TOPOLOGY CONTROL IN WIRELESS SENSOR NETWORKS

Mathias Becquaert, Bart Scheers, Ben Lauwens
Royal Military Academy
Department CISS
Renaissancelaan 30
B1000 Brussels, Belgium
E-mail: mathias.becquaert@mil.be, {bart.scheers, ben.lauwens}@rma.ac.be

ABSTRACT

Ad hoc networks are considered to be a very promising technology. Despite the importance and the variety of possible applications, they still suffer from some basic problems. One of the most important issues of wireless sensor networks is their restricted amount of energy resources, and therefore their restricted autonomy. This paper tries to reduce that problem by introducing topology control into the Wireless Sensor Networks (WSN). Many studies have been performed on topology control. This paper gives an overview of the most important algorithms for topology control. The performance of the different algorithms will be compared and their practicability will be discussed. Most of the literature about the topic of topology control stays focuses only on the theoretical part. This paper goes further and discusses an implementation of an algorithm for topology control on a real sensor network.

INTRODUCTION

A Wireless Sensor Network (see figure 1) is a special kind of ad hoc network. An ad hoc network is formed by a group of mobile clients communicating, using wireless connections. Wireless sensor networks are in most cases composed by a huge number of clients sending measurement results to a central node, called the sink. These networks possess some specific characteristics. In most cases they are dense, the network topology can change, the nodes only have a restricted view on the entire network, the energy resources are often batteries and therefore limited in lifetime. In order to improve those networks, an answer has to be found to the problem of the restricted power supply.

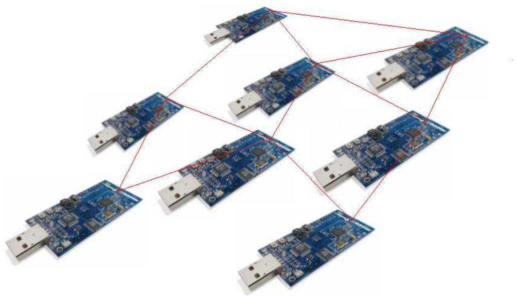


Figure 1. Wireless Sensor Network

In the paper we will first introduce topology control and explain how topology control can reduce the energy consumption while sending messages. In the literature many topology control algorithms are discussed. In this paper, some of those algorithms will be presented and classified according to their practicability. The topology control algorithms are implemented in Matlab in order to study and compare their ability to reduce the overall power consumption when all nodes in the network are sending messages to a sink. The purpose of this study was to pinpoint the most promising topology control algorithm to be implemented on a real sensor network. In the last part of the paper we will focus on the implementation of that algorithm on a real sensor network, using Tmote Sky modules as sensor nodes and Contiki as operating system on those nodes. The paper will conclude with some important remarks and recommendations for future studies.

USE OF TOPOLOGY CONTROL

In this paper topology control is introduced into Sensor Networks in order to reduce the power consumption in the nodes. In a wireless environment, the received signal strength may be expressed as [1]

$$P_{Rx} = P_{Tx} + G_{Tx} + G_{Rx} + L \qquad (1)$$

where P_{Tx} is the transmitted power, G_{Tx} and G_{Rx} are the transmit and receive antenna gains and L is the path loss in dB. In general, the path loss of the transmission channel is governed by

$$1/D^{\alpha}, \qquad (2)$$

where α is called the attenuation factor. Hence, the power necessary to send a message over a certain distance D is not a linear function of that distance, but is proportional to D^{α}. In practice, α takes a value between 2 and 6 [1]. Figure 2 shows the impact of using an intermediate node to send a message from *u* to *v*, admitting that the attenuation factor equals 4. In the first case, the necessary power is proportional to D^4, whereas in the second case the power is proportional to only $D^4/8$.

