of the total number of nodes in the network. This is demonstrated as the probability of random failure to such important nodes is low compared to the large size of the network where a higher failure probability is expected for the majority of nodes having a low degree. Since failure of a high degree node contributes to the network's vulnerability, it is considered an intentional attack type of failure. As a result, the strength of SF networks relies heavily on the underlying premise that the probability of random failure is high compared to targeted attacks. From a WSN perspective, failure probability analysis differs from the above due to the constraints, limitations and vulnerability of the sensor nodes. Therefore, a WSN that is required to operate in harsh environmental conditions can draw inspiration from the SF model in order to increase

error tolerance against random failure. Other issues regarding the implementation of the BA SF model to WSNs are more structural and involve the following:

- The SF degree distribution follows the rich get richer phenomenon which cannot be implemented to the sensor nodes as that implies some nodes will have a large number of connections which results in rapid energy consumption.
- SF models do not consider any geographical constrains in forming connections between nodes. On contrary, sensor nodes are limited in range and nodes outside a certain range cannot establish a connection.
- The efficiency of SF networks relies on its property of small average path lengths between nodes which is a result of long edge links. The existence of long edge links in a WSN contributes to rapid power depletion which results in an unbalanced energy use and therefore is not considered a good design.
- The vulnerability of SF networks to targeted attacks is a result of the low clustering coefficient between nodes with high degrees and such nodes for a WSN contribute to the most important nodes as they are the cluster-heads. Therefore, increasing their clustering coefficient is important.

Chapter 6 Energy Model

6.1 Energy Model Overview

The goal of the energy models presented here is to produce energy efficient sensor networks through a scale-free degree distribution. The inherit properties of self-organization, fault-tolerance, efficiency and robustness found in networks with a scale-free degree distribution leads to our choice. Therefore, we derive two theoretical models to produce scale-free WSNs when establishing connections between sensor nodes. The basic idea for both models is to construct the WSN according to the connectivity (node degree), and remaining energy of each sensor node, in order to produce scale-free networks which have a performance of random error tolerance as well as a good energy distribution among the nodes. The difference between both models is how the energy is depleted over time.

In the first model, we assume that nodes remaining energy value are different but remain constant throughout the time evolution of the algorithm. In the second model, we also assume that each node has a different remaining energy value but this value decreases over time each time a node establishes a connection. Therefore the second model accounts for the dynamical behavior of sensor nodes remaining energy. And therefore the time evolution of node degree depends on its energy.

In order to derive both models, we use Barabasi and Albert (BA) scale-free evolution model (3) and adapt it to conform to the constraints of sensor nodes with means to build an energy efficient communication topology. The BA Scale-free model is used to generate networks through two main processes: growth and preferential attachment. Growth is the process responsible for network growth. Preferential attachments guides the way nodes increase their degree which follows the "rich gets richer phenomenon". This means that the higher the degree of the node the more probable this node will increase in degree. However, in addition we consider factors of node transmission radius and reaming energy value in our model in order to produce energy efficient and fault-tolerant WSNs. The relationship between the BA inspired models is captured in figure 6.1. In the BA model, the probability of a node to establish connection is related to the degree as $p_i = k_i / \sum_{j \neq i}^{N} k_j$. In our derived models this probability depends on the degree and remaining energy values as $p_i = k_i E_{ri} / \sum_{j \neq i}^{N \in LA} k_j E_{rj}$.

For CEM, $E_{ri,\,j} \in f(E)$ and for DEM $E_{ri,\,j} \in f(E(t), K(t))$.

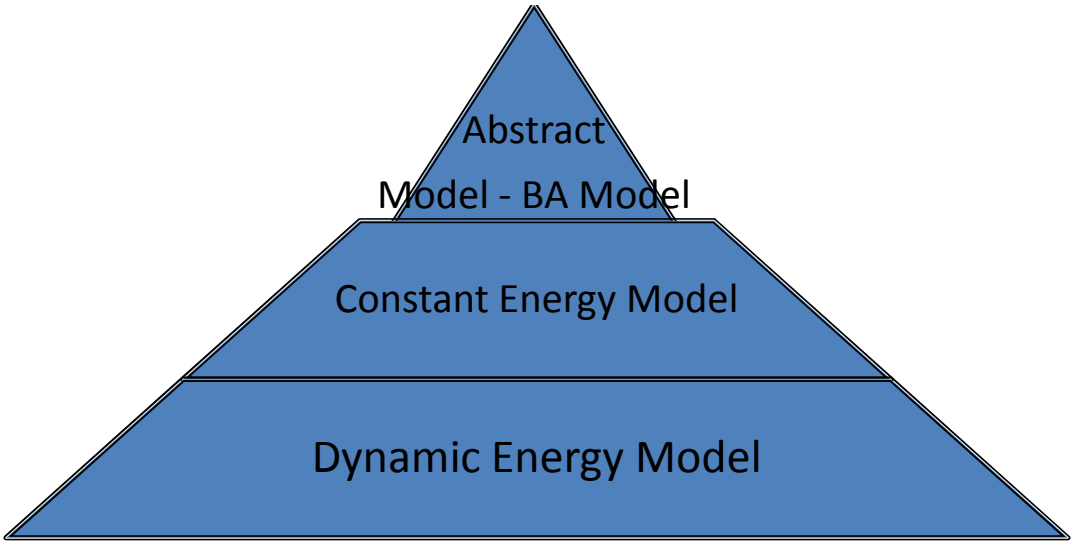

Figure 6. 1 Relationship between the three models

It is important to understand that nodes in a WSN operate at different spatial and temporal levels. Basically, when nodes are deployed in the environment, each node occupies a different location. Different locations lead to varying data gathering operations. This leads to different communication loads. Finally this all results in different energy consumption for each node. The energy consumption can be assumed to be unknown since the environment is unknown to begin with. Therefore, each node assumed location and energy value are important for model derivation.

6.1.1 Homogeneous vs. Heterogeneous Structures for WSNs

In the context of WSN, homogeneity in the communication network structure means that almost all sensor nodes are topologically equivalent, like regular lattices or random graphs. For

a large graph size, the degree distribution can be approximated by Binomial or Poisson shaped graphs. This implies that each of the possible $\frac{N(N-1)}{2}$ interaction links is present with equal probability which in a WSN represented an issue which can lean to undesirable events such as overload or congestion on a node and increased energy consumption.

Oppositely, heterogeneous interaction structures for WSN displays a power law degree distribution. This is more desirable as it avoids the aforementioned problems and enables fast and reliable information transfer between sensors devises while optimizing energy and increase fault-tolerance.

6.2 Energy Model Framework

We present the basic structure foundation of our energy model here. This structure allows us to derive our energy model to produce energy efficient and fault-tolerant WSNs. In practical applications, nodes in a WSN tend to spend most of their energy in communication which is transmitting data after the network organizes into a desired network structure. The proposed model is intended to optimize nodes remaining energy after two stages of the WSN operational phases which are deployment and data-collection. Below we present a description of the WSN conditions when using the proposed model:

1. Sensor nodes are deployed in a region according to some deployment algorithm or it can just be random deployment. However, for simplicity we assume that there are no

nodes that are isolated. So nodes are expected to fall within each other's transmission range. Usually, the deployment algorithm takes care of this issue.

2. After the network is organized, nodes collect data from the environment, and transmit the data among themselves. Therefore, the remaining energy of each node is varied. So, nodes are all given initial random values of remaining energy.

3. The degree of a node is limited to the number of nodes that exist within its transmission radius. So, the neighborhood of each node which is usually referred to as its local area connections is bounded by a small constant. Since each sensor node has a transmission range, the number of nodes that can communicate with it depends on the nodes that rely within its range.

The way which the model generation is established is based on simulating the following evolutionary processes:

1. **Growth:** starting with a small number of nodes, we introduce new nodes to the network with number of edges (v)< number of nodes already in the network ($\boldsymbol{v_0}$). We specify an upper bound on the maximum number of edges a node can take on (maximum degree).

2. **Preferential attachment:** a new node will establish connections with nodes already in the network with a probability that is proportional to the node degree and its remaining energy.

6.3 Constant Energy Model

Based on the model framework described in section 6.2, we derive the Constant Energy Model (CEM) with the goal to produce energy efficient WSNs by implementing a scale-free degree distribution of the network. The goal is to measure the performance of remaining energy from the degree distribution and centrality measure distributions perspective since they are important measures to in the study of complex networks. Such distributions can describe certain properties of the resulting network such as speed of information transfer, reliability and robustness.

Two assumptions are carried on for this model. First, nodes remaining energy value is varied but left constant. Second, in addition to the node degree value, the probability a node establishes a new connection depends on its remaining energy value which is assumed here to be a constant value. In order to solve the mathematical equations, continuum theory (57) is used to simplify the calculus. In brief, continuum theory is a method developed by Barabasi and Albert, and it can be used for networks where they undergo large number of time steps. The list of parameters used for this model is as follows:

N: Total number of nodes in the network.

K_x : Degree of node $\mathrm{x}, \mathrm{x} \in [1, \mathrm{N}]$.

v_x: Number of edges of incoming nodes (initial degree of node x).

<κ>: Average degree of the whole network.

LA: (Local Area) : number of nodes in the local area of a newly coming node. That value depends on the radius (range) of each sensor node which decreases by time.

E: Remaining energy of a node x.

$\bar{E}$: Expected value of E .

We calculate the degree distribution of our CEM Model as follows:

Similar to the calculations described for the BA scale-free model (3), we represent the probability of node x to establish a new connection through the probability:

$$P(sf) = \frac{(Ek_x)}{\sum (E_j)k_j} \ , j \in LA \qquad (6.1)$$

We calculate the change in node degree over time as:

$$\frac{\partial(kx)}{\partial(t)} = v_x E \qquad (6.2)$$

Where e, is the number of new edges introduced to the network.

$$\sum \ f(Ej)kj = (LA) < k > \ \bar{E} \qquad (6.3)$$

We calculate the avg. degree of the whole network, the expected value of remaining energy and the avg. number of nodes in each new comer local area.

The calculations presented here follow the BA calculations done in order to estimate the degree distribution (3) (9). Based on continuum theory, the calculations are simplified. In the case of large scale networks that undergoes a large number of time steps, the average degree can be calculated as:

$$< k >= 2 \times \frac{number\ of\ edges\ per\ node\ (v) + number\ of\ edges\ at\ the\ begining}{number\ of\ nodes\ at\ the\ begining + number\ of\ time\ steps\ (t)}$$

$$< k > \approx 2v \qquad (6.4)$$

This implies that the average degree of the network is approximately double the number of edges per node.

Based on the value of LA, the local area can be either fixed or dynamic. A fixed value indicates a limit on the number of nodes in the new comer's local area which implies a limit on the degree which is basically the maximum degree of the node ($Kmax$).

The time evolution of node degree can be calculated as:

$$\frac{\partial(kx)}{\partial(t)} = \frac{Ekx}{2\ (LA)} \qquad (6.5)$$

$$\frac{\partial(kx)}{kx} = \frac{E}{2\ (LA)}\partial(t) \qquad (6.6)$$

$$kx = e^{\frac{E}{2(LA)}+C} \qquad (6.7)$$

The calculation of the constant C can be explained as follows: at a given time, a new node is introduced to the network with a fixed number of edges. So we can write the following:

$$kx(tx) = \mathrm{v}_x \rightarrow c = \ln \mathrm{v}_x - \frac{E}{2(L-A)}\, tx \tag{6.8}$$

$$kx = e^{\frac{E}{2\,(L-A)}(t-tx)}\mu_x \tag{6.9}$$

The probability that a node x degree is less than *k* is:

$$p(k_x(t) < k) = p\left(\frac{E}{2(LA)\bar{E}}(t - t_i) < \ln\left(\frac{k}{\mu_x}\right)\right)$$

$$= p\left(t - \left[\frac{2(LA)\bar{E}}{E}\ln\left(\frac{k}{\mu_x}\right)\right] < tx\right) \tag{6.10}$$

If nodes are added at equal time intervals → probability density for node x at t_x (time) is given by: $p_x(t_x) = \frac{1}{\mu_0 + t}$

$$\rightarrow p(k_x(t) < k) = 1 - p\left(t_x \leq t - \frac{2LA\bar{E}}{\mathrm{E}}\ln\left(\frac{k}{\mu_x}\right)\right)$$

$$= 1 - \frac{\left[t - \frac{2LA\bar{E}}{E}\ln\left(\frac{k}{\mu_x}\right)\right]}{\mu_0 + t} \tag{6.11}$$

The Pdf for nodes with remaining energy E:

$$\rightarrow p(k_E) = \frac{\partial p(K_x(t) < K)}{\partial k} = \frac{1}{\mu_0 + t}\frac{2LA\bar{E}}{E}\frac{\mu_x}{k} \tag{6.12}$$

$$p(k) = \int_{E_{min}}^{E_{max}} P(E)P(k_E)\, dE \;\; = \int_{E_{min}}^{E_{max}} \rho\, \frac{1}{\mu_0 + t}\, \frac{2L\bar{E}}{E}\, \frac{\mu_x}{k}\, dE \tag{6.13}$$

Where $P(E)$ is the distribution of E, and E_{min} , E_{max} are the limits of the energy distribution.

$$A = \int_{E_{x,min}}^{E_{max}} P(E) \frac{1}{\mu_0 + t} \frac{2L\bar{E}}{E} \, dE \qquad (6.14)$$

In general, scale-free networks have power law degree distributions which can be approximated by:

The distribution: $p(k) \sim Ak^{-\gamma}$ where, $2 < \gamma < 3$. The average degree $< k >$is well defined and bounded. The variance is: $\sigma^2 = < k^2 > - < K >^2$. The variance value depends on maximum limit on the degree k_{max}as: $< k >= \int_{k_i min}^{k_{max}} k^2 p(k) \sim k_{max}^{3-\gamma}$.

Therefore, based on the calculations observed in Eqs. (6.13) and (6.14), we can see that the algorithm can result in a scale-free degree distribution with a degree exponent $\gamma = 1$ and an A value given in Eq. (6.13). In the simulation section we confirm the calculation results in order to validate our theoretical results.

6.4 Dynamic Energy Model

The constant energy model presented above can produce energy efficient networks. However, nodes in the WSN can consume their energy very fast if their degree is high. Therefore, to provide more realistic results and to better model the energy dynamics for each sensor node, nodes' energy values cannot stay constant. Here, we present a Dynamic Energy Model (DEM)

with means to balance the energy consumption of all nodes in the network in a better way. Therefore the main difference here is the limit given on the degree of a node which depends on its remaining energy value. The remaining energy value also depends on the degree value.

The model gives each node a maximum value of degree ($K_{\max,i}$) and a maximum value of remaining energy (E_{rmax_i}). These values are dependent on each other. The dependency between both variables can be observed in the following relationship:

$$K_{max,i} = XE_{rmax} \quad (6.15)$$

Where X defines the relationship between the maximum degree node i identifies relative to the maximum remaining energy value of that node.

The remaining energy value of node i can be estimated as the initial remaining energy of that node after subtracting the amount of energy consumed as in:

$$E_{r,i} = E_{0,i} - Y \quad (6.16)$$

Where Y: defines (or approximates) a function that depends on the degree and the rate of power consumption of node i. Basically it is the rate of change in energy value over time which can be computed as the time derivative of the remaining energy.

$$\frac{\partial E_r}{\partial t} = Y \quad (6.17)$$

Similar to the CEM described in section (6.3), the degree distribution and the remaining energy distribution are calculated and used as a way to evaluate the performance of the resulting network.

6.4.1 Model Variables

We categorize the variables we use to derive the DEM presented here into input, computational and output variables. The dependency between variables is shown through numerical equations which captures how variables change over time relative to each other.

Input variables

n_0: Initial number of nodes in the network.

LA: Number of nodes in each local area (local-area-connections).

e_0: Initial number of links between n_0 nodes.

$(e_0 < n_0)$. *Note: In the simulation model, each node is initialized with a very small degree and the number of links in the network = 2k/n which must be <* n_0.

$E_{r,0}$: Initial remaining energy value of node i.

$k_{x,0}$: Node initial degree. Each node has an initial degree (i.e. number of edges) that is less than the number of nodes in the local area.

E_{max}: Maximum energy value of a node.

Computational variables

$N_{t=i}$: Current number of nodes in the network i.e. to compute number of nodes at every time-step (if discrete model) or (at any time ti for continues model).

$K_{total,t}$: Degree of the network at a given time.

ΔE: The change in energy for any node at a given time.

E_r: Remaining energy value of a node.

The total change in energy is:

$$\Delta E = E_{max} - E_r \tag{6.18}$$

And the energy change per time step is:

$$\Delta E(t) = E_t(x) - E_{t-1}(x) \tag{6.19}$$

Note: for the deterministic model, there is no change in energy. We do not account for energy as the node degree increases.

ΔK: The change in degree for a node at time t:

$$\Delta K = k_t(x) - k_{t-1}(x) \tag{6.20}$$

$$\Delta K(x) = \begin{cases} +1 \dots\dots if\ C1 \\ o \dots\dots if\ C2 \\ -v \dots\dots if\ C3, v \geq 1 \end{cases}$$

C1: node x established a new connection

C2: node x has no new connections

C3: node x loss of 1 or more connections. The value of v vs. Er is critical to determine if node x power is critical or one of its neighbors is damaged.

