Topology Control Mechanisms in MANETs*

BY

Joel Andres Granados Moreno
EAFIT University
{joel.granados@ieee.org}

ADVISORS

Marco Carvalho
Institute for Human & Machine Cognition
{mcarvalho@ihmc.us}

Edwin Nelson Montoya
EAFIT University
{emontoya@eafit.edu.co}

JUDGES

Roberto Hincapie
UPB University
{roberto.hincapie@upb.edu.co}

Jorge Mario Londoño
UPB University
{jorge.londono@gmail.com}

EAFIT University
Medellin - Colombia

November 27, 2006

1

"...
If you can force your heart and nerve and sinew
To serve your turn long after they are gone.
And so hold on when there is nothing in you
Except the Will which says to them 'Hold on!'
..."
Rudyard Kipling, If Poem.

Table of contents

List of figures

Glossary

Introduction

The recent mobility trend brought on by impressive potential systems that can be developed using mobile technology is changing the way applications are used. With the advent of devices that have their own power supply, contain integrated communication systems and allow for user mobility, come new applications that increase the quality of human life. The characteristics of these applications and their subsystems are crucial elements to their success in society and make part of the current field of study of many research groups.

The physical restrictions imposed by the natural qualities of wireless mobile networks make the creation of a suitable infrastructure a real headache. There are numerous challenges when handling wireless communications: The interference caused when two signals are transmitted simultaneously and in close proximity, the energy consumption of mobile devices that have limited power capability, and the lack of routing models able to cope with the dynamic and unpredictable nature of mobile networks are just some of the problems that have to be addressed.

The before mentioned challenges are a continuing field of work. To reach a feasible solution there have to be various approximations that concentrate their efforts in solving specific difficulties with efficient proposals. Topology control is one of these approximations; its objective is to manage the pattern of the elements that compose a network. Given that lots of network qualities are directly dependant on network topology, this field of study is rapidly becoming an important aspect in wireless mobile networks. From a general point of view, the problem consists in creating and maintaining an optimum topology that takes into account energy consumption and communication interference while at the same time providing maximum network performance.

XTC is an algorithm that addresses the creation of an optimal topology in static environments but does not consider its performance in long-lived dynamic networks. In other words, XTC will create a topology for statically positioned nodes. If the network has mobile devices capable of modifying the underlying order with which the topology was created, there is a need for an additional element devoted to keeping the topology up to date. The creation and testing of this additional element is the problem faced in this dissertation.

State of the art

Currently there is a great number of wireless communication technologies that are leading the way towards the future of telecommunications. These new technologies have begun to enter the market by providing new and improved products and services. Recent statistics [15] show that wireless device sales have a tendency to grow over time. This increase is due to diverse consumer trends mainly related to the specific functionality of mobile devices and can also be tied to the increasing number of available wireless technologies. The wireless market is growing and is innovating constantly to address new needs.

Current mobile devices have an array of functionalities and services that make them very attractive for the end user. The possibility to access Internet resources, intercommunication between devices that allow file sharing as well as game interactions and secondary functionalities like taking pictures and listening to music are some elements that are putting massive quantities of wireless devices in the hands of consumers. Considering these functionalities there is bound to be an increase in the mobile device market. According to [8], the number of cellular users has been doubling every one and a half years, with 23 million in 1999 to 860 million users in 2002. It also mentions that if the current rate of growth continues, the total number of mobile device users will exceed the total number of fixed-line users.

There are some aspects that need to be addressed to continue with the velocity of growth that the wireless market has experienced in the last two decades. The objective is to combine a group of elements that will make the wireless market grow, not only in terms of number of users but also in terms of type of services. The wireless market has to create new technologies that directly address user needs to maintain or increase its rate of growth.

Diverse wireless technologies are constantly emerging to address consumer needs. Personal Area Networks (PANs) basically address short range communications for cellular phones, PDAs and personal computers where the main goal is to have a network with a range of maximum 10 meters around a person. Networks like Blue-Tooth and Zig-Bee are currently being used for this purpose. Wireless LANs (WLANs) are used for medium range communication of mobile devices like personal computers and PDAs. The main goal is to allow a user to move in a certain geographical area covered by the WLAN signal while at the same time having connectivity through the main network. Networks like WiFi are used in this area. Finally there is the Wireless WAN approach (WWAN) that covers large areas and allows for a high performance connections. WiMAX would be the emerging technology for WWAN.

To give the end user a robust portfolio of services, the before mentioned technologies have to be integrated in such a way that they present a seamless network to the end user [8]. The current state of these wireless technologies and the relation that they have with their wired cousins tend to indicate that the network of the future will be composed of a high speed backbone with peripheral LANs attached to it [8]. Moreover, structures like WANs and PANs will extend the scope of the network to the mobile user [8]. To this effect, wireless technologies have to work towards a shared objective instead of an individual effort like they have been doing since they first appeared.

It is common to suppose that there is an underlying backbone network, like that of the cellular communication system, that will support wireless communication. In some situations this assumption can be inaccurate. In wild life exploration, war like endeavors or emergency situations there is a clear lack of a communication backbone. It is in these cases where a different type of network is needed. A self-organizing, self healing, autonomous network is needed to create "on the spot" solutions for these type of communication problems. Moreover, this type of network could serve as an option to some more mundane communication necessities, like office meeting or entertainment communication infrastructures. In any case Mobile Ad-hoc NETworks (MANETs) can play an important part in the wireless technology integration for the advancement of future communication systems.

Characteristics of Mobile Ad-hoc NETwork

MANET is a self-organizing (ad-hoc) network that is made up by mobile nodes that communicate by using multi-hop wireless links. These networks start off as structure-less entities that order themselves into a network like structure.

Dynamic Topology

One of the most important characteristics of MANETs is the relative movement of the nodes with respect to their neighbors. In other words, a node's neighbors have a high probability of having a different velocity and direction at any given time. Because of this characteristic the network topology is in constant change. Routing algorithms that depend on the topology have to constantly change their representations to keep up. This characteristic not only renders a huge challenge for routing protocols, but for all other network systems the variability can get to a point in which a situation can arise where there is a group of nodes that is totally disconnected from the main network. Depending on the type of mobility of the network, these islands can present themselves very frequently.

Heterogeneous Transmission Range

Heterogeneity regarding range of transmission is an important factor in determining the type of link in a network. It is very likely to encounter unidirectional links if transmission range is not homogeneous. If node u that has a greater transmission capacity than node v, then node u will be able to communicate with node v, but node v will not be able to communicate with node u. This becomes a problem when trying to, for example, return acknowledge messages (ACKs).

Multi-hop Communication

Networks usually span throughout geographical areas that are greater than that of most mobile node communication ranges. If node communication is to be possible, each device has to take on three simultaneous roles: Information generator, information receiver and information forwarder. In other words, each network node will be a host and a router at the same time. These are generally called multi-hop networks due to the number of hops a messages has to go through to get to its destination. Multi-hop networks are necessary because of the restricted range of transmission and their ability to work around interfering structures. Two nodes might not be able to communicate directly because of an interfering structure. This can be handled with a multihop network where the transmitting node sends the message to the destiny node through an intermediate group of nodes, thus avoiding the obstacle. There are additional considerations with respect to MANET multi-hop links: communication depends on the intermediate participants. Any movement or change of state of any of the intermediate nodes has the potential to break the multi-hop link. This adds to the fragility of multi-hop links and to the general chaos of the network.

Energy Constraints

Originating from various MANET characteristics, node size needs to be reduced. As a result, space is limited for the systems of a mobile node (MN). This is most relevant to the energy unit due to the difficulty of developing a small battery with high energy storing capacity. Therefore the energy that the nodes use for their computational and communication activities comes usually from a small coin battery or in the worst case a small solar panel. Basically the energy available is very limited for any activity inside a MANET device. All the supporting systems and the applications are forced to implement energy saving policies in search of a longer lasting network. Additionally, energy consumption is made even more critical because the nodes not only act as information generators and receivers but as information forwarders. This means that besides the traffic that is generated by them or for them, nodes have to handle traffic for the rest of the network. This directly translates to an increment in energy use [2].

Ad-hoc Characteristic

MANETs are by definition self-organized. Cellular wireless networks have highly configured sections that serve as the backbone to the whole system. Base Stations (BS) and other elements centralize the communications activities while wireless links are only used in the last mile. Ad-hoc network don't have the same advantage. Due to mobility and constant changing topology, the network has to self-organize into new structures that respond to rate of change.

MANET Functionalities

What are MANETs for? Given their specific characteristics MANETs can handle diverse and hostile situations. Possible scenarios consist of office information interchange, communication infrastructure for emergency and tactical purposes as well as educational and home uses. In general, MANETs are used where communication infrastructure is either not needed or not available and mobility is necessary or crucial in the specific situation in which the network is being used.

Office

A busyness meeting is a good example of a situation where no infrastructure is needed. A meeting usually consists of a small group of people that gather to discuss a certain subject and requires the assistants to share data or information. No outside communication infrastructure should be used if the meeting only concerns people in a reduced area, like an office. It would be easier to use an auto-configurable network among the devices present in the meeting so information can be exchanged faster and more effectively? Moreover, and assuming that the information being shared is classified, would it not be more secure to use a small auto-configured network that avoids passing through a larger communication structure that might compromise the integrity of the information? The basic needs of a meeting in terms of networks are very clear: A temporary MANET that is deployed at the moment the meeting begins and auto-destructs at its end. This network would have to maintain communication between all the devices present throughout the duration of the meeting. It would also need to be secure where all the devices present are authorized and the possibility of gathering information by intercepting the signal is reduced. The ability to auto-configure, in this case, has to be stretched to the maximum due to the non-technical type users. Businessmen can not be expected to spend 15 minutes of their time configuring a network. This has to be done automatically by the network in the least amount of time. The auto-configuring phase must use standards to interconnect all types of devices. Gathering all the before mentioned aspects, the meeting MANET would render a communication tool that is secure, fast and easy to use.

Emergency

An emergency is a prime example of a situation where no communication is available and is desperately needed. In an emergency situation information sharing is crucial. The current state of a person's vital functions, the structural integrity of a building that is about to collapse, and the state of hazardous and possibly life-threatening material are just some of the types of information that would be transmitted throughout the network. Although an emergency can occur inside an area that already has a functioning communication infrastructure, it may be compromised due to the emergency itself. The collapse of a building, a massive car accident or any situation where human life is threatened would require a massive use of a communication infrastructure. It will most likely have to endure a high quantity of traffic and frantic emergency messages. In this case a fast-deployable, auto-configurable and trustworthy network like a MANET is needed to address the communication of the different parties involved in the emergency.

There is also a possibility for emergencies to occur where there is a lack of infrastructure altogether. In developing countries, cell phone services are available in large populated areas but not used in small communities. Moreover, the small towns that are located in hard to reach places make for the construction of a temporary communication infrastructure a real hassle. Added to this, is the fact that the emergencies in these areas are mostly natural disasters where building a network becomes even more difficult. Consider for example a very large land slide in a rural area. The only cell Phone base station that existed was damaged beyond repair by the land slide. All the town is covered in mud making any kind of physical work almost impossible. There is a considerable probability that there are survivors throughout the town and there are lots of sensor devices that can aid in finding the last few remaining inhabitants, there are also rescue teams that need to communicate (voice, data, video ...) among themselves to manage the rescue effort. So what characteristics does the network need to have? The network has to be deployed and running in a matter of minutes, this includes the positioning of network devices and network configuration. Given that the devices are exposed to extreme environmental changes they need to withstand great amounts of stress from the natural surrounding and last for the complete length of the rescue effort. All things considered the network should be resilient to environmental change, easily deployable and long

lasting. MANETs are the type of networks that address these necessities and are most suited to be used in these precarious situations.

Tactical Networks

In the same way that MANETs can have an effect on emergency situations, they can serve as a important asset in war endeavors. Historically MANETs were first studied as a tool to improve tactics and survivability [8]. From the beginning these networks have been related to the United States Department of Defense (DoD). Given both the unpredictability of terrain characteristics and lack of optimum communication platforms in tactical activities, the DoD has researched ways of dealing with these aspects for a long time. The Dynamic nature of war is a perfect place to use MANETs and visualize their full potential.

The need to monitor massive amounts of terrain with inexpensive small devices that have a high tolerance is a common requirement for any military activity. These devices must be easily deployable, highly tolerant and preferably invisible to the enemy. The objective is to have a solid multipurpose communication platform that allows monitoring of the enemy. MANETs can also aid the communication between ground troops by addressing the line of site (LOS) problem. This problem arises when two devices can not communicate with each other because they don't have a direct LOS. In this case other MANET devices that are located on the field can serve as information forwarding routers to allow communication. In most tactical and monitoring operations where network longevity is at best unpredictable, MANET devices must make their best effort to provide longevity to the network. In tactical networks where communications might be undertaken in enemy territory or where there is a possibility of an enemy getting a hold of one or several devices, it is critical to enforce security policies to assure that information is handled by authorized personnel. For tactical situations in general a device has to have four main characteristics. It has to be highly tolerable to physical punishment by the environment as well as by the soldier that carries the device. It also needs to be highly adaptable to different tactical situations that might arise in the field. It must be capable of enduring long periods of time in active mode generating or forwarding important information. And it should use a policy driven framework that handles the information securely.

Monitoring

Although some monitoring activities have been mentioned in the emergency and tactical networks sub-sections there are more specific monitoring capabilities to describe. [22] has a list of projects that use MANET-like networks to achieve their monitoring objectives. Two projects are mentioned for their mobility characteristics.

[22] CORIE is an Environment Observation and Forecasting System (EOFS) that was deployed in the Columbia river estuary. Throughout the river's shores the researchers located static communication posts that served as information gathering devices. Sensors were released into the river on-board floating devices, the sensors then transmitted information to the onshore devices that, in turn, forwarded it to a central server. There was a challenge with sensor communication, in some instances the line of sight was obstructed by the river waves and the communication was compromised.

[22] also mentions the possibility of using MANETs for health monitoring systems. The monitoring of human physiological data, tracking doctors and patients and managing the administration of drugs in a hospital are just some of the possible uses that [22] describes. The paper also comments on the outcome of embedding devices in the human body. It specifies that the devices must be very small, safe, reliable and maintenance free. These are situations in which MANETs can be used as the main communication infrastructure given its autonomous, self-organizing qualities.

Monitoring the social behavior of a group of animals is another application in which MANETs can have a significant role. Assume a pack of wild dogs where one of them carries a satellite link and the rest of the pack carries small measuring devices spread throughout their bodies. Each dog can have several sensors used to measure temperature, blood pressure and levels of chemical substances. The objective is to use the self-configuring and autonomous qualities of MANETs to manage a network that forwards all the sensed information through the individual carrying the satellite collar. In this way there is a large quantity of information gathered with the least amount on interference for the animals.

Ubiquitous & Pervasive Computing

Ubiquitous and pervasive computing is the future of computer and communication technologies. These approaches will allow a world where there are no heavy, annoying, user-unfriendly contraptions specifically designed to show people the information they need. It will be a world where information is present in all the objects that a person handles. It will be a world where information seeks the owner and not the other way around. It will be a world where you can get the daily news from the coffee table of the shop around the street and pay for the "news paper" in the same way. It will also be a world of uncountable challenges for the computer and communication communities.

Basically, ubiquitous computing consists of having computational devices everywhere. In the office machines would be located in chairs, tables, walls, light fixtures. So acquiring or generating information could be done from any place in the company. At home all the consumer electronics would have an array of actuators and sensors to respond to any signal from their environment. It would be possible to turn off the stove by sending an e-mail, or by phoning your house or just by saying "turn off stove" as your leave your home. A person could play a game of chess on any shop table in the city, and, if he or she were to remember that an e-mail needed to be sent, it could be done without having to move from the shop table.

Communication between the thousands of devices that might be present in a given area can be a titanic task to undertake. Not only is there a great number of devices present, but most of them are moving and communicating with their static counterparts. An unpredicted increase in the number of devices can occur at any time and the network must adapt seamlessly to the change. MANETs can address this and other challenges and will become an important part of future networking.

A pervasive system is one that blends in with its surroundings. Its a system that is present, but is camouflaged to hide its facade. Presently computational devices are non-pervasive, actually its quite invasive. An office computer is big and it is immediately noticeable. A mobile phone rings interrupting whatever a person is doing. A portable computer has to be carried around, taken care of, plugged in when the batteries fail and protected against thieves. People ultimately modify their behavior to own a portable computer. With pervasive computing, one would not be carrying the machine that facilitates the management of information. With pervasive computing the means to manage information "follow" the user and fit seamlessly in to the environment. Pervasiveness is not only about user interfaces, it also addresses the amount of effort needed to use a technology. When a technology requires large amounts of configuration by the user, it becomes invasive instead of pervasive, the network should auto-configure to fit the needs of the user. It is not necessary for a user to know how WiFi links should be configured. The ad-hoc nature of MANETs give them the capability of taking care of all the grueling configuration processes and in this way increasing the pervasiveness of the system.

In summary MANETs will be useful in the communication of mobile devices, in the scalability of ubiquitous networks (referring to networks that interconnect ubiquitous devices). It will also play an important role in increasing the pervasiveness of future systems.

Technical Challenges

For MANET development there is a list of difficulties that researchers have to successfully overcome: Wireless communication originates a great amount of interference related problems. These are mainly addressed by the first and second layer of the OSI (Open System Interconnect) stack that consisting of a list of MAC (Medium Access Control) and PHY protocols. Movement is another aspect that has an effect on most of the lower layers of the OSI model. Movement introduces additional interference and increases the difficulty of routing messages through the network. Energy consumption and general resource management is critical in MANETs. Given the reduced quantity of available resources, devices have to use diverse processes to make the best of what they have. In addition to all the before mentioned problems, that are mostly related to the behavior of network elements, there is a difficulty with the way MANETs are simulated. The lack of an accurate mobility model that describes the behavior of real mobile entities and the lack of a realistic wave propagation models that correctly describe attenuation, interference, multiple path and others phenomena, distance simulations from reality. These challenges will be described in this section.

Media Access Control

Media access control (MAC) is a part of OSI's data link layer. Its function is to decide who will gain access to the transmission medium at any given time [21]. It is also in charge of some error correction, synchronization and implementing reliable point to point and point to multi-point connections [12]. In a wireless network, the transmission medium is, at times, a resource that is shared between adjacent wireless nodes. Therefore there are mechanisms to optimize the use of this medium and mitigate the problems related to sharing.

