CONSTRUCTING ENERGY EFFICIENT BLUETOOTH SCATTERNETS FOR WIRELESS SENSOR NETWORKS

A THESIS
SUBMITTED TO THE DEPARTMENT OF COMPUTER ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
Sain Saginbekov
August, 2004

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. İbrahim Körpeoğlu(Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Özgür Ulusoy

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Ali Aydın Selçuk

Approved for the Institute of Engineering and Science:

Prof. Dr. Mehmet B. Baray
Director of the Institute

ABSTRACT

CONSTRUCTING ENERGY EFFICIENT BLUETOOTH SCATTERNETS FOR WIRELESS SENSOR NETWORKS

Sain Saginbekov
M.S. in Computer Engineering
Supervisor: Assist. Prof. Dr. İbrahim Körpeoğlu
August, 2004

The improvements in the area of wireless communication and micro-sensor technology have made the deployment of thousands, even millions, of low cost and low power sensor nodes in a region of interest a reality. After deploying sensor nodes in a target region of interest, which can be inaccessible by people, people can collect useful data from the region remotely. The sensor nodes use wireless communication and can collaborate with each other. However, sensor nodes are battery powered and therefore they have limited energy and lifetime. This makes energy as the main resource problem in sensor networks. The design process for sensor networks has to consider energy constraints as the main factor to extend the lifetime of the network.

The wireless technology used for communication among sensor nodes can affect the lifetime of the network, since different technologies have different energy consumption parameters. Bluetooth, being low power and low cost, is a good candidate for being the underlying wireless connectivity technology for sensor networks tailored for various applications. But in order to build a large network of Bluetooth-enabled sensor nodes, we have to first form a Bluetooth scatternet. The topology of the Bluetooth scatternet affects the routing scheme to be used over that topology to collect and route informaton from sensor nodes to a base station. And routing scheme, in turn, affects how much energy is consumed during transport of information. Therefore, it is important to build a Bluetooth scatternet wisely to reduce and balance the energy consumption, hence extend the lifetime of a sensor network.

In this thesis work, we propose a new Bluetooth scatternet formation algorithm to be used in Bluetooth-based sensor networks. Our algorithm is based on first computing a shortest path tree from the base station to all sensor nodes and then solving the degree constraint problem so that the degree of each node in the network is not greater than seven (a Bluetooth constraint). We also propose a balancing algorithm over the degree constrained tree to balance the energy consumption of the nodes that are closer to the base station. The closer nodes are the nodes that will consume more energy in the network since all traffic has to be forwarded over these nodes. Our simulation results show that our proposed algorithm improves the lifetime of the network by trying to reduce the energy consumed during data transfer and also by balancing the load among the nodes.

Keywords: Wireless Sensor Networks, Bluetooth, Scatternet, Routing, Shortest Path Tree.

ÖZET

BLUETOOTH TABANLI SENSOR AGLARI İÇİN ENERJİ ETKİN SCATTERNETLERİN OLUŞTURULMASI

Sain Saginbekov
Bilgisayar Mühendisliği, Yüksek Lisans
Tez Yöneticisi: Yrd. Doç. Dr. İbrahim Körpeoğlu
Ağustos, 2004

Şu ana kadar kablosuz komunikasiyon ve micro-sensor alanında olan ilerlemeler sayesinde yüzlerce hatta binlerce ucuz ve az enerji harcayan sensorlardan oluşan ağların kurulması ve kullanılması mümkün hale gelmiştir. Bu tür ağları oluşturan sensorlar bir alana doşendikten sonra, o alan ile ilgili bilgiler bir merkezde insanların alana gitmesini gerektirmeden toplanabilmekte ve işlenebilmektedir. Fakat, bu şekilde kullanılan sensorların tek enerji kaynağı pillerdir ve bu sebeple sensorlar sınırlı enerji kaynağına sahiptir. Bundan dolayıdır ki enerji, sensor ağları için dikkatle kullanılması gereken en önemli kaynaklardan biridir.

Sensor ağlarında kullanılan iletişim teknolojisi genelde kablosuzdur. Bu amaç için kullanılabilecek bir çok kablosuz ağ teknolojisi günümüzde mevcuttur (mesela, Bluetooth, 802.11, ZigBee, gibi). Kullanılan kablosuz ağ teknolojisi, sensor ağlarının yaşam süreleri üzerinde etkili olmaktadır. Bunun başlıca nedeni değişik teknolojilerin değişik miktarlarda enerji harcamalarıdır. Günümüzde populer hale gelen Bluetooth teknolojisi, düşük enerji harcayan ve düşük maaliyete sahip olan bir teknoloji olarak sensor ağlarında kullanılmak için oldukça elverişli bir teknolojidir. Sensor ağlarının yaşam sürelerini etkileyen faktörlerden bir başkası olarak, toplanan verinin sensor düğümlerinden bir merkeze, yani baz istasyonuna, aktarılmasında kullanılacak yolları belirleyen yönlendirme metodlarının da önemi büyüktür.