Output variables

E_r: Remaining energy of a node *and remaining energy distribution.*

Total change in energy of a node=

$$\Delta E = E_{max} - E_r \quad (6.21)$$

$k_t(x)$*:* Degree of a node *and degree distribution.*

6.3.2 Behavior and Algorithm

Similar to CEM, we use the same two processes for network evolution: growth and preferential attachment. Based on the conceptual model, the behavior of the system over time is captured through calculating how variables change through the energy model described in this section. Continuum theory is used to examine the degree distribution created by the proposed algorithm. This algorithm enables network growth to be treated as a continuous process in order to allow simplification using calculus. For networks that undergo large number of time steps, this approximation should match closely with discrete network growth.

In order to account for energy consumption due to communicating with other nodes, we assume that a node will consume a certain amount of energy (*x Jules*) per degree unit (*1 degree*). Therefore, we consider the energy consumption rate denoted by (E_{RT}) to be a static value.

The algorithm is described as follows:

At each time interval (t), variables are calculated as follows:

- **At $t_{i=0}$ or $t = o$**

  $E_{max}(x_n) = E_{r,t=0}(x_n)$ …… $n \in [1, N_0]$

  N_0: # $nodes\ initially\ in\ network$

  $E_{max}\ for\ (x_n)$: $constant\ value\ or\ follows\ a\ distribuion$: $time\ invarient\ value$

  $\Delta E(t = 0) = 0$

  $E_{r,x_n}(t = 0) = E_{max,x_n} = E_{r,x_n}(t = 0)$ …… $n = \{1,2,3, \dots, N(t = 0)\}$

  $K_{t=0}(x_n) = initial\ degree\ of\ node\ n\ is\ given\ a\ small\ value \in [1, N(t = 0) - 1]$

  $E_{r,t=i}(x_n) = E_{r,0}(x_n)$ …… $n \in [1, N]$

- **At t_{i+1} or $t > 0$**

At time $(t = i + 1)$, a new node $(x_{n=t+N0})$ is added to the network with number of edges less than the number of nodes already in network. $K_{t=i+1}(x_{n=t+N0}) < N_0$. (For simplicity we write it as $K_t\,(y_n)$ to indicate the newly introduced node as y, i.e.

$$K_t(x_{n,i}) = K_{t-1}\,(x_{n,i}) + K_t\,(y_n)\left[\frac{E_{r,t-1}(x_{n,i})K_{t-1}\,(x_{n,i})}{\sum_{i=1}^{N} E_{r,t-1}(x_{n,i})}\right] \qquad (6.22)$$

For all nodes in network, compute:

$$n\ is\ node\ index{:}\ x_n is\ node\ value\ y_n\ is\ the\ new\ node\ coming\ in\ the\ network,$$

$$can\ be\ written\ as\ x_{n=N(t)"index"}\ where\ N(t)$$

$$K_{x_{n=i}}(t) = K_{x_{n=i}}(t-1) + K_y\,(t)\left[\frac{E_{r,x_{n=i}}(t-1)K_{x_{n=i}}\,(t-1)}{\sum_{n\neq i,n=1}^{N(t-1)} E_{r,x_n}(t-1)K_{x_n}\,(t-1)}\right] \quad (6.23)$$

$$\forall\ x_n = \{1,2,3\ \ldots.,N(t-1)\}$$

$$\Delta k_{x_{n=i}}(t) = K_t\,(y_n)\left[\frac{E_{r,x_{n=i}}(t-1)K_{x_{n=i}}\,(t-1)}{\sum_{n\neq i,n=1}^{N(t-1)} E_{r,x_n}(t-1)K_{x_n}\,(t-1)}\right] \qquad (6.24)$$

So, the total number of nodes in the network is $N(t) = N(t-1) + 1$ or $N(t) = N(t-1) + i$

And the change in energy is:

$$\Delta E_{x_n}(t) = \Delta k_{x_n}(t)\,E_{RT}$$

$$= K_t\,(y_n)\left[\frac{E_{r,x_{n=i}}(t-1)K_{x_{n=i}}\,(t-1)}{\sum_{n\neq i,n=1}^{N(t-1)} E_{r,x_n}(t-1)K_{x_n}\,(t-1)}\right] \qquad (6.25)$$

Equation (6.24) basically describes how the degree of a node that already exists in the network i.e. all the nodes at a given time which is $t < t_i$ can change in response to the degree of the new node introduced. So, at time t_i, if we pick a node from the set of nodes $\in x_{n,}(t_{i-1}) \ldots \ldots . \forall i$. The degree of that node can be estimated as: (the value of the node old degree) + [the degree of the new node introduced times a probability that depends on the ratio of its remaining energy and degree values over all nodes in the network]. In general, we can write:

$$k_t(x) = k_{t-1}(x) + \Delta k(x) \text{ and } \Delta k(x) = \Delta k_t(x) \quad (6.26)$$

$$\Delta k_t(x) = k_t(y_n) \left[\frac{E_{r,t-1}(x) k_{t-1}(x)}{\sum E_{r,t-1}(x_n) k_{t-1}(x_n)} \right] \quad (6.27)$$

$$\forall x_{n,} \neq x, x_n \in (N_0)$$

$$E_{r,t}(x) = E_{r,t-1}(x) - \Delta E_t(x) \quad (6.28)$$

$$\Delta E_t(x) = \Delta k_t(x) E_{RT}$$

$$E_{r,t}(x) = E_{r,t-1}(x) - [(k_t - k_{t-1)} E_{RT}] \quad (6.29)$$

This model is tested under different distributions of nodes initial values of degree and remaining energy in order to study the performance results from a degree distribution and evolution perspective. Additionally, the number of nodes in the local area is given random values as well as constant for all nodes in order to see the effect of both cases. We vary the upper limit on a node degree in order to observe the resulting outcome. Also, we vary the

lower limit on a node minimum allowable energy value that can enable a node to establish new connections. Therefore, we give the degree and energy values of nodes threshold values that can be adjusted depending on different applications in order to allow users to conduct different experiments. As a result, the network can be adapted for different applications. Since WSNs are usually used for different purposes, and depending on the application in hand, different priorities are given to expected performance results.

Chapter 7 Simulation Model

In this chapter, we present the approach we use for the simulation model. The purpose is to study the performance of the networks generated under CEM and DEM when varying parameters such as: number of nodes in the network, the local area of each node, remaining energy values, and maximum degree. For a WSN, a simulation model requires a proper representation of the senor node geographical layout. Basically, we need to characterize the environment which represents the sensing field, data points to be sensed and sensor devices. Also, we need to keep track of each sensor node values of remaining energy and degree over time. In order to properly characterize the network, we consider the following issues:

a. *Local area of each node.* For each node (i), this is the number of ($j \neq i$) nodes in its local area that they can communicate with each other.

b. *Density deployment of sensor nodes relative to location.* Which nodes are in dense area and which are sparse.

c. *Data-driven formulation*. Analyze the outcome of different deployment schemes. The data-driven conditions can be set through this.

d. *Monitor the degree of each node over time*. In order to analyze the time evolution of nodes in the network.

e. *Monitor the energy of each node over time*. In order to analyze the energy changes of nodes in the network over time.

In order to study the above issues, we use matrices and vector calculus to formulate the WSN model. Matlab was used as a software tool to build and simulate the model. Below we explain in details of the methodology and framework of the simulation model.

7.1 Methodology

First: in order to represent the sensing field, we use a matrix of size (k*k) where k is the total number of nodes in the network. Therefore, we give a vector representation of sensor nodes as:

1. We formulate matrix [M1], where its rows and columns represent sensor nodes.
2. Each node will be represented by a column vector composed of 0 or 1 entries (components).
3. The dimensionality of each vector must equal to the total number of nodes in the network.
4. The matrix is symmetrical with 0 diagonal values because a node cannot connect to itself.

The following is a sample $[M1]$ matrix:

Node vectors $[\boldsymbol{V}1\ \boldsymbol{V}2\ \boldsymbol{V}3\ \boldsymbol{V}4\ \boldsymbol{V}5]$, i.e.

$$[M1] = [\boldsymbol{V}1 \quad \boldsymbol{V}2 \quad \boldsymbol{V}3 \quad \boldsymbol{V}4 \quad \boldsymbol{V}5] \tag{7.1}$$

$$[M1] = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (7.2)$$

So a sensor node vector (ex. $V_{i=1}$,) $[M1]$ enables us to characterize the node state since it can show information about:

1. The number of nodes within node 1 radius as the sum of 1's in the vector i.e. $\sum_1^r(r, c = 1) = 1$. This is basically the size of the vector (# nodes in local area). Where *r, c* are rows and columns of the matrix respectively.

2. If node 1 has a number of nodes in local area equal to a high value, this implies that this node is in dense area. Else, if the number of nodes in local area is low, this impels that this node is in sparse area.

3. For $V_{i=1}$, a value of 1 in its j'th component (ex. $j = 5$) means that vector node $V5$ is within the transmission/communication range of node 1.

4. Similarly, a value of 0 at $j = 3$ for example, means that nodes 1 and 3 are not within each other's transmission/communication range.

5. If we look at the whole network represented by the matrix, which is composed of all the column vectors, we can find out if the network follows more of a hierarchical structure or not. The way the *1*s are distributed in the matrix can reveal great information about the density deployment of sensor nodes.

6. The data-driven conditions can be set by initializing [M1] to different values.

Basically [M1] is the matrix used to represent the environment where sensor nodes are located. We provide more information on [M1] in the next section.

Second: in order to keep track of each sensor node values of remaining energy and degree over time we formulate two matrices: $[\mathrm{MK}]$ to keep track of the degree and $[\mathrm{ME}]$: to keep track of the energy.

The degree matrix $[MK]$:

1. Columns represent sensor nodes; again we consider vector notation of each node. So columns are numbered from $i = 1$ to total number of nodes, each node is identified by its number.

2. Rows are time steps. Time t starts from *0* to (total number of nodes +1). So rows are $j = \{1,2,3 \dots, i + 1\}$.

3. Values of the matrix elements represent the current degree for $[MK]$ or current energy for $[ME]$ of a node. So the value in matrix element $(1, 3)$, for example, shows the degree or energy of node 3 at t$=0$. Similarly, the value in $(5, 7)$ shows the degree or energy of node 7 at t$=4$.

7.2 Model Construction

7.2.1 Simulation Framework

The simulation model is built for a network of size $i \leq 1000$ sensor nodes. The idea is to make this model adaptable so that the user can specify different parameters in order to test different scenarios. Therefore, the model variable values are controllable by the user.

In order to mimic a real environment situation for WSNs, we establish the formulation explained in this section which is heavily based on matrix calculations. As a result, Matlab is used as a programming platform for building and simulating the model.

To represent the environment of sensor nodes on a geographical layout, we need to know which nodes are close in radial proximity with each other so they can establish connections. Therefore, based on the formulations presented in the previous sections, if we assume $i = 1000$, then each node is represented by a column vector of size $j = i = 1000$. The vector is composed of $0\ and\ 1$ values. For all vectors V_i, where $i = 1\ to\ 1000$, each component of V_i corresponds to a j value, where j is the dimension of the vector ($j = i = 1000$).

In order to build the network of sensor nodes, we follow these steps:

- Step 1: Construct matrix $[M1]$ of size $(i * i)$ where we set *i*=1000=total number of sensor nodes.

- Step 2: treat each column of $[M1]$ as a node i.e. $V1$ is node 1, $V2$ is node 2… etc.

- Step 3: Construct $[M2]$ to reflect the number of 1's in each vector node. Basically we need to compute $[M2] = [\boldsymbol{V}1]^T[M1]$ which should be of size (1×1000). Each value in [M2] is important for later steps. The value in each index $(1, c)$ is the number of nodes in the local area of a node $i = c$.

- Step 4: Initialize each sensor node at time $(t = 0)$.

  a. Construct matrix $[MK]$ where columns C_i are nodes $i = 1\ to\ 1000$ and rows $Rj's$ are time steps $t = \{0,1,2, \dots ,1000\}$. So the size of $[MK]$ is $= (i * (i + 1))$ since t starts at 0.

     - At $t = 0$, only the first row is filled with k_i values. Each node *i* has an initial k_i

       o The first node is initialized as $k_{i=1} = 0$; we arbitrarily pick other 7 nodes and initialize their degree to a small value which has to be less than the total number of nodes chosen (here 7) and each node initial degree must be smaller than the number of nodes in the local area. So, $k_i < M2_i$. The rest of the nodes are given $k_i = 0$. This will take care of the growth process where the network begins with a very small number of nodes and links.

       o So, $[MK]$ has the first row initialized to represent ($t = 0$).

     - For $t \geq 1$, that is rows ≥ 2 in $[MK]$ are initialized to 0.

  b. Construct matrix $[ME]$ where columns are nodes $i = \{1,2,3, \dots ,1000\}$ and rows are time steps $t = \{0,1,2, \dots ,1000\}$. $[ME]$ has the same size as $[MK]$ $(1000 * 1001)$

- At $t = 0$, only the first row ($r = 1$) is filled with E_i values
    - For E_i values: we need to try different values (for now we use random values between $[0.3,1]$ Jules).
    - Fill $(r = 1, c = i)$ for all $i = \{1,2,3, \dots ,1000\}$ with random numbers $\in [0.3,1]$.
- All rows starting from $r = 2$ (i.e. $t = 1$) are *0*s.

- Step 5: Now we start the simulation. The number of time steps (t) = number of nodes(i) + 1. Here, $i = 1000$.

  So at $(t = 1)$:

  Pick node $i = t = 1$

  1. Assign $MK(r = 2, c = 1) = k_1$ $(ex.\ k_1 = 2)$.
  2. Assign $ME(r = 2, c = 1) = E_1$ $(ex.\ E_1 = 0.7)$
  3. From Step 2 search $V_{i=1}$, to find the indexes j where the value of $(r = j, 1) = 1$. These locations point to the vector nodes ($j \neq i$) where they can connect to it. So for example if node 1 had the vector value:
     a. $\boldsymbol{V}_1 = [0\ 1\ 0\ 1\ 1\ ]$, that means nodes 2,4,5 are within its local area.
     b. Use $\boldsymbol{V}_2, \boldsymbol{V}_4, \boldsymbol{V}_5$

c. If we initialize node 1 with values of $A_1 = (2, 0.7, 3)$ corresponding to $(k_1, E_1,$ the 1st value in $[M2])$, $k_1 =$ value in $MK(\mathrm{r} = 1, \mathrm{c} = 1), E_1 = ME\ (\mathrm{r} = 1, \mathrm{c} = 1)]$)

4. For each vector nodes 2,4,5 compute the following probability:

$$Probability\ (i) = \left[\frac{E_{r,x_{n=i}}(t-1)K_{x_{n=i}}(t-1)}{\sum_{n\neq i,n=1}^{N(t-1)} E_{r,x_n}(t-1)K_{x_n}(t-1)}\right], i = \{2,4,5\}$$

$(t-1) = 0$, means use the initial conditions values which are in the 1st rows of MK and $[ME]$ (first row vectors of $[MK]$ and $[ME]$.

Basically,

$$1st: probability\ (i = 2) = \left[\frac{E_{r,x_2}(t-1)K_{x_2}(t-1)}{\sum\ \{(E_{r,x_4}(t-1)K_{x_4}(t-1) + E_{r,x5}(t-1)K_{x_5}(t-1)\}}\right]$$

$$2nd: probability\ (i = 4) = \left[\frac{E_{r,x_4}(t-1)K_{x_4}(t-1)}{\sum\ \{(E_{r,x_3}(t-1)K_{x_3}(t-1) + E_{r,x5}(t-1)K_{x_5}(t-1)\}}\right]$$

$$3rd: probability\ (i = 5) = \left[\frac{E_{r,x_5}(t-1)K_{x_5}(t-1)}{\sum\ \{(E_{r,x_3}(t-1)K_{x_3}(t-1) + E_{r,x4}(t-1)K_{x_4}(t-1)\}}\right]$$

Since $k_1 = 2$, then we pick the max *2* values of ($Probability\ i$). If $k_1 = 1$, pick only the one maximum value ($Probability\ i = 1$).