In general MANETs communicate using omnidirectional antennas. When a message is transmitted, the signal spreads out in a sphere-like shape that moves outward with the center at the initial transmission point. This characteristic is one of the reasons for signal attenuation. A device does not receive the totality of the sent signal, it only detects the part of the total signal that is present around its receiving antenna. The relationship of the received signal with the sent signal is given by Equation 1

$$PD_r = \frac{P_t}{4 * \Pi * R^2} \tag{1}$$

$$PD_r = \text{Power density of reception}$$
$$P_t = \text{Power of transmission}$$
$$R = \text{Distance between nodes}$$

In any case, an omnidirectional approach is used in Mobile Ad-hoc NETworks because of its convenience, simplicity and lower cost [2]. Given the types of antennas used for these transmissions a complex system of hardware parts is not required. Therefore the acquisition, configuration and use become very simple. Moreover, and again because of its simplicity, the cost is reduced.
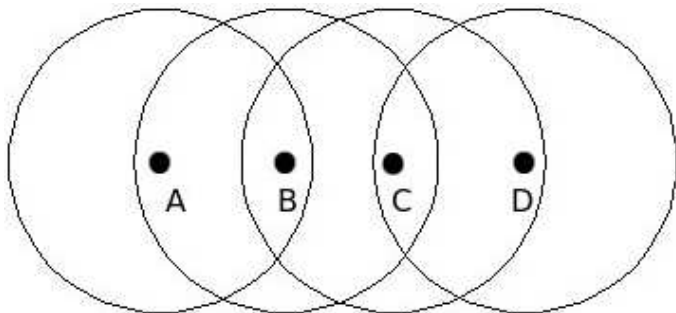
Figure 1. Hidden terminal, Exposed terminal

Networks that use omnidirectional antennas have two very particular difficulties. The hidden terminal problem and the exposed terminal problem. The two are situations in which the wireless medium is not used at its maximum potential. They occur when nodes are involved in simultaneous wireless transmissions.

The hidden terminal problem consists of undetected interference. Assume node B can be reached by the signals of node A and C (Figure 1). Further assume that none of the nodes are transmitting. After sensing the medium and making sure that there are no other transmissions in its vicinity, node C begins transmitting a message to node B. At the moment of the transmission node B and node D receive the signal and know that they can't transmit until C's signal has stopped. But node A has no idea that node C has begun a transmission towards node B. Moreover node A has a message for node B and given that it does not hear any transmissions on its receiver, it immediately starts transmitting. As node A's signal arrives at its destination it creates interference with the signal that is coming from C. In this way the messages from nodes A and C are not received correctly and have to be retransmitted. If another method

of accessing the medium was devised the interfering signals would be avoided and the transmission resource could be used for other purposes.

The exposed terminal problem is similar to that of the hidden terminal problem in the sense that it also describes a misuse in the medium. Assume that node B wants to transmit a message to node A. The first thing it does is listen to the medium to make sure that there are no current transmissions. After making sure that it is possible to transmit, node B begins transmission towards node A. The message is not only received by node A but also by node C. Therefor C is convinced that the medium is being used. Further assume that immediately after node B starts its transmissions, node C becomes ready to transmit a message to node D. However node C is convinced that it can't use the medium because of node B's transmission. When in reality if node C were to transmit there would be no real interference for any of the receiving nodes. Node A would receive node B's messages without any interference from node C because node C's signal does not reach node A's position. And in the same way, node D would receive node C's message without any interference from node B. The total throughput of the network would increase if the apparently interfering signals could transmit simultaneously.

The IEEE 802.11 standard defines a mechanism by which the medium is reserved [4] and the problem of the hidden terminal is mitigated. Unfortunately this method only addresses the hidden terminal problem and not the exposed terminal one. As already stated the main objective is to reserve the medium to avoid signal collisions. If all the neighbors that have a possibility of interfering with a transmission are correctly informed about said transmission, they will abstain form sending any kind of message. After the message has been sent the neighbor nodes then become eligible to transmit. But how exactly does the system avoid the hidden terminal problem?

The system depends on two messages: ready-to-send (RTS) and clear-to-send (CTS). These messages successfully reserve the medium and increase the possibilities for a successful transmission. Consider the hidden terminal example where node C (Figure 1) transmits to node B and is interrupted by node A. If the RTS/CTS mechanism were to be implemented, node A would not start transmission until the end of node C's transmission.

At the beginning of C's transmission it will send a RTS message to B. But B will not be the only one to hear the RTS message. All the neighbors that are in C's range will hear the message as well. Moreover, the message will contain an estimated duration time and the nodes that receive it will know that they can't transmit any kind of messages for that period of time. After receiving the message, B sends a CTS to C confirming that the transmission is possible. As with the RTS message, the CTS is heard by all of B's neighbors, including A. The nodes that receive the CTS message are now aware of the transmission and will not attempt to send any kind of message. It is in this way that node A is restrained from sending a message to node B while C sends a message to B. The process is called handshaking.

The RTS/CTS mechanism does a great job at solving the hidden terminal problem but it does nothing to address the exposed terminal problem. In fact it forces the devices into an exposed terminal situation. If B were transmitting toward A using the RTS/CTS mechanism, node C would not be able to transmit to node D. Not only because node C is constantly listening to node B's message but because of the RTS message received by C when B started transmission. In other words the RTS message forbids node C from transmitting immediately to node D.

There are also some situations where the mechanism fails to accomplish its purpose. When a network has signal range heterogeneity, that is, when all the nodes have different transmission ranges, the possibility for a breakdown of the mechanism grows. Suppose that node X wants to transmit a message to node Y which has a short transmission range. Following the RTS/CTS mechanism node X sends and RTS message, after receiving the CTS from Y it starts transmitting. Unfortunately for node Y, node Z has a large transmission range that contains node Y. Furthermore Z is not contained in node Y's range so it does not receive the CTS message. If node Z were to start transmitting it would interfere with X's signal. This means that the RTS/CTS system does not work properly in networks that have range heterogeneity.

Node mobility is also an important factor that contributes to the demise of the RTS/CTS mechanism. Suppose that there is a handshake between X and Y. This suggests that all the nodes that were present when the RTS and CTS messages were sent, will not transmit. Further assume a mobile node Z that moves into a position in which its transmission range contains the receiver in the $X - Y$ transmission but it does not hear the ongoing transmission. When node Z begins to transmit it will cause interference in the receiving node. This is yet another case where the RTS/CTS fails to deliver interference free communication.

Medium access control processes have to continue to grow toward a mechanism where the principal communication resource is used in an optimal way. New policies have to be engineered to cope with wireless network needs. Policies that fairly assign resources to the network. Policies that take into account the network context in their calculations. Policies that can handle large quantities of devices all interacting by using the same medium. Secure policies to increase the trust in wireless communication and in that way continue the growth of the wireless market.

Routing

Routing in MANETs, as in any type of network, is a basic tool for communication. There are, however, some additional challenges with MANET routing protocols that are not present in conventional wired networks. Movement is a critical aspect of MANETs that has a profound effect on routing protocols. Due to the constant change in the underlying network topology the routing protocol must devise a mechanism that delivers a message to any node. Moreover this message can come from any part of the network. The routing protocol not only has to consider topology fluctuations but it also has to address energy issues. A routing protocol that uses all the energy resource to successfully rout messages is not acceptable because it considerably reduces the lifetime of the network. Also, due to the growing number of types of devices, there is a high probability of range heterogeneity that routing protocols must handle. Moreover they could be able to use this heterogeneity to increase the routing capacity of the network buy using the higher capacity nodes as an advantage [1].

It is also important to consider the relevance that routing protocols have on infrastructureless networks. Routing protocols are the ones which allow MANETs to be multi-hop structures, and it is one of the characteristics that makes MANETs so useful in diverse situations. In other words, without routing protocols, without the multi-hop characteristic, MANETs would not be eligible for many of the situations seen in the functionalities sections.

Routing protocols can be categorized using various characteristics. If they are categorized using the overall network topology used for routing purposes, the routing protocols would be separated into three groups: Flat routing protocols, Hierarchical routing protocols and geographical or location-based routing protocols [1]. On the other hand, if they are categorized by the moment in which information was gathered to calculate the routes, then there would be also three groups: Proactive approach, Reactive approach and Hybrid approach [13], a combination of the proactive and reactive approaches.

The flat routing protocols are the ones that render a topology structure in which all the nodes are part of the same network tier. In this type of routing protocol all the nodes have the same communication capabilities and functions. All the network participants receive, create and forward messages in the same way. It is a flat structure because there is only one layer to which all the nodes belong.

Hierarchical routing protocols are the ones that organize the network into clusters [13] and layers. A cluster is a group of nodes that are in a specific area and can communicate amongst themselves. In a determined layer there can be many clusters. Two elements from different clusters can communicate by using an upper layer as a bridge. The number of layers that a protocol handles depends solely on the design. The upper most layer of the structure has to have the capability to connect the rest of the nodes. For inter-cluster communication to take place a node is selected amongst the group to serve the purpose of a bridge to another cluster. The node that is selected is usually a node with increased computational,

communication and energy capacity. The chosen node is also in charge of receiving any message from the upper cluster and communicating it to its neighbors.

Hierarchical routing can be a good approach towards routing in MANETs but it has a downside. Given that the cluster representative is the node with the best characteristics, the group can become dependant on that node. If the cluster representative does not have a replacement that has similar characteristics the cluster has the possibility of losing communication. Moreover the cluster representative is more prone to failing because it is handling all the incoming and outgoing traffic of the cluster. This means it is using great amounts of energy as a communication bridge.

Location based routing protocols are the ones that use geographical information to find the path to another node in the network. The nodes basically use the position of their neighbors as input for their routing mechanism [1]. In [1] there is a description of two possible ways to get neighbors position. A Global positioning System (GPS) is one of the most logical ways to get a fixed position. The problem with these systems is that they take up space, use battery power and some of them have accuracy issues that don't make them very suitable for MANET use. Another way to get an approximate relative position is to deduce the distance to the transmitting node by analyzing the strength of the incoming signal [1].

Aside from the three groups that are characterized by the topology, there are three groups characterized by the moment in which routing information is gathered. A proactive approach is constantly gathering data and updating information. The routing algorithm continually probes the network looking for changes in the topology. In this way when it is time to rout a message the node will have all the information available to make the correct decision. Unfortunately these types of protocols need a lot of energy and in some cases the information that is collected is never used and the energy allocated for the recollection is practically lost.

The reactive approach is activated only when a necessity arises. The protocol waits for the node to have a message that needs to be routed. It is only when the node has expressed its necessity to send the message, that the protocol starts looking for the rout to the destination. After the rout is found the message is sent and the protocol begins waiting once more. These types of protocols can be very energy efficient because they don't constantly recollect information. The downside to these types of protocols is that they might introduce an unnecessary delay in the transmission of the message. Depending on network traffic, distance (in hops) to the final destination and on the way the protocol finds the rout, the sending node has to wait for a determined period of time before the message is actually sent.

Finally there is the hybrid approach that includes both reactive and proactive mechanisms. These protocols need to assess when it is best to use the proactive approach and when it is best to use the reactive approach. The main idea is to have the accuracy and speed of the proactive approach, with the energy saving qualities of the reactive approach.

Mobility Models

MANET technology is only recently made its way towards the surface of telecommunication research. Due to the fact that its such a new area there are no real, fully operational MANET applications yet. Moreover, systems in which to test the newly developed models are practically inexistent or inaccessible. And its due to this that the work done in MANETs is mostly simulated. Several simulators are used: ns2, OPNET, QualNet, GloMoSim, just to name a few. Even though these are important tools in the advance of telecommunication research they are still very limited with respect to mobility models.

The objective of a mobility model is to accurately represent the movement of an entity in a certain environment. So if a car were to be represented, only two-dimensional movements that are restricted by predetermined pathways would be considered. The underlying problem with these types of models is that they lack the accuracy to represent the movement of a mobile node. To the best of our knowledge there is no mobility model that can predict movement without using a random function. Models that introduce special movement characteristics have been engineered but they still depend on a random distributions.

There is another possibility besides random function based models. It has to do with trace files, these are structures that are directly created from a real experience. A trace file for a car can be created by measuring its position every other second. In other words each file line will contain a position in a certain moment, and the file as a whole will contain the movements of the car in a laps of time. This file can then be used to model the movement of an entity inside a simulated environment. To the best of our knowledge no real trace files have been measured that represent the behavior of mobile nodes inside a MANET. In this section a series of mobility models will be described.

Mobility models are characterized in several ways depending on certain model aspects. One of these characterizations is based on the relation between modeled elements. Using this characterization, mobility models can be separated into two groups: Entity Mobility Models and Group Mobility Models. The first describes the movement of each mobile node based only on the its individual information. There is no direct relation between MNs other than the mathematical formula that describes their movement. The second type of Model describes the behavior of a group of MNs as a whole; an individual directly depends on the values of the rest of MNs in the model. Another possible characterization is based on the origin of the movement of the MN. The models are divided into two groups: Models based on trace files and synthetic mobility models based on random functions. In this section the mobility models based on the first characterization will be described.

Random Walk Mobility Model [18]: This model like most of the Entity Mobility Models defines a process that repeats itself throughout the whole simulation. This model uses four variables to describe the node's movement: position, direction, velocity and time of movement. The nodes move inside a simulated environment represented by a two dimensional square. Before the beginning of the simulation the nodes are given random positions, direction, velocity and time. The position is a pair that expresses the position in a two-dimensional plane. The x value of the pair will be a random value from 0 to the maximum value of x and y is calculated in a similar way. The direction is a random angle from a range of $0°$ to $360°$. The velocity is a random value from 0 to a predefined max speed value. Finally the time for the first movement is randomly calculated.

When the simulation begins the MNs move with the calculated values. When the time expires in the movement of a specific node all the values are recalculated and the node keeps moving with the newly found values. Keep in mind that all the nodes will have different time values and will recalculate depending on the random distribution.

These type of models can be models in 2-D, 3-D and N-Dimensional systems. Another important thing to mention is that the time in which the values are valid can be replaced with a distance value. Finally when a node encounters the edge of the area it bounces with the same angle with which it hit the boundary.

Random Waypoint Mobility Model [11][6]: This mobility model chooses a position inside the simulated area at random, afterward it selects a random velocity and begins moving. When it arrives at its destination it pauses for a random determined amount of time. The nodes have two states, a movement state in which the node has a velocity and a direction, and a static state where the node is waiting for a counter to time out. In this model the MN begins in the static state by choosing a waiting time value between zero and a maximum waiting time.

Random Direction Mobility Model [18]: After a MN's initial position is chosen, its direction and velocity are randomly chosen. The MN stops when it has reached the border of the model, at that moment a new direction and velocity are calculated. Note that the direction has to be between $0°$ and $180°$. This was created to solve the Density Wave problem. This problem consists in the tendency of elements to cluster around the center of the simulated space.

Boundless Simulation Area Model [18]: With this mobility model there are two differences with respect to to the before mentioned ones. The other models calculate their new position using an equation that has velocity, direction and current position as its variables. The Boundless Simulation Area Model not only includes those variables but adds the previous direction and velocity of travel to the equation. In this way the new position will be related to the current state and states in the past.

17

This model has another element that differs considerable from the other models and its the element that gives it its name. The area in which the MNs move has no boundaries. This is achieved by uniting the horizontal borders to form a cylinder and putting together the cylinder borders to form a torus like shape. In this way the MNs move in a area that apparently has no end.

After a MN's initial position is chosen; its direction and velocity are calculated following a function that includes the velocity and direction that the MN had in the previous moment. The destination position is then determined using this formula. The model's reaction to a MN reaching the border of the modeled space is different from the rest of the models. Basically when a MN comes upon the side of the modeled it does not bounce, instead it disappears from the boundary only to reappear on the opposite side. So if a MN were to hit the right side of the area at a hight of 4 units of space, it would appear in the left side of the area at the same hight.

Gauss-Markov Mobility Model [18]: This model was created with the objective of modifying the level of randomness of a model. There is a value α which fluctuates from 0 to 1 that defines the randomness of the model. If α takes the value of 1 the model will be totally linear and if the variable takes the value of 0 it will be totally random. Randomness is given by the Gaussian distribution. The values in between give several levels of randomness.

After an MN's initial position is chosen; its direction and speed are calculated with an expression that includes mean velocity, mean direction, a random velocity, a random direction, α, the previous velocity and the previous direction value. All these values are necessary for the calculation of the new destination and direction variables. Regarding the MN hitting the border, this model has a different approach; by changing the mean direction value, the model is able to smoothly redirect the MN towards the inside of the simulated area.

Probabilistic Measure of Random Walk [18]: This model is based on a probability matrix that dictates what the MN's next position will be. There are three possible values that can be given to each 2-D coordinates (X and Y): the previous position, the current position and the next position. The next position is determined by a constant speed and a random direction value. The probability matrix determines the probability for each coordinate to take one of the three possible values. This model is useful because it can accurately describe human and MN behavior. This is true because these behaviors usually follow a statistical distribution. The difficult part is finding the probability matrix that describes specific movements.

City Section Mobility Model [18]: After a MN's initial position is randomly chosen; the model randomly selects a destination and calculates the shortest path. After selecting the speed of the element, it begins to move and pauses for a random amount of time when it reaches its destination. The process is repeated from the current position. The modeled space resembles city streets and restrain the MN's movements.

Group Mobility Models [18]: Although there are several models that fall in this group, there is one that has the capability of describing the others. The *Reference Point Group Mobility Model* is a generic definition of what a group model should be. The model has three elements. The first is the MN of the model. The second consists of reference points that restrict the movement of each MN, in other words the MN follows the reference points through the modeled world and can only move around it. Moreover, there is one reference point per MN. The third element is the logical center of all the reference points. When this center moves around in the modeled world all the reference points move with it and consequently so do the MNs. First the logical center moves, then the position of each of the reference points is calculated with a Group Motion Vector GM. After the position of the reference point is established, the position of the Mobile element is found using a Random Motion Vector RM. The model will have two types of movements: the group movement, where the reference points move with the logical center and the MN movements where the movement is determined by the reference points.

The mobility models based on randomly generated values describe situations in which the nodes move with respect to the random function. In real situations MNs do not move randomly, each node, having a purpose or functionality, will have a movement pattern that comes from its immediate and long term

objectives. People's movements are described by their immediate objectives. Moreover a person can have an objective that requires lack of movement all together. To the best of our knowledge the current mobility models used in MANET research do not use any type of movement that is oriented toward objective. It is because of this reason that the models do not give a very accurate representation of MN movement. And must be used as a tool to predict outcomes and not to test true behavior.