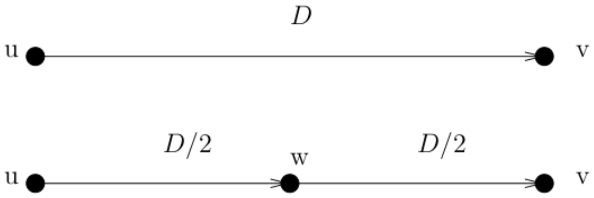


Figure 2. Example of the use of topology control

Hence, the aim of topology control is to eliminate (forbid) long distance links, without compromising too much the connectivity of the nodes to the rest of the network. By adapting the power to send a message over shorter links, permitted by the topology control algorithm, the overall power consumptions for sending messages will be reduced. The drawback of topology control is the reduction of the throughput and the increased delay due to the introduction of additional intermediate hops, although this will be in most cases not an issue in sensor networks.

In literature many algorithms for topology control have been studied. They can be subdivided depending on whether they are calculated in a centralized or a distributed way. If the topology is calculated in one coordinator node, it is called a centralized algorithm. This implies that the coordinator node needs to have an overview of the whole network before it can start the calculation of the topology. In practice, this can be a problem. In a distributed algorithm, the topology is calculated locally in every node. Only a local view, e.g. on the neighbours, is needed. A second classification can be made depending on the parameter used to calculate the topology. This parameter can be the distance between the nodes (e.g. derived from their positions) or the path loss between the nodes. Using the path loss automatically implies that the algorithm takes the environmental influences into account. Note that in practice, the path loss can be easily derived from the Received Signal Strength Indicator (RSSI) if the transmitted power P_{Tx} is known. Thus, if one wants to implement an algorithm for topology control, it is preferable to choose an algorithm that is at the same time distributed (locally calculated), and based on the path loss between nodes.

ALGORITHMS FOR TOPOLOGY CONTROL

Algorithms for topology control are based on graph theory. The ones studied in the scope of this paper are [2]: the (Local) Minimum Spanning Tree, the Relative Neighbourhood Graph, the Gabriel Graph, the Delaunay Triangulation [3], the Yao Graph, the XTC algorithm [4] and the Minimum Power Topology (see figure3).

The first subplot of figure 3 represents 10 nodes, randomly distributed over an area of 100m by 100m. The second subplot, the unit disk graph, represents all possible links between the nodes, given that the maximum transmission range is 50m. The other subplots represent the graphs obtained by a given topology control algorithm.

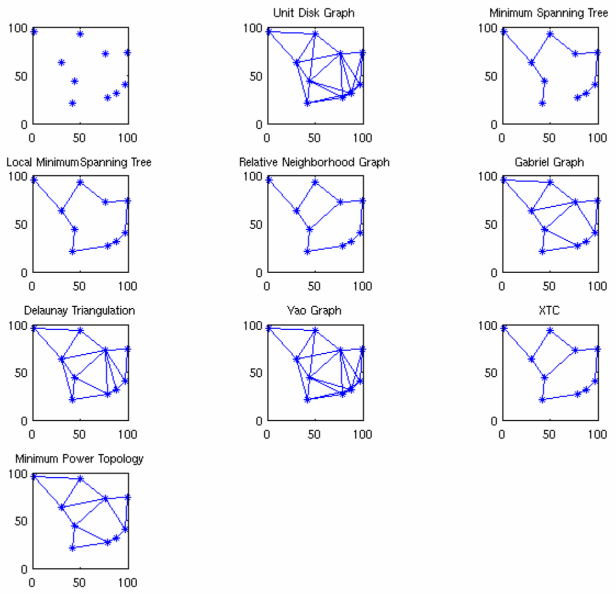


Figure 3. Algorithms for topology control

The Minimum Spanning Tree (MST) algorithm connects all nodes avoiding the formation of closed circuits. MST is not local and not based on path loss (RSSI). A local version of the algorithm exists and the algorithm can be based on the path loss (i.e. Local MST).

The Relative Neighbourhood Graph connects every two nodes *u* and *v* if there is no node *w* implicating that the connection *uv* forms the longest side of the triangle *uvw*. In other words, if there is no node inside the intersection of the circles centred on the nodes with a radius that equals the distance between the nodes.

The Gabriel Graph connects two nodes *u* and *v*, on condition that there is no node inside the circle centred on the centre between *u* and *v* and with a diameter that equals the distance between those nodes.

The Delaunay Triangulation, is the dual of the Voronoi Diagram [3] and connects the nodes *u* and *v* if a circle with *uv* as chord exists without any other node inside.