Bluetooth teknolojisinin bir sensor ağında altyapı olarak kullanılması için önce scatternet adı verdiğimiz bir Bluetooth ağının oluşturulması gerekmektedir. Bir scatternet oluştururken ise bir çok değişik objektif gözönünde bulundurulabilir. Fakat, sensor ağları için en önemli objektif, oluşturulan scatternetin, verinin sensorlardan baz istasyonuna taşınması sırasında az enerji harcanması için uygun bir topolojiye sahip olmasıdır. Bu tez çalışmasındaki amacımız, sensor ağları için oluşturan Bluetooth scatternetlerinin mümkün olduğunca enerji verimli olarak oluşturulması için gerekli algoritmalar geliştirmektir. Bu amaçla geliştirdiğimiz algoritma, önce her bir düğümün baz istasyonuna olan bağlantısını mümkün olan en kısa yoldan yapıp, sonra eğer varsa yediden fazla komşusu olan dügümlerin komşu sayısını en fazla yedi olmak üzere indirgemeye dayalıdır. Bu şekilde, her bir dügümün oluşturduğu veri baz istasyonuna en az enerji ile taşınmış olacak, ve aynı zamanda oluşturulan topoloji Bluetooth teknolojisi kullanılarak gerçekleştirilebilecektir (bir Bluetooth düğümü en fazla 7 tane komşuya sahip olabilir). Yine baz istasyonuna bağlı düğümlerin yükünün dengeli olması için, önerdiğimiz algoritma ağaç şeklinde olan scatternetinin birinci seviyesindeki düğümlerde harcanan enerjiyi dengelemeye çalısmakta, ve bu şekilde en çabuk ölecek olan düğümün hayatını uzatmaya çalışmaktadır. Yaptığımız simulasyon sonuçları algoritmamızın Bluetooth tabanlı sensor ağlarının hayatlarının uzatılmasında etkili olduğunu gostermektedir.

Anahtar sözcükler: Kablosuz Sensor Ağları, Bluetooth, Scatternet, yol belirleme, kısa yol ağacı.

Acknowledgement

I am very grateful to my supervisor, Assist. Prof. Dr. İbrahim Körpeoğlu, from whom I have learned a lot, for his encouragement during my two years of graduate study and for his comments and suggestions in the supervision of the thesis.

I would like to express my thanks and gratitude to Prof. Dr. Özgür Ulusoy and Assist. Prof. Dr. Ali Aydın Selçuk for evaluating my thesis.

I would like to express my special thanks to my parents and brothers for their endless love and support throughout my life. Without them, life would not be that easy and beautiful …

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The fast improving technology enables manufacturing of new products everyday. These new products and technologies help our lives to be more comfortable and enable development of some interesting applications that were not possible earlier. For example, one such application, remote monitoring of large regions (in the order of several kilometer squares) is enabled by today's technology; and with the use of such an application people can sit at their offices and monitor regions for some interesting events without being on the field anymore. This technology of today, which is becoming more mature everyday, is called wireless sensor network technology. It is enabled mainly by the advancements in the area of micro sensor devices and low-cost/low power wireless communication technologies.

A wireless sensor network, generally speaking, consists of one or more base stations and many tiny sensor nodes that are deployed over a target region to monitor. A wireless sensor node basically consists of three parts: a sensor device, a processor, and a radio chip; and therefore is capable of sensing, computing, and communicating. Multiple sensor nodes can communicate and collaborate with each other. Depending on the sensor technology, each sensor node, or some special sensor nodes can also talk with a base station that is located at a fixed point and that is used to collect all the information produced by sensor nodes. The communication technology is usually an RF based wireless technology. Wireless communication provides flexibility and self-configuration, and enables mobility if required. Micro sensor nodes are very small in size, and they are battery powered. Therefore they have limited amount of energy to consume during operation on the field. The cost of each sensor node is not so low at the moment, but it is expected that further improvements in technology will enable inexpensive wireless sensor nodes to be produced in large quantities in near future [6]. Then the sensor nodes will be commodity items purchasable by everyone and by every organization. Such a pervasive availability will enable people to develop a large number of interesting applications in different areas, such as home automation, healthcare, business, military, civil, and transportation sectors. For example, a very interesting and useful application can be environmental monitoring which is monitoring a region for concentration levels of various chemicals, for air pollution, for humidity, and so on. In short, wireless sensor networks (WSN) will be the essential part of our daily lives in the near future.