Topology Control

In this dissertation network topology is defined as the underlying pattern of connections in a network. Star, bus, fully connected and tree are just some types of topologies that are valid in wired networks. The star topology is the one that interconnects all the network nodes by means of a "central" node to which every other node in the network is connected. In this way all messages have to pass through this central node to get to any other place in the network. The bus topology is the one that uses a communication bus as its main transport medium. All the nodes share this medium and implement an algorithm that controls the access to it. The fully connected topology is when all the nodes are connected with every other node in the network. Finally the tree topology is based on a tree structure. The main node, the root, is located at the highest hierarchical point of the tree. Each node that is in a lower tier from the root node is considered a leaf node. These are just a few examples of the network topologies in wired networks.

In wireless networks the pattern in which the nodes are connected define some interesting characteristics. For wireless transmission it is true that the energy of transmission grows quadratically with the transmitted distance. As Figure 2 shows, if the relation between energy (E) and distance (r) were direct the selection of certain topology structures would be more optimum than others. In the figure two simple topologies are presented. The first topology communicates two opposite nodes while at the same time uses certain amount of energy. Topology two, on the other hand, communicates the same two nodes, but it does this in an energy efficient manner. It basically saves half of the energy that is used in the first topology. This figure basically shows that if short distance communication links are chosen, energy is effectively saved.
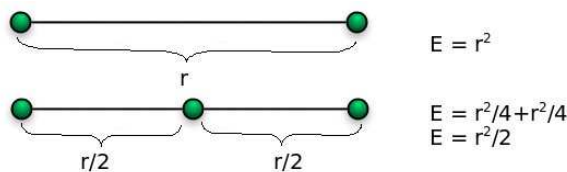
Figure 2. Quadratic relation consequences.

A topology control mechanism (TCM) is a system that modifies the network topology to accomplish certain objectives. Considering that the change in the wireless network topology can result in a optimization of energy use, TCMs can then control the topology in such a way that short range links are used instead of long range ones. In this specific context TCMs are cataloged as resource management tools.
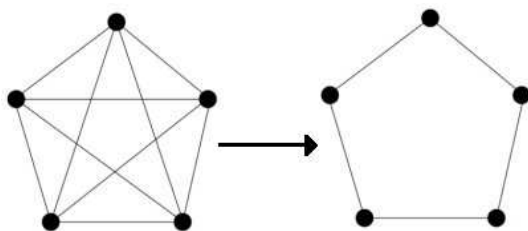
Figure 3. Topology Control Process.

The TCM selects a sub-group of all the possible network links. The process is expressed in Figure 3. Before executing the TCM there is a topology that is made up of all the possible links (fully connected). With information gathered from the network the initial topology is transformed into a simpler more energy efficient structure. This structure is then fed to the routing algorithm. This would locate the TCM right below the routing mechanism. The two systems can also work in unison by sharing relevant information. There could be a case where the routing algorithm might want to select certain links that the TCM cannot modify or it can give the TCM some additional data for the topology calculation. In any case the routing mechanism will use the topology rendered by the TCM.

What is described in Figure 3 is the general process followed by TCMs. This process can be done with a distributed or central algorithm. The distributed algorithm would be executed in each node individually using information collected from the immediate vicinity of each node. One of the possible follies of these types of algorithms is inaccuracy, because local nodes might have incomplete information of the network. Consequently topology that is not optimum might be rendered.

On the other hand centralized algorithms depend on powerful central processing units that handle most of the calculations. The centralized algorithm recollects as much information as possible from the network before calculating the topology. When it has finished, it proceeds to deliver the calculated topology to all the nodes in the network. Due to the complexity of this process, these algorithms require a lot of time to calculate a topology. Moreover, and considering the role that central processing units have, systems become totally dependant on the behavior of one node. Such is the relation that if one of these units goes off-line, the topology would not be calculated and communication would be compromised.

To implement any type of TCM, a way of measuring the quality of connections is necessary. Ultimately, TCMs need to know which connections are the best to create the final topology. It is in this way that TCMs use link quality in their calculations. Link quality is simply a metric that somehow characterizes a link as "good" or "bad". Given the quadratic relation between energy and distance, the metric is necessarily related to distance. It can also be related to other variables like traffic load and amount of errors per unit of time.

In the following paragraphs various TCMs are described. These models can be used for environments that have no variability. The objective is to execute these algorithms at network deployment and use the created topology for the full life of the network.

The Relative Neighborhood Graph (RNG): This is a local geographic approach. A link between node u and node v is selected if no node w exists such that w is closer to u than v, and w is closer to v than u. In other words, if $\text{MAX}(d\{u,w\}, d\{w,v\}) < d\{u,v\}$ then the link is not selected, where $d\{u,v\}$ is the distance between node u and v [20], [13] and [16].

Gabriel Graph: This is also a local geographic approach to topology control. A link between node u and node v is selected if no node w exists within the circumference that intersects both nodes u and v and has a diameter equal to the distance between the two nodes $d\{u,v\}$. Formally speaking the link between node u and v will be selected if no node w exists such that $d^2\{u,w\} + d^2\{v,w\} \leq d^2\{v,u\}$ [20] and [13].

Yao Graphs or θ-graphs: These are two approaches that are very similar and basically share the same objective. As the ones before them, these algorithms are local and geographic. The basic idea is to separate the area surrounding node u in sectors and choose the closest neighbor node for each one. In Yao's case, the sectors would be equally separated rays and in θ-graphs' case they would be sections of a fixed angle. The key to an optimum $G\prime$ topology is to limit the number of sections or rays with a constant. Coincidentally in both the approaches the constant is $k = 6$, so the Yao approach would have maximum 6 rays and the θ-graphs approach would have maximum 6 areas with an angle of $\prod /3$ [20] and [13].

XTC: This is a local approach that uses link quality as input for the algorithm. Each node selection is made from the local node's point of view. In general, a node u will choose v as its neighbor if no node w exists that is "better" than v and can be reached easier from node v than from u [19]. This approach will thoroughly be explained in the next section.

Interference

TCM and interference go together in MANET research. It is believed that these mechanisms reduce interference by minimizing the number of active links in a network. Researchers intuitively conclude that given the reduction in transmission range that is brought on by the use of a modified topology, the possibility of interference occurring in a network is reduced. In most cases researchers make this assumption without formal proof. In this section the common misconception of interference in TCM is described.
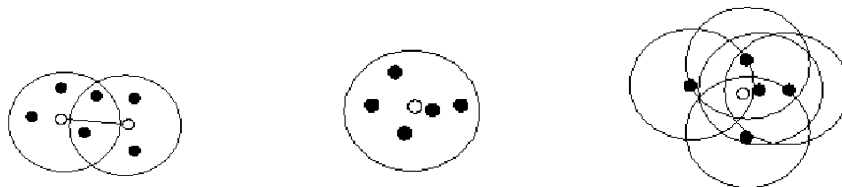
Figure 4. Link Interference. **Figure 5.** Individual Interference. **Figure 6.** Neighbor Interference.

At the moment there is a general misconception regarding the relation between interference in wireless networks and TCMs. It basically consists in assuming that the reduction of node degree brought on by TCMs automatically delivers an Interference-optimal network. This is hardly the case for most algorithms, as they do not use rigorous mathematical models to find the minimum interference network. Usually they concentrate on other network qualities such as spanner properties, node degree and edge overlapping among others. Most documents that don't use a model are left with assuming a possible outcome regarding network Interference.

A mathematical model is important because it provides a rigorous method to identify the best system amongst a set of possibilities, facilitating the search for an optimal network. There are various approaches to the interference that deal with its meaning; In Figure 5 a model that defines interference as the number of nodes that are in the transmission range of a given node is described. Figure 4 considers the number of nodes that are in a region created by two communicating nodes as the metric for interference. In Figure 6 interference is described as the number of nodes that can reach a given node [10][7]. All these have a specialized algorithm that find the optimum network based on the respective mathematical model. In conclusion the misconception lies in making assumptions about the Interference qualities of a network without having the proper model to describe it.

Energy Conservation

Energy use is an important factor in the success or demise of Mobile Ad-hoc NETworks. Node energy capacity and the amount of energy that is used in a unit of time defines the duration of a network. Mainly due to their size, MANET nodes do not have a large energetic capacity. They usually carry a small battery or a solar cell that will drain quickly if misused. In situations where MANETs use proactive routing protocols, the node has to constantly use its energy supply to keep up with the change of it's environment.

Most of the progress in this area can be approached from an electronics point of view. The design, development and deployment of energy efficient microprocessors is an important factor in the successful growth of MANET as a leading technology. These microprocessors don't necessarily have to be the ones that use the least amount of energy. They do, however, need to implement a system by which they can respond to environmental changes to actively save energy. In other words the microprocessor should be capable of increasing its clock speed when ever the node is involved in important activities. It should also be capable of reducing or maybe stopping the clock when little or no activity is being done by the node.

Another way that energy might be saved is by using energy aware systems that modify the node's behavior in order to optimize overall network energy use. An individual module in charge of modifying the behavior of the node in such a way that energy is ultimately saved would be possible. Another

approach that can be considered is energy saving models that are distributed along all subsystems. In this way routing, transmitting, processor calculations and all the other node activities would have their own energy saving module. Compiling applications is also a process that can include energy saving concepts. Code would be compiled in such a way that when it is executed it would not misuse energy resources.

[14] talks about a routing protocol that collects information on what they call the energy map of the network. The energy map contains the energy state of every node in the network. That is, how much energy is left in each node. Using this information the routing algorithm calculates the message routes in such a way that the message traverses through the regions where there is a high concentration of energy. The routing protocol does this in order to allow the nodes with less energy to stay active longer and therefore increase the longevity of the network. This model is a prime example of energy saving mechanisms that are distributed among the node's systems. In this case the mechanism is lodged into the routing protocol.

[5] lists an Logical Link Control (LLC) sub-layer approach based on the reduction of the number of messages that the nodes sends in specific network situations. The approach is based on a modified Automatic Repeat Request (ARQ) method that is used to reduce the number of sent data when low performance of the network is detected. The studies show that, under certain conditions, the modified ARQ mechanism can save energy. In the same article, the authors state that energy can be saved in what they denominate the OS/Midleware layer. They state that mobile devices can take advantage of the fact that wireless networks are systems that have sporadic computational activities. In this sense the nodes can shutdown during periods in which they are not active. It also mentions a mechanism that exploits the DRAM design by implementing a paging mechanism that saves up to 55% (compared to not using it) of energy when utilized. The article also talks about Power Partitioning. It basically consists of separating the computational effort between the mobile nodes. A node with greater computational capacity can be selected inside the network to take care of the complex operations of the network. Other approaches consider application specific variables. In video streaming for example, where transmission accounts for more than one third of the total energy, it is said that energy can be saved in the encoder by reducing the size of the video or to disregard certain packets at transmission time to reduce the total tranmitted information.

In [3] the authors consider the energy saving capabilities of compressing data. The article mentions that the saving capabilities depend on the compression and decompression mechanisms. It states that, depending on the mechanism, some approaches would not save energy at all because the compression and decompression processes use more energy than that saved by the transmission of the compressed data. Moreover, the results of the tests in the article show that the used energy in compression and decompression depend on the hardware in which it is executed. So the authors propose an asymmetrical mechanism in which information is compressed by the best-of-bread mechanism at the starting point, and decompressed by a best-or-bread mechanism at the end point. They end the article with the conclusion that data compression and its relation with energy optimization is far from trivial. They also mention that it depends on CPU energy use, memory access, network activity as well as application specific characteristics.

XTC Algorithm

XTC is a general purpose algorithm that calculates an energy efficient topology based on a metric. The basic idea behind the algorithm is to take a graph G and transform it into a graph G'. The resulting graph G' adds certain characteristics to the network that allows it to use its energy resources more efficiently by selecting a subgroup of links. The purpose of this section is to give the reader some specific background information about the algorithm that is used in this dissertation. It also becomes relevant because most of the dissertation is emphasized in the information recollection stages of the XTC algorithm.

To understand just how energy is saved by selecting a subgroup of network links, the relation between transmission energy and transmission range must be clarified. In general the energy needed to transmit grows quadratically with the distance. This is true due to the propagation characteristics of an electromagnetic signal in free space. For all omnidirectional antennas the energy used on transmission is spread out in a sphere like shape with the center at the point of transmission. In other words the energy with which a receiving node detects an incoming transmission is just a small part of the total spread energy. Taking all this into account the equations described in Figure 2 do not accurately describe the quadratic relation, Equation 2 is a more accurate expression [17].

$$PD_r = \frac{P_t}{4 * \Pi * R^2} \tag{2}$$

PD_r=Power density of reception
P_t=Power of transmission
R=Distance between nodes

The relation is clearly quadratic with respect to the distance.

Keeping the relation between the transmission energy and the transmission range in mind, an optimum situation can be deduced. If energy of transmission grows quadratically with distance then it becomes better to use various short hops to communicate with a distant neighbor than to use long communication links. Communicating from one end of the network to the other would take a large number of short range hops, but the transmission energy would drastically be reduced compared to that used in a few long range links.

Considering the specific relation described in Equation 2, it is not difficult to devise a situation in which choosing short hops instead of a long one does not render an energy efficient topology. Assume three nodes A, B and C. Further assume that the three are inside the range of the others. The distance between A and C is 4 Mts, the distance between A and B is 3 Mts and the distance between B and C is 3 Mts. According the the model proposed, the 4 Mts connection between A and C would be much better served if it were to go through B. But this is far from the truth. The energy from A to C is calculated using the following expression $E_{a-c} = 4 * \prod * 4^2 * E_t$. Furthermore the energy from A to C using the intermediate node is given by $E'_{a-c} = 2(4 * \prod * 3^2 * E_t)$. It is simple to deduce that $E_{a-c} < E'_{a-c}$. This is just a specific situation in which the model does not work at its full potential and does not mean that that is it's average behavior. Furthermore, XTC not only saves energy by using intermediate nodes, [19] also states that energy is saved by avoiding interference. The article itself lacks the mathematical proof needed to make this conclusion and it does not use any of the models described in the interference subsection. In any case, the specific energy saving capabilities of the resulting topology is not the main focus of this dissertation and can be considered in future work.

Energy management is extremely important in MANETs due to the energy restrictions of the devices. Mechanisms that save energy by modifying certain aspects of the underlying structure can increase the longevity of the network and increment MANET dependability. The models that use energy saving mechanisms like XTC can guarantee a long lasting service in diverse scenarios. This is not just an improvement on energy efficiency, but can also be considered as an aspect that makes MANETs more appealing to the customers. In other words XTC like mechanisms increase the viability of MANETs and will push towards the development of massive self-organizing mobile structures.

One of the special characteristics of XTC is that it is a localized algorithm. This means that it runs in each node and does not need the assistance of a centralized computational entity. XTC is an algorithm that is much faster in calculating a topology because it avoids having to send and receive information from the central control unit. However the calculated topology might not be as good as the one calculated by centralized topology control algorithms because these have much more information than the localized one and might make better decisions in that sense.

23

XTC selects links based on link metrics that are calculated with information that has previously been recollected. The XTC algorithm can basically be separated in two main parts: First, the recollection phase, in which the local node sends and receives information to and from its neighbors to create the XTC data structure. The second part would be the actual execution of the algorithm.

Lets suppose that the link metric for an assumed network depended only on distance. In this case the objective of the algorithm is to choose the "closest" neighbors in such a way that, with the chosen group, the local node has the capability of communicating with all its other neighbors. This is not an easy task. The local node can't just choose the neighbors with the shortest distance. If it were to do this, there would be a possibility of having unreachable neighbors. A sub-group of neighbors has to be created in such a way that they are closest to the local node allowing for communication with the rest of the unselected neighbors.

The before mentioned analysis can be extended to metrics that not only depend on distance but consider other variables. In real situations where nodes have a large variety of elements that modify the quality of a link, a different metric system must be devised. The interference from signals that bounce from the floor and arrive in the receiving antenna in a different phase could possibly be included in the metric. The signal absorption properties of physical elements that are "in the way" of a signal are also eligible as metric information. Any element that modifies the reception quality of a transmission has the possibility of being included as a metric variable. The XTC becomes extendable in the sense that it can calculate a topology using any metric. With this in mind the links that are chosen are not necessarily the "closest" to the local node but are the "best" with respect to a determined metric. Finally the metric used in any TCM must be measurable.

The link quality values (metrics) need to be organized in a data structure for the algorithm to correctly calculate the local topology. The data structure is an ordered set that contains neighbors. The set is ordered with respect to the link quality values that the local node has of each reachable neighbor. Consequently the set needs to have the link quality for each neighbor. In Figure 7 a set of the neighbor nodes of node 2 is described. Keep in mind that node 2 is not the only one that has this set in its memory, all the network nodes need to have this structure. This means that all of node 2's neighbors will have a similar structure but with different values. This structure will be denominated node 2's set.

address(1)	address(5)	address(4)	address(3)	address(6)	...
metric = 8	metric = 7	metric = 4	metric = 4	metric = 3	...

Figure 7. Node 2's set.

But the set in Figure 7 is incomplete. Additionally to the information that is contained in Figure 7. the local node must have the sets from its neighbor nodes. So the data structure will look more like Figure 8 than Figure 7. In other words the information needed to calculate XTC is the set of neighbors that was ordered using link quality and the neighbor sets that are received through incoming messages.

address(1)	address(5)	address(4)	address(3)	address(6)	...
metric = 8	metric = 7	metric = 4	metric = 4	metric = 3	...
order(1)	order(5)	order(4)	order(3)	order(6)	...

Figure 8. Node 2's complete set.

The original XTC algorithm is divided into three stages. The first two are information recollection stages and are the ones that create the totality of the data structure needed to calculate XTC. The third stage is the algorithm itself and it needs the data structure created in the first two stages.

The objective for the first stage -neighborhood ordering- is to generate a structure like the one described in Figure 7. This structure will not have all the information needed to run the XTC, but it is the first step towards recollecting all the information. In this stage all nodes transmit beacon message with maximum power and wait to receive the beacon messages from their neighbors. For each message that is received the local node calculates the link quality for the neighbor that sent it and inserts the neighbor address in the initial structure. When the node has received all its neighbor's messages it will have created a structure that describes the immediate neighborhood. The created structure not only contains the

neighbor's address and the metric related to the neighbor, but it also expresses the order in which the neighbors are organized with respect to the metric.