The Minimum Power Topology is an algorithm that connects the nodes *u* and *v* if the following expression holds true for any other node *w*:

$$P(uv) \leq P(uw) + P(wv) \qquad (3)$$

P(xy) is the power needed in order to send a message over the distance |*xy*|. In order to obtain *P(xy)*, a model for the path loss is needed. That model follows the formula (2). In practice, *P(xy)* can be obtained by the RSSI parameter in the receiving node (Received Signal Strength Indicator).

The Yao Graph is a direction based algorithm. The space around every point is subdivided into cones. In each cone the node selects the closest node to connect itself.

The XTC graph [4] is obtained in three stages. Firstly every node gives a ranking to each neighbour. This ranking can be done based on different parameters, like distance or path loss. Secondly the neighbours exchange their ranked list of neighbours. Finally, the nodes select some connections. The selection is made as follows: every node neighbour is examined in the order according to the ranking. If a candidate ranked an already examined node higher than the node, the node will not connect with that candidate. For symmetrical transmission channels, the selection of a link is consistent in both directions, meaning that if *u* selects a link to *v*, *v* will also select the link towards *u*. This algorithm can easily be based on the path loss between two nodes and it is at the same time a distributed algorithm.

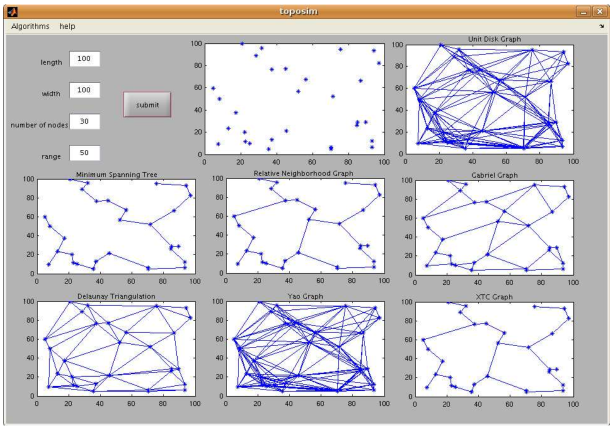
STUDY OF THE ALGORITHMS FOR TOPOLOGY CONTROL

In order to study and compare the different algorithms described in the previous section, a program has been written in Matlab, implementing the graphs obtained by the different algorithms. The user interface of the program is shown in figure 4.

The comparison of the algorithms is based on three parameters: the practicability, the node degree and the energy efficiency.

To have a practical topology control algorithm, which can be implemented on a real WSN, two conditions must be fulfilled by preference: first the algorithm must be distributed. Secondly the algorithm must be based on the path loss and not on the distance between the nodes, as in practice the path loss is more easily obtained than the distance between the nodes. From all the above discussed algorithms, only the Local MST, the

XTC algorithm and the Minimum Power Topology algorithm are distributed and based on the path loss.



length 100
width 100
number of nodes 30
range 50

Figure 4. Program implementing the topology control algorithms

The second parameter is the node degree [2]. The node degree is defined as the average number of neighbours of each node. Fewer neighbours will mean a smaller probability for interference between neighbours. However the node degree may not be too small to guarantee a sufficient degree of redundancy in the network. Figure 5 shows the mean node degree for a number of nodes *N* going from 10 to 30, for different algorithms. It seems that the mean node degree tends to a constant value for the XTC, MPT and MST algorithms. MST seems to be a bad choice for WSN, because of the lack of redundancy in the network. The node degree tends to 2 by definition. Hence, if one link is down, the network will immediately be divided into two non-communicating parts.

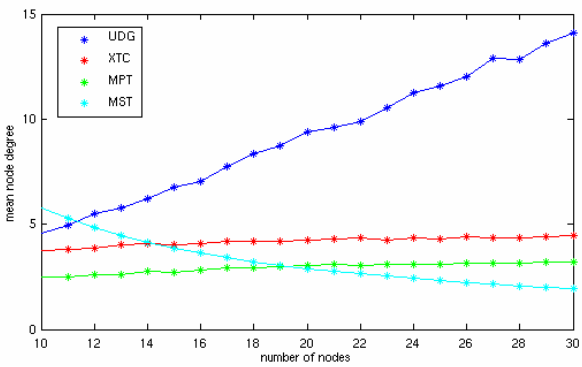


Figure 5. Mean node degree versus node density

A third way to compare the topology algorithms is based on the energy efficiency. This will be the subject of the next section.