Example for nodes 2,4,5:

- Node 2 values= $[9,0.7,6]$, where $9 = \mathrm{MK}(\mathrm{r} = 1, \mathrm{c} = 2), 0.7 = \mathrm{ME}(\mathrm{r} = 1, \mathrm{c} = 2)$

- Node 4 values= $[3, .3, 8]$, where $3 = \mathrm{MK}(\mathrm{r} = 1, \mathrm{c} = 4), .3 = \mathrm{ME}(\mathrm{r} = 1, \mathrm{c} = 4)$

- Node 5 values= $[2, .3, 10]$, where $2 = \mathrm{MK}(\mathrm{r} = 1, \mathrm{c} = 5), .3 = \mathrm{ME}(\mathrm{r} = 1, \mathrm{c} = 5)$

- We take values in $[\mathrm{MK}]$ and $[\mathrm{ME}]$ corresponding to indexes $(\mathrm{r} = \mathrm{t} + 1, \mathrm{c} = \mathrm{i})$ for $\mathrm{t} = 0\ and\ i = 2, 4, 5$

$$prob\ (i = 2) = \left[\frac{.7*9}{(.3*3)+(2*.3)}\right] > prob\ (i = 4) = \left[\frac{.3*3}{(.7*9)+(2*.3)}\right] > \mathrm{Prob}\ (\mathrm{i} = 5) = \left[\frac{.7*9}{(.3*3)+(.7*9)}\right]$$

Therefore, node 1 will connect with nodes 2 and 4. Therefore, update nodes degree and energy values in $[\mathrm{MK}]$ and $[\mathrm{ME}]$ from $t = 0\ to\ t = 1$ as: (Note: do not update node 5 since it was not chosen)

- In $[\mathrm{MK}]$ update:

    a. node $i = 2$ is updated in location $(r = 2, c = 2)$ as:

        i. $\mathrm{MK}(2,2) = \mathrm{MK}(\mathrm{r} = 1, \mathrm{c} = 2) + 1$

            - This is exactly: $\mathrm{K2}(\mathrm{t}) = \mathrm{k2}(\mathrm{t} - 1) + 1$

            - $\mathrm{K}(\mathrm{r}, \mathrm{i}) = \mathrm{k}(\mathrm{r} - 1, \mathrm{i}) + 1, \mathrm{where}\ \mathrm{i} = \mathrm{c}$

b. The same applies for node $i = 4$, where it gets updated in location ($r=2$, $c=4$) as:

i. $MK(2,4) = MK(1,4) + 1$

- In [ME]

a. For $i=2$

i. $ME(2,2) = ME(1,2)$ - RT

RT: is rate of change in energy (example 0.01). Similarly we can use the notation $E2(t) = E2(t\text{-}1)$ - RT

b. For $i = 4$

i. $ME(2,4) = ME(2,4)$ - RT. Assume $RT = .01$

- Increase time step by 1 as: $t=t+1$

7.2.2 Sample Numerical Example

In an adjacency matrix a value of 1 indicates the existence of a link between two nodes and 0 does not indicate a link; the local area of each node is represented. Similarly, 1 represents the existence of a node in the local area and 0 does not indicate a link.

The matrix $M1$ below represents the topology structure that shows how nodes are distributed in the field.

$$[M1] = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

From $[M1]$, nodes 2 and 3 are within the local area of node 1 as $M1(2,1) = M1(3,1) = 1$, whereas nodes 1 and 5 are not within each other's local area.

The number of nodes in the local area of each node is (sum of each column):

$$D1 = [\ 2 \quad 2 \quad 4 \quad 2 \quad 2\ ]$$

At time $t = 0$, the first rows corresponding to $[\mathrm{ME}]$ and $[\mathrm{MK}]$ are initialized in this example as:

$$[ME] = \begin{bmatrix} \mathbf{0.8225} & \mathbf{0.4058} & \mathbf{0.7668} & \mathbf{0.3996} & \mathbf{0.7712} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$[MK] = \begin{bmatrix} \mathbf{0} & \mathbf{1} & \mathbf{3} & \mathbf{1} & \mathbf{1} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

At$t = 1$:

First: the value of $\mathrm{MK}(1,1)$ is initialized with a value $< \mathrm{D}(1)$.

$$[MK] = \begin{bmatrix} \mathbf{0} & \mathbf{1} & \mathbf{3} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

So, node 1 is given an initial degree value $\mathrm{MK}(2,1) < \mathrm{D}(1)$. (We need to check for 1<2)

Second: check $\mathrm{M1}$ column 1 for entry values =1. Get their locations which point to the selected nodes in local area. In this example it corresponds to nodes (2 and 3).

Third: since $(\mathrm{MK}(2,1)=1) < (\mathrm{D}(1)=2)$ then we need to select 1 out of 2 nodes (in this example 2 or 3) and the probability of the winner node follows:

Check for condition **A**: Starting with node 2: if $\mathrm{MK}(1,2) >= (\mathrm{dth\ ex}=3)$ OR $\mathrm{ME}(1,2)$, then node 2 is eliminated. For node 3: if $\mathrm{MK}(1,3) >= (\mathrm{dth\ ex}=3)$ OR $\mathrm{ME}(1,3)<=0.1$, then node 3 is eliminated.

- If $\mathbf{A}$ is true, then $\mathrm{MK}(\ 2,\ 2\ \mathrm{to\ end}) = \mathrm{MK}(1,2\ \mathrm{to\ end})$.

If $\mathbf{A}$ is false, which means nodes are within acceptable range values of degree and energy, the probability calculations are computed as in the previous section.

Similarly, the [MK] and [ME] are updated at different time intervals such as:

$$[ME] = \begin{bmatrix} 0.8225 & 0.4058 & 0.7668 & 0.3996 & 0.7712 \\ 0.7725 & 0.4058 & 0.7168 & 0.3996 & 0.7712 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$[MK] = \begin{bmatrix} 0 & 1 & 3 & 1 & 1 \\ 1 & 1 & 4 & 1 & 1 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$[ME] = \begin{bmatrix} 0.8225 & 0.4058 & 0.7668 & 0.3996 & 0.7712 \\ 0.7725 & 0.4058 & 0.7168 & 0.3996 & 0.7712 \\ 0.7225 & 0.3558 & 0.6668 & 0.3996 & 0.7212 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$[MK] = \begin{bmatrix} 0 & 1 & 3 & 1 & 1 \\ 1 & 1 & 4 & 1 & 1 \\ 2 & 4 & 7 & 1 & 2 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

And the time varying adjacency matrix is updated such as:

$$[DAM] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Chapter 8 Results and Analysis

In this chapter, we study the performance of networks generated by both CEM and DEM models, and analyze the time evolution of the connectivity and degree distribution. In figures 8.1-8.16 we study how the factors of remaining energy distribution, number of nodes in the local area and node incoming time influence the connectivity growth of a node.

In our numerical experemints and simulation, we examine the behavior of CEM and DEM under two different distributions of remaining energy: Uniform and Normal. Additionally, we vary the parameter values of maximum degree, minimum remaining energy, and the amount of energy decrese when a node establishes a new connection. Additionally, the size of the network determined by the total number of nodes in the network is varied for different simulation runs. Therefore, different scenaros are set by varying different system values. The list of parametres are represented in table 8.1.

For a Uniform distribution, if the limits where a and b then the mean is (58):

$$\frac{1}{2(a+b)} = \bar{E} \qquad (8.1)$$

For a Normal distribution (59):

$$PDF = \frac{1}{\sigma\sqrt{(2\pi)}} e^{\frac{(x-\mu)^2}{2\sigma^2}} \qquad (8.2)$$

And the mean is: μ=$\bar{E}$ and variance is: σ^2.

Parameters
Total number of nodes in the network K
Number of nodes in the local area LA
Maximum allowable degree per node. Degree threshold dth
Remaining energy of nodes Er
Energy depletion per degree increase E
Energy reduction per degree $ered$
Energy threshold ets
Remaining energy distribution

Table 8.1: Network Parameters

1. NORMAL ENERGY DISTRIBUTION of remaining energy values and using CEM

The network starts with a small number of connected nodes. Starting at t=0, at each time step, a new node is introduced with a random and small number of edges that is less than the number of nodes in the local area. The limits of the remaining energy distribution are: 0.3 and 1 Jules. The remaining energy limits are controllable variables which mean that the user has the ability to specify the limit values. For each of the simulation runs, the values of total number of nodes K, The degree threshold dts and remaining energy threshold ets are indicated for each Figures 8.1-8.3.

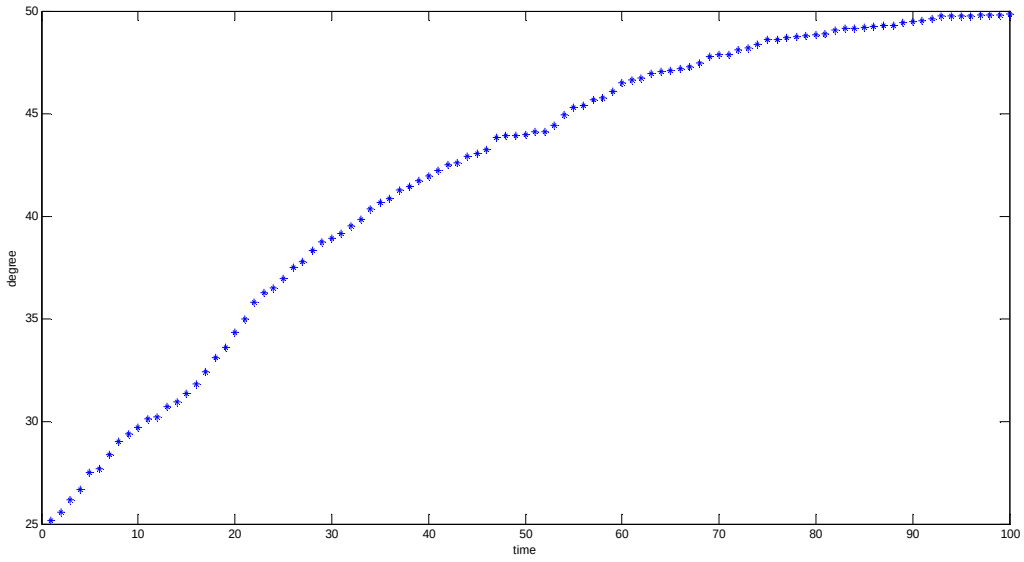

Figure 8. 1 Time evolution of connectivity vs. time when nodes remaining energy value does not decrease for (k=100, ets=0.2)

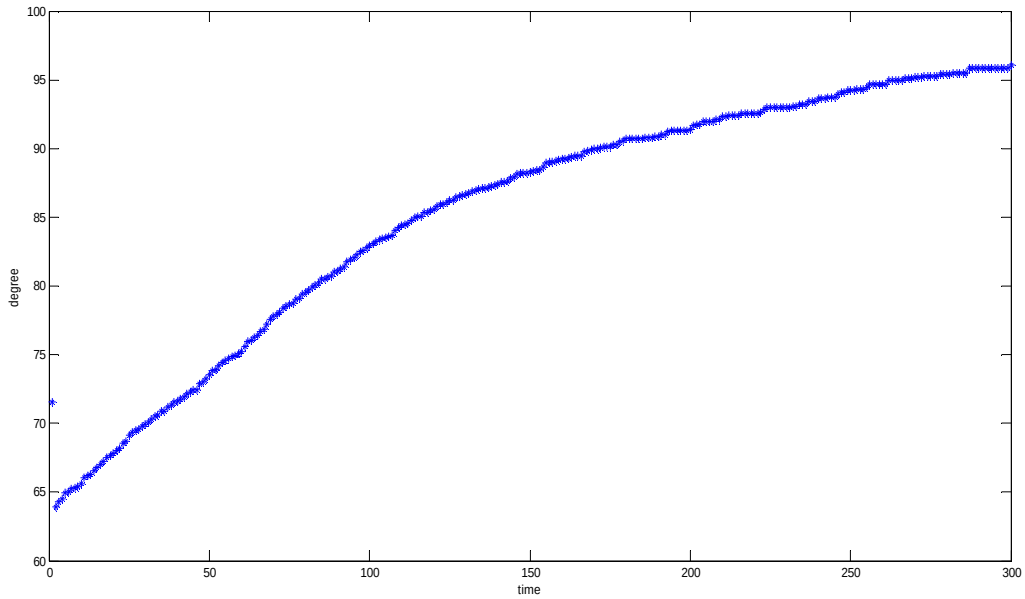

Figure 8. 2 Time evolution of connectivity vs. time when nodes remaining energy value doesn't decrease for (k=300, ets=0.2)

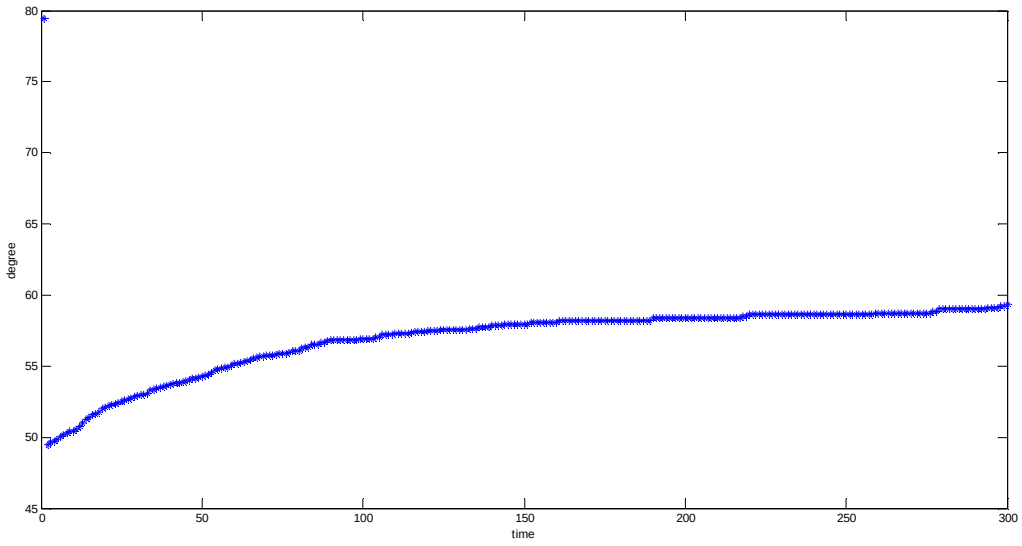

Figure 8. 3 Time evolution of connectivity vs. time when nodes remaining energy value doesn't decrease for (k=300, dts=k/5, ets=0.2)

2. UNIFORM ENERGY DISTRIBUTION of remaining energy values and using CEM

The network starts with a small number of connected nodes. Starting at t=0, at each time step, a new node is introduced with a random and small number of edges that is less than the number of nodes in the local area. The limits of the remaining energy distribution are: 0.3 and 1 Jules. The values of the total number of nodes K, the degree threshold dts and remaining energy threshold ets are indicated next to the figures 8.4 and 8.5.

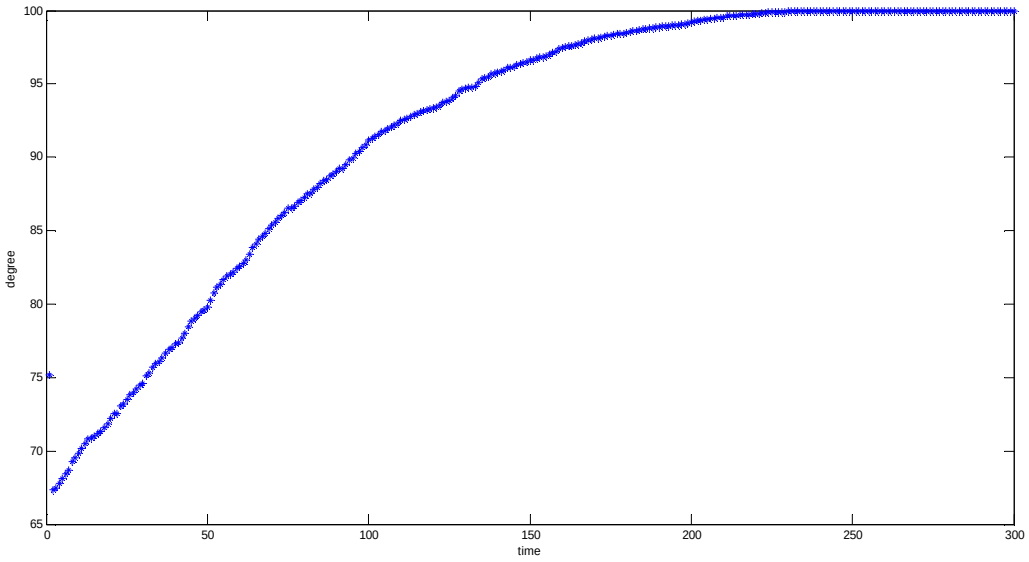

Figure 8. 4 Time evolution of connectivity vs. time when nodes remaining energy value doesn't decrease for (k=300, dts=k/3, ets=0.2, ered=0.2)

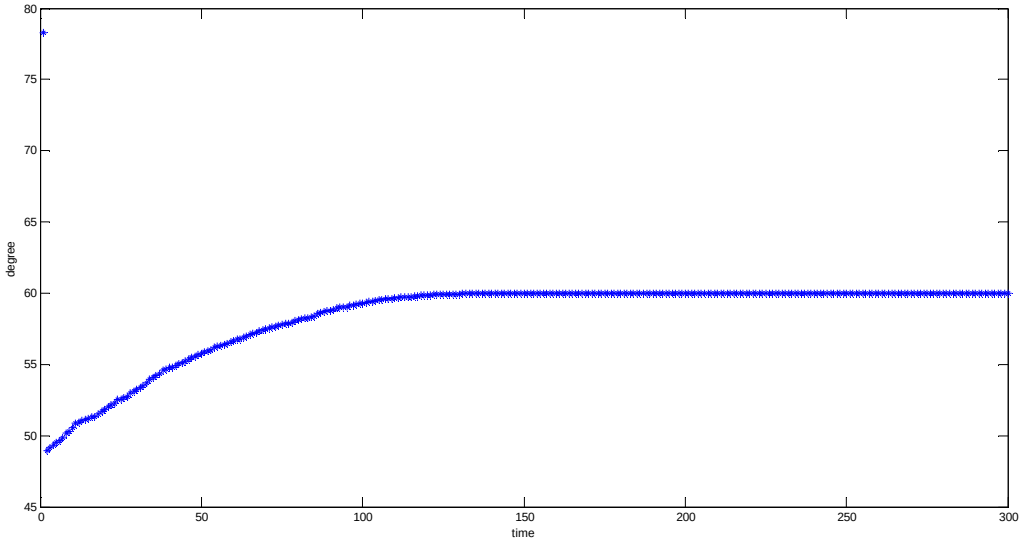

Figure 8. 5 Time evolution of connectivity vs. time when nodes remaining energy value doesn't decrease for (k=300, dts=k/5, ets=0.2)

From the results of Figures 8.1- 8.5, we can see that based on the number of nodes in the local area that is the choice of LA, the degree of a node is influenced. Clearly, if we choose a small value of local-area, a node will establish connections with a higher probability than of larger local-area value. As a result, a node with a small local-area value will increase its degree faster than a node with a higher value.