The second stage -neighbor set exchange- is where the final data structure is created. The only information that the local nodes needs to form a structure like the one described in Figure 8 is the order from its neighbors. When the local nodes begins stage two it sends its newly calculated set with maximum power, while at the same time receives the incoming messages that contain the set from the neighbors. For each incoming message the local node will place the received neighbor set in the corresponding list offset and it will continue to do this until it has filled the data structure completely. After the data structure is completely filled the local node can proceed with the third and final step in the XTC process, the topology calculation.

The third stage -edge selection- is the execution of the XTC algorithm, it basically consists of separating the neighbor list into two non intersecting lists: the chosen neighbor list and the rejected neighbor list. The following is the pseudocode of the algorithm followed by an explanation of each line and data structure.

Local node u
    1. Establish a neighbor list $\prec_u$ for u
    2. Transmit $\prec_u$ and receive lists from neighbors
    3. $N_u = \{\} \wedge \tilde{N}_u = \{\}$
    4. While ($\prec_u$ contains unprocessed elements)
    5. $v =$ next unprocessed element
    6.   if $(\exists_w \mid w \in (N_u \cup \tilde{N}_u) \wedge w \prec_v u)$
    7. $\tilde{N}_u = \tilde{N}_u \cup v$
    8.   else
    9. $N_u = N_u \cup v$

Where:
$\prec_u$: Represents $u's$ order
N_v: Chose neighbor list
$\tilde{N}_u$: Rejected neighbor list
$x \prec_u y$: Its an expression that means that element x coms before y en in $\prec_u$

Figure 9. XTC Algorithm.

The first line is where node u creates its list of neighbors ordered by link quality. On line 2 node u transmits its neighbor set and receives the sets of its neighbors. Line 3 initializes two sets: the chosen neighbor set N_u, containing the nodes that u will communicate with, and the rejected neighbor set $\tilde{N}_u$, containing the neighbors that are reachable but don't have a direct communication with u. Line 4 expresses the fact that the algorithm traverses u's neighbor list $\prec_u$. In line 5 an element of $\prec_u$ is assigned to v. Line 6 - 9 is where node v gets either chosen or rejected. "Informally speaking node u will form a direct communication link with node v if there is no better node w that can be reached more easily from v than from u itself"[19].

According to [19] after the execution of the XTC algorithm the resulting graph G' has three characteristics that make it optimum. The article also makes an addition for two of the characteristics making them better when compared to other topology control algorithms like Gabriel Graphs or Relative Neighborhood Graphs. It is these characteristics that make XTC an ideal candidate to use in dynamic environments where node movement is involved.

The first characteristic is connectivity. This is basically a relation between the initial graph G and the resulting graph G'. The rule states that if two nodes u and v are connected in G, then they must be connected in G'. Due to calculations done in XTC, the number of hops connecting the two nodes should be larger in G' than in G. Moreover if connectivity is not enforced, the G' graph may end up with islands of nodes that are not connected to the main network. In conclusion the lack of connectivity might compromise the capability of reaching network nodes.

There is an additional property to the connectivity characteristic in [19]. The article states that the topology rendered using the XTC algorithm has an additional aspect that further optimizes the connectivity of G'. The characteristic is called spanner. It is a restriction of the relation between the connectivity in G and G'. To describe spanner the article uses a general expression of optimal path cost that depends on individual link cost. Let c be the cost of the optimal path between u and v in G. So the cost of the optimal path between the nodes in G' is $f(c)$. The article states that if the function $f(c)$ is bounded from above by a linear function in c, the resulting graph is spanner. The article concludes saying that though the XTC does not provide a strict spanner property, it shows promising results.

The second characteristic is symmetry. In networks that have range heterogeneity there is a high probability of finding links that are not symmetric. A link that is asymmetrical is one where communication is possible in one direction only. If a node with a small radius of transmission receives a message from a node that is out its range, it will not be able to send a direct message back. This becomes a problem when immediate ACK packets are required for communication, like those used in the 802.11 MAC layer in a technique called positive acknowledgment [4]. In [19], the author proves that the graph calculated by the algorithm is completely symmetric.

The third and last characteristic is called sparseness. For a resulting graph G' to be considered sparse it is enough for the number of links in the network to be in the order of the number of nodes in the network. Let $G' = \{E', V'\}$, where V' is the group of network nodes and E' is the group of links in the network. So for a topology to be sparse the following must be true:

$$\text{SIZE}(V') = g(\text{SIZE}(E'))$$

Where SIZE() is the function that returns the amount of elements in a group. This last characteristic also has an additional property. Not only does the before mentioned equation have to be true for the resulting network, but the degree of the local node must be bounded from above by a constant as well. In [19] it is proven that the maximum number of possible links for a local node is 6.

To end this section on the XTC algorithm, a brief comment on the relationship between XTC and Relative Neighborhood Graphs (RNG) is given. According to [19] there is a small difference between XTC and RNG. It talks about a specific situation in which RNG includes a link and XTC does not. It basically states that if there are two or more neighbors that are at exactly the same distance from the local node, the RNG selects all the neighbors, where the XTC selects only the first one that is analyzed. Though the specific situation is possible it is highly improbable.

RNG fails when two neighbors are at the same distance from the local node. This is true for simple mathematical models that do not consider the chaos of real situations. Even if two nodes were to be in at the same distance from a local node, it is highly unlikely for the local node to give the same metric to both of them. This is true because the shape of an omnidirectional signal is not a perfect sphere and a node receiving a signal from two different sources that are at the same distance does not necessarily mean that it will receive the transmission in the same conditions. In conclusion there is a very slim probability for the specific situation to occur, therefore XTC and RNG should be considered as equals.

Dynamic Environment Considerations

One of the most important characteristic in MANETs is variability of network elements. It is an important aspect because it calls for a series of mechanism that try to cope with the chaos created by variability itself. The challenge imposed by MANETs to the various protocols and systems of the OSI layers is not to be taken lightly. For these type of networks to be usable in real situations it is necessary to embrace the existing variability and develop open system type solutions [4]. Solutions in which there is a constant flux of energy through the system [4], or a constant flux of information.

Variability is considered to be the change of value in time. It is the change of specific variables or states that represent certain aspects of the network. The specific aspect or aspects of the network that a variable represents depends solely on the design of the system. A variable, for example, can represent the mobility in the immediate vicinity of a device. Zero (0) might represent a static environment, while one (1) would represent complete chaos. On the other hand, and by means of mathematical calculations, a variable can also include various network aspects. Equation 3 is an example of variability of a network.

$$A(m) + B(h) = v \tag{3}$$

Where A, B are values from 0 to 1 that represent the importance that is given to the variable in the equation and $A + B = 1$. m is the mobility value and h is the value that represents range heterogeneity. The important aspect to remember is the fact that variables represent network aspects and that variability can be measured by the change of these variables in time.

Keeping the variability concept present and bearing in mind that the main objective is to create a system that uses topology control to save energy in dynamic environments, some aspects of XTC in dynamic environments are mentioned. XTC is designed to calculate a topology but not to maintain one. If the algorithm were to be used in dynamic environments without any change to its structure, its performance would be poor, if not useless. The problem lies in the non-repeatability of the algorithm. It would perform great at network deployment, but the calculated topology would be rendered useless as network elements vary. In other words, given that the network representation does not change, the calculated topology can not be used for communication.

If any TCM is to be used in dynamic environments like MANETs, the network representation has to constantly change with the network itself. For this to be possible two elements need to interact inside the new dynamic TCM (D-TCM). The first element is the one that calculates the topology, this element takes existing environment information and transforms it into a working representation. The second element is precisely the one that provides network information to the first one. These two elements need to be used constantly by D-TCM to maintain a representation.

To assure a correct representation of the real network, the mechanism used for the recollection of the information ought to be very accurate. To the best of our knowledge it is impossible, in dynamic environments, to have a representation that is in every sense equal to reality. The time that goes into gathering all the information combined with the time that it takes the algorithm to calculate the representation, creates an inevitable difference between what is represented and what is really there (assuming node mobility). The successful maintenance of the topology depends on how large this difference is. In other words the final objective regarding the representation of the network is for it to be as close as possible to the authentic state of the network.

Let NR_n be the represented network at time T_n. Also assume that AN_n is the Authentic Network at time T_n. Suppose that NR_n and AN_n are represented with a mathematical structure very similar to the one used in XTC. Let diff() be the function that calculates the difference between the two structures. Assume that S_0 is the state of the network at time T_0 and that TC_0 is the calculated topology for that specific state. Further suppose that the algorithm spends ΔT_1 units of time gathering all S_0's information and ΔT_2 units of time calculating TC_0, this means that the algorithm will spend a total of $T_t = \Delta T_1 + \Delta T_2$ calculating the topology. So at T_1, exactly T_t units of time after T_0, the network state will have changed to S_1 but will be represented with TC_0 instead of TC_1. In other words the local node will use a topology calculated for S_0 in S_1. If diff(AN_1, NR_0) is large, the node will not be able to use its current TC_0 for communication purposes. It becomes important in dynamic environments to address the difference between what is represented and what is really there. In conclusion, the key to maintaining a representation of the network in dynamic environments is to constantly recollect and analyze information with processes that can execute in short intervals of time. The recollection and analysis processes will be described more accurately in the following paragraphs.

When focusing on the information recollection process, it is important to identify the origin of the information. The communication interface is the place where network information is originated. It not only receives all the the messages that are sent to the local node, but also eavesdrops on other conversations that take place in the vicinity. This empowers the local node to gather information on its immediate environment by analyzing the incoming messages.

The type of information that the local node can gather is diverse. It can be as simple as identifying all the local neighbors by maintaining a table where the rows represent nodes for which messages have been received. These rows would have a time to live (TTL) that would represent the time for which the information is valid. On the other hand, it can be as complex as identifying a pattern of arrival and departure from the communication range. By the use of pattern identifying algorithms and statistical models, the local node would be capable of predicting the behavior of neighbors. In this way the communications might increase in accuracy. Simplicity or complexity of the model depends solely on system design.

The information gathered in the local node comes from the analysis of received data. The communication interface provides a great deal of possibilities for data recollection. The time of arrival of a message, the energy with which certain message was received, the address of the sender, the total number of messages that a certain node has sent and the amount of errors that have occurred with a neighbor are just some examples of data that can be used. This data can be organized, filtered and manipulated so that it ends up representing other, more complex, concepts like local mobility or range heterogeneity.

If the information recollection process is used to define types of recollection strategies, there are basically two ways of gathering information: a proactive approach and a reactive approach. When using a reactive approach, the information recollection process depends on environmental activity. That is, information is only recollected when it is available through the communication interface. In other words, if no communication takes place in the immediate neighborhood of a local node, the information recollection process will not be executed. This is a very unpredictable situation due to the manner in which the messages arrive. The node can have a good network representation if there is a constant flow of messages, but will render an out-of-date representation if its neighbors do not transmit. It may want to communicate, but since there has not been any external activity, it will not be able to do it accurately. Finally, the reactive approach does not specify any method of getting information aside from that of waiting for incoming messages it is this characteristic that makes it unusable for MANETs.

The proactive approach is the second type of information recollection strategy. It consists in an additional effort from the network nodes to share information with the objective of having enough data to increase the possibility of calculating an accurate representation. By constantly sharing data among the neighbors, this approach in some way ensures the accuracy of the information to a certain point. But it does have a downside. The energy allocated for proactively recollecting information might be lost when the resulting representation is not used. In networks where transmission is rare and the topology is seldomly used, the efforts made are senseless. In a network of sensors, for example, where communication occurs sporadically there is no sense in implementing a proactive approach because the network representation that is created by said approach would be underutilized and the energy used to create the representation would be lost. In this case a reactive approach that responds to the necessity of each node is more pertinent.

The proactive recollection of information can be implicit or explicit. It is explicit when information request messages are used. A local node informs all its neighbors by means of a request for information messages that it needs to update its representation. When a neighbor receives this message it responds with an information message containing relevant data. In this way the local node refreshes the information of the neighborhood and can have an accurate representation whenever it needs it. On the other hand there is the implicit proactive approach that does not need the information request message. It interchanges information by means of a global information interchanging policy used in all the network. In other words each node has a series of rules or policies that commit it to sharing its state periodically using an information interchange message where the state of a node is transported. Suppose a simple policy where all nodes sends their states once every unit of time. Further suppose that the unit of time is equal for all nodes. In this way a node will receive neighbor information periodically without having to explicitly ask for it.

Having looked at the information recollection strategies, the information analysis process will be described. As with the first one, the information analysis process can be divided into two groups: a proactive approach and a reactive approach. Each one with its pros and cons that depend mostly on the energy and time spent calculating the network representation. Moreover the two systems; the information recollection strategy and the information analysis process, have to work in unison to create an accurate and usable network representation.

The proactive approach for information analysis consists of repeating network calculations constantly. Of course, for the repetition to be effective the information provided by the recollection module needs to be accurate, or else the calculation will be repeated using the same information and the rendered representation wont change. This approach has the possibility of overly using energy resources, in the sense that it continually makes an effort to maintain a network representation that may not be used. However an accurate network representation is available whenever it is needed and can be used immediately if required. In other words this approach might misuse energy but allows for an immediate access to the topology structure.

The reactive approach for information analysis basically waits for the moment in which the representation is needed and only after it sees the need, it will calculate the representation. The "need" comes from a requirement of the upper layers. Only when a message needs to be sent is the network representation calculated. This approach is in opposition to the characteristics of the first one. It will generally make good use of energy resources because it only runs a calculation when it is needed. Moreover the time from moment the representation is requested to the moment the representation is calculated could be long. This approach not only has to analyze the data but it also has to recollect all the information that is needed for the calculation and the time spent in these activities might slow down the overall communication process.

In any case, for the two information analysis approaches and for the two information recollection strategies the used energy, the accuracy of the network and time spent in the processes become crucial to the success or demise of the system. Moreover the characteristics of proactive and reactive approaches can be summarized in one idea: For reactive approaches there is a relatively good use of the energy resource and poor response time. And for the proactive approaches there is a relatively bad use of energy resource and good response time.

Variables

To describe the behavior of a D-TCM and/or to compare two approaches there needs to be measurable variables that describe key elements inside a specific model. These variables will allow the model to be characterized as successful or unsuccessful considering certain assumed parameters. Therefore, the selection of these variables becomes vital to the accuracy and relevance of the conclusions made in this dissertation.

To begin the description of the variables, the context in which the topology control model is developed must be considered first. Let the reader be reminded that the basic premise of a TCM is to control the network topology towards a predetermined objective. The objective itself can be any specific characteristic that has a direct relation with the links between nodes in a network. However the current context narrows the pool of possibilities down to just one objective: to save transmission energy in search of prolonging the life of the network.

Energy

Energy saving TCMs select links from an initial graph which have the best link quality. In static environments the selection process is done once in the whole life of the network. This means that the energy used to execute the mechanism is not relevant to the longevity of the network. On the other hand, in the dynamic environments, where the mechanism needs to be repeated constantly to have an accurate repre-

sentation of the network, the energy used by the mechanism becomes important. Given that the process of calculating a representation is going to be repeated indefinitely throughout the life of the network, the energy used in the calculations can compromise the longevity of the network. By modifying the calculations to reduce their energy consumption, energy can be saved in the first phases of D-TCMs as it is saved when using the calculated topology.

It is in the repetition of the recollection process and the execution of the algorithm that additional energy saving strategies can be implemented. The idea is to reduce, as much as possible, the number of messages that have to be sent in order to recollect information for the D-TCM. Moreover the saving of energy through the reduction of the number of steps that have to be executed in the algorithm itself should be considered as well. Consequently there will be two possibilities to save energy in dynamic environments: the first being the reduction of the amount of communication effort being done in the recollection phases and the second is the reduction of the amount of effort that is made in the execution of the algorithm itself.

With respect to energy consumption in wireless devices there are basically four states for a communication interface: transmitting, receiving, idle and sleep [9]. The transmitting state is when the communication interface is actively transmitting a signal. The receiving state is when the interface is actively receiving a signal. The idle state is when the interface is neither receiving nor transmitting but it has the possibility to do either. And finally the sleep state is when the communication interface is neither receiving nor transmitting and it does not have the possibility of transmitting [9].

Considering the four states, a mechanism to save energy can be devised for every state. In which the specific characteristics of the state are used towards an optimum utilization of the energy resource. To this objective, this dissertation only considers mechanisms that have to do with the state of transmission. That is, the measured variable for energy consumption will be transmission energy. Keep in mind that the energy spent in transmitting exceeds that of other states, and therefore any avoided transmission will translate into considerable amount of time for the other states. Moreover this dissertation refers only to the transmission energy and makes no analysis on the other communication interface states. Other energy saving policies that are concentrated in the idle, reception and/or sleep states can be considered for future work, following this dissertation.

Finally, the energy spent in calculating the topology, which is a state of the node and not of the communication interface, will be considered equal to that of the other states (idle, sleep and reception) in the sense that it too can be modified towards an energy saving infrastructure, but will be left to future analysis. Moreover the energy used in calculation purposes is also very small compared with the energy used in a transmission. This means that for every avoided transmission a considerable amount of time for calculations is gained.

In conclusion, the first of the measurable variables that is going to be used in this dissertation is Energy of Transmission of the control messages in the D-TCM. This variable is directly related to the number of bits that are transmitted and to the distance of the transmission. If this distance is not modified for any message, the energy can be measured by counting the amount of bits that were transmitted in a given unit of time.

Accuracy

But its not as simple as just saving energy. If the only variable to optimize were energy consumption the solution would be trivial. The node would simply abstain from sending any kind of control message and from executing any kind of algorithm. This, of course, goes against any conception of a usable structure and would render an unusable network. In static environments the process to create a workable structure is executed at network deployment and, as stated before, does not have a great effect on total energy consumption. On the other hand for dynamic situations where the network is constantly changing, it is important to maintain a balance between the accuracy of the network and the energy used to obtain that accuracy. The objective then becomes to use as little energy as possible while still maintaining an accurate representation of the network.

Representation accuracy is also very important for the model to be successful in real situations. It is based on this accuracy, that energy is or is not saved with the generated topology. If what is generated is very different form the real state of the network the energy saving qualities of the calculated topology can not be predicted.