Low-cost, low-power, wireless connectivity, and self-organization features of the sensor nodes will make it feasible to deploy hundreds or even thousands of them to a target region. This kind of deployment of large quantities of low-cost, low-power sensor nodes brings some benefits compared to deploying a few sophisticated sensor nodes: more robust, more fault-tolerant, more reliable, more flexible, and more accurate network and information gathering and processing will be possible [32, 6]. However, this kind of deployment faces also some challenges. These kind of sensor nodes that are randomly deployed and left unattended are powered by batteries, and recharging or changing the batteries may be impossible since the nodes may be deployed in inaccessible terrains. Therefore, the lifetimes of this kind of sensor nodes will be limited with the lifetime of their batteries. According to [25], battery capacity only doubles in every 35 years. Since the battery technology is not improving as fast as computing and communication technology, to extend the lifetime of sensor nodes and the whole network, various methods have to be used to conserve energy as much as possible.

In a sensor node, battery energy is drained for sensing, computing, and communicating. Significant amount of energy is consumed during communication [1, 29]. Therefore, wise methods for selecting routes, the paths through which data has to be transported from a source to a sink, have to be used.

The choice of wireless communication technology has an effect on the network lifetime since different wireless technologies use different transmit powers. Different wireless communication technologies can be considered for sensor network applications. However, the ones that consume much energy and that cost much are not suitable for sensor networks. Therefore, when choosing one of the existing wireless technologies, people should look at their cost and the energy consumption values.

Another factor that greatly impacts the energy consumption is the routing protocol that a network uses. In sensor networks the routing protocol should be energy-efficient unlike in traditional ad-hoc networks. In ad-hoc networks, most of the attention was paid on the mobility, delay, etc.

Bluetooth is one of the wireless technologies available today. Bluetooth can be used as an underlying wireless communication technology for sensor network applications. Its low-cost, low-power, small size are the main features that make it a good alternative wireless technology for sensor networks.

In this thesis work, we propose a Bluetooth scatternet formation algorithm for sensor network applications that prolongs the lifetime of a network. It is scalable and self-healing algorithm. In our algorithm we used an existing shortest path tree algorithm where a root of a tree is a base station. Since shortest path tree algorithm does not consider the degree of a node, a node in a tree can have more than seven degrees, which causes the formed tree not feasible for Bluetooth scatternet. A Bluetooth node can have at most seven neighbors. So, our algorithm after solving the degree constraint of a node in a shortest path tree, balances the energy consumption of the nodes, which are one hop away from the base station as well.

Our algorithm is run at the base station. Thus, before the formation of scatternet, base station has to know the information about the nodes' coordinates, neighbors, or distances between any two nodes. Therefore, our algorithm lengthens the scatternet formation time that is equal to time for collecting information about the nodes and running time of our algorithm.

Since in sensor a network, time is not the main design consideration but energy conservation is, it is worth to make a trade off between time and energy.

Our simulation results show that the resulting Bluetooth scatternet consumes for about 170 to 350 units more than the lower bound of energy consumption per round, which happens when the scatternet is formed in 6-ary tree manner, for different network sizes. Moreover, our balancing algorithm reduces the energy consumption of a node, which consumes the highest amount of energy, for about 30 % to 50 % and prolongs the lifetime for about 40 % to 100 % depending on the network size.

The rest of the thesis is organized as follows. Chapter 2 gives information about sensor networks and Bluetooth technology in addition to related works done so far. In chapter 3, we define the problem statement and the network model. In chapter 4, the proposed solution approach is described in details. Chapter 5 shows results obtained from our simulations. And finally, in chapter 6, we conclude the paper and define future works that can be done on this area.

Chapter 2

Background and Related Work

2.1 Sensor Networks

The technological advances in wireless communication and hardware have enabled the deployment of large number of sensor devices in diverse areas to monitor and control the events of interest. These devices are called sensors. Each sensor node has the ability of sensing, processing, and communicating. Sensor nodes have small amount energy. They are mostly powered by small batteries. Limited amount of energy of sensor nodes put constraints on their processing ability. Sensor nodes cannot process much as the nodes that have unlimited source of energy. However, the combination of large number of sensors into one network enabled cooperation and distributed processing which make a sensor network a powerful system compared to a single powerful node. The aim of a sensor network may depend on the application. An important class of applications includes collection of environmental data from a target field into one point, which is usually called a base station, where the data is analyzed and interpreted.