The simulations also revealed that some algorithms result in a same graph (see figure 4). A first observation is the similarity between the graphs obtained by the Relative Neighbourhood Graph and the XTC algorithm. This can be easily explained. The Relative Neighbourhood algorithm will not connect to nodes *u* and *v* if a node *w* exists so that $|uw|<|uv|$ and $|vw|<|uv|$. Using the RSSI parameter at the receiving node for estimating the path loss between the nodes will not change the ranking of the nodes: $|uw|^{\alpha}<|uv|^{\alpha}$ and $|vw|^{\alpha}<|uv|^{\alpha}$. Hence the XTC algorithm will yield the same ranking of the nodes. Therefore, the XTC algorithm is not a new algorithm, it is similar to the Relative Neighbourhood Graph, but uses a different parameter to rank the nodes.

Also the Gabriel Graph, based on distances between the nodes, and the Minimum Power topology algorithm, based on the path loss between the nodes, results in the same graph. Indeed, no matter the value of the power attenuation factor in the path loss, the condition to select a connection stays the same for both algorithms.

XTC VERSUS MINIMUM POWER TOPOLOGY (MPT)

So far the comparison of the algorithms showed that the best choice for a topology control algorithm is the XTC or the Minimum Power Topology algorithm.

To compare the energy efficiency of the XTC and the MPT algorithms, a random constellation of *N* nodes on a 100m by 100m area was generated over 50 times, with *N* varying from 5 to 30. The transmission range of the nodes was set to 50m. For each constellation the XTC and the MPT topology was determined and the mean power needed to send a message from all nodes to all the other nodes was calculated on both topologies. From this mean power, two parameters can be derived [4]: the “power ratio” and the “power stretch factor”.

The power ratio is the ratio of power used with an algorithm for topology control to the power without using topology control. Hence it is a measure for the gain in overall transmission power consumption obtained by the topology algorithm.

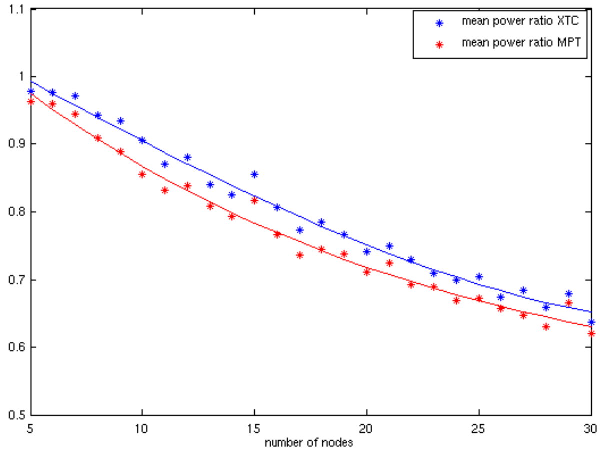


Figure 6. Mean power ratio versus node density

Figure 6 shows the results for the power ratio for ascending node densities (N varying from 5 to 30 nodes). In the simulation, the attenuation factor for the path loss was set to 2. From figure 6 it can be seen that the MPT algorithm is slightly more efficient than the XTC algorithm, although the difference is small. In both cases the gain in power consumption increases with *N* and goes up to 35% for *N*=30. For attenuation factors α=4, this gain is even more important and can be as high as 75% for N=30.

The stretch factor is the ratio of the power needed when using topology control to the minimally possible needed power. The more the mean stretch factor tends to 1, the more the algorithm is efficient. Figure 7 shows the mean stretch factor for the two topologies.