A way to measure accuracy of the representation is by using a diff() function. This function takes two arrays and calculates the difference between them. The first array would represent the node representation of the network and the other would represent the network itself. Let A be the array for the real network and B the array for the representation contained in the node. Both these arrays contain all the neighbors organized in order of link quality. They don't necessarily have to have the same amount of items. A diff() function can be developed where the same element in the same position of the array is the only possibility for there not being a difference. In other words if $A[i] = B[j] \land i = j$, no difference is counted, in all other cases a difference unit is counted (Example 1).

Example 1.

$\text{Alfabet} = \{a, b, c, d, e, f\}$

$A = \{a, d, e, f\}, B = \{b, d, c, e, f\}$

$d = \text{Difference value.}$

$1. A[0] \neq B[0] \Rightarrow d \leftarrow 1$

$2. A[1] = B[1] \Rightarrow d \leftarrow 1$

$3. A[2] \neq B[2] \Rightarrow d \leftarrow 2$

$4. A[3] \neq B[3] \Rightarrow d \leftarrow 3$

$5. B[4] \Rightarrow d \leftarrow 4$

$\text{diff}(A, B) = d = 4$

It is important to mention that the accuracy metric, as calculated by the diff() function, is impossible to measure in real situations. This is true because the current state of the network can't be measured in real time. This variable is valid only in simulations where the acquisition of the real state of the network can be implemented as a "snapshot" function. In any case, it is important to consider this variable because it gives an idea of how close the represented topology might be with respect to the real network.

There is another way to measure the representation accuracy. By looking at the problem from a different perspective the measurement becomes a matter of telling time. Let NR_n refer to the representation at time T_n and AN_n refer to the real state of the network at T_n. Further assume that S_0 is the state of the network at time T_0 and that TC_0 is the calculated neighborhood for S_0. Further suppose that the algorithm spends ΔT_1 units of time gathering all S_0's information and ΔT_2 units of time calculating TC_0, this means that the algorithm will spend a total of $T_t = \Delta T_1 + \Delta T_2$ calculating the neighborhood. So at T_1, exactly T_t units of time after T_0, the network state will have changed to S_1 but will be represented with TC_0 instead of TC_1.

From this point of view the problem is visualized as the relationship between three variables: S_0 the initial state where the information is recollected, S_1 the second state where the calculation is made, and T_t the time between the two states. The optimum solution for this problem is for $T_t = 0 \land S_0 = S_1$, this optimum solution is inconceivable due, mainly, to the time it takes to recollect the information. However, a model can be described based on T_t where the objective is to reduce this value.

To calculate T_t a basic principal of S_0 needs to be understood. S_0 is not one state that represents all the values for all the nodes at one given point in time. Rather it is a state that represents all the values for all nodes at different times. The information for each neighbor is recollected at different moments in time, so S_0 would be a group of elements gathered with a different time of arrival. Example 2 clarifies the idea.

Example 2.

Optimum S_0:

e_1	e_2	e_3	e_4	e_5	e_6	e_7
t_0	t_0	t_0	t_0	t_0	t_0	t_0

$\Big\}$ All the information arrives at the same time.

Real S_0:

e_1	e_2	e_3	e_4	e_5	e_6	e_7
t_1	t_2	t_3	t_4	t_5	t_6	t_7

$\Big\}$ All the information arrives at the different times.

$$T_t = \frac{\sum T_n}{n}$$

T_n: Time between the arrival of element n and the calculation of the topology.

n: Number of elements.

In conclusion and based on Example 2, S_0 is not a state in t_0, but a state from t_0 to t_n. Keeping this in mind, T_t is not calculated by measuring the time difference between t_0 and the time of the topology calculation. It is calculated by adding all the time differences from all the elements in the S_0 structure and dividing it by the number of elements, like described in Example 2. The time difference between the arrival of the information and the calculation of the topology can be the second measurable variable.

For the before mentioned analysis to be true there is an underlying assumption. There needs to be variability in the environment for this model to be valid. There are situations where $T_t \gg 0 \wedge S_0 = S_1$. When there is no variability of the network the two states (S_0, S_1) will remain equal and will have no relation to T_t's value. But considering the type of networks that are being discussed in this dissertation (MANETs), these situations are very unlikely. However this time metric will not be used in this dissertation. So the difference between the represented topology and the real network becomes the second measurable variable.

Time of execution

This metric is different from the one discussed in the Accuracy section. It refers to the time of execution of D-TCMs and has no direct relation with accuracy. If the D-TCM takes a long time to execute then it will not be able to perform in situations where there is a large rate of change. Remember that when variability increases the mechanism must be repeated at a higher rate. So, if the D-TCM takes a long time to execute, it will eventually fail when the rate of change requires an execution time less than that of the D-TCM. On the other hand a mechanism that executes in a short amount of time, performs well in more situations than the one that takes longer.

The variable measures the time it takes the local node to calculate a topology. It considers the XTC execution time and the time for all D-TCM control message, not only the messages that are originated at the local node, but all the control messages that are transmitted in its vicinity. In other words, the metric counts the time it takes the local node to send and receive D-TCM control messages. Time of execution is measured in this way because it is assumed that, for a D-TCM to work correctly, all neighbors must completely send all control messages. So even if the local node has finished sending its control messages, it must wait to receive all the messages that come from its neighbors. Two times are counted: the time it takes the local node to send all control messages to the neighborhood and the time it takes the local node to receive all control messages from the neighborhood.

In general, a D-TCM that takes the shortest amount of time calculating the topology is required. This objective can be accomplished by reducing the size of the messages or by reducing the amount of messages that have to be sent to recollect information. A modification can also be made to the algorithm itself so it can execute as few steps as possible and in that way save precious time that might be needed to send other type of information.

In general TCMs require three qualities. The first consists of using small quantities of energy in the recollection process. The second is accuracy of the represented structure with respect to the real network. The third consists in reducing the time of execution to accommodate networks with a high rate of change. With these qualities as underlying objectives for the model, an optimum mechanism that saves energy while maintaining connectivity can be developed.

Models

XTC is an algorithm that is very successful in exploiting wireless transmission elements to increase network longevity. It avoids having to maintain a centralized node which manages all the calculations and successfully spreads the calculation effort throughout the whole network. It's an algorithm that has an optimum behavior in static environments. But, as described in [19], lacks the scalability necessary for dynamic environments. It is designed in such a way that it can only be used in static environments. It is from here that the necessity of modifying the original algorithm becomes clear. As next generation networks begin to depend on MANET-like structures to increase their capability, it becomes more important to create new and innovative ways to handle increasing network variability. The models described in the next section try to adapt a very successful static strategy into one that can be used in dynamic environments.

Two models are described with the basic idea of introducing repetition into the algorithm. This is achieved by the use of intervals of time. These not only give the periodicity of repetition but also control all the events related to the information recollection and algorithm execution phases. In other words it is through intervals that the repetition needed for dynamic environments is implemented.

Model 1

Model 1 (M1) is a first approach to an algorithm that tries to create autonomous nodes inside a dynamic network. Autonomous in the sense that the local node is the one that decides when certain information is needed and will take action when it deems necessary without waiting for any peer confirmation. Another aspect that was considered in M1 is the similarity to the original XTC concept. The algorithm tries to differ as little as possible from the original algorithm with the hope of having similar results to those in static environments.

The Model description will begin from a local nodes point of view. In other words, from a node that is gathering information to calculate its topology representation. Afterwards the behavior of the node providing the information will be described. The descriptions are based on Figure 10.
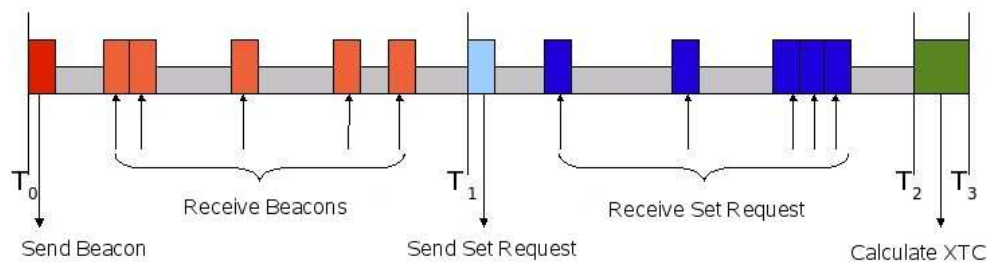
Figure 10. Model 1 description of process.

As seen in Figure 10 the three stages of the original algorithm are maintained. In the first stage the local node constructs the set that contains all neighbors, this stage begins at T_0 and ends at T_1. The second stage is where the local node receives its neighbor's sets, this stage begins at T_1 and ends at T_2. And finally the third stage consists of using the recollected information which was gathered in the two first stages to execute the XTC algorithm. The first two stages implement a waiting period that is used to receive the neighbor responses.

In the first stage the local node sends a beacon message (BM) with full power containing its address. Immediately after the initial message is sent the local node listens for the response of its neighbors. When a neighbor receives the beacon message from the local node it has to respond immediately with a response to beacon message (RBM). The RBM also has to be sent with full power so the receiving end can accurately measure link quality. The RBM basically carries the neighbor address and other relevant information to calculate link quality. For the remaining stretch of stage 1 the local node listens for messages that are directed to it. When a messages is received, it proceeds to calculate the link quality for the neighbor which sent the message and inserts it in the local set. Let the reader be reminded that the order in which the neighbors are inserted in the local set depends solely on the calculated link quality. In other words the neighbor with a "good" link quality will appear at the beginning of the set and in the same way the neighbors with a "bad" link quality will appear at the end of the set.

The second stage is very similar to the first one in that it behaves as a request and response system. The objective of the second stage is to recollect all the neighbor sets and it does this in the same way as the first stage. At the beginning of the second stage the local node sends a set request message (SRM) with full power, this message contains the local node's set and demands a response from every neighbor that receives it. This response contains the set of the neighbor and will be denominated response to set request message ($RSRM$). In this way, after sending the initial SRM, the local node waits for the $RSRM$ from its neighbors.

At the end of the two stages the local node will have the data structure needed to calculate XTC. The algorithm itself does not require further transmission and the execution does not take as long as the other stages. Moreover the XTC algorithm itself is unchanged and executes in the same way as it would for static environments. At the end of the third stage the local node has a fresh topology representation and will start the whole process again.

Additional to the transmissions that occur at the beginning of the first two stages, the local node responds to any SRM or BM message that it receives. In other words it will transmit an $RSRM$ and a RBM for each neighbor inside its immediate neighborhood. So, assuming that N represents the number of neighbors for the local node, the total number of sent messages for M1 in one interval would be expressed in Equation 4:

$$T_{m1} = (BM + SRM + N * RBM + N * RSRM) \tag{4}$$

Moreover, the size of the messages that contain a set (SRM, $RSRM$) depends on the amount of neighbors. This is because the structure sent in these messages is a set of neighbors and its length depends on the number of neighbors. Equation 5 is more accurate in the sense that it describes the amount of bits transmitted in one iteration of M1. The amount of transmitted bits becomes important when analyzing energy consumption and total time spent transmitting in one interval. Let AS be the size in bits of the node ID (address), OH the size in bits of the overhead for each message and N the number of neighbors for a local node. T_{bl1} will be the total number of transmitted bits for a local node in M1.

$$AS + OH = BM$$
$$(AS * N) + OH = SRM$$
$$(AS + OH)N = RBM$$
$$(AS * N + OH) * N = RSRM$$

$$BM + SRM + RBM + RSRM = T_{bl1}$$
$$AS + OH + (AS * N) + OH + (AS + OH)N + (AS * N + OH) * N = T_{bl1}$$
$$AS + 2 * OH + (2 * AS + 2 * OH) * N + AS * N^2 = T_{bl1} \tag{5}$$

34

Model 2

Given the vast amount of messages needed for M1, Model 2 (M2) is proposed with the sole purpose of reducing the quantity of transmitted data without compromising the result of the rendered topology. For this model the 3 stage approach is disregarded while the interval of repetition is maintained. M2 eliminates one stage and implements an approach that assumes cooperation of all the nodes inside a given transmission area. The Model is expressed in Figure 11.
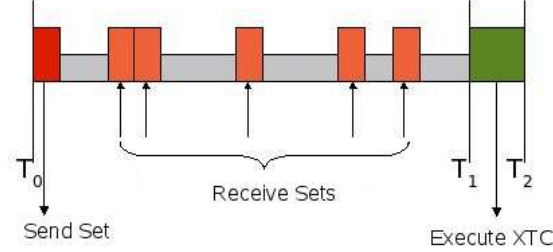

Figure 11. Model 2 description of process.

Instead of having 3 stages, as in M1, this model only has 2. Of the two proposed stages, one is in charge of information recollection. The other executes the XTC algorithm with whatever information is recollected from the fist stage. At the end of stage 2 of M2, the whole process is repeated. The model basically works as follows. All activity is inside a time interval that starts at T_0 and ends at T_2. Stage 1 (S_1), is for information recollection purposes, and stage 2 (S_2), is where the XTC algorithm is executed. At the beginning of S_1 a messages is sent with full power to all the nodes in the immediate vicinity. This message contains the network representation housed in local memory and is the only message sent by the local node, moreover it does not force its neighbors to respond. For the rest of the first stage the local node waits for all its neighbors to send their messages. For this model to function correctly it is assumed that all control messages are received in side the first stage interval and the intervals in all network nodes have the same duration.

Control messages are essential in this process and are the ones that allow the network to be visible to the local node. They are the ones that describe the network. Too few control messages might mean a poor network representation, but too many of them can fill the interval to a point where no other type of message can be sent. The sole objective of a D-TCM is to increase the productivity of the network. If the amount of data messages sent in an interval are reduced because of the amount of control messages the interval time must be increased. Due to the importance of the data messages in the MANET structure, all the control messages have to fit in the interval while at the same time leaving ample space for data messages.

Compared to the first model, M2 manages to reduce the number of control messages. If the two models where directly compared. That is, if the time between T_0 and T_3 from M1 were the same as the time from T_0 to T_2 from M2; the difference regarding the amount of transmitted messages between M1 and M2 is considerable. The total number of messages sent in an interval in M2 (T_{m2}) is expressed in Equation 6.

$$T_{m2} = SRM \tag{6}$$

As in the first model the size in bits of the messages can be calculated. For M2 it is much simpler because only one message is sent per interval. The number of bits transmitted for one interval in M2 (T_{bl2}) is expressed in the next equation. Note that the number of bits transmitted for M1 (Equation 5) grows quadratically with the number of neighbors. On the other hand the number bits transmitted for M2 has a linear relation to the number of neighbors (Equation 7).

$$AS * N + OH = T_{bl2} \tag{7}$$

The objective is for the two proposed models to be compared against the original XTC (O-XTC) algorithm. This will be done by assuming that the O-XTC can be repeated indefinitely throughout the life of the network without phase synchronization. The model used to compare all the models, the behavior and analysis of each model are described in the next section.

Change Model

The comparisons done in this dissertation will be analyzed with the assumption that the rate of change of the environment is constant and can be expressed as discrete units of change. Assume that the rate of change is represented by a two dimensional plane. The x axis represents time and the y axis represents the amount of change. So a function like $f(x) = c$, where c is constant, would define a situation without change. Variability is represented with functions that vary the y value, like that one described in Figure 12. This figure describes a situation in which the variability is linear and constant. Moreover it represents the total change with respect to the initial state in (0,0). So, according to the figure, the difference between the initial point in interval T_1 and the initial point in interval T_4 is $3k$. It is also important to note that at all the time intervals are equal and the same amount of change occurs in each one. These time intervals will be denominated intervals of validity (IV) because each one defines a time in which no apperant change occurs and an accurate analysis can be done inside said interval.
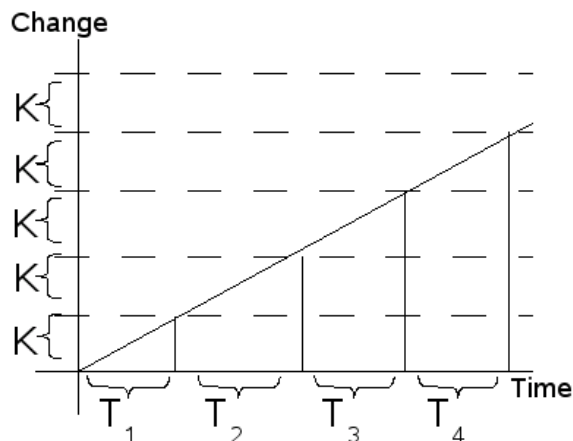

Figure 12. Change representation.

As stated before, to analyze the different D-TCM models involved in this dissertation an interval is assumed where change is not relevant. The local node needs to have the certainty that the representation rendered from the recollected information will be useful for a period of time. This would suggest that the whole recollection phase, the topology calculation phase and the moment in which the topology is used should fit inside the IV. Therefore the actual state of the network has to be very similar to the state in which the information is recollected. If this is not true, the system will fail to give current information, rendering it useless for any type of topology based decision.

As described in the Figure 12, change is continuous with respect to time. That is, inside any interval there are changes occurring to the network. At the end of each interval these changes are represented by an accumulated value k. This value expresses the total amount of change present in one interval. It is assumed that k is the maximum tolerable change for a D-TCM. That is, if a network presents a change that is less than the value of k, it is considered that that network did not actually change. For example a node that moves a couple of millimeters will not have an effect on the validity of the representation nor the performance of the model and therefore no change is considered in these cases. Moreover, for each k value there is a related time interval T_n. If the three main events: information recollection, topology cal-

culation and topology use occur inside the time interval defined by k (IV), then the calculated topology will correctly predict the network and can be used to make topology based decisions.

It is important to clarify that change needs to be linear for the analysis to be valid. If it is non linear, like the one expressed in Figure 13, the time intervals need to fluctuate to account for the variability. This non linear situation will not be considered because, for it to be possible, an algorithm that handles time interval modification based on amount of change has to be implemented. This algorithm not only needs to manage the activity inside the local node but needs to address the fact that network nodes will possibly have different time intervals at a specific time. In other words a situation like the one presented in Figure 13 requires models that are different from the ones presented in this dissertation.
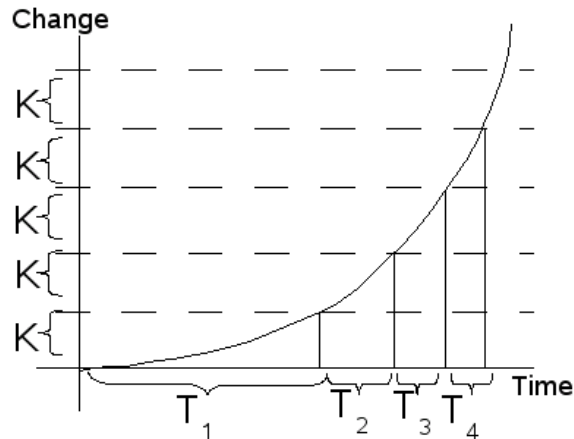
Figure 13. Non linear change representation.