Sensor networks can be used in different applications on the ground, in the air, and under water [3]. Sometimes people need to monitor and collect data from the area that is not possible to be accessed by people, and from the area that does not have an installed infrastructure. These types of applications can benefit from the wireless sensor networks. To establish such a network, wireless sensor nodes can be dropped from a plane, or via some other methods, to the target region. Then the sensor nodes form a network without any manual installations. This is called self-organizing capability. This is different than establishing a wired sensor network, which is usually done manually. Wired sensors have to be connected with wires and should be reconnected by people when some nodes fail for some reason. However, the advantage of wired sensor networks is that the nodes does not have to be operated using batteries in cases where there is access to power-line network. Hence, wired sensor networks may have longer network lifetimes. Therefore, trying to prolong the lifetime of wireless sensor networks is an important objective, which may be irrelevant for wired sensor networks. In fact, energy conservation is one of the main issues that has to be addressed in wireless sensor networks [5]. Another challenge in designing protocols for wireless sensor networks is due to re-configuration requirement. When some changes happen in the network, like node failures, link break-ups, etc., the network has to re-configure itself to adapt to the changes. Moreover, since wireless sensor nodes can be deployed in an area which cannot be accessible by people, such as toxic fields, the nodes have to form a network in ad-hoc manner.

Figure 2.1 shows a sample sensor network consisting of several sensor nodes and one base station. Since not all sensor nodes can be in the communication range of the base station, a node has to forward its sensed data through one of its neighbors towards the base station. This requires each sensor node to act also as a router to relay the data of other nodes.

Various wireless communication technologies can be considered as a communication technology in sensor networks. However, not all fulfill the requirements of the sensor networks and sensor nodes. Sensor nodes should have mainly the following properties: low cost, low power, and small size. Not all wireless technologies are suitable to be incorporated into low power, small sized and low cost sensor nodes. But, Bluetooth is one of the candidates among all the available wireless technologies that can meet the requirements of wireless sensor networks. As stated in [8], today's available hardware platforms for sensor networks (i.e. sensor nodes) can be divided into four classes: special purpose sensor nodes,

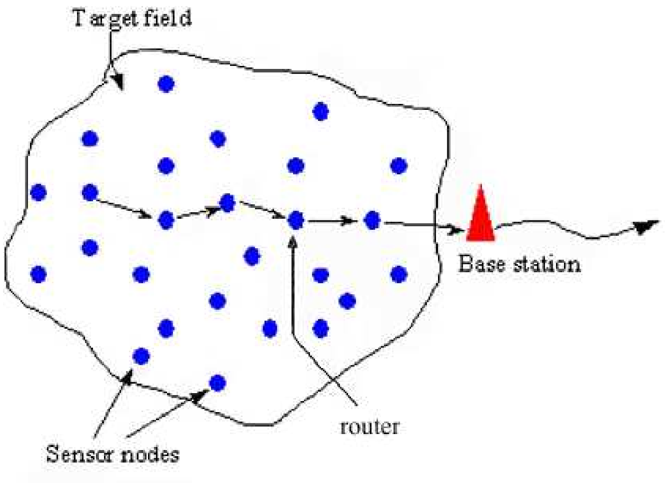

Figure 2.1: A sample sensor network

generic sensor nodes, high-bandwidth sensor nodes, and gateway nodes. The power requirements of those nodes increase respectively. Bluetooth technology can be used as part of sensor nodes that fall into the third class. Current sensor nodes that use Bluetooth as an underlying communication technology are *BT node* (developed in 2001) and *Imote 1.0* (developed by Intel Research in 2003). The less energy consuming ZigBee standard is to be used in sensor nodes that are included in the second class. ZigBee provides a data rate in the order of 250 Kbps, whereas Bluetooth provides a raw data rate of 1 Mbps. Therefore, Bluetooth is better for sensor network applications that are bandwidth demanding. Bluetooth is also low power, addressing one of the most important issues in sensor networks. However, ZigBee and its co-operative technology IEEE 802.15.4 aim to operate communication devices at much less power than the Bluetooth technology.

The next section gives more detailed information about Bluetooth.

2.2 Bluetooth Technology

Devices such as laptops, mobile phones, PDAs are becoming essential part of our daily lives. To connect these devices together, the common way is use of cables. But it is very tedious for people to connect the cables every time they need communication among these devices, and to disconnect the cables again after finishing with the communication need. The solution developed in recent years for this tedious cable work is to use a short-range, low cost, and low power wireless communication technology to replace cables. So was the initial aim of Bluetooth: cable replacement.