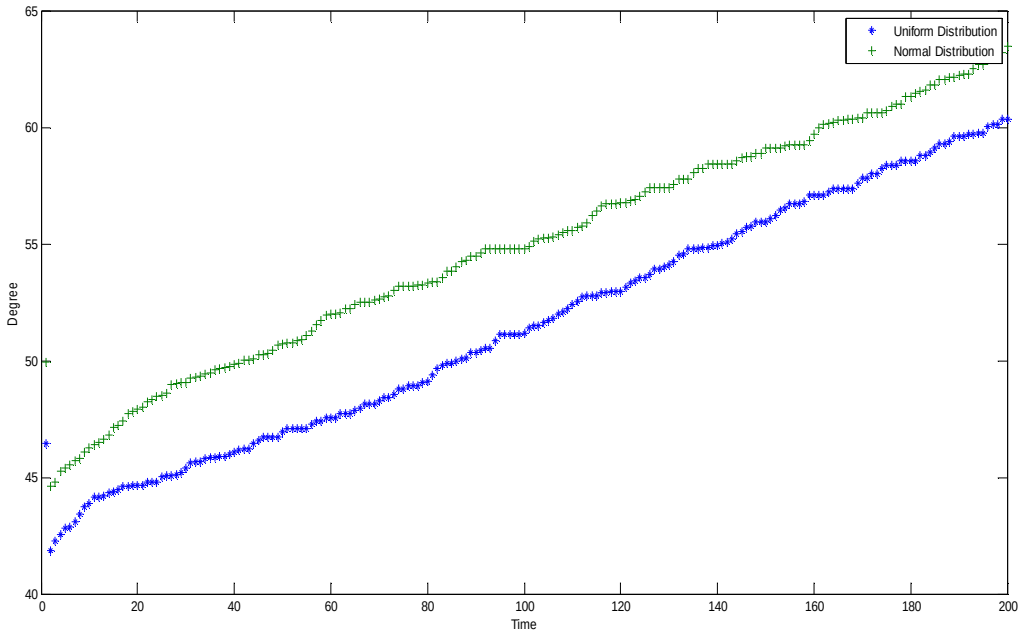

Figure 8. 6 Uniform Vs. Normal Distributions degree time evolution for (K=200, dts=K/2, ets=0.3) using CEM

Figure 8.6 shows the impact of the remaining energy distribution on the degree growth. Different expected values produce different connectivity growth. However, different energy distributions all results in a scale-free degree distribution.

3. NORMAL ENERGY DISTRIBUTION of remaining energy values and using DEM

The network starts with a small number of connected nodes. Starting at t=0, at each time step, a new node is introduced with a random and small number of edges that is less than the number of nodes in the local area.

1. Total number of nodes in the network K=300.
2. Number of time steps =300.
3. Degree threshold (dts) was used as an upper limit on the node degree which value was varied (indicated on each figure).
4. Energy threshold (ets) = 0.2.
5. Remaining energy values were assumed between 0.3 and 1 Jules.
6. Each time a node establishes a connection which is an increase in degree by 1 a node decreases energy by 0.01 Jules.

In Figures 8.7 and 8.8, we show the results of the simulation using DEM. In Figure 8.7, we can see that the remaining energy value is slightly above the threshold (0.1) and as time increases; the average remaining energy of all nodes in the network is more balanced.

Therefore, from this perspective the algorithm can produce more balanced energy results using the degree of nodes as a performance measure. In Figure 8.8, we can see the time evolution of nodes degree. Since the upper limit on node degree was set to $k/2$, the maximum node degree was below that value.

Most relevant here, the positive effect of the model where the average degree of the network increases in a balanced way as well which implies that there are no nodes consuming their energy completely as a result of rapid increase in degree. At each time step, the average degree as well as the average remaining energy of nodes, of the network can be observed from both Figures 8.7 and 8.8.

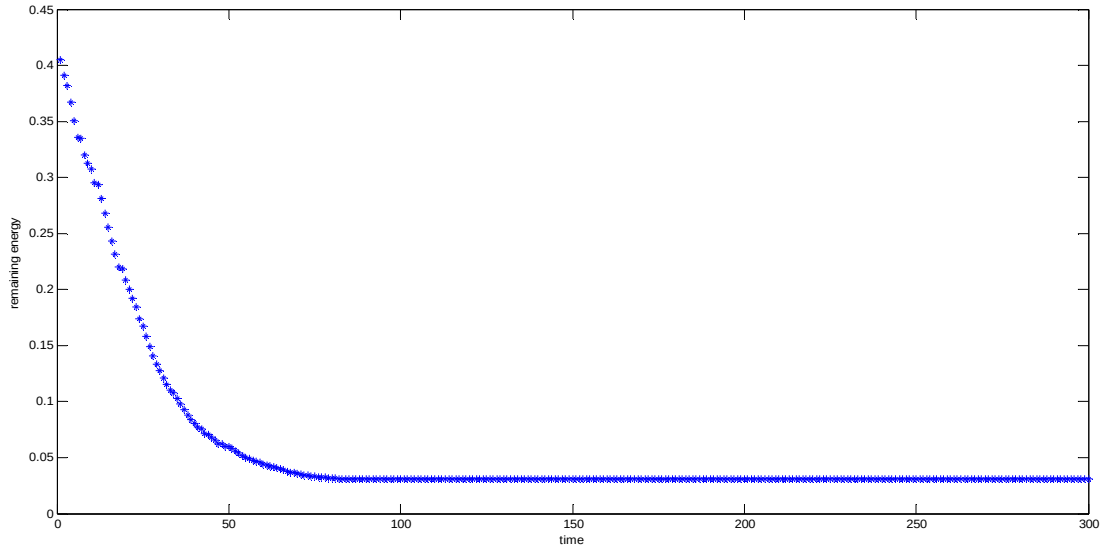

Figure 8. 7 Remaining energy of nodes over time when: k=300, dts=k/2

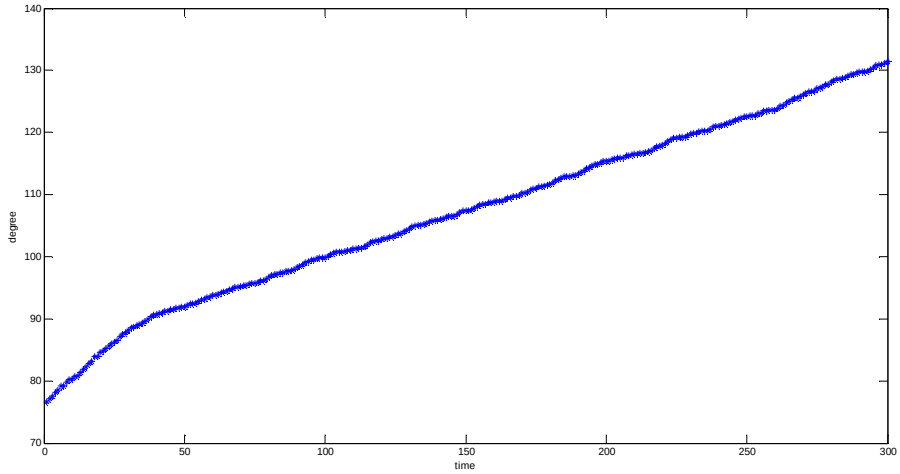

Figure 8. 8 Time evolution on the connectivity Ki(t) vs. t for: k=300, dts=k/2, ets=0.2

4. UNIFORM ENERGY DISTRIBUTION of remaining energy values and using DEM

The network starts with a small number of connected nodes. Starting with t=0, at each time step, a new node is introduced with a random and small number of edges. Below we show the parameter values used for the simulation results in this section:

1. Total number of nodes in the network K=300.
2. Number of time steps =300.
3. Degree threshold (dts) was used as an upper limit on the node degree which value is indicated on each figure.
4. Energy threshold (ets) was 0.2.

Figures 8.9 - 8.16 below shows the results of the simulation.

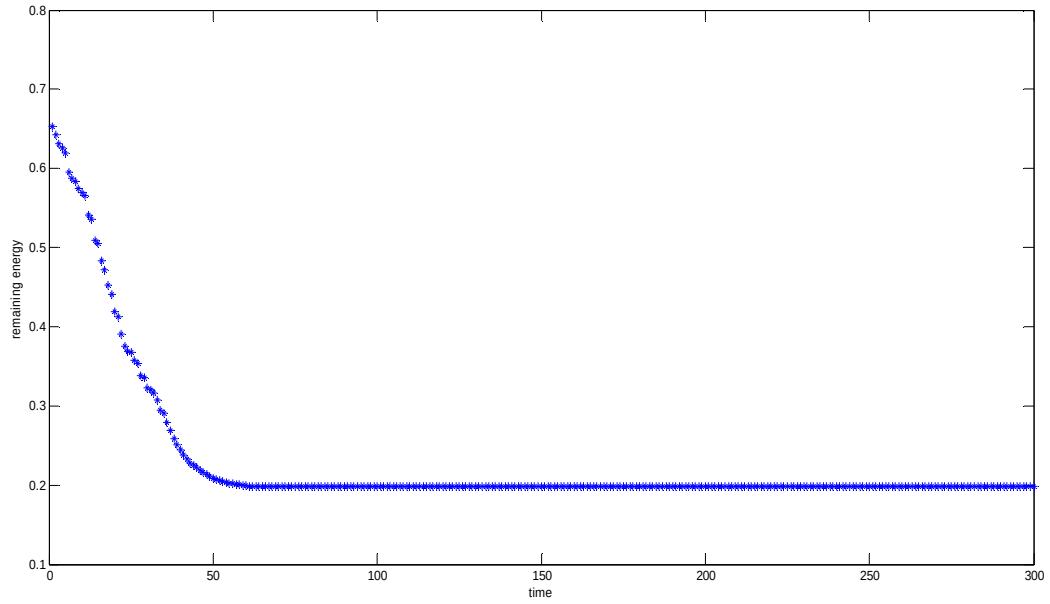

Figure 8. 9 Remaining energy of nodes over time for: k=300, dts=k/2, ets=0.2

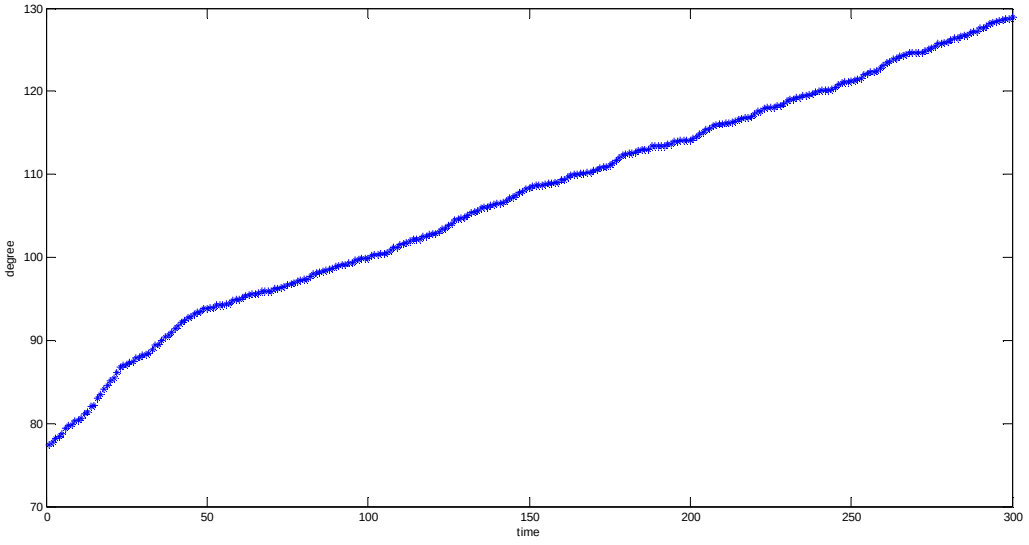

Figure 8. 10 Time evolution on the connectivity Ki(t) vs. t for: k=300, dts=k./2, ets=0.2

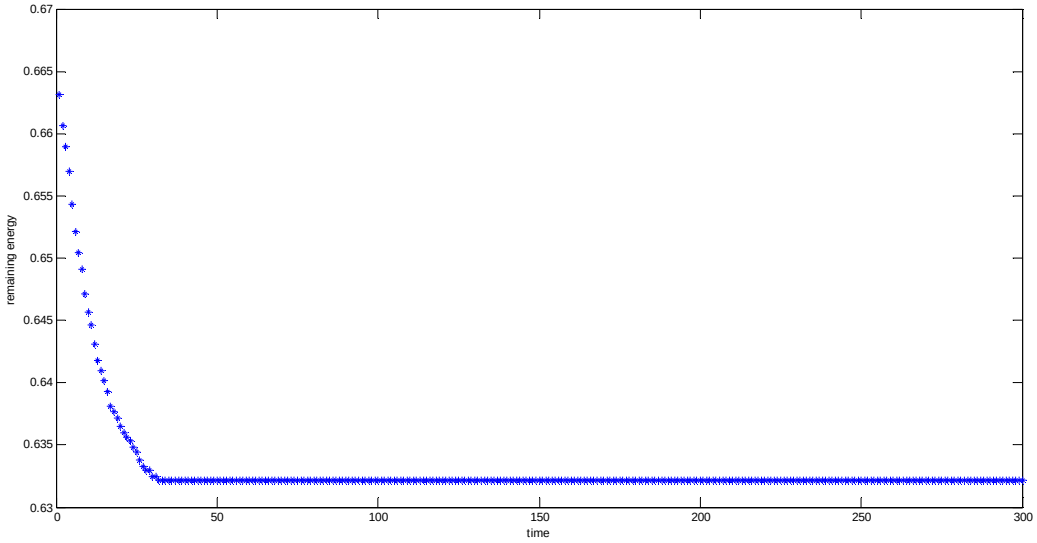

Figure 8. 11 Remaining energy of nodes over time for: k=300, dts=15, ets=0.2

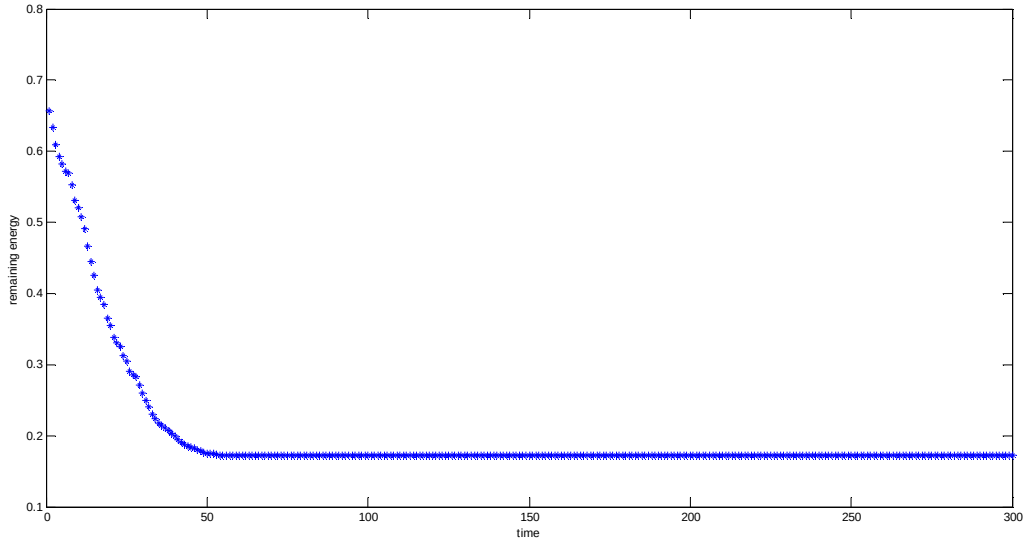

Figure 8. 12 Remaining energy of nodes over time for: ets=0.2, no limit on k

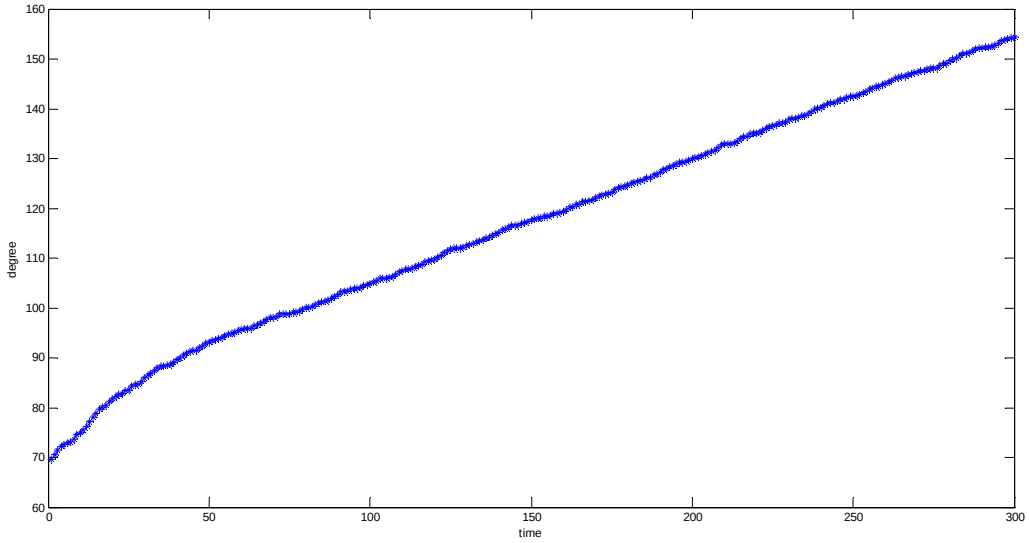

Figure 8. 13 Time evolution on the connectivity Ki(t) vs. t for: ets=0.2, and no limit on k