But the change model is not a series of discrete time intervals that define the moment in which the three main topology control events should occur. It is best described by Figure 14 where an IV follows the present state of the network. As mentioned before the IV must contain the three topology control events: recollection, calculation and use. T_0 represents the actual state of the network and is continually moving forward. ΔT describes the duration of the IV which depends on k, the maximum tolerated amount of change. If the models manage to include the three main events inside the IV, then the model will be successful in its objective of saving energy by replacing long range links with short range ones.
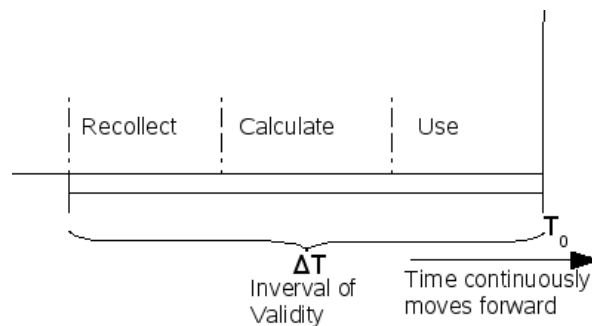
Figure 14. Interval of validity.

At the moment that time moves forward the recollected information becomes unusable. In Figure 15 the information that was recollected between $(T_0 - \Delta T - n)$ and $(T_0 - \Delta T)$ becomes unusable for the current state of the network in T_0. In this case the models have to constantly recollect, calculate and use the network information to avoid using old information and representations. In conclusion all the recollected information in any TCM implemented for dynamic environments and based on the IV model has to be inside the IV.

37
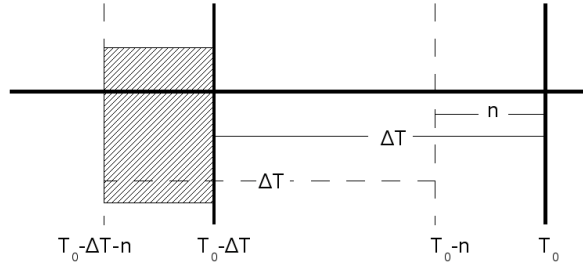
Figure 15. Time moves forward.

The models proposed in this dissertation need to be defined using the IV described in this section. The intervals of repetition of a model do not need to be equal to the IV. For both of the models it is true that all the recollected data needs to be inside the IV, moreover there needs to be a designated time in which the calculated topology is used. Two general assumptions will be described to understand the behavior of the models in the IV.

The first assumption deals with the recollected information. All recollected data must be inside the IV. The sets received from the neighboring nodes need to be inside said interval. In other words, any data contained inside a node, that is used for topology control calculations, must not be older than the IV. The models must behave in such a way that they comply with this assumption.

The second assumption deals with the use of the calculated topology. A period of time must be allocated where the network representation has a chance of being used. Moreover, the period of time in which the calculated topology is used, must be inside the same IV in which the data was recollected. In other words the network in which the calculated topology is used cannot differ significantly from the network in which the data was recollected.

Model 1, Model 2 & Original XTC in Change Model

Before making any comparison the two proposed models and the O-XTC are described inside the Change model. The objective is to make clear the individual characteristics of each model and then compare them using the Change model.

Model 1

Before mentioning the manner in which M1 is going to be used inside the IV, the information recollection process inside the IV will be thoroughly dissected. As mentioned before, M1 consists of three stages, two of which are information recollection stages and the third and last one is where the XTC is executed. The first stage is where the set representing the neighborhood is created. Stage two is where the sets are shared between nodes. For this specific analysis, the time in which the XTC is executed will not be considered. Also for this analysis the first stage will be denominated S_1 and the second stage S_2. As the third stage is not considered in the analysis, it will not be mentioned.

M1 recollects information in S_1 to create the local set, afterwards it proceeds to recollect all the neighbor sets in S_2. The neighbor sets together with the local set make up all the information needed to execute the XTC algorithm. As seen in Figure 16 the information recollection phase goes from t_0 to t_2. It is exactly at t_2 that the XTC algorithm is calculated. Note that the information recollected in S_1 of interval 1 (I_1) does not change until t_3, when the local node refreshes the local set with the information recollected from S_1 of interval 2 (I_2). In other words the local set created at the t_1 will be valid until t_3 when it is replaced by a new set, Figure 16 indicates it with a darker tone.

38

In Figure 16 the interval for M1 is visualized, the XTC algorithm is executed at t_2 and the two stages needed for the execution come before said time. Regarding the local set created with information from the neighborhood, there is no data that is recollected before t_0. This is because the petition for beacon messages in S_1 comes after t_0 and before t_1. In other words, in a worst case scenario the oldest piece of data regarding the local set will be recollected right after t_0.
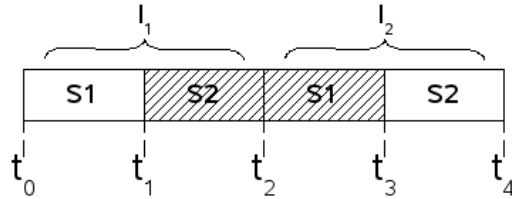

Figure 16. Model 1 general behavior.

So far S_1 of M1 has been described. Equally important is S_2 of M1. The second stage is where the neighboring sets are recollected. In this case the behavior of the neighboring nodes is analyzed together with the local node. Attention is placed on the neighboring nodes because the data that is gathered in S_2 originates in them.

Figure 17 shows a situation where local node A receives data in its S_2 from node B, when it is at the end of its S_1. Let the reader be reminded that what B is sending A was created in t_1 at the end of B's first stage and might have information that dates back to t_0. So in the situation expressed in Figure 17 the information contained in A dates back to t_0. This of course is the worst case scenario because if the petition were to be made a moment after t_5 the oldest contained data would be from t_2.
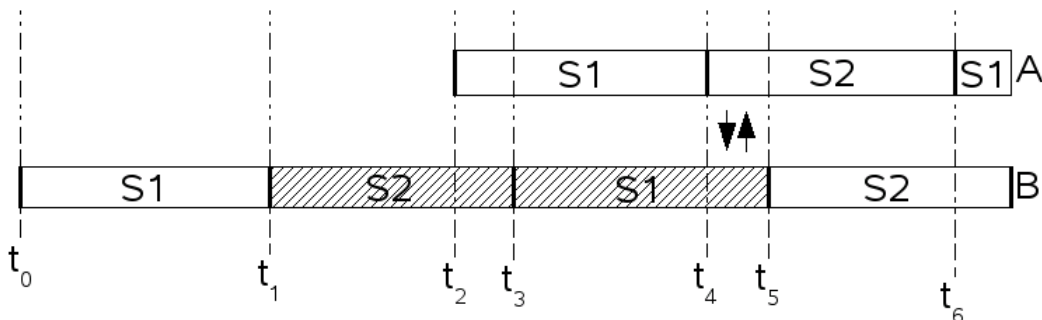
Figure 17. Possibility for Model 1 information interchange.

The oldest possible data element received in S_2 of node A would be that of a neighbor node that communicates a set with the characteristics depicted in Figure 17. In conclusion, at worst, the data contained and used for the XTC calculation at t_6 of node A would be as old as two times the interval of M1 as depicted in Figure 17.

So it would seem that the IV has to be at least as long as two times the length of the interval of M1. This much is gathered from the information recollection analysis. But there is still one more aspect of the Change model that has not been mentioned and that increases that length. The fact that the calculated topology in t_6, Figure 17, needs to be used requires a longer IV. The question then becomes how much time after t_6 is needed to use the calculated topology.

The topology calculated at t_6 must be used in a network that has no considerable variability. Moreover, the next representation is calculated in the second stage of M1 that comes after t_6. So the representation calculated at t_6 must be valid until that time because no new representation is calculated before it. In other words the IV needs to be extended at least for another M1 interval.

A more accurate representation of how M1 behaves inside the IV is expressed in Figure 18. Assume that node A calculated XTC at T_{2A}, so at current time, node A, is using the calculated topology from T_{2A}. At the same time the node is going through the process of information recollection to calculate the next topology in the next M1 iteration. Note that the information used to calculate the topology has information that dates back to T_{0A}. However this time does not surpass T_0 that is the edge of the IV and is still valid inside the network. Also consider that as T_A grows, T_{0A} gets closer to T_0, but when $T_{0A} = T_0$ a new topology is calculated and the oldest data used will be from T_{1A} instead of T_{0A}.
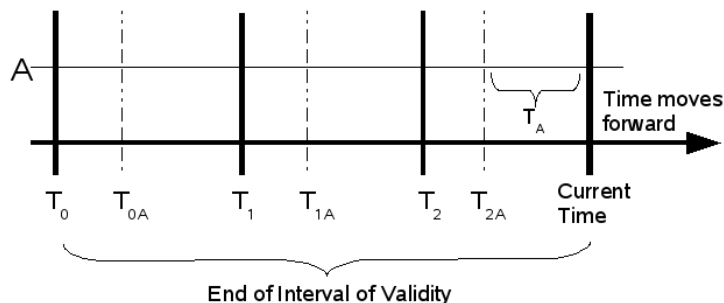
Figure 18. IV in a time continuum.

In conclusion to use M1 inside an IV, 3 M1 intervals are necessary. In this way the validity of the data is not compromised and a time to use the calculated topology is assured.

Model 2

M2 will be described in the same way as M1. As in the M1 analysis the time for the execution of the XTC algorithm will not be considered. This would mean that M2 would be stripped down to only one interval of repetition. The objective is to describe the behavior of M2 intervals inside an IV.

As in M1, M2 creates its set with the messages that it receives from its neighbors in its information recollection interval. All the messages that are received contain neighbor sets, but they also contain information to calculate link quality and create the local set. In this way the local node ends up with a local representation of the network at the end of the interval. At the same time, and using the same messages, the local node recollects the sets of its neighbors. So at the end of a M2 interval it will have received enough information to execute the XTC algorithm. But how relevant is the received information? how old is the recollected data? These questions are answered to describe the behavior of M2 inside an IV.
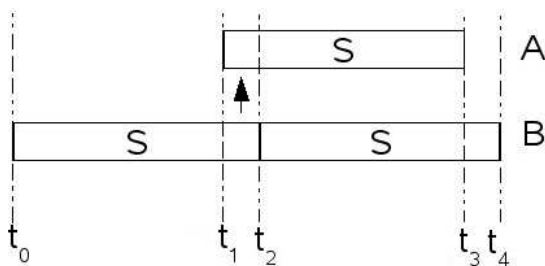
Figure 19. Worst case scenario Model 2.

The worst case scenario can be seen in Figure 19. Node A is just starting its M2 interval at t_1, and before t_2 it receives the first neighbor set. The message is received from B that is about to end its interval at t_2. Given that B has already reached t_2, it transmits the set that it calculated using information from the previous stage that started at t_0 In other words the message that is being transmitted in Figure 19 has information that is as old as one M2 interval. Therefore, the calculated topology in t_3 will have information that dates back to t_0. This means that two M2 intervals should fit inside the IV to assure the validity of the data used by the local node.

40

Remember that the local node not only recollects data but also needs time to use the calculated topology. Figure 19 shows that for information recollection purposes the IV has to be at least as long as two times the length of a M2 interval. For the use of the calculated topology the same analysis used in M1 is applied in M2. As a consequence the time necessary to use the calculated topology inside a M2 interval is one M2 interval. In conclusion the total number of M2 intervals that must fit inside an IV to calculate and use a topology is three.

Figure 18 also depicts the behavior of M2 in a time continuum. The topology that is being used in the figure was calculated in T_{2A}. According to the analysis done to the M2 structure, the data that is contained in a calculated topology is, in the worst case scenario, as old as two M2 intervals. In other words the oldest possible piece of data would date back to T_{0A}. Note that T_{0A} is still contained inside the IV and will not contain invalid data. As in M1, when T_{2A} reaches T_2 the XTC algorithm is calculated and a new topology is generated. At that time the oldest data for the new topology would be located at T_{1A}. In conclusion to use M2 inside an IV, 3 M2 intervals are necessary. In this way the validity of the data is not compromised and a time to use the calculated topology is assured.

Original XTC

As in M1 and M2 the time of execution of XTC is not considered. In other words, the number of phases that the O-XTC will have in the analysis is two. Furthermore, to analyze O-XTC in the Change model, phase synchronicity must be assumed. That is, that all network nodes are in the same phase at any moment. This is done to assure that the messages are received correctly. If full phase synchronicity is assumed O-XTC is a very optimum approach.

By definition the algorithm ensures that inside the interval of repetition all the data will be valid. In other words, O-XTC will render an accurate topology using just one interval, it does not have to wait for two intervals like the two other models. This is mainly due to the structure of the approach. It ensures that in the first phase all the network shares all relative information for the creation of the set in every node. So, after the first phase, all the nodes will have an accurate representation of the network, assuming a network based on the Change model. The same applies for the second phase, it too will share all the relative information within the phase, and at the end of the first two stages the algorithm will have enough information to execute the XTC algorithm.
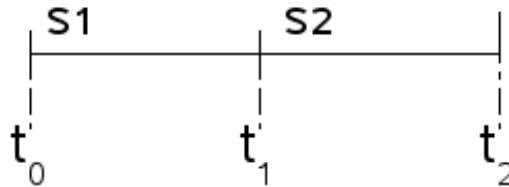
Figure 20. Original XTC in Change model.

Figure 20 describes O-XTC in more detail. At t_2 XTC is calculated with the information recollected in S_1 and S_2. Given that the two stages are synchronized, O-XTC can assure that all the information of the network is received after t_0 and before t_2. Additionally to the recollection process, the node has to use the calculated topology. As in the other models an extra interval is used to allow the node to use the calculated topology. In conclusion, two O-XTC intervals are needed to accurately calculate and use the rendered topology.

To analyze and compare the O-XTC model to the others, it needs to be described in terms of number of total bits transmitted ($T_{\mathrm{blO-XTC}}$). Equation 8 describes this value.

$$SRM + BM = T_{\mathrm{mO-XTC}}$$
$$(AS * N + OH) + (AS + OH) = T_{\mathrm{blO-XTC}} \tag{8}$$

Model behavior

At this point it is clear that the three models have different behaviors inside the IV. On the one hand M1 and M2 need to be executed three times inside the IV, while O-XTC has to be executed only two times. In this way the models assure that the generated topology can be used without any problems regarding data accuracy. Additionally the transmission behavior of the three models has been characterized by three equations. Equation 5 describes M1's behavior. From this equation it is concluded that the amount of bits transmitted grows quadratically with the number of nodes. Moreover, equation 7 describes M2 as a linear relation with the number of neighbors. And finally, Equation 8 describes O-XTC as a linear relation with respect to number of neighbors.

The information for each model has to be adapted to the IV. In other words if M1 is repeated three times, then the amount of bits used is not expressed by equation 5 but by equation 9. T_{IV1} represents the total bits transmitted when using M1 in an IV.

$$T_{IV1} = T_{bl1} * 3 \tag{9}$$

In the same way, the number of bits transmitted by M2 in an IV is not described in equation 7. Given that it has to be repeated three times, the expression that describes the total transmitted bits is given by equation 10. T_{IV2} represents the total bits transmitted when using M2 in an IV.

$$T_{IV2} = T_{bl2} * 3 \tag{10}$$

Finally equation 11 describes the amount of bits needed to render an accurate topology using the O-XTC model. Remember that this model must be repeated twice for the generated topology to be valid.

$$T_{I\,VO-XTC} = T_{blO-XTC} * 2 \tag{11}$$

Considering equations 9, 10 and 11 an analysis can be developed based on accuracy of representation, energy use and time of execution. The analysis of model accuracy is made using the change model and does not consider the time of use. In other words, it is only given from an information recollection perspective. The use of the calculated topology does not matter in this analysis because accuracy can be measured up to the end of the recollection period. In other words, the phases that come after the recollection phase do not influence the accuracy calculation. Therefore, conclusions regarding representation accuracy can be taken from an analysis that only considers the information recollection phase of the models.

Following the accuracy analysis, energy consumption and time of execution can be identified. Ultimately a Model that uses the least amount of energy and can render a topology in the least amount of time is required. It is important to note that, for these two metrics, the models will be compared assuming that they execute in the same IV and therefore their accuracy is equal. So the models can accurately be compared using the time and energy metrics without worrying about the accuracy of the representation.

Accuracy in Models

To compare the models using the accuracy metric, common ground has to be laid down. In general for all phases that require a send and receive process (SRP) there is a common value that bounds the minimum time of execution. In all SRP phases the execution time is required to be at least as large as the round trip time (RTT) of a message. This needs to be true to allow the successful interchange of information. Each phase in M1 and O-XTC models are considered to be one SRP phase. In the same way, M2's phase is also considered as a SRP phase. In other words, S_1 and S_2 of M1 can be compared with the M2 interval. In the same way they can be compared with S_1 and S_2 of O-XTC, as described in Figure 21.

Figure 21. Model comparison.

Figure 21 describes the way that the models are related. According to the analysis, M1 has to be executed two times in order to recollect all the information. The same applies for M2, the only difference between the two models is that M2's phase of recollection only has one SRP phase, where as M1's recollection phase has two SRP phases. In other words, M1 would take double the time to render a topology compared to M2 or O-XTC.

Considering the restriction described in Figure 21 and the behavior described by Equation 5, M1 would not be comparable to the other models. In terms of accuracy it might behave similarly to M1 or O-XTC but the large number of messages that the local node must transmit (Equation 5) for every neighbor; and the fact that M1 must be repeated twice to recollect the information, leads us to believe that M1 is not comparable to the other models in terms of resource optimization. In other words, M1 can render an optimum topology but the energy necessary to do so is out of proportion with the other models and therefore it will not be included in the accuracy analysis.

For the sake of analysis assume that the IV is two times the interval of repetition of M2. This means that two intervals of repetition of M2 fit inside an IV and one interval of repetition of O-XTC fit inside the IV. Remember that, based on the Change model, the amount of change inside the IV is neglected. Now assume that this amount of change can be expressed by a percentage P. $P\%$ then becomes the percentage of allowable change for both models. Therefore the amount of change for every interval is $\frac{P\%}{2}$. In other words the amount of change from t_2 to t_3 (Figure 21) is $\frac{P\%}{2}$.