But the usage scenarios of Bluetooth are not limited with cable replacement. Bluetooth technology also supports devices to form ad hoc networks on-the-fly. This can be done by forming piconets and scatternets.

Another promising application area of Bluetooth technology is sensor network. Sensor networks require the sensor nodes to be very low power and low cost. This requires the nodes to have very low power and low cost radio chips. The cost of Bluetooth radio chips is decreasing everyday, and therefore Bluetooth is becoming a good candidate to be used as the communication technology between the nodes of a wireless sensor networks. Bluetooth's power requirements make it suitable for various sensor network applications, although not for all.

Bluetooth is operated at universally available unlicensed ISM (Industrial, Scientific, and Medical) Radio Frequency (RF) band (2400-2483.5 MHz). The band is divided into 79 channels. Each channel width is 1 MHz. Bluetooth uses Frequency Hopping Spread Spectrum(FHSS) scheme. The radio hops through 79 channels using a pseudorandom hopping sequence. Each channel is divided into time slots each longing 625 μs. The hoping rate is 1600 hops per second. Thus, one slot can hop 1600 frequency channels in a second. The communication between devices is based on Time Division Duplex (TDD) scheme. Each device can send a packet by alternating slots. One packet can be as long as one, three, or five slots. Different throughput can be achieved as a function of packet size (in slots) in the direction of master to slave and slave to master, with Forward Error Correction (FEC) and without FEC (see Table 2.1 [2]). Bluetooth supports synchronous and asynchronous links. The synchronous connection-oriented (SCO) link is used primarily for voice and they are transmitted through reserved intervals. A piconet can support up to three SCO links. SCO packets are not retransmitted. The Asynchronous connectionless (ACL) link is used primarily

Table 2.1: Achievable channel throughput for different packet sizes

Packet size (in slots)		Throughput in Kbps (with FEC)		Throughput in Kbps (no FEC)	
In slave direction	In master direction	In slave direction	In master direction	In slave direction	In master direction
1	1	108.8	108.8	172.8	172.8
3	1	378.2	54.4	585.6	86.4
5	1	477.8	36.3	723.2	57.6

for data. ACL link can use the remaining slots on the channel. Unlike SCO, to ensure data integrity ACL packets are retransmitted.

The are three different Bluetooth device classes. Each class has a different transmit power (hence a different transmission range):

- *Class 1 device*: The communication range is 100 meters and transmit power is 100 mW (20 dBm)
- *Class 2 device*: The communication range is 50 meters and transmit power is 2.5 mW (4 dBm)
- *Class 3 device*: The communication range is 10 meters and transmit power is 1 mW (0 dBm)

2.2.1 Piconets and Scatternets

The smallest network that can be formed with Bluetooth-enabled devices, so called *piconet*, can contain up to 8 nodes, one master and up to seven active slaves, which share a common radio channel. A Bluetooth layer-2 connection has to be established between a slave and a master node before any data is exchanged in between. The starter of a connection will take the role of master. Master defines which frequency-hopping sequence the members of the piconet will

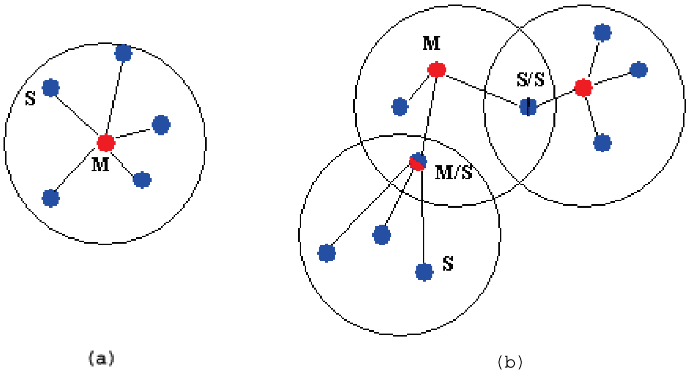

Figure 2.2: a) Piconet containing one master and five slaves; b) Scatternet containing three piconets with one M/S and one S/S bridge.

follow. If more than seven devices want to communicate with master, master tells some current active devices to switch to low power park mode (will be explained below) and invites others to be member of its piconet. Each slave of a piconet can transmit data only through the master of the piconet. The nodes cannot communicate if they are not in the same piconet and unless there is a scatternet formed incorporating the nodes . Communication between nodes in different piconets is possible when a scatternet encompassing those piconets is constructed. A *scatternet* can be formed from two or more piconets by using some of the nodes as bridges between piconets. A bridge node can connect two or more piconets; but usually it connects two piconets together. A bridge node connecting two piconets can be a master in one piconet and a slave in the other piconet. Such a bridge node is called a master-slave (M/S) type of bridge. Similarly, a bridge node can be a slave in both of the piconets it connects together. Such a bridge node is called a slave-slave (S/S) bridge. A bridge node then takes part in each piconet it is connected to in a time-sharing basis. (see Figure 2.2). This means, after participating in one piconet for some time, the bridge node switches to another piconet's frequency hopping sequence and participate in that other piconet. The more piconets the bridge node connects together, the more time it takes for the bridge node to start participating in the same again.