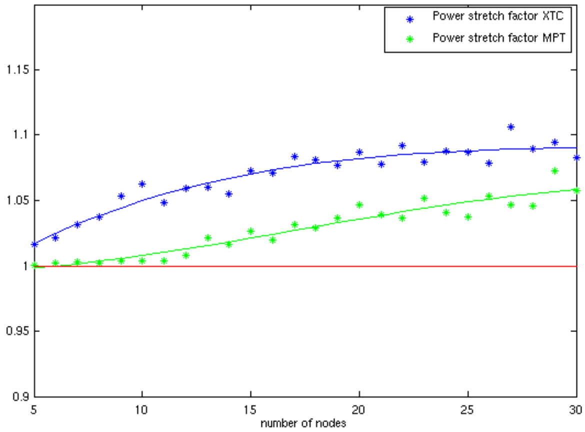


Figure 7. Mean stretch factor versus node density

As can be seen on figure 7, the MPT algorithm is again slightly more efficient than the XTC algorithm for an attenuation factor equals 2.

IMPLEMENTATION ON A REAL SENSOR NETWORK

Now that it is possible to compare and therefore to choose a good algorithm for topology control, it is very interesting to try to implement such an algorithm into a real network stack, in order to confirm the theoretical results and to analyse if it is possible to implement the algorithm into a real WSN.

The hardware platform that is used as building block for the WSN is the Tmote™ Sky platform from Moteiv [5]. The Tmote Sky platform is a wireless sensor node based on a TI MSP430 microcontroller with an 802.15.4-compatible [6] radio chip CC2420 from ChipCon [7], with programmable transmission power. The Tmote Sky platform offers a number of integrated peripherals including a 12-bit ADC and DAC and a number of integrated sensors like a temperature sensor, 2 light sensors and a humidity sensor.

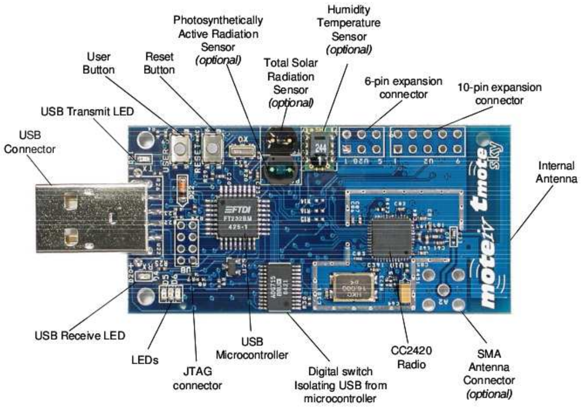


Figure 8. Tmote Sky platform

The real-time operating system used on the nodes is Contiki 2.0 [8]. Contiki is an open source multi-tasking operating system for networked systems. It is designed for embedded systems with small amounts of memory. Contiki permits to use of two different communication stacks: uIP and Rime. uIP is a light version of the TCP/IP stack. Rime is a communication stack existing of a lot of very light layers, easy to understand and adapt. Therefore Rime is used in our implementation.

In general a topology control algorithm is implemented between the second and third layer of the ISO model. It uses the link layer to connect with its direct neighbours and exchange information with them. This information will be used to build the topology of the network. The network layer will only see the neighbours permitted by the topology control.

For the implementation, the XTC algorithm was chosen. The principle of the implementation is as follows: each node periodically broadcast messages to its direct neighbours. In a first step these messages are used to build an up-to-date list of possible neighbours and their corresponding RSSIs. In a second step this list is send to all neighbours. With this information every node is able to calculate the XTC algorithm and to decide to which nodes it will connect. Only the permitted connections are put visible for the Rime network layer.

In order to verify the implementation, the program Netsim was used. Netsim permits to emulate the code as if it was written on the nodes. The results of this implementation are shown on the next figures.

In the Rime stack, every node receives a hopcount towards the sink, indicated with a circle on the figures. This hopcount is used to decide how to send the message to that sink. When a node has to relay a message, it will send it to a node with a lower hopcount and therefore closer to the sink. Figure 9 shows a network with the hopcount of the nodes, without using a topology control algorithm. Figure 10 shows the same network, but this time using the XTC algorithm for topology control. It can be seen that the hopcounts at the nodes are adapted because the topology control will forbid the long connections in the network.

With this implementation it is shown that it is possible to use an algorithm for topology control into wireless sensor networks.