In Figure 21, for M1 and O-XTC the calculated topology will be the same. This is because the O-XTC needs only one interval to recollect all the information and M2 needs two intervals to do the same. At time t_4 the two models will have calculated the same structure. Remember that the period in which the calculated topology is used is not being considered.

Figure 22. Comparison between M2 and O-XTC.

But if we wait for half of an IV the node will change to the state that is expressed in Figure 22. M2 has just calculated XTC with the information from the previous information recollection process. Moreover, M2, at current time, has a valid representation of the neighborhood. O-XTC on the other hand, is about to start stage 2 and does not have a valid representation of the neighborhood. If a petition were to be made in t_3 for the calculated topology, O-XTC would return the topology calculated at t_2 that, in the worst case scenario, has information from t_0, while M2 would return a structure calculated at t_3 that, at

43

worst, has information from t_1. Remember that the IV goes from t_3 to t_1. Since O-XTC has information that dates back to t_0 it will have an inaccuracy of $\frac{P\%}{2}$. Moreover M2 will not have any inaccuracy because its information dates back to t_1, time that is inside the IV. The worst case scenario is expressed in Figure 23 where the calculated topology is requested just before the two models calculate the XTC algorithm. In M2's case it will have an inaccuracy of $\frac{P\%}{2}$ (from t_1 to t_2) while O-XTC has an inaccuracy of $P\%$.
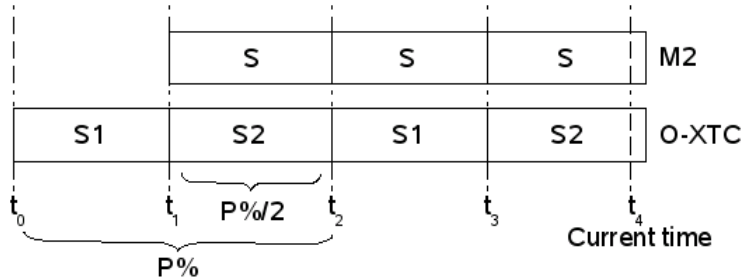

Figure 23. Accuracy in models.

Based on the analysis contained in this section and on the change model, M2 presents less inaccuracy than O-XTC. Energy and time of execution being static in both models, M2 is more accurate in all cases than O-XTC. Therefore M2 will be a more accurate approach to a successful D-TCM that truly takes advantage of the energy saving characteristics of the XTC algorithm.

Considering the worst case scenario described in Figure 23 and if the IV is incremented in such a way that it includes three M1 intervals (from t_1 to t_4 Figure 23) instead of two, the inaccuracy in M2 would be eliminated while O-XTC would still have an inaccuracy of $\frac{P\%}{2}$. In any case M2 is a model that is much more accurate than the O-XTC.

Energy in Models

For the energy and the time analysis, it is assumed that all the models produce the same topology representation. In other words, all the compared models will have the same IV in which all the recollection process is implemented. The objective is to measure the energy used by each model to render a valid topology. M1 will again be disregarded in the analysis because it does not make an optimum use of the energy resource. Instead M2 and O-XTC will be directly compared to see which one saves the most energy.

In the comparison, M2 is executed for every phase of O-XTC, as described in Figure 21. Phase 1 of O-XTC starts at t_2 and ends at t_3, in the same way the first M2 interval is inside this time line. Phase 2 of O-XTC starts at t_3 and ends at t_4, as well as the second interval of M2. Equal accuracy between O'XTC and M2 needs to be assumed. However, in the accuracy analysis it has been shown that O-XTC is not as accurate as M2. To equal the accuracies, O-XTC will send a set in each of its phases. In this way two O-XTC algorithms are running in parallel in each local node. While one O-XTC algorithm is executing S_1 the other executes S_2 in parallel. With this strategy a set will be transmitted for every phase of O-XTC. As a result, to render an accurate topology, M2 executes twice, one execution after the other. And O-XTC executes twice in parallel.

In mobile networks there are many node states that use different amounts of energy. As stated before, nodes have 4 basic energy states; transmitting, receiving, idle and sleep. Of the four states, transmission is the one that uses the most energy. It's due to this and the fact that the recollection phases of the two models are based on transmitted messages, that the energy metric is based only on transmitted bits. To analyze the energy consumption in each model the equations that express the number of transmitted bits have to be related to the amount of energy needed to transmit one bit. Let E_b be the energy needed for one bit to be transmitted, $E_{O-\mathrm{XTC}}$ the energy used in O-XTC and E_2 the energy used in M2. To find the energy consumption in each model it is enough to multiply E_b with equations 7 and 8.

In Equation 12, $T_{\text{blO}-\text{XTC}}$ is multiplied by 2 to account for the parallel execution of two O-XTC models. In the same way, in Equation 13, the T_{bl2} is multiplied by 2 to account for the serial repetition of the M2 model. In Equation 14 energy consumption in M2 and O-XTC is compared. It can clearly be seen that M2 model uses less energy to render a valid topology than the O-XTC.

$$T_{\text{blO}-\text{XTC}} * 2 * E_b = E_{O-\text{XTC}}$$
$$[(AS*N+OH)+(AS+OH)]*2*E_b = E_{O-\text{XTC}} \tag{12}$$

$$T_{bl2}*2*E_b = E_2$$
$$(AS*N+OH)*2*E_b = E_2 \tag{13}$$

$$E_{O-\text{XTC}} \gg E_2$$
$$[(AS*N+OH)+(AS+OH)]*2*E_b \gg (AS*N+OH)*2*E_b$$
$$(AS*N+OH)+(AS+OH) \gg (AS*N+OH) \tag{14}$$

It is observed in the third step of Equation 14 that the E_b value is canceled. This can be attributed to the fact that there is a direct relation between energy and number of transmitted bits. In other words the transmitted energy can be expressed directly as number of transmitted bits. Therefore in Equation 14, even though E_b is not present, it compares the energy transmission qualities of both models. Remember that equal transmission characteristics for all models is assumed. Equation 14 positively identifies M2 as the Model that optimizes energy use as the number of neighbors grow.

Also take into account that a totally synchronized model is assumed for O-XTC. In real situations a synchronization mechanism that uses control messages needs to be implemented to develop a working O-XTC model. This means that additional energy needs to be allocated in the O-XTC model to cope with synchronization tasks, increasing considerably the total used energy.

Time in Models

The time needed to execute each model is also an important metric. It defines the specific environments in which each model can be used. Let the reader be reminded that the IV is defined by a certain amount of change. So when the IV is very short, the amount of change is considerably large. Moreover, when the IV is long, change in the environment is low. Therefore, models that take a long time to execute can only be used for environments with low variability, whereas models that can execute in short amounts of time can be used in low and high variability environments. In other words a model is more optimum as its time of execution decreases.

The way to measure the time that a model needs to completely execute depends on the time needed to transmit one bit. This value, in turn depends on transmission characteristics like signal frequency and physical bit coding. As with the energy metric, time is measured by counting the amount of transmitted bits and multiplying them by the time it takes the local node to transmit one bit. Remember that transmission homogeneity is assumed. In other words, that all nodes have the same transmission characteristics and therefore, all of them require the same amount of time to send a bit.

The fact that the nodes share a transmission medium needs to be considered. To correctly measure the time needed for each of the models it is not enough to just measure the transmitting time of the local node. All activities that make use of the shared medium must be taken into account. From a local node's point of view there are three types of messages that are included in the time calculations. Those that are transmitted by the local node, those that are sent by neighboring nodes and addressed to the local node and those that are sent by neighboring nodes and are not addressed to the local node. All these messages are relevant to the time calculation because non of them can occur at the same time.

The primary objective of any D-TCM model is to create a topology representation. To accomplish its objective all of the nodes in the neighborhood must have executed the D-TCM algorithm completely (information recollection and algorithm). This means that all of the messages related to the D-TCM information recollection phases of all the neighborhood nodes need to be transmitted. In other words all of the messages of all the neighborhood nodes need to have a "slot" in which no other transmission is undertaken. The time used in transmissions that are interfered is not considered.

To measure the time needed to execute a model all the control messages inside a neighborhood should be taken into account. To accomplish this it is necessary to include the messages sent by the local node and the messages sent by the neighbors. In this way the total medium use will be measured and consequently the total time it takes for a model to execute will be known. Let T_b be the time it takes any node to transmit a bit. Using equations 7 and 8 the total time of execution can be calculated. Each equation is multiplied by T_b to represent the total time that a local node uses to transmit all of its messages. Moreover it is multiplied by the number of nodes in the neighborhood to express the fact that no two messages can be transmitted at the same time. The relation of each model with the IV is maintained in the time equations. Also assume that $T_{O-\text{XTC}}$ represents the total time needed to execute O-XTC and T_2 is the total time needed to execute M2.

$$T_{\text{bl}O-\text{XTC}} * 2 * T_b * N = T_{O-\text{XTC}}$$
$$[(AS * N + OH) + (AS + OH)] * 2 * T_b * N = T_{O-\text{XTC}} \tag{15}$$

$$T_{bl2} * 2 * T_b * N = T_2$$
$$(AS * N + OH) * 2 * T_b * N = T_2 \tag{16}$$

$$T_{O-\text{XTC}} \gg T_2$$
$$[(AS * N + OH) + (AS + OH)] * 2 * T_b * N \gg (AS * N + OH) * 2 * T_b * N$$
$$[(AS * N + OH) + (AS + OH)] \gg (AS * N + OH) \tag{17}$$

As in the energy analysis the difference in time comes down to the relation between number of bits. Equation 17 positively identifies M2 as the model that optimizes time as the number on neighbors grows. In other words M2 will behave "better" than O-XTC in high mobility situations and will have a similar behavior as the variability lessens.

Further Analysis

Up until now the models have been compared using the energy and time equation. To further understand the model, the variables in the equations assume a value. In other words the models will be analyzed in the light of a real protocol. Blue-Tooth is a good choice because of its piconet, scatternet structures. It also has a very robust stack that increases the possibility of it being used in MANET environments. Therefore the equation variables will take on Blue-Tooth values.

Theoretically Blue-Tooth 2.0 has a bit rate of 3 Mbs, and with distances of 100 Mts it will use at most 0.1 mW, that is 0.1×10^{-3} Jouls/sec. All this information allows the calculation of the energy per bit value (E_b) and the time used to transmit one bit (T_b).

$$\frac{0.1 \times 10^{-3} \text{Jouls}}{\text{sec}} \div \frac{3 \times 10^6 \text{bit}}{\text{sec}} = E_b$$
$$\frac{0.1 \times 10^{-3} \text{Jouls}}{3 \times 10^6 \text{bit}} = E_b$$
$$0.0\widehat{333} \times 10^{-9} \frac{\text{Jouls}}{\text{bit}} = E_b$$
$$3.\widehat{333} \times 10^{-11} \frac{\text{Jouls}}{\text{bit}} = E_b \tag{18}$$

$$1 \div 3\text{Mbs} = T_b$$
$$3.\widehat{333} \times 10^{-7} \text{sec} = T_b \tag{19}$$

To give value to each model three additional variable have to be defined: the bits in the overhead (OH), bits in the address (AD) and the number of neighbors to the local node. For this specific model OH=150 bits, AD=50 bits and the number of nodes will vary from 1 to 20. This analysis will give the reader a sense of the behavior of the models using Blue-Tooth values.
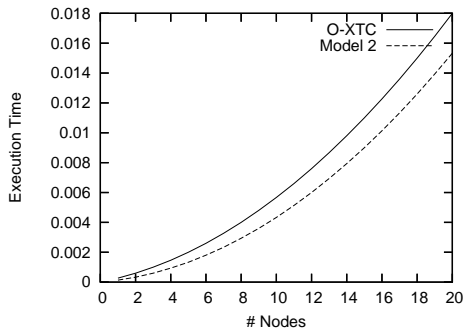

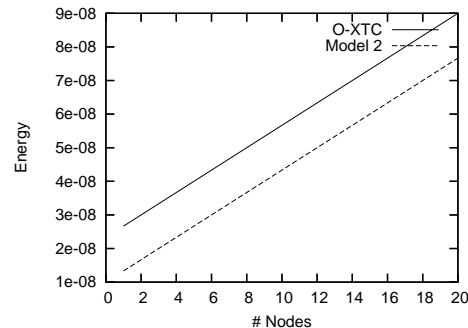**Figure 24.** Execution times.

Figure 25. Energy use.

Figure 24 and 25 show the relation that the Models have with respect to the number of nodes. Figure 24 shows the difference between O-XTC and M2 with respect to the execution time. It describes O-XTC and M2 as models that have non-liner relation with the number of neighbors. M2 grows with O-XTC, but for every node, it executes in less time than O-XTC. This suggests that as the number of neighbors grow, the performance of M2 will be better than O-XTC.

Using the information provided in Figure 24, an IV for each model can be speculated. Actually since the behavior of each model is so similar, the interval of validity for each number of neighbors will be similar. For example if the IV were 0.01 seconds, O-XTC would render a valid topology with 14 neighboring nodes or less, but if the number of nodes increase the model would fail for that particular number of nodes. With M2 the number of nodes supported for an IV of 0.01 seconds would be 15. Not a very noticeable difference with respect to the O-XTC model.

Considering the energy figure, assume that there is a situation where the network must last at least 2 months. The engineers have calculated that the individual nodes must not spend more than 7×10^{-8} Jouls/sec to comply with the 2 month restriction. The individual behavior of the two models are seen in Figure 25. M2 can handle 19 neighbor nodes. O-XTC shows similar behavior, with the 7×10^{-8} restriction it is capable of handling 14 neighbor nodes. In general, and since the functions are linear, the difference between the number of nodes that the models can handle will always be 5. That is, if O-XTC model can handle 4 nodes, M2 will be able to handle 9.

Simulation

The analysis gave lots of insight on the possible behavior of the models. When compared to O-XTC, M2 showed similar behavior. Though M2 is more optimum, the difference showed in Figures 24 and 25 is not very significant. It is important to remember that this model describes the theoretical behavior of the models in assumed situations. It is difficult to accurately predict the real behavior of the models with the analysis alone. As a result, simulations were implemented to further describe and study the behavior of each model in dynamic environments.

Since the simulations are not based on the Change model, they do not use the time of calculation variable. However, the simulations measure the two other variables: accuracy and used energy. Energy is measured in the same way as in the analytical model. The bits that are transmitted are multiplied by the amount of energy it takes to transmit one bit. Accuracy is measured as described in Example 1 of the Variables section. It is the difference between the represented topology and the current "snap-shot" of the network. This comparison would not be possible in real networks because the "snap-shot" with which the local representation is compared, is, to the best of our knowledge, impossible to calculate. In the simulation, on the other hand, the "snap-shot" presents no problems because the current values can be accessed as needed.

The use of a simulator as a tool to support the conclusions of this dissertation needs additional analysis regarding linearity. In the section called "Change model" variability is described and situations are shown in which the proposed models are valid. The section also states that non-linear environments are outside the scope of the dissertation because the proposed models do not have mechanisms able to manage the variability of the I.V. To be consistent with this fact, all elements of the simulations must be linear. To this end, all node movement and traffic have constant values throughout the simulations. This, however, is not enough to assure linearity in the group behavior. The movement and traffic from individuals can be linear, but when all the movement is considered as a whole, the variability of the network varies in time. There could be a situation where most of the nodes are static and no variability is sensed. But there could also be a situation in which most of the nodes move with a certain average velocity, in which case there would be a sensed variability. Moreover, these two situations can present themselves one after the other, changing the I.V. In other words, the simulations have a non-linear behavior with respect to the I.V. variable.

There is an apparent contradiction between the simulation and the analysis sections that can be solved by looking at the problem from another perspective. It is true that the simulation variability is not linear, but the maximum change per unit of time is (assuming constant velocity and traffic). Furthermore, to have sound conclusions from the simulations, the models must be able to handle the maximum change in the simulated network. In other words the difference between the represented network and the current network, calculated by the models, should not exceed a predefined threshold. If the difference were to exceed this threshold the calculated topology would be useless and would not contribute in maximum change environments.

The definition of this threshold is far from trivial. A number representing the tolerable "difference" must be calculated. If the difference variable is less than the threshold, all systems that use the topology have a tolerable behavior. But if the difference variable is greater than the threshold, systems begin to behave in unpredictable and unacceptable ways. Moreover the threshold would vary from system to system. For a routing protocol the threshold might be greater than that of a multimedia transport protocol. In any case, the definition of such a variable is out of the scope of this dissertation and threshold compliance will be assumed for all simulations.

Also consider that what the simulations are showing is the relation that M2 and O-XTC have with respect to energy and accuracy variables. Its objective is not to characterize models as usable or non-usable based on an accuracy variable. With this in mind the condition that characterized the simulations as non-linear can be relaxed without losing any relevance in the conclusions.
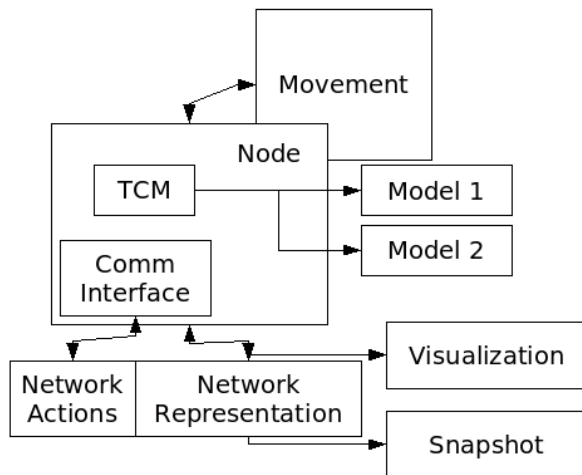
Simulation Model

Figure 26. Simulation Model.

48

The simulator's modules are described in Figure 26. The node module is the one that contains most of the functionality for the nodes. It has submodules that serve different purposes inside the simulation. The TCM submodule is located inside the node and is the generic interface for all TCM models. It is this submodule that contains the calculated representation. To access the representation, the node must make a petition to this submodule. The TCM submodule can be connected to any TCM that implements the interface definition. The communication interface submodule is also located inside the node and it serves as an interface to the network module outside the node. Any outgoing or incoming message must go through the communication interface.