2.2.2 Low Power Modes

Bluetooth supports different power saving modes. This is a very important property that can be utilized if it is used as the underlying communication technology for wireless sensor networks.

As mentioned above, a slave node in a piconet can be an active slave or a parked slave. Park mode is a lower power mode. In addition to these two modes, active and park modes, Bluetooth has two other low-power modes: sniff mode and hold mode. These modes are incorporated into Bluetooth technology considering the observation that the devices using Bluetooth do not always need to be active after forming a piconet or scatternet. A Bluetooth node can, for example, go into sleep after transmitting and receiving data until the next period of transmission and reception.

Sniff Mode is one way of conserving energy in a node. In this mode, the master and a slave agree for certain regular times. The master sends packets to the slave at those agreed times. Slave listens for the packets again at those agreed times. If the slave does not receive packets, then it continues being in sniff mode, otherwise, it receives all other following packets from the master and then goes into sniff mode. A slave node spends less energy in sniff mode compared to being in active mode. In this way, a slave node saves energy by reducing its radio duty cycle. The amount of saving depends on the time interval between agreed times.

Hold Mode can be used when a slave wants to do other things. For example, before switching to another piconet, a slave can go into hold mode in the current piconet. This mode also depends on the hold time interval agreed by master and a slave node. The difference of this and sniff mode is that, in hold mode a slave does not need to listen packets from a master until the end of the hold time. Therefore, a slave can conserve more power in hold mode than in sniff mode. The amount of energy saving depends on the hold time.

Park Mode is not considered as a mode for an active slave. In park mode, we can consider the slave as inactive. This mode is similar to sniff mode in that the node in park mode do not communicate and do not participate in the piconet When a slave node is in park mode, its temporary MAC address (a 3 bit value) can be used by some other node in active mode. In this way, the number of slaves associated with a master (i.e. slaves in a piconet) can be more than seven. However, the number of active slaves in a piconet can never exceed seven. An active slave can be in one of the three modes: active, sniff, or hold mode. A slave node can be in park mode and active mode alternatively.

In sensor networks nodes usually do not transmit and receive data continuously, but at regular times or when an event occurs. This implies that a node's radio transceiver does not have to be turned on always. If Bluetooth is used as the communication technology between sensor nodes, the Bluetooth radio transceiver can be put into sniff or hold modes when there is no data to be sent between sensor nodes. This way a node can save energy at the radio chip. For example, a slave sensor node can switch to hold mode after transmitting its data to the corresponding master node. When in hold mode, the slave node can then participate in another piconet and send data to the master of that other piconet. Participation in another piconet may not be always necessary. A slave can go into hold mode just to sleep and save energy during inactivity.

2.2.3 Scatternet Formation Algorithms

If a Bluetooth network consisting of more than eight nodes is needed, which is certainly the case in sensor networks using Bluetooth, a Bluetooth scatternet has to be formed. What a scatternet is and its constituting components are very well defined in Bluetooth standards, but how to form a scatternet and the topology of the resulting scatternet is not specified in the standards and therefore this is a research problem. A solution of the scatternet formation problem includes the shape (topology) of the resulting scatternet and a step-by-step algorithm specifying how to construct that scatternet. The topology of the scatternet determines the connectivity between the nodes (which node connects to which other nodes). Information about a scatternet has to include also the type of roles the nodes of the scatternet assume. A node can be a master, a slave or a bridge node. Again a bridge node can be an M/S bridge or an S/S bridge, or it can be a bridge connecting more than two piconets.

Various Bluetooth scatternet formation algorithms with different goals have been developed so far. Each algorithm may have different objectives. Some studies aim to have an easy routing in the constructed scatternet, whereas some studies consider bandwidth efficiency, running time, etc., as the main metric of performance. In this section, we describe some of the existing scatternet algorithms.