5. NORMAL ENERGY DISTRIBUTION of remaining energy values and using CEM

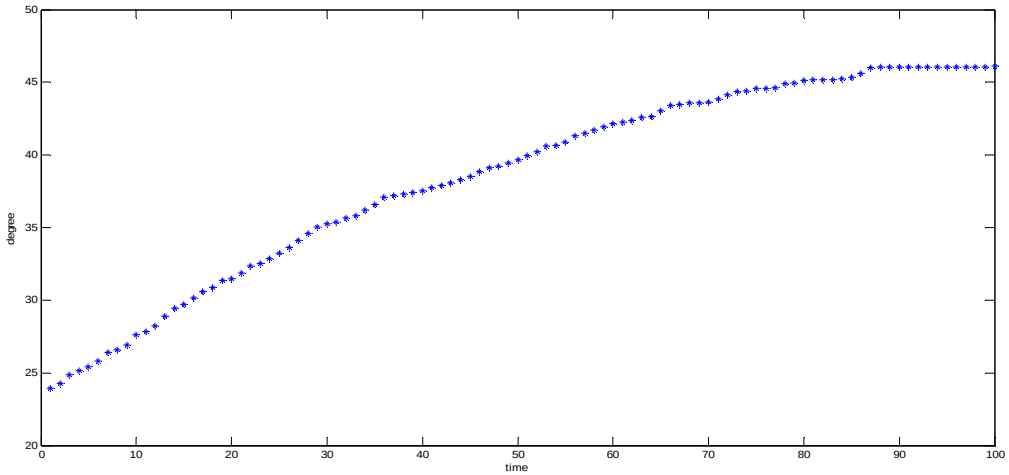

Figure 8. 14 Time evolution of connectivity vs. time when nodes remaining energy value doesn't decrease

6. Starting with a UNIFORM ENERGY DISTRIBUTION of remaining energy values and using CEM

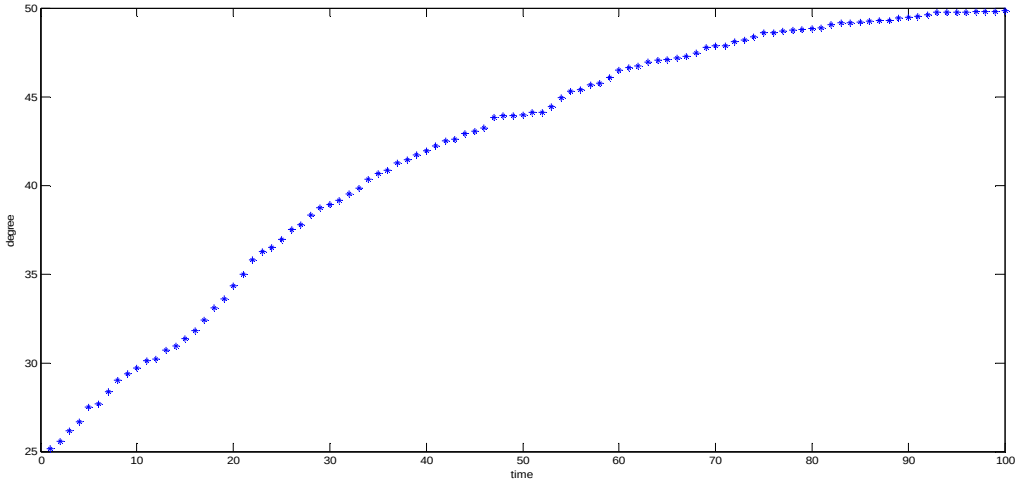

Figure 8. 15 Time evolution of connectivity vs. time when nodes remaining energy value doesn't decrease

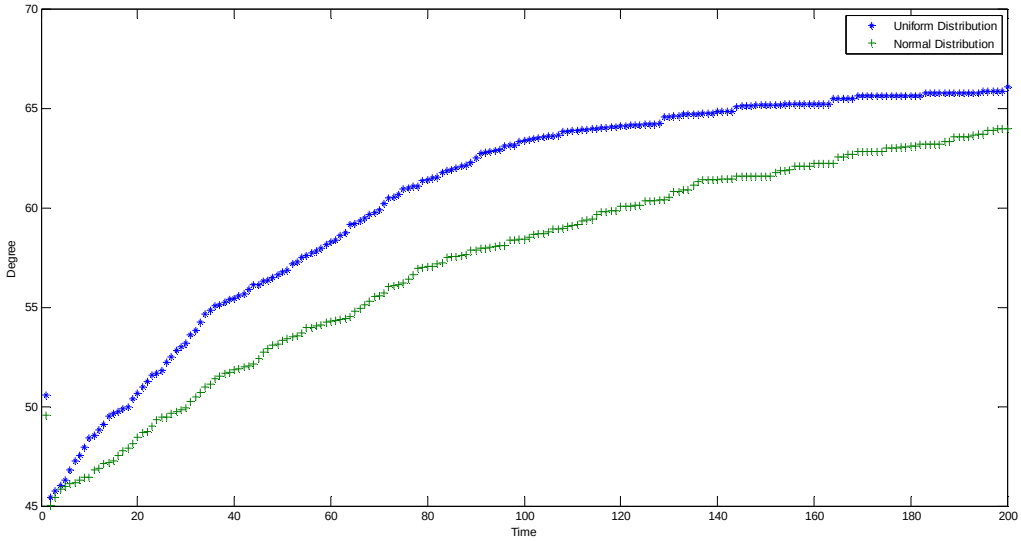

Figure 8. 16 Uniform vs. Normal distribution for (k=200, dts=k/3,ets=0.2)

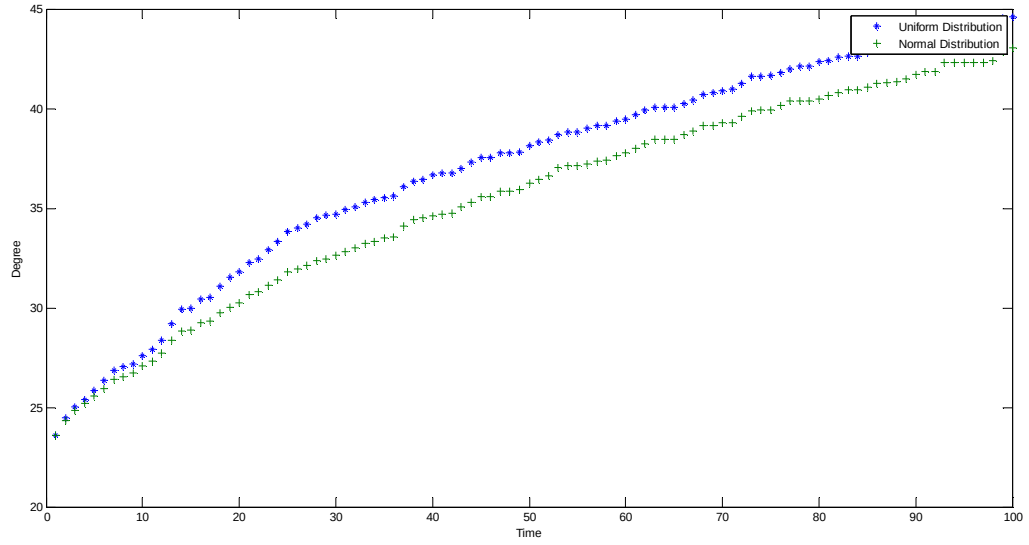

Figure 8. 17 Uniform vs. Normal distribution degree evolution over time for (k=100, dts=k/2,ets=0.2)

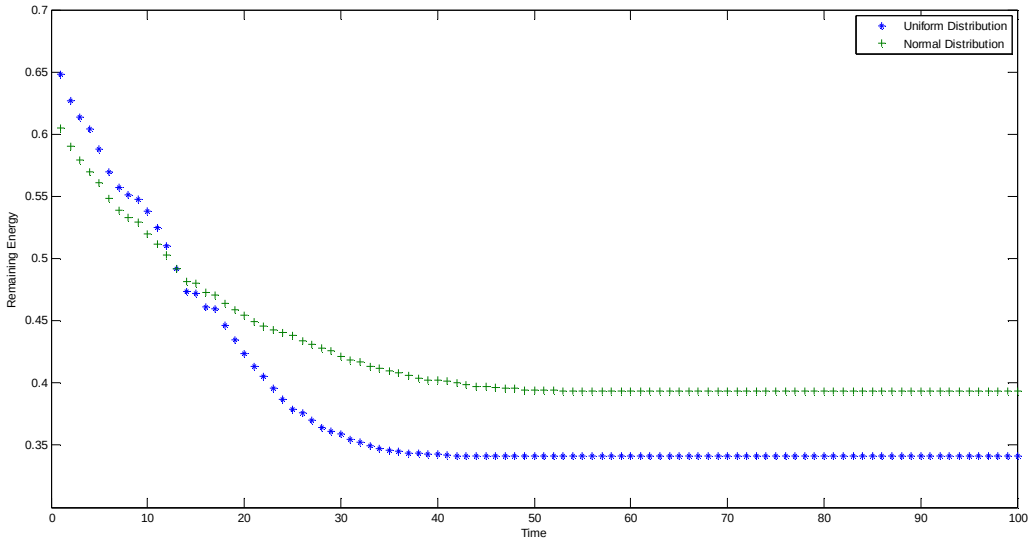

Figure 8. 18 Uniform vs. Normal distribution remaining energy over time for (k=100, dts=k/2,ets=0.2)

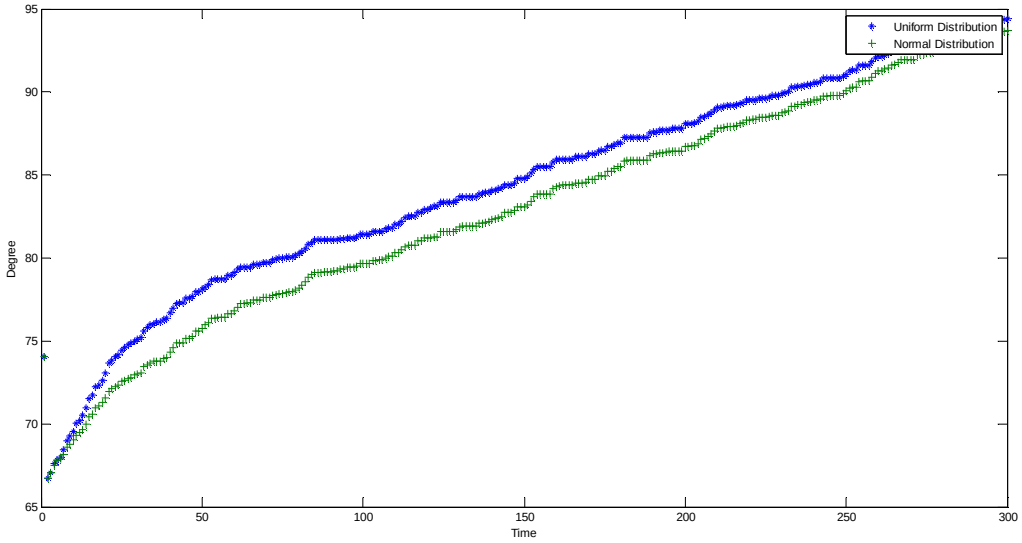

Figure 8. 19 Uniform vs. Normal distribution remaining degree evolution over time for (k=300, dts=k/2, ets=.1,ered=0.05)

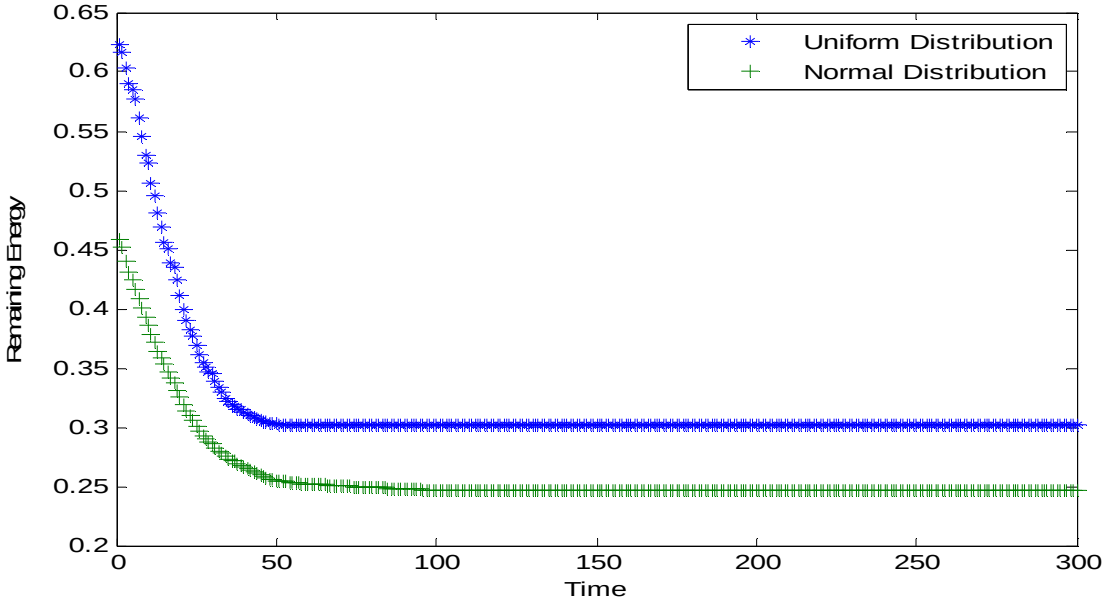

Figure 8. 20 Uniform, vs. Normal distribution remaining energy over time for (k=300, dts=k/2, ets=.1, ered=0.05)

Therefore, from the results we can see that the higher the limit on the degree k, means the smaller the maximum allowable degree, the more remaining energy of nodes. When k is increased up to the total number of nodes then the remaining energy approaches its threshold which is fine since we specify the value. And the nodes in the networks with a high degree value will end up increasing their degree slower because of the constraint on the maximum degree.

Since the maximum degree was varied, the speed at which nodes will increase in degree also depend on it. As a result, the model prevents the nodes to consume their energy unexpectedly. Further this leads to more balanced energy consumption of the whole network. The less restrictions, the maximum degree a node can take, and therefore the more the network will behave in a scale-free way. However due to power limitations of WSNs, to achieve more tolerant results against intentional attack type of error, the network must limit the high degree nodes from consuming their energy very fast.

7. Network Properties

The main focus here is to explore the network shown in figures 8.19 and 8.20, and analyze the network measures of: betweenness centrality (Figure 8.21), closeness centrality (Figure 8.22), average neighborhood connectivity (Figure 8.23) and degree distribution (Figure 8.24).