The node has variables that describe its position, these are located inside the node. The module that is in charge of changing these node variables is called the movement module and is located outside the node. This module describes the movement for all the nodes in the simulation. It also describes the periodicity with which the node's positions are changed. A Random Walk Mobility Model and a simple Group Mobility Model were implemented for the simulations. Only the Random Walk model was used.

The network module is the one that has the description of all the network. It manages an array of nodes that represent the network. Moreover, the message delivering system is implemented in this module. The node's communication interface submodule communicates directly with the network module to transmit any message. When a message is transmitted the network module calculates the range of the transmission based on the energy of transmission that the node specifies. The network module then delivers the message to the nodes that are inside the range of the sent message.

The visualization module is implemented for graphic visualization purposes. Its function is to periodically probe the network for the positions of the nodes and their represented network. The visualization module gives a complete view of the simulated area, the movement of the nodes and the connections that the nodes have amongst themselves.

The snapshot module is used to take snapshots of the network and keep them in memory for posterior use. Its interest is in the node position variables and the node representation of the network. It is with these snapshots that, at the end of each simulation, the accuracy of a model is calculated.

Finally the TCM modules (M1, M2) are the ones that house each TCM, Each mechanism is implemented to be completely self contained. In other words, all the variables and the related processes are contained inside the individual module.

Measured Variables

As in the previous analysis, the simulations measured certain variables that describe how optimum one model is compared to the other. The experiment's objective is to evaluate the behavior of O-XTC and M2 in MANETs. The experiment will use consumed energy and network representation accuracy to compare the behavior of the two algorithms.

Energy consumption is the first variable and it will be measured by following all the outgoing control messages in a selected node. Only the part of the message that is strictly related to topology control information will be considered. The calculation is done by multiplying the bits in a message (BPM) with the energy required to transmit a bit (EPB). In this way the total message energy (TME) is recorded each time a message is sent (Equation 20). When the simulation ends the total energy spent (TES) by the node in topology control messages can be measured by adding all the TME's recorded as expressed in Equation 21.

$$TME = BPM \times EPB \tag{20}$$
$$TES = \sum TME_n \tag{21}$$

Let NR_{0u} be the neighborhood for node u at time t_0 calculated using one of the models, let AN_{0u} be the real neighborhood for u at time t_0. Let $\text{diff}_u(NR_{0u}, NA_{0u})$ be the difference between the two lists in u. The function counts the number of elements that are in one list but not in the other. The used diff() function is described in Equation 22.

$$\text{diff}(a, b) = [\text{COUNT}(w \mid w \in a \wedge w \notin b) + \text{COUNT}(w \mid w \in b \wedge \notin a)] \tag{22}$$

To count the total difference of the network at a specific time, all the diff() values from all the nodes in the network must be counted. Equation 23 describes the total difference in the network at time t_m (TD_m) and n represents all the nodes in the network.

$$TD_m = \sum \text{diff}_n() \tag{23}$$

In each simulation the TD_m value was calculated every determined unit of time. So at the end of each simulation there is a group of TD_m values. These values were used to calculate an average value (AD) showed in Equation 24. where m represents all diff() calculations done in the simulation and k represents the number of times the diff() function was calculated. AD becomes the second metric used in the simulations.

$$AD = \frac{\sum \text{TD}_m}{k} \tag{24}$$

Energy Model

Energy is important because lots of network aspects depend on it. The metric used to classify nodes, the transmission range, the used energy in a message and the method used to decide weather a message is received or not, are all important aspects that depend on the energy concept. Let $OTRP_u$ be the Optimal Transmission Reception Power for node u, in other words $OTRP_u$ is the value that is used to decide if a transmission is received or not. Let RE_{ut} be the power density with which a transmission t is received in u. Basically when $RE_{ut} > OTRP_u$, t is received successfully otherwise t will not be received. In the simulation model $OTRP_n$ has the same value for all the nodes and will be considered as $OTRP$ from this point forward.

In the energy model, RE_{ut} depends on the transmission distance and energy with which neighbor v transmits the message. Let TE_{vt} be the transmission energy that node v used for transmission t and let r_{vu} be the distance from node v to u at the time of transmission. The power density with which node u receives node v's transmission is given by Equation 25 [17].

$$RE_{ut} = \frac{TE_{vt}}{4 * \prod * r_{vu}^2} \tag{25}$$

With respect to message transmission there are two modules that interact: the node module and the network module. The node module creates the message and defines TE_{vt} to be used for transmission. Using TE_{vt} and $OTRP$ the network module can calculate the maximum radius of the transmission and therefore identify the nodes that are influenced by the transmission. Let r_t be the maximum radio of transmission given by Equation 26. After the r_t is calculated the RE_{nt} for all the n nodes that were influenced is calculated using Equation 25 on each node. All of the above calculations are made base on [17].

$$r_t = \sqrt{\frac{TE_{vt}}{4 * \prod * OTRP}} \tag{26}$$

Link Quality Model

In the simulation, link quality is considered to be the energy of transmission. That is, the link quality for a neighbor is represented by the energy needed to send a message to that neighbor. Given that the energy used to send a message to a neighbor can intuitively be considered the effort that the local node has to endure to successfully transmit that message, the energy of transmission is a usable metric for link

quality. The neighbor that is closest will be the one that needs less energy of transmission and the one that is farthest needs more energy to be reached. So the smaller values are considered to be the best quality links opposed to the larger values that represent links that are far away. The link quality calculation is described in Equation 27. Where LQ_{uv} is the link quality that u has of v ($LQ_{uv} = TE_{ut}$). Remember that transmission energy depends solely on node position, no obstacle interference or noise interference values are considered. In general each time a local node u receives a messages from v it calculates LQ_{uv}.

$$LQ_{uv} = \frac{OTRP * TE_{vt}}{RE_{ut}} \qquad (27)$$

Mobility Model

To simulate MANETs it is necessary for the nodes to be able to move through the simulated environment. The node movement is made possible by a module in the simulator that modifies the position of the nodes periodically. While there are lots of mobility models [19] the simulator only uses Random Walk Mobility Model. The behavior of the nodes when these come in contact with the simulated area limits, is to bounce off with the same angle as they arrived.

Transmission Model

Not all wireless transmission environments are the same. Individual characteristics like maximum range, power use and bit rate depend mostly on signal frequency. To specify the transmission characteristics, the model requires the maximum range of transmission (r_t), Power needed to transmit a distance of r_t for one second (ES) and bit rate (BS). As stated before the transmission model is based on Blue-Tooth 2.0 values. Equations 28 and 29 show how the simulator calculates $OTRP$ and the maximum energy necessary to transmit one bit MAX_B.

$$MAX_B = \frac{ES}{BS} \qquad (28)$$

$$OTRP = \frac{MAX_B}{4 * \prod * r_t^2} \qquad (29)$$

Additional Simulator Aspects

Considering that the main objective of this simulator was to provide a proof of concept, the decision was made to develop it as simple as possible. There are two characteristics that the simulator does not consider inside its simulations. These characteristics might render behavior that moves away from reality, however it can give insight on problems with the models that can be solved before developing a real prototype.

The simulator does not implement an interference model. This basically means that nodes that are in close proximity can receive messages simultaneously from different sources. Additionally Time in the simulator was not modeled as a discrete variable. However the simulator was implemented in Java and therefore it does have the default behavior of its threads with respect to simultaneous events.

Simulation Process

Each simulation is a group of nodes that have the same velocity range and mobility model (Random Walk Mobility Model) but in general move in different directions, different speeds and different initialization points. The simulated area is of $200\,\text{mts}^2$, each simulation runs for 5 minutes. The transmission characteristics of the simulations are very similar to Blue-Tooth theoretical values. A range of 50 Mts with 100 mW of power spent per second and a bit rate of 0.8 Mbs. Additionally all the nodes have a velocity within a range of $5m/s$ to $10m/s$. Finally, the time for each O-XTC phase is 300 ms and the time for the M2 interval is 300 ms.

A total of 10 experiments were run for M2 and for O-XTC. M1 was not considered because of its lack of optimum performance. The number of nodes are modified from 10 to 20 and energy and accuracy are measured. Another important aspect to mention about the simulations is that O-XTC was not implemented in parallel like in the change model. That is, O-XTC is executed only once per interval in the simulations. This might have a significant effect on the accuracy of O-XTC and will definitively have an effect on the amount of used energy. Finally the O-XTC's synchronization is implemented using a wait period that involved all the network nodes. In other words, all the network nodes finish each phase for the algorithm to continue executing, in this way making sure the phases are synchronized.
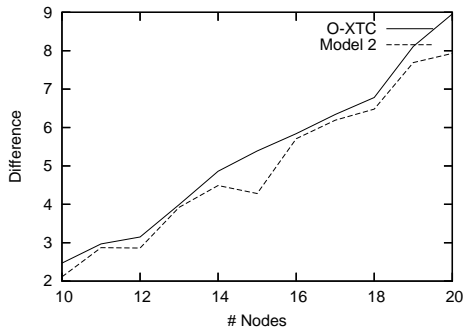

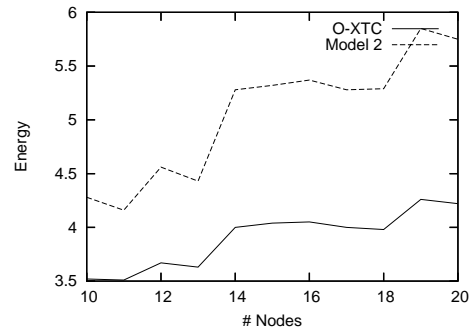**Figure 27.** Accuracy in models.

Figure 28. Energy use in models.

Figures 27 and 28 show that O-XTC can render a relatively better network representation than its counterpart. Let the reader be reminded that Figure 27 shows a situation in which the velocity of the nodes have a range from $5m/s - 10m/s$. A relatively slow simulation to portray the true differences between the two models. Figure 29 shows the behavior of the two models in a situation where velocity ranges from $20m/s - 25m/s$ in a high density situation (14-20 nodes). It is here where the true difference between the two models is visualized. With a network that presents low variability the models have lots of time to calculate a representation, so there wont be a great difference between them in terms of accuracy. But as variability increases the time in which the representation is calculated becomes important and the model that takes less time executing will be the one that renders the best topology. In Figure 29 M2 calculates a topology that is superior to that of O-XTC.
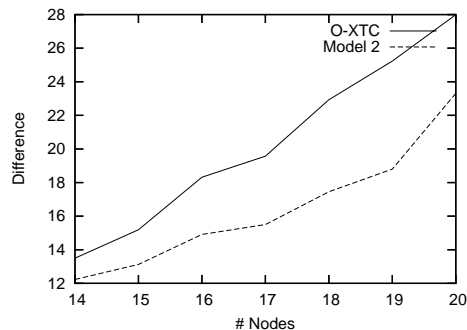

Figure 29. Accuracy in high variability.

Figure 28 describes M2 as an energy hungry model and O-XTC as one that optimizes the use of energy. Be aware, however, that Figure 28 can be misleading in the sense that it describes an O-XTC model that only executes once and is synchronized using methods that cannot be implemented in real situations. Moreover and based on the change model analysis, O-XTC would have to be executed twice to render a topology as accurate as M2. This means that, for the O-XTC model to behave in the same way as the M2 with respect to accuracy, the O-XTC must spend at least twice the energy described in Figure 28. Additionally, O-XTC would have to implement a mechanism by which all network nodes are synchronized, increasing even more the amount of energy used. After adding the two additional elements to O-XTC, it would be very similar to M2 like described in Figure 25.

Conclusions

Conclusions for this dissertation depend solely on the assumptions for each model. It is assumed for M2 that all nodes have the same interval time. The same is assumed for the O-XTC, but additionally all nodes in the network must be synchronized in the same phase. With this in mind we present our conclusions.

In situation with low variability, like that one described in Figure 27, the two models have a similar behavior regarding accuracy of representation. Figure 27 shows a network that is moving at a very slow rate ($5m/s - 10m/s$) and accuracy is very similar between. Since the network changes so slowly the two models have time to make the same calculations. Only when the network changes rapidly does the M2 model optimize accuracy.

Figure 28 shows how the two models behaved with respect to energy consumption. M2 is portrayed as a non efficient algorithm with respect to O-XTC. The reason for the difference between the energy use in the two models is the repetition of each one in the simulation. M2 has an interval of 300ms that is repeated twice for every O-XTC interval. O-XTC, on the other hand, has two phases that last 300ms each. And considering that in the first phase O-XTC only transmits a BM, when M2 transmits a set message, the O-XTC model will transmit less bits than M2 and therefore use less energy. But if it is assumed that the O-XTC executes twice in parallel, the two models will behave similarly in energy utilization, like described in Figure 25. Also consider that the results shown in Figure 28 describe the behavior of an O-XTC model that lacks the synchronization mechanism. In other words, its energy use will increase with the implementation of this mechanism.

The reason to execute O-XTC twice in parallel was to give it the same accuracy that was reached with M2. Keep in mind that, in the worst case scenario the accuracy difference would be $P\%$, or an O-XTC interval (Figure 23). But if the network had a variability with an IV that was, at least, two times the length of the interval of O-XTC, O-XTC would not have to execute twice in parallel to keep up with the movement. This is true because, even though O-XTC is not executing twice in parallel, it is executing twice inside the IV.

Looking at Figure 28 an area between the function that describes the energy use of O-XTC and the function that describes the energy use of M2 can be visualized. Assume a function f is located over the O-XTC function. f describes the behavior of O-XTC including the synchronization mechanism. If f is below the M2 function, then O-XTC will be the best mechanism for low variability MANETs.

But as variability increases the IV gets shorter. If the IV is not at least two times the length of the O-XTC interval, then the model begins to render a inaccurate topology. It is in these situations where M2 becomes relevant. M2 becomes a valid alternative in high variability environments where it spends more energy than O-XTC but renders a more accurate topology. So, for D-TCMs to accurately generate a workable topology in high variability environments, M2 would be the only choice between M1, M2 and O-XTC. O-XTC could still be used in high variability environments but it would calculate an inaccurate topology compromising the energy saving capabilities of XTC.

The two models assume that all nodes in the network must have the same interval at all times. This implies that, for O-XTC model, all nodes have the same time interval for each phase. In the same way, in M2, all the nodes in the network must have the same time interval. So what happens when network variability changes? Don't the intervals have to change with the network? Future work consists in implementing a dynamic interval mechanism that measures the variability of the networks and changes the interval time. With this, a mechanism to measure network variability must be implemented. Moreover a mechanism that handles a network in which all nodes can have different time intervals is necessary. An additional system for the synchronization of O-XTC must also be considered. The comparison between the upgraded systems will positively identify which of the two models is best suited to serve as a D-TCM for MANETs.

Bibliography

[1] Jamal N. Al-Karaki and Ahmed E. Kamal. Routing techniques in wireless sensor networks: A survey. Technical report, Dept. of Electrical and Computer Engineering. Iowa State University., Ames, Iowa 50011.

[2] Somprakash Bandyopadhyay Amitava Mukherjee and Debashis Saha. *Location Management and Routing in Mobile Wireless Networks*. Artech House, 2003.

[3] Kenneth Barr and Drste Asanovic. Energy aware lossless data compression. *Proceedings of MobiSys 2003: The First International Conference on Mobile Systems, Applications, and Services*, May 5-8 2003.

[4] IEEE-SA Standards Board. Part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications. Technical report, IEEE, 2003.

[5] Prathima Agrawal Christine E. Jones, Krishna M. Sivalingam and Jyh Cheng Chen. A survey of energy efficient network protocols for wireless networks. 2001.

[6] Hannes Hartenstein Cristian Bettstetter and Xavier Perez-Costa. Stochastic properties of the random waypoint mobility model. Technical report, Kluwer Academic Publishers., 2004.

[7] Tomas Johansson and Lenka Carr-Motyckova. Reducing interference in ad hoc networks through topology control. *DIALM POMC*, September 2005.

[8] Jennifer J.-N. Liu and Imrich Chlamtac. *MOBILE AD HOC NETWORKING WITH A VIEW OF 4G WIRELESS: IMPERATIVES AND CHALLENGES*. 2004.

[9] Cintia B. Margi and Katia Obraczka. Instrumenting network simulators for evaluating energy consumption in power-aware ad-hoc network protocols. In *MASCOTS*, 2004.

[10] Roger Wattenhofer Martin Burkhart, Pascal von rickenbach and Aaron Zollinger. Does topology control reduce interference? In *MobiHoc*, May 2004.

[11] Emilio Leonardi Michele Garetto. Analysis of random mobility models with pde's. *MobiHoc*, May 2006.

[12] Ramjee Prasad and Marina Ruggieri. *TECHNOLOGY TRENDS IN WIRELESS COMMUNICATIONS*. Artech House, 2003.

[13] Rajmohan Rajaraman. Topology control and routing in ad hoc networks: A survey. Technical report, College of Computer Science, Northeastern University, Boston MA 02115, USA.

[14] Loureiro Geraldo R. Mateus Olga Goussevskaia Max do V. Machado Raquel A. F., Mini Antonio A. F. and Jose Marcos Nogueira. Data dissemination based on the energy map. *IEEE Communication Magazine*, July 2005.

[15] Gartner Press Room. Quick statistics. http://www.dataquest.com/press$_g$artner/quickstats/phone.html, 2003.

[16] Kenneth J. Supowit. The relative neighborhood graph, with an application to minimun spannint trees. Technical report, University of Illinois at Urbana Champaign,, Urbana, Illinois, 1983.

[17] Wayne Tomasi. *Sistemas de Comunicaciones Electronicas*. Prentice Hall Hispanoamericana, S.A., 2 edition, 1996.

[18] Jeff Boleng Tracy Camp and Vanessa Davies. A survey of mobility models for ad hoc networks research. *Wireless Communication and Mobile Computing (WCMC)*, 2(5):483–502, 9 2002.

[19] Roger Wattenhofer and Aaron Zollinger. Xtc: A practical topology control algorithm for ad-hoc networks. Technical report, Department of Computer Science, ETH Zurich, 8092 Zurich, Switzerland.

[20] Xiang Yang Li Wen Zhan Song, Yu Wang and Ophir Frieder. Localized algorithms for energy efficeint topology wireless ad hoc networks. *MobiHoc*, 2004.

[21] Wikipedia. Media access control sublayer. http://en.wikipedia.org/wiki/Data$_l$ink$_l$ayer#Media$_A$ccess$_C$ontrol$_s$ublayer.

[22] Ning Xu. A survey of sensor network applications. Technical report, Computer Science Department, University of Southern California.