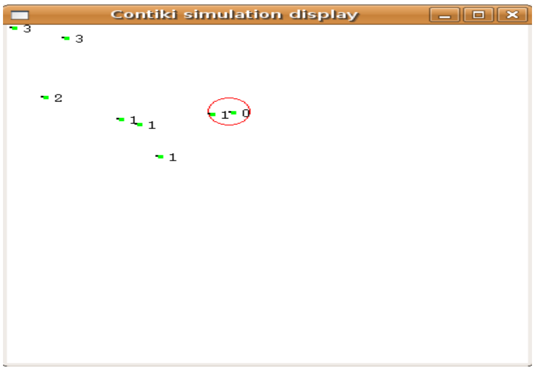


Figure 9. Netsim: simulation without topology control

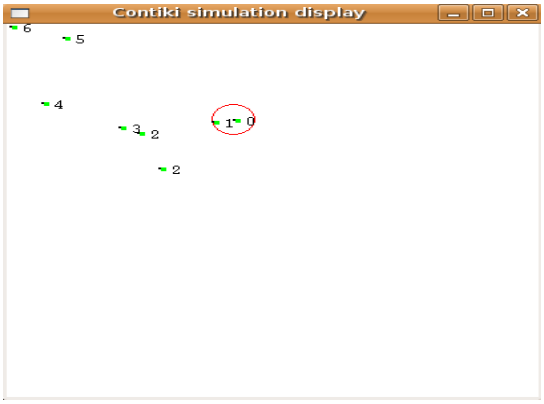


Figure 10. Netsim: simulation with topology control

CONCLUSIONS AND REMARQUES

In the literature many theoretical studies on topology control can be found, but few comment on a practical implementation of those algorithms. This paper first makes a study of the commonly used topology control algorithms for WSN. It was found that the Minimum Spanning Tree algorithm is not interesting due to lack of redundancy in the connectivity. The other algorithms have the same mean node degree and hence offer the same redundancy. The algorithms can also be subdivided depending on whether they are calculated in a centralized or a distributed way. From a practical view of point, it is preferable to have a distributed algorithm. Further the algorithms can use for its calculations the distance or the path loss between the nodes. The latter is preferred as it can be easily obtained using the RSSI parameter in the receiving node.

The two most promising algorithms are XTC and MPT. They are both distributed and based on path loss and have the same node degree. XTC and MPT are further compared by their ability to reduce the overall energy consumption while forwarding messages to a given sink. The performance of both algorithms is found to be similar. Simulations showed that a reduction in power consumption of 35 % is obtained for a WSN with 30 nodes on a 100m by 100m area, a transmission range of 50 m and an attenuation factor of 2 for the path loss. The gain in power consumption even increases with higher values of the attenuation factor.

In this paper an implementation of the XTC algorithm on a WSN is presented. It is shown that such an implementation is feasible on a real sensor network.

But many problems and questions stay unanswered. Indeed, the study did not look into the overhead created by the topology control algorithm, nor the additional power needed to processes the messages in the intermediate nodes. However it is shown that an implementation of a topology control algorithm into a real sensor network is possible and useful.

BIBLIOGRAPHY

[1] T.S. Rappaport: *Wireless Communications, principles and practice*, Second edition, Prentice Hall, 2002

[2] I. Stojmenović: *Handbook of Sensor Networks, Algorithms and architectures*, Chapter 10, Wiley series on parallel and distributed computing, 2005.

[3] Kristof Van Laerhoven:, *Voronoi Diagrams & Delaunay Triangulation*, http://www.comp.lancs.ac.uk/~kristof/research/notes/voronoi/

[4] R. Wattenhofer, A. Zollinger: *XTC, A Practical Topology Control Algorithm for Ad-Hoc Networks*, Technical Report 407, Department of Computer Science, ETH Zurich, 2003.

[5] *Tmote Sky datasheet*, Moteiv corporation, 2006, http://www.moteiv.com/products/

[6] IEEE 802.15 WPAN™ Task Group 4: *IEEE Std 802.15.4-2003*, IEEE, 2003, http://www.ieee802.org/15/pub/TG4.html

[7] *CC2420 datasheet*, Chipcon products from Texas Instruments, 2004, http://www.chipcon.com/

[8] *The Contiki Operating System- Home* , Adam Dunkels, http://www.sics.se/contiki/