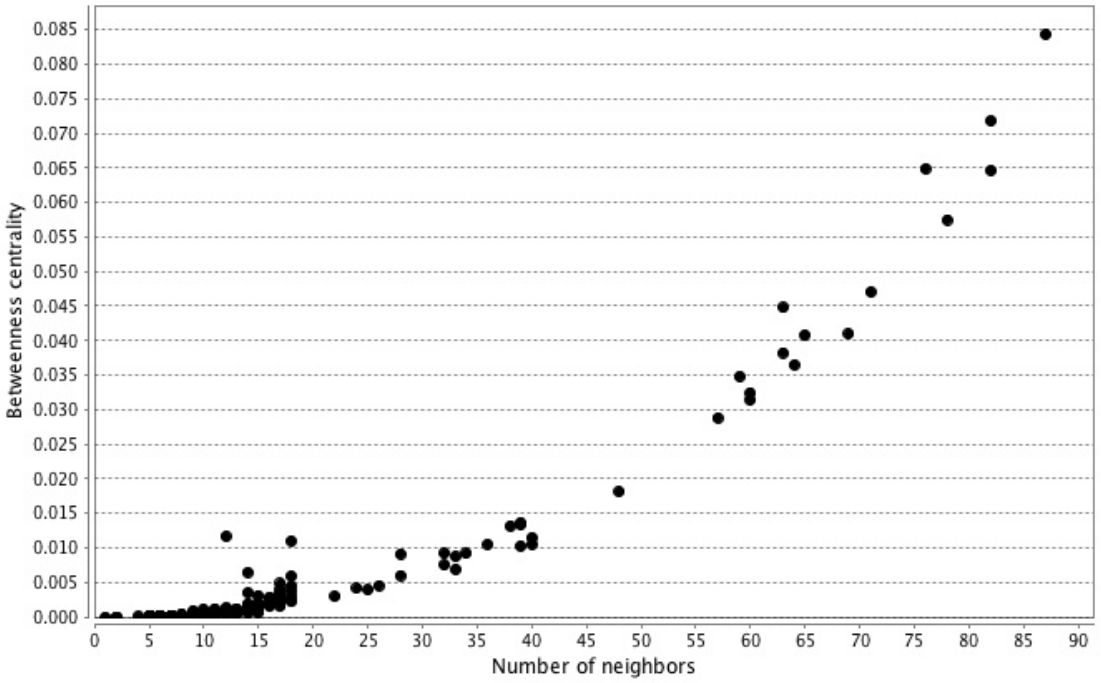

Figure 8. 21 Betweenness centrality for the network in Figures 8.19 and 8.20

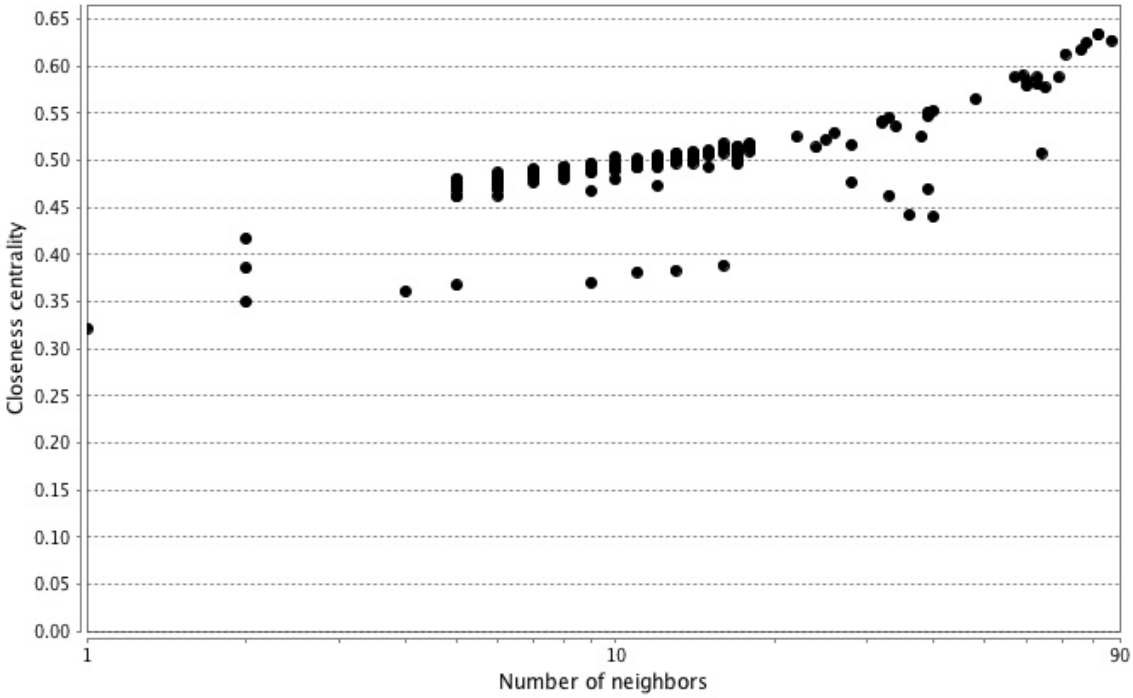

Figure 8. 22 Closeness centrality vs. number of nodes

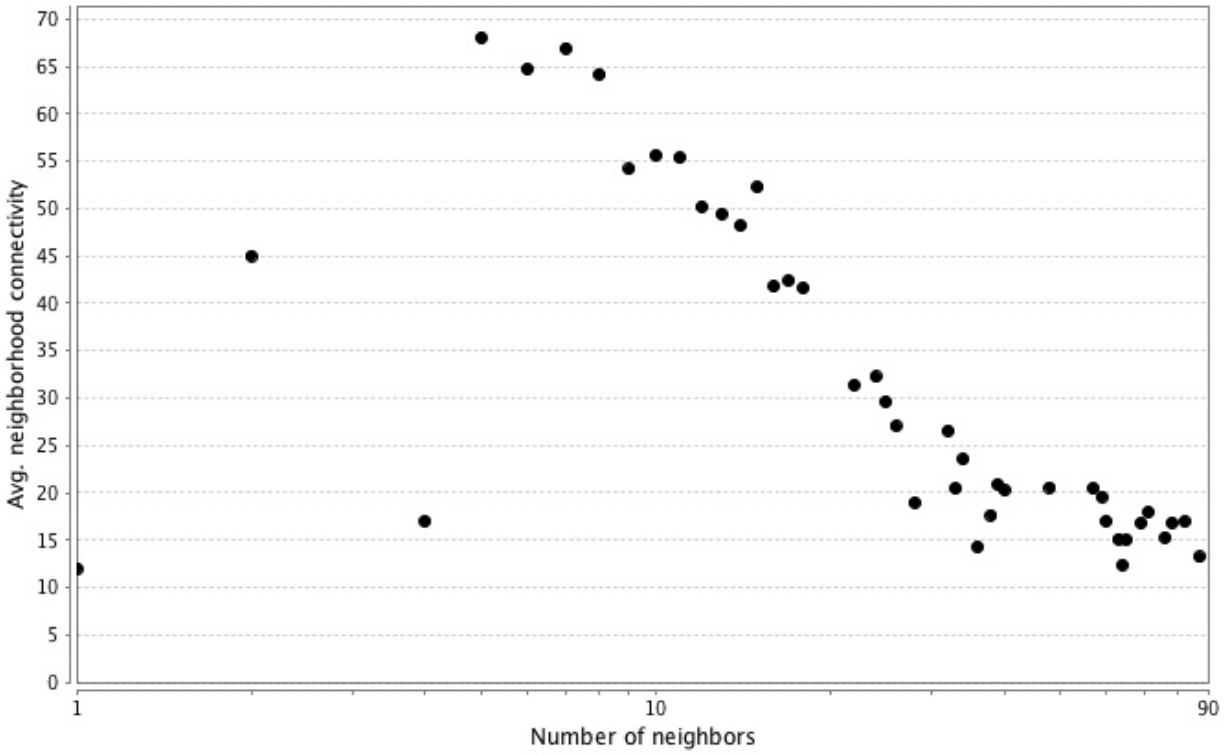

Figure 8. 23 Average neighborhood connectivity

Degree related centrality measures might be criticized because they only take into account the immediate ties that a node has, or the ties of the node's neighbors, rather than indirect ties to all others. In some cases, a single node might be tied to a large number of others, but those others might be disconnected from the network as a whole. In a case like this, the actor could be quite central, but only in a local neighborhood.

The betweenness centrality of a node is a measure of how central the node is in the network. It can be calculated as the number of shortest paths from all the nodes to all others that pass through that node. More than connectivity, the betweenness centrality is useful to measure the

load on a node. For a sensor node that is a very important measure since it can help avoid undesirable situations such as overload or congestion on a node. From Figure 8.23 we can see that only few nodes have high betweenness value which indicated higher loads on a node, whereas the majority of the nodes have lower betweennes. Since most nodes have low betweenness then we can see that the model can result in a network structure where majority of nodes have low loads. This makes the network more robust.

In general, closeness centrality methods highlight the distance of a node to all others in the network by focusing on the distance from each node to all others. Depending on how one wants to think of what it means to be "close" to others in any networked system, a number of slightly different measures can be defined. Closeness centrality is a measure of how fast information spreads from a given node to other reachable nodes in the network. From Figure 8.22 we see that nodes in general have a good coefficient value. Most nodes have few connections, yet the pattern of their direct and indirect ties allow them to access all the nodes in the network more quickly than anyone else. They have the shortest paths to all others, which mean that they are close to everyone else. They are in an excellent position to monitor the information flow in the network since they have the best visibility into what is happening in the network.

The neighborhood connectivity of a node gives the average connectivity of all his neighbors. Since the neighborhood connectivity distribution is a decreasing function in *k* (Figure 8.23),

edges between low connected and highly connected nodes prevail in the network (60). This means that sensor nodes with higher remaining energy value are connected to nodes with low remaining energy value. Most relevant here, the positive effect of this in addition to good clustering coefficient value results in increasing the network redundancy level. If a node runs out of power, the network won't become disconnected.

The degree distribution is shown in figure 8.24. As expected from the calculations described in section 6 on DEM, the degree distribution might be versatile. However, specifying initial network parameters accordingly may results in a scale-free network.

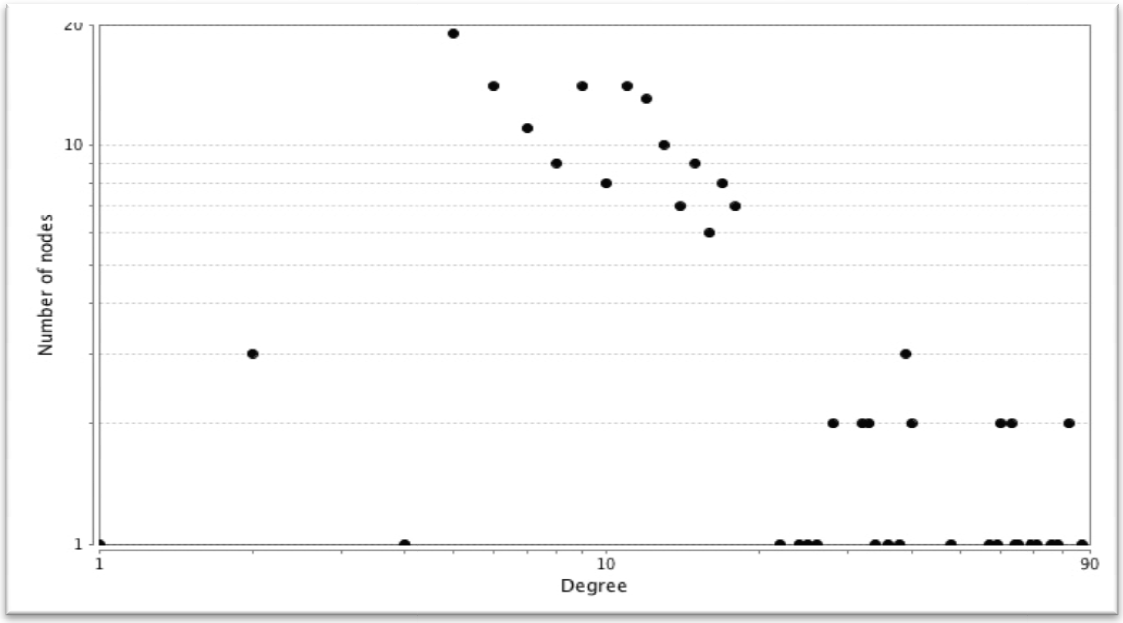

Figure 8. 24 Degree distribution

The main focus here was to explore the robustness of the resulting network from a network measure perspective so that it can tolerate failure of some sensor nodes. The centrality

measure results suggest that the network can tolerate damage to some nodes, hence will not result in disconnecting the network. This implies that the network can be designed in order to eliminate situations where there is missing information, delay in signal or congestion.

Network Visualization Example

With respect to the issue of how the degree distribution can affect the power efficiency of the WSN, we analyze the network structure that results when using DEM for 300 node-clusters. The size of each cluster is 50 which is the number of nodes that belong to the cluster. In Figure 9.25 we present a visual example of one cluster. And in Figure 8.26 we show the whole network of 300 node-clusters.

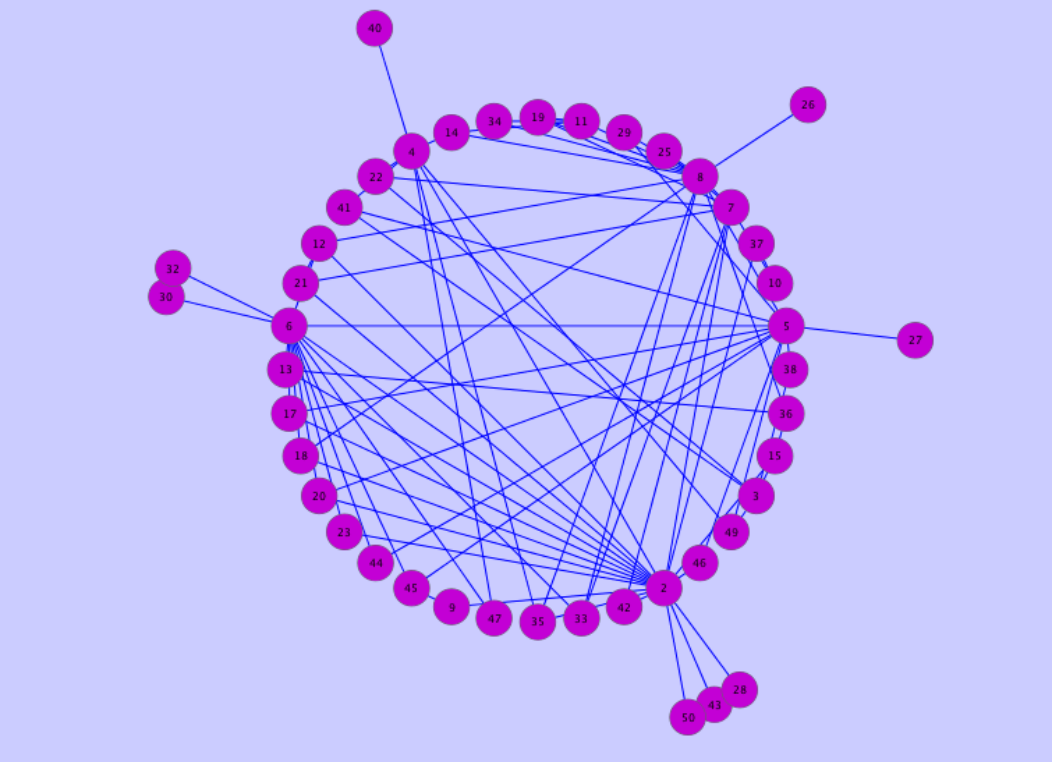

Figure 8. 25 A cluster of 50 nodes constructed using DEM

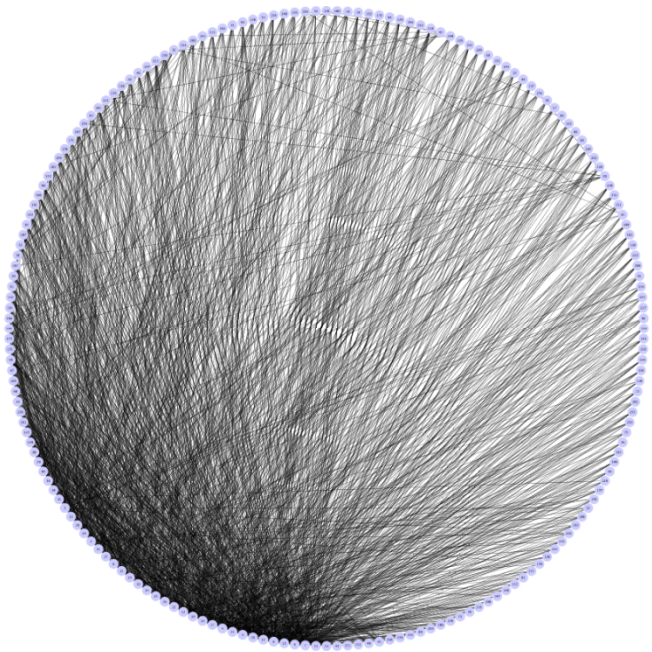

Figure 8. 26 The network of 300 clusters constructed using DEM

In this example, we study the relationship between node orientations, node degree and the network behavior in terms of energy efficiency of one cluster. We presuppose that nodes in the cluster are deployed in the environment as shown in figure 8.25. Therefore, each node occupies a position in the environment. Nodes with one link identify other clusters, or different base stations. Nodes with high degrees represent the cluster-heads. When using the proposed model, the local area of each node is predestined by the user. As we can see, the majority of nodes have a low degree, where only few nodes have a higher degree. However, even the high degree nodes such as node 2 for example, have a maximum degree of 19. The results of the simulation show that nodes have an average remaining energy value of 0.4 Jules. Two nodes had a remaining energy value of 0.1. And the average degree of the whole network was 5. The average clustering coefficient was 0.5. This indicates that the network has good clustering so that there are multiple communication paths between nodes. A higher value is not desirable for the WSN since it can result in rapid energy consumption due to increasing connections. Therefore, the network can tolerate stressful conditions such as congestion or overload on a node. This shows that the network has good results in terms of balanced remaining energy value of nodes as well as good clustering. Moreover, the network results in better redundancy leading to increased fault-tolerance ability.