In [35], with the proposed algorithm, a tree-shaped scatternet, so called Bluetree, is formed. In this protocol an arbitrary node, blueroot, is selected. That node connects all its neighbors as slaves. Blueroot will be assigned the role of master. And then, each child of blueroot will act as a master to connect its neighbors as slaves. And this procedure will be repeated until the leaf nodes are reached. Since, some of the nodes can have more than seven slaves, the tree is reconfigured by another protocol. By the observation of authors, if a node has more than five neighbors then at least two of them are neighbors to each other. Using this observation, they claim that all nodes can have no more than five slaves. Results show that the number of roles that each node can assume is limited to two. This reduces piconet switching overhead.

In [26], authors proposed a protocol to extend the lifetime of a scatternet. Two different energy conservation techniques are used. The idea of the first one is to change the master/slave role of a piconet. Since all data transmissions of a piconet are done through a master, master node of a piconet must consume more energy than its slaves, and thus, its energy is drained soon. Therefore, to prolong the lifetime of a master, master must give its role to another slave. Master node decides to give its role to a slave, which has maximum amount of energy, when its energy becomes less than a specified value. Then the new master informs other slaves about its role. The second technique uses the prior knowledge of distance between the master and slaves to chose the transmit power and conserve energy.

In [14, 13], main consideration was on scatternet construction time and the number of messages transmitted during this time. In a resulting scatternet, any node is a member of at most two piconets and the number of piconets close to be minimal to avoid network bottleneck and minimize inter-piconet interference, respectively.

There are also other different scatternet construction algorithms with different goals like robustness, connection delay, etc. [24, 28].

2.2.4 Advantages and Drawbacks of Bluetooth When Used for WSN

Bluetooth uses license-free ISM RF band at 2.4 GHz. Globally available license-free ISM band is useful for deploying sensor nodes with Bluetooth everywhere in the world without need for line-of-sight communication between sensor nodes. This band, however, is also used by other technologies such as microovens, 802.11 devices, etc. This can pose an interference problem to Bluetooth enabled sensor nodes during communication. Bluetooth combats with this interference problem using FHSS scheme. Under this fast frequency hopping scheme, a different 1MHz wide frequency band is used in a piconet at every 625 microseconds. In this way the chance of using the same 1 MHz wide frequency band with other nearby devices, hence interference, is reduced dramatically.

Support for low power modes in Bluetooth enables sensor nodes to save energy when not communicating. This is a very important feature of Bluetooth that is addressing one of the main issues in sensor networks: energy conservation.

The initial price target for Bluetooth radio chips was in the order of 5 dollars. This is not an acceptable price target for sensor networks that will consist of thousands of nodes. However, the cost of Bluetooth radio chips is falling as the technology becomes more mature and advanced. Additionally, not all sensor networks need to consist of thousands of nodes. There is also need for sensor networks that require high-rate data communication and consist of hundreds of nodes. The cost of Bluetooth chips for such networks will fall into an acceptable range.

Table 2.2: Advantages and Drawbacks of Bluetooth Technology

Advantages	Drawbacks
Low power, low cost, and small size	Piconet can have at most eight nodes
Can be operated everywhere	Range is 10 meters
Has three low power modes	Lengthens delay
Secure	
1Mbit/sec data rate	

Bluetooth is initially considered for portable devices around human beings. These devices include mobile phones, PDAs, watches, headsets, etc. All these devices have a small form factor. Therefore, it was also requirement for Bluetooth to have a small form factor. This objective is also in line with the objective of having the sensor nodes also small devices. Sensor nodes will be deployed in large numbers with low cost. Therefore, they need to be small in size to not clutter the environment much and to reduce the material costs.

Another issue in sensor networks is security. It is important for some sensor network applications, such as the military ones, to transport data securely from sensor nodes to a central location [30]. Blueooth has also features addressing this issue. Bluetooth devices can authenticate each other before communicating any data. Bluetooth supports encryption. Additionally, FHSS scheme has benefits for security although its main goal is to reduce interference between piconets and between different technologies. It is hard for a stranger to listen an FHSS radio channel, because of fast frequency hopping to different channels following a pseudo-random frequency pattern which can only be guessed if the stranger knows some of the parameters of the piconet (like master's BT address) [19]

Before using Bluetooth technology in a sensor network, however, several issues have to be addressed. Since sensor nodes may be densely deployed in a target field, and each master node in a Bluetooth network (scatternet) can connect to at most seven slave nodes, the Bluetooth based sensor network has to be formed considering this constraint of bounded degree. Moreover, Bluetooh-enabled devices have to form a piconet before exchanging data among them. Formation of a piconet requires establishment of layer-2 connections between the master and the slaves, and therefore causes extra delay in the network formation phase.