The number of links between clusters is determined by the cluster state. The cluster state is defined by two values the average degree of the cluster and the average remaining energy. However, these values are dynamic and change every time step. The network shown in Figure 8.26 is an example of what can result after the evolution of each cluster when using DEM.

Therefore, the results of each cluster construction under DEM can be used to construct the network of 300 clusters. Therefore, after each cluster determines a cluster head, the links to other clusters are established.

Finally with respect to the ability to provide self-organization for the network, these results were consistent with the view that the model enables us to fine tune different system parameters based on each application requirement. The model is capable to provide adaptable parameter specifications. The time-driven event behavior was captured through the growth process, whereas data-driven events were formulated based on the topology matrix [M1] and different system parameters and the effects was observed in the preferential attachment process. Therefore, we can see the dependency between both modes of behaviors as we change parameters the two processes growth and preferential attachment are affected. Our implications suggest that the models avoid the problem of having high degrees for nodes that are old in the network. Usually in a scale-free network the older the node is in the network, the higher its degree. However, our results show that older nodes in the network do not follow that phenomenon. This avoids stressful situations such as rapid energy consumption for a sensor node. Therefore, the results show the degree is scale-free but balanced for the whole network. This indicates that the algorithm considers the energy limitations when establishing connections.

8. Analysis on Network Robustness

In this section we discuss how we can build robust WSNs from a network measure perspective when considering factors of energy efficiency, fault-tolerance and local-area of each node. Based on our findings we found that for each node the size of the local-area and the maximum degree impact the robustness of the system. Since WSNs have many limitations of range and power, using smaller sizes of local-area implies decreasing the sensor node communication range. Basically, sensor node energy should not be consumed for longer range links if a node can establish connections with neighboring nodes. Using our proposed approach our network results in good betweenness centrality distribution, which implies that most nodes had low loads. And that coefficient increases with the degree (Figure 8.21). So higher nodes had higher values, which suggest that only cluster-heads can tolerate higher loads as they are usually equipped with more powerful hardware capabilities. The closeness centrality distribution (Figure 8.22) had a good coefficient value which indicated that there was fast information flow between a node and all other nodes in the network. Therefore, using lower values for the number of nodes in the local-area i.e. lower transmission range did not impact the speed of information flow. So we can still get good results when using lower range links for the majority of nodes and limiting the long range links to nodes with higher degree as they represent cluster-heads. This agrees with our initial choice of using a hierarchical structure of the network using clusters of nodes. The relationship between network robustness, the size of the local area, energy efficiency are captured in the figures below (8.27-8.29).

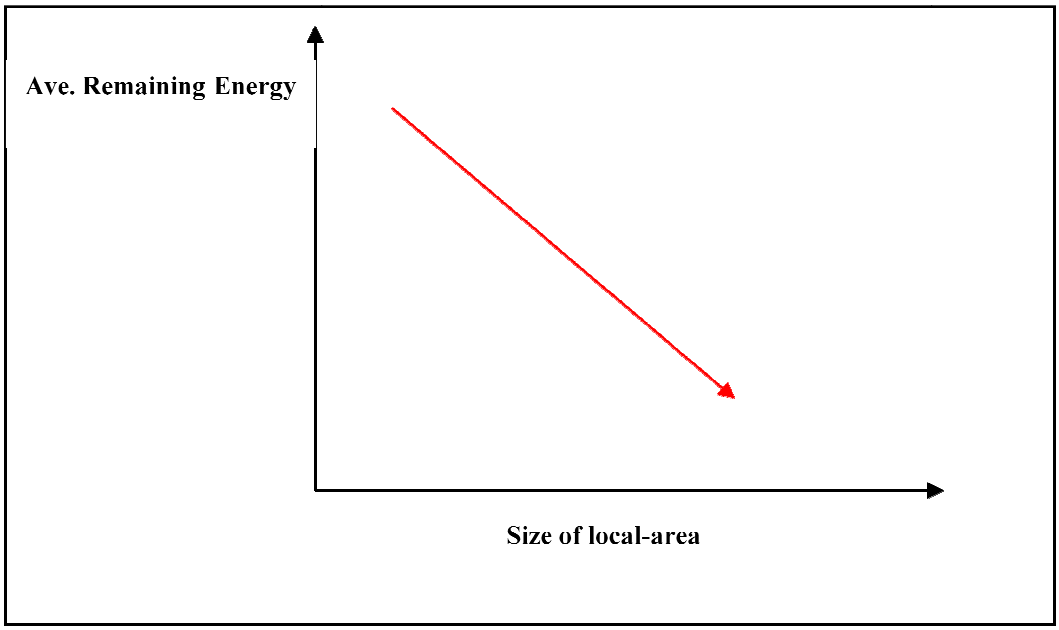

Figure 8. 27 Relationship between the local-area size and avg. remaining energy

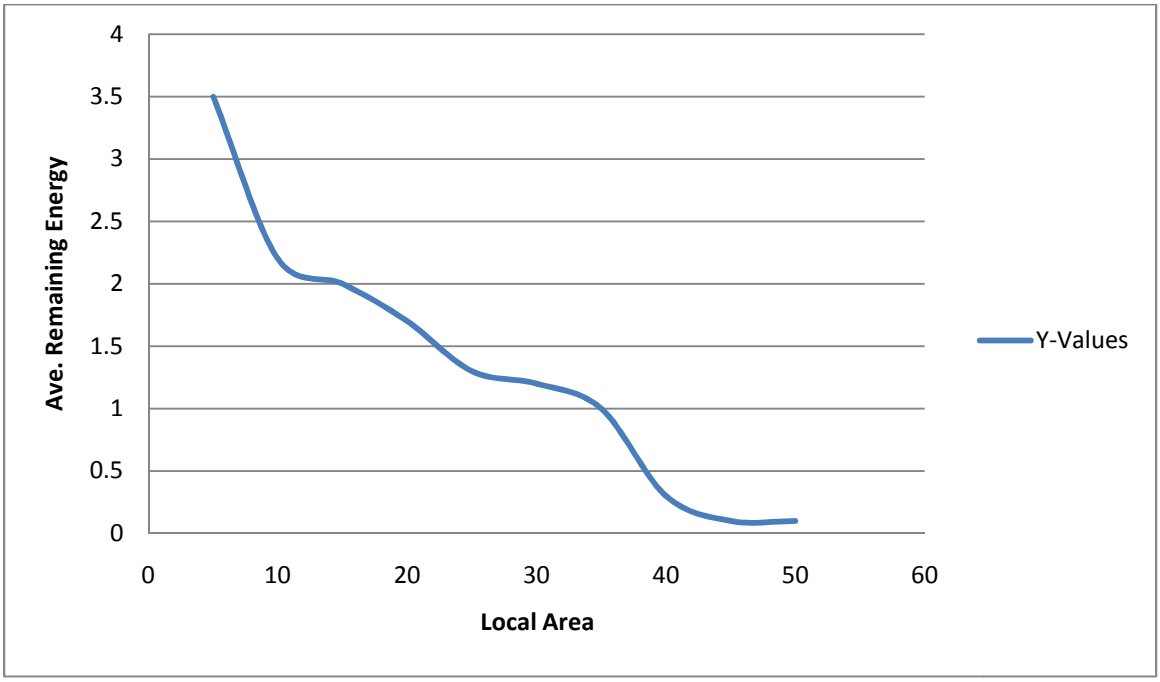

Figure 8. 28 Size of local-area vs. avg. remaining energy

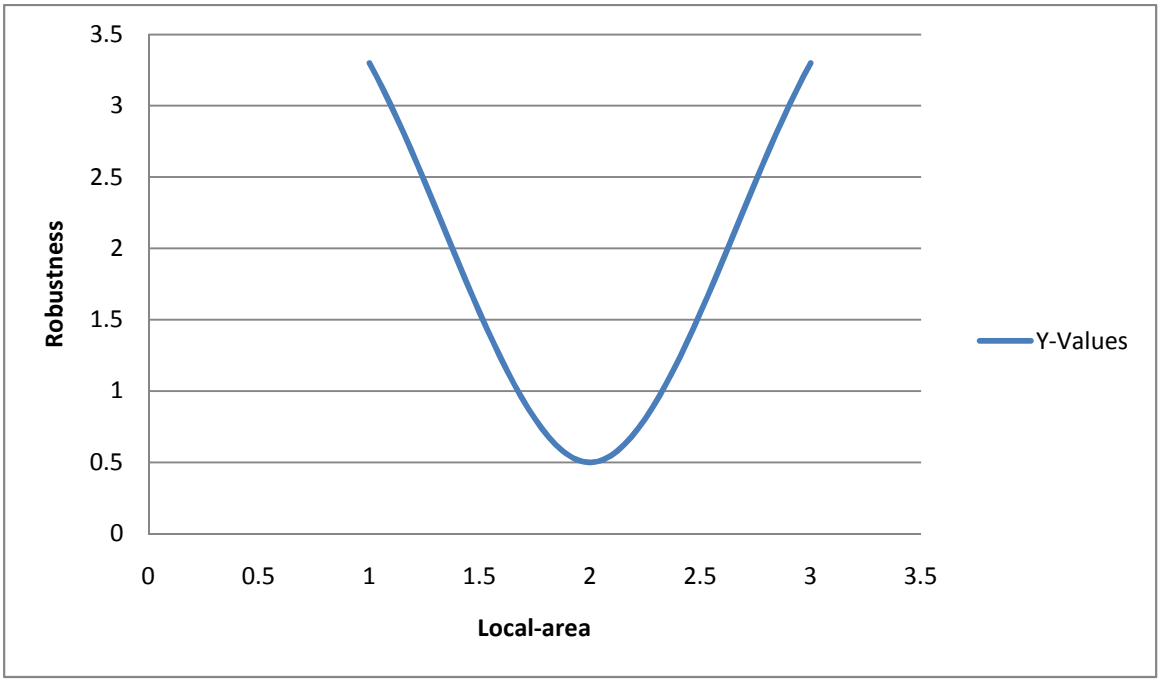

Figure 8. 29 Relationship between robustness and local-area size

We indentify critical locations in the resulting network in order to determine the vulnerability measure and further analyze the robustness. Our approach uses only the topology of the network elements without the use of attribute or labeled information. The method we use combines graph modeling with connectivity analysis and two topological graph measures: betweenness centrality and clustering coefficient values. Centrality measures are describe the structural importance of each node in the system graph. Nodes with higher centrality values have a larger impact on other nodes. The three most commonly used measures are degree, closeness and betweenness. These measures were first introduced in social network analysis (61). Closeness and degree are considered radial measures since they set properties that stem

from a given node. The degree and closeness centralities in WSNs can describe the reachability of a node in the network. In other words this measure shows how easily information can reach that node. Centrality of a given vertex in the graph can be described by the betweenness measure which can be measured based on the number of paths that pass through it. Therefore, betweenness in a WSN describes the extent to which the node is needed as a link in the chains of nodes to facilitate the spread if information in the network in order to reach the base station. If a sensor node with high betweenness is removed from the network, many flows of information are disrupted or must take longer detours. The importance of a vertex in its immediate neighborhood can be measured by the clustering coefficient. The clustering coefficient gives an estimate of how well connected is the immediate neighbourhood of a given vertex.

The framework for the identification of critical locations is based on the following procedure: Nodes that have one or more of the following properties:

1. Nodes with a high betweenness.

2. Nodes with low clustering coefficient.

We choose a high betweenness value to be $\geq .075$ and a clustering coefficient $\leq .1$. First we pick the nodes with high betweeness and assign a value of .5. Second, if the corresponding clustering coefficient values of these nodes were low then we assign a value of .5. Then a node adds both values. The results show that 5 nodes had a value of 1 which corresponds to nodes

with high degree. Therefore, from our assumptions we consider those nodes to be cluster-heads.

Chapter 9 Conclusions and Future Work

In this dissertation, we give two methods for modeling WSNs. The first model, CEM, enables nodes in a WSN to establish connections based on values of remaining energy and degree for existing nodes within its radius. Simulation shows performance results that can organize the network in an energy efficient way. The model can result in a scale-free network which have features that allow for fault- tolerance against random failure of the sensor nodes; therefore, offering more robust WSNs. In the second model, DEM, we decrease the remaining energy value of each node based on increasing the degree. Therefore, this algorithm can provide more realistic results and make the energy consumption for all the nodes in the network more balanced than the CEM.

Results show that older nodes in the network do not necessary increase their degree unexpectedly. Part of this is related to the maximum limit given on the degree and the other part is related to the minimum remaining energy threshold which prevents nodes from establishing new connections. As a result, the models take in to consideration redundancy of the network against undesirable events such as over load on a node which can lead to energy depletion.

Indeed, this makes the proposed models able to provide adaptable parameter specifications. This enable to fine tune different system parameters based on each application requirements.

The models proposed in this dissertation aim to offer a new framework towards WSN control by considering the degree distribution and evolution properties. By using the universal principles found in many natural and biological networks we hope that improved Quality of Service expectations can be achieved as a result of self-organization through a scale-free degree distribution.

Future work can look at the degree distribution of all the clusters in the network. Moreover, forming the whole network of clusters at the same time the clusters are being constructed would make the control of the network more dynamic. So the clusters and cluster heads are dynamically adapting.

In regards to maintaining the network that is formed using our approach, our suggestion would be after forming the clusters, and the network of clusters to continuously update the network state and adapt the connections accordingly. Therefore, this may lead to self-healing capabilities. This will further lead to greater self-organization results.

Further work will also explore whether our findings reflect endorsement of a specific situation. One way to test this is to manipulate attacks on the system by destroying certain percentage of nodes in order to amplify the vulnerability of the network before organizing the network under the suggested models.

References

1. *Modeling and control of complex systems.* **Petros A Ioannou, Andreas Pitsillide.** 2008.

2. **Bar-Yam, Yaneer.** *MAKING THINGS WORK: Solving Complex Problems in a Complex World.* 2005.

3. **Boccaletti, S., et al.** *Complex networks: Structure and dynamics.* s.l. : Physics Reports, February 2006. pp. 175–308. Vol. 424.

4. **Hara, Takahiro.** *Wireless Sensor Network Technologies for the Information Explosion Era.* s.l. : Springer, 2010.

5. *A Survey of Energy Conservation, Routing and Coverage in Wireless Sensor Networks.* **Wang Bin, Li Wenxin, and Li Liu.** s.l. : Active Media Technology: 7th International Conference, Amt 2011, Lanzhou, China, 2011.

6. *Large-scale Network Discovery: Design Tradeoffs in Wireless Sensor System.* **Deepak Ganesan, Bhaskar Krishnamachari, Alec Woo.**

7. *Optimization of Routing Protocol Robustness for Wireless Sensor Network.* **Bin Zeng, Jun Wei, Tao Hu.**

8. *A topology discovery algorithm for sensor networks with applications to network management.* **B. Deb, S. Bhatnagar, and B. Nath.** s.l. : Circuits and Systems, 2005. 48th Midwest Symposium, 2005, Vol. 2, pp. 1346 - 1349.

9. *Complex networks: Structure and dynamics.* **S. Boccalettia, V. Latorab,c, Y. Morenod,e, M. Chavezf , D.U. Hwang.** s.l. : Physics Reports 424, 2006, pp. 175 – 308.

10. *Error and attack tolerance of complex networks.* **R. Albert, H. Jeong and A. L. Barabasi.** s.l. : Nature, 2000, pp. 406-378.

11. **Estrada, Ernesto.** *Network Science: Complexity in Nature and Technology.* s.l. : Springer, 2010.

12. **Kazem Sohraby, Daniel Minoli, Taieb F. Znati.** *Wireless Sensor Networks: Technology, Protocols, And Applications.* s.l. : Wiley-Interscience, 2007.

13. **Haenselmann, Thomas.** *Wireless Sensor Networks: Design Principles for Scattered Systems.* s.l. : Oldenbourg Verlag, 2011.

14. **Hara, Takahiro.** *Wireless Sensor Network Technologies for the Information Explosion Era.* s.l. : Springer, 2010.

15. *Wireless Sensor Network Issues-An Overview.* **P. Vijayakumar1, and V. Ravichadran.** s.l. : International Journal of Wireless Networks and Communications, 2009, Vol. 1, pp. 17-28.

16. *A Survey of Energy-Efficient Scheduling Mechanisms in Sensor Networks.* **Xiao, Lan Wang and Yang.** 5, s.l. : Mobile Networks and Applications, 2006, Vol. 11.

17. *FUNDAMENTAL PERFORMANCE LIMITS OF WIRELESS SENSOR NETWORKS.* **ZHIHUA HU, BAOCHUN LI.**

18. **Economides, Fotis Diamantopoulos and Anastasios A.** A performance study of DSDV-based CLUSTERPOW and DSDV routing algorithms for sensor network applications. *1st International Symposium on Wireless Pervasive Computing.* 2006, pp. 1-6.

19. *A Distributed Topology Discovery Algorithm for Wireless Sensor Networks.* **Georgeff, Iavor.** s.l. : Sensing Technology. ICST. 3rd International Conference, 2008.

20. **Ferrari, Gianluigi.** *Sensor Networks: Where Theory Meets Practice.* s.l. : Springer, 2010.

21. *Special issue on wireless multimedia sensor networks Computer Networks.* s.l. : Computer Networks, 2008, Vol. 52, pp. 2529–253.

22. *A SURVEY ON POWER CONTROL ISSUES IN WIRELESS SENSOR NETWORKS .* s.l. : 4TH QUARTER 2007, VOLUME 9, NO. 4 SURVEYS IEEECOMMUNICATIONS, 2007.

23. *A survey on topology issues in wireless sensor network.* **M. Li, B. Yang.** 2006. International Conference on Wireless Networks (ICWN). pp. 503-509.

24. *M. Handy, M. Haase, and D. Timmermann.* Low energy adaptive clustering hierarchy with deterministic cluster-head selection : 4th International Workshop on Mobile and Wireless Communications Network, 2002.

25. *A performnance study of DSDV-based.* **Economides, Fotis Diamantopoulos and Anastasios A.** 2006.

26. **Daohua Yuan, Zhishu Li.** An Optimized Topology Control Algorithm for Mobile Ad Hoc Networks. *Lecture Notes in Computer Science.* s.l. : ADVANCES IN WEB-AGE INFORMATION MANAGEMENT, 2003. Vol. 2762.

27. *Power Control and Clustering in Ad Hoc Networks.* **Kuma, Vikas Kawadia and P. R.** s.l. : IEEE INFOCOM, 2003.

28. **Zeeshan Hameed Mir, Deepesh Man Shrestha, Geun-Hee Cho.** Mobility-Aware Distributed Topology Control. *ICOIN 2006, LNCS 3961, pp. 257–266.* 2006.

29. *Digital Radio shows great potential for Rail: The advantages of digital radio communication for railways are so great that the only uncertainty lies in the speed at which the various hurdles to its introduction will be overcome - Signalling.* s.l. : International Railway Journal, 2001.

30. s.l. : CBTC World Congress Presentations, Stockholm, Global Transport Forum, 2011.

31. *Topology Control in Heterogeneous Wireless Networks: Problems and Solutions.* **Hou, Ning Li and Jennifer C.** s.l. : IEEE, 2004.

32. **Kazem Sohraby, Daniel Minoli, Taieb Znati.** *Wireless Sensor Networks: Technology, Protocols, and Applications.* 2007.

33. **Shad Roundy, Paul Kenneth Wright, and Jan M. Rabaey.** *Energy Scavenging for Wireless Sensor Networks: With Special Focus on Vibrations.* 2004.

34. **Lewis Girod, Thanos Stathopoulos, Nithya Ramanathan, Jeremy Elson,Deborah Estrin, Eric Osterweil, Tom Schoellhammer.** A System for Simulation, Emulation, and Deployment of Heterogeneous Sensor Networks. s.l. : Center for Embedded Networked Sensing, University of California, Los Angeles .

35. *A Theory for Maximizing the Lifetimeof Sensor Network.* **Joseph C. Dagher, Michael W. Marcellin, Fellow, Mark A. Neifeld.** s.l. : IEEE, 2007.

36. *Lightweight Deployment-Aware Scheduling for Wireless SensorNetwork.* **KUI WU, YONG GAO, FULU LI, YANG XIAO.** s.l. : Mobile Networks and Applications, Springer Science, 2005.

37. *ASCENT: Adaptive self-configuring sensor networks topologies.* **Alberto Cerpa, Deborah Estrin.** s.l. : IEEE Transactions on Mobile Computing, 2004.

38. *A Survey on Topology Control in Wireless Sensor Networks.* **Zheng Gengzhon, Liu Qiumei.** s.l. : Second International Conference on Future Networks, 2010.

39. *Efficient Coverage and Connectivity in Wireless Sensor Networks.* **Popesc, Adrian-Daniel.** s.l. : University of Toronto, 2009.

40. *Integrated coverage and connectivity configuration in wireless sensor networks.* **Xiaorui Wang, Guoliang Xing , Yuanfang Zhang , Chenyang Lu , Robert Pless , Christopher Gill.** s.l. : In SenSys '03: Proceedings of the 1st international, 2003.

41. *Advanced Low Energy Adaptive Clustering Hierarchy.* s.l. : (IJCSE) International Journal on Computer Science and Engineering, 2010, Vol. 02, pp. 2491-2497.

42. *Scheduling sleeping nodes in high density cluster-based sensor networks.* **Jing Deng, Yunghsiang S. Han, Wendi B. Heinzelman, Pramod K. Varshney.** 6, s.l. : Mobile Networks and Applications, 2005, Vol. 10.

43. *Geographic Adaptive Fidelity and Geographic EnergyAware Routing in Ad Hoc Routin.* **Sinchan Roychowdhury, Sinchan Roychowdhury.** 2, 3, 4, s.l. : Special Issue of IJCCT, International Conference [ACCTA-2010], 2010, Vol. 1.

44. *Ethernet Topology Discovery without Network Assistance.* **Richard Black, Austin Donnelly, C´edric Fourne.** s.l. : Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP'04), 2004.

45. *HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks.* **Younis, O., Fahmy, S.** s.l. : IEEE Transactions on Mobile Computing, 2004.

46. *Collective dynamics of 'small-world' networks.* **Watts DJ, Strogatz SH.** s.l. : Nature, 1998.

47. Wikipedia. [Online] http://en.wikipedia.org/wiki/Milgram_experiment.

48. *Scale-free topology for large-scale wireless sensor networks.* s.l. : 3rd IEEE/IFIP International Conference in Central Asia, 2007.

49. *Evolution of wireless sensor networks.* **Chen Li-jun, Chen Dao-xu, Xie Li, Cao Jian-nong.** s.l. : Wireless Communications and Networking Conference. WCNC IEEE, 2007.

50. *The Reliability Performance of Wireless Sensor Networks Configured by Power-Law and other Forms of Stochastic Node Placement.* **Mikan Ishizuka, Masaki AIDA.** s.l. : IEICE TRANS, 2004, Vols. E87-B.

51. *A Scale-Free Routing Algorithm in Wireless Sensor Networks.* **Xu Ye, Wang Zhuo.** s.l. : Second International Conference on Future Generation Communication and Networking, 2008.

52. *Energy and Coverage Aware Routing Algorithm in Self Organized Sensor Networks.* **Salzmann, J., Kubisch, S., Reichenbach, F., Timmermann, D.** s.l. : Fourth International Conference on Networked Sensing Systems, 2007. INSS '07, 2007.

53. **Newman, M. E. J.** *Networks: An Introduction.* s.l. : Oxford University Press, 2010.

54. **Caldarelli, Guido.** *Scale-Free Networks: Complex Webs in Nature and Technology.* s.l. : OXFORD UNIVERSITY PRESS, 2007.

55. **David Joyner, Minh Van Nguyen, Nathann Cohen.** *Algorithmic Graph Theory.* 2009-2010.

56. *Random graphs with arbitrary degree distributions and their applications.* **Newman, Mark. E. J. and Watts, S. H. Strogatz and D. J.** s.l. : Physical Review, 2001.

57. *EVOLVING SCALE-FREE NETWORK MODEL WITH TUNABLE CLUSTERIN.* **BING WANG, ZHONGZHI ZHANG, HUANWEN TANG, ZHILONG XI.** s.l. : International Journal of Modern Physics B, 2008.

58. Wikipedia. [Online] http://en.wikipedia.org/wiki/Normal_distribution.

59. Wikipedia. [Online] http://en.wikipedia.org/wiki/Uniform_distribution_(continuous).

60. *Specificity and Stability in Topology of Protein Networks.* **Sergei Maslov, Kim Sneppen.** s.l. : Science, 2002.

61. *Centrality in social networks: Conceptual clarification.* **C, Freeman L.** s.l. : Social Networks, 1979 , Vol. 1.

62. **Jun Zheng, Abbas Jamalipour.** *Wireless sensor networks: a networking perspective.* s.l. : John Wiley & Sons, 2009.

Appendix A.

Uniform Energy Distribution with no Energy Decrease

```
clear all
% k is the number of nodes
k=300;
%degree threshold
dts=k./5;
%Energy Threshold
ets=0.2;

%Steps to create topology matrix (topo)
R1=round(rand(k,k));

A1=R1;

M1=A1-diag(diag(A1));
M1U=triu(M1,1);

topo=M1U+M1U';

%find the diagonal of M1U*M1U'
D1=sum(topo);

%degree matrix MK
```

```
MK=zeros(k,k);


for j=2:k
    MK(1,j)=randi(D1(j)-1,1);
end



%remaining energy node matrix
ME=zeros(k,k)
for j=1:k
    ME(1,j)=.3+(1-.3).*rand(1,1)
end




for t=2:k
%MK(2,1) at time = 1
MK(t,t-1)=randi(D1(t-1)-1,1)


%finding probablity for each active node
x=topo(t-1,1:k);
l=find(x==1)
P=zeros(1,k);


for i=1:k
    if x(i)==1
```

```
        %energy and degree threshold condition
        if ME(t-1,i)>ets
            if MK(t-1,i)<dts
                temp1=MK(t-1,i).*ME(t-1,i);
                temp2=(sum(MK(t-1,l).*ME(t-1,l)))-temp1;
                P(i)=temp1./temp2;
            end
        end
    end
end

temp1=P
win=[];
%choosing the winner nodes - nodes with highest energies

for i = 1: MK(t,t-1)
    if any(P)~=0
        [temp2,I]=max(temp1);
        win=[win I];
        temp1(I)=0;
    end
end

MK
win
P

%update values of winner nodes
```

```
MK(t,1:k)=MK(t,1:k)+MK(t-1,1:k);
MK(t,win)=MK(t,win)+1;

for i=1:k
    if MK(t,i)>dts
        MK(t,i)=dts;
    end
end

%update of energy
%reduction erud
ered=.2;

ME(t,1:k)=ME(t-1,1:k);
% if any(ME(t-1,:)>ets)==1
% ME(t,win)=ME(t,win)-ered;
% end



end
ME
MK
close all

figure
plot(1:k,mean(ME'),'*')
xlabel('time')
```

```
ylabel('remaining energy')


figure
plot(1:k,mean(MK'),'*')
xlabel('time')
ylabel('degree')


X=[]
for i=1:k
    X=[X; 1:k]
end


figure
surface(X,X,MK)
xlabel('time')
ylabel('node')
zlabel('degree')


figure
surface(X,X,ME)
xlabel('time')
ylabel('node')
zlabel('energy')


figure
surface(X,MK,ME)
xlabel('time')
ylabel('degree')
```

```
zlabel('energy')
%  figure
%  hist(mean(ME))
%  ylabel('frequency')
%  xlabel('remaining energy')
%
%  figure
%  hist(mean(MK))
%  ylabel('frequency')
%  xlabel('degree')
```

Appendix B

Normal Energy Distribution with no Energy Deacrease

```
clear all
% k is the number of nodes
k=300;
%degree threshold
dts=k./5;
%Energy Threshold
ets=0.2;

%Steps to create topology matrix (topo)
R1=round(rand(k,k));

A1=R1;

M1=A1-diag(diag(A1));
M1U=triu(M1,1);

topo=M1U+M1U';

%find the diagonal of M1U*M1U'
D1=sum(topo);
```

```
%degree matrix MK


MK=zeros(k,k);


for j=2:k
    MK(1,j)=randi(D1(j)-1,1);
end



%remaining energy node matrix
ME=zeros(k,k)
for j=1:k
    ME(1,j)=.3+(1-.3).*randn(1,1)
end




for t=2:k
%MK(2,1) at time = 1
MK(t,t-1)=randi(D1(t-1)-1,1)


%finding probablity for each active node
x=topo(t-1,1:k);
l=find(x==1)
P=zeros(1,k);


for i=1:k
```

```
    if x(i)==1
        %energy and degree threshold condition
        if ME(t-1,i)>ets
            if MK(t-1,i)<dts
                temp1=MK(t-1,i).*ME(t-1,i);
                temp2=(sum(MK(t-1,l).*ME(t-1,l)))-temp1;
                P(i)=temp1./temp2;
            end
        end
    end
end

temp1=P
win=[];
%choosing the winner nodes - nodes with highest energies

for i = 1: MK(t,t-1)
    if any(P)~=0
        [temp2,I]=max(temp1);
        win=[win I];
        temp1(I)=0;
    end
end

MK
win
P
```

```
%update values of winner nodes
MK(t,1:k)=MK(t,1:k)+MK(t-1,1:k);
MK(t,win)=MK(t,win)+1;

for i=1:k
    if MK(t,i)>dts
        MK(t,i)=dts;
    end
end

%update of energy
%reduction erud
ered=.05;

ME(t,1:k)=ME(t-1,1:k);
% if any(ME(t-1,:)>ets)==1
% ME(t,win)=ME(t,win)-ered;
% end



end
ME
MK
close all

figure
plot(1:k,mean(ME'),'*')
```

```
xlabel('time')
ylabel('remaining energy')

figure
plot(1:k,mean(MK'),'*')
xlabel('time')
ylabel('degree')

X=[]
for i=1:k
    X=[X; 1:k]
end

figure
surface(X,X,MK)
xlabel('time')
ylabel('node')
zlabel('degree')

figure
surface(X,X,ME)
xlabel('time')
ylabel('node')
zlabel('energy')

figure
surface(X,MK,ME)
xlabel('time')
```

```
ylabel('degree')
zlabel('energy')
%  figure
%  hist(mean(ME))
%  ylabel('frequency')
%  xlabel('remaining energy')
%
%  figure
%  hist(mean(MK))
%  ylabel('frequency')
%  xlabel('degree')
```

Appendix C

Normal Energy Distribution with Energy Decrease

```
clear all
% k is the number of nodes
k=10;
%degree threshold
dts=k./2;
%Energy Threshold
ets=0.2;


%Steps to create topology matrix (topo)
R1=round(rand(k,k));


A1=R1;


M1=A1-diag(diag(A1));
M1U=triu(M1,1);


topo=M1U+M1U';


%find the diagonal of M1U*M1U'
D1=sum(topo);


%degree matrix MK
```

```
MK=zeros(k,k);


for j=2:k
    MK(1,j)=randi(D1(j)-1,1);
end



%remaining energy node matrix
ME=zeros(k,k)
for j=1:k
    ME(1,j)=.3+(1-.3).*randn(1,1)
end




for t=2:k
%MK(2,1) at time = 1
MK(t,t-1)=randi(D1(t-1)-1,1)

%finding probablity for each active node
x=topo(t-1,1:k);
l=find(x==1)
P=zeros(1,k);

for i=1:k
    if x(i)==1
```

```
        %energy and degree threshold condition
        if ME(t-1,i)>ets
            if MK(t-1,i)<dts
                temp1=MK(t-1,i).*ME(t-1,i);
                temp2=(sum(MK(t-1,l).*ME(t-1,l)))-temp1;
                P(i)=temp1./temp2;
            end
        end
    end
end

temp1=P
win=[];
%choosing the winner nodes - nodes with highest energies

for i = 1: MK(t,t-1)
    if any(P)~=0
        [temp2,I]=max(temp1);
        win=[win I];
        temp1(I)=0;
    end
end

MK
win
P

%update values of winner nodes
```

```
MK(t,1:k)=MK(t,1:k)+MK(t-1,1:k);
MK(t,win)=MK(t,win)+1;

for i=1:k
    if MK(t,i)>dts
        MK(t,i)=dts;
    end
end

%update of energy
%reduction erud
ered=.05;

ME(t,1:k)=ME(t-1,1:k);
if any(ME(t-1,:)>ets)==1
ME(t,win)=ME(t,win)-ered;
end



end
ME
MK
close all

figure
plot(1:k,mean(ME'),'*')
xlabel('time')
```

```
ylabel('remaining energy')


figure
plot(1:k,mean(MK'),'*')
xlabel('time')
ylabel('degree')


X=[]
for i=1:k
    X=[X; 1:k]
end


figure
surface(X,X,MK)
xlabel('time')
ylabel('node')
zlabel('degree')


figure
surface(X,X,ME)
xlabel('time')
ylabel('node')
zlabel('energy')


figure
surface(X,MK,ME)
xlabel('time')
ylabel('degree')
```

```
zlabel('energy')
%  figure
%  hist(mean(ME))
%  ylabel('frequency')
%  xlabel('remaining energy')
%
%  figure
%  hist(mean(MK))
%  ylabel('frequency')
%  xlabel('degree')
```