Table 2.2 summarizes the advantages and drawbacks of using Bluetooth and that we have discussed so far.

2.3 Routing in Sensor Networks

As we discussed earlier, energy efficiency and conservation is one of the most important issues that has to be considered in wireless sensor networks. This is not different for sensor networks based on Bluetooth.

Energy efficiency in sensor networks can be achieved in different ways. One way is using energy efficient routing schemes in transporting data from sensor nodes to the base station. Routing scheme affects the energy efficiency, since it determines the paths that packets will follow from sensor nodes to the base station, and the parameters of a path (the number of edges on the path and edge costs) determines how much energy is spent in transporting data over that path. The best routing scheme to be used depends on the objectives in terms of energy. Different objectives may dictate the use of different routing schemes.

There is an abundant amount of work in the literature on routing. However, those studies concentrate on either routing in static networks like Internet, routing in infrastructure based wireless networks like cellular networks, or routing in mobile/wireless ad hoc networks. The objectives and routing metrics used in those routing schemes are very different than the objectives of routing schemes for sensor networks. In traditional wireless ad-hoc networks, routing schemes are designed mostly to achieve good quality of service (QoS) parameters like low delay, high throughput, transparent mobility, etc. In sensor networks, these factors are not the main considerations. The main issue to consider in sensor networks is energy conservation. Moreover, the data flow in ad hoc networks can be from any node to any other node, while in sensor networks the data flow is usually either from sensor nodes towards base station or from base station towards

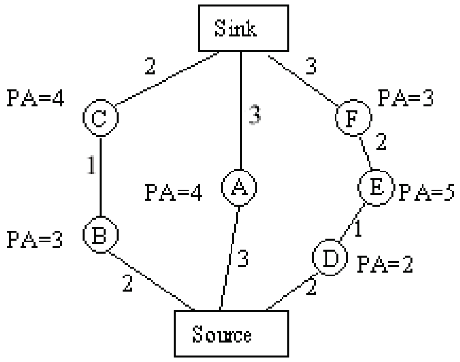

Figure 2.3: Different routing strategies

sensor nodes. Sensor nodes usually do not communicate with each other in end-to-end manner. The routing schemes designed for ad-hoc networks, therefore, are not very suitable to be used in sensor networks. We need routing schemes that are based on energy for sensor networks. This is also the case for Bluetooth based sensor networks.

2.3.1 Routing Strategies

Various energy-efficient routing strategies can be implemented for wireless sensor networks considering energy consumption as the main resource problem. We list some of them below. Figure 2.3 shows a sample sensor network using which the schemes can be described. On the figure, the available remaining energy in a node is denoted with PA (power available) and the weight on a link is the cost of transmitting a data packet over that link.

- *Maximum Available Power Route (MAPR):* The total available powers, sum of each node's PA in the route, of each possible route are calculated and the one, which has the maximum power is selected. In sample network it is (Source-D-E-F-Sink). This approach is selected to prolong the lifetime of a network. However, since this route does not consider the link cost, it can choose longer path.

- *Minimim Energy Route (MER):* Among the possible routes between the source and the sink, the one which consumes the minimum energy is selected. In figure, it is (Source-B-C-Sink). This way of routing consumes minimum energy. However, if the same route is used many times, the nodes on that route may die faster.
- *Minimum Hop Route (MHR):* Among the possible routes between the source and the sink, the one which have minimum number of nodes is selected. In figure, it is (Source-A-Sink). This will be the same as MER if the cost of each link will be equal. MHR can be used when the link costs do not differ so much.
- *Maximum Minimum Available Power (MMAP):* Among the possible routes between the source and the sink, the one whose minimum available power is the largest than the minimum available powers of other routes. In a network, it is again (Sink-A-Sink). This method extends the lifetime of a node, which has the minimum available power.

2.4 Related Work

Upto now, we have background information about wireless sensor networks and Bluetooth and routing in general. But we did not discuss some relevant work that is very close to the problem area we are working on: Bluetooth based sensor networks and routing.

As mentioned in previous sections, in the design of routing protocols for mobile ad hoc networks, the main factor considered is not energy conservation but other factors like packet delay, control traffic overhead, etc. In [33], different algorithms proposed for ad hoc networks [22, 20, 9, 10, 34, 21, 23, 11] are classified according to their relevancy and efficiency when applied to personal area networks and sensor networks.

Besides schemes that are adaptations of earlier schemes developed originally for ad-hoc networks, there are also routing schemes that are developed for sensor networks specifically. The goal of these schemes has been energy efficiency and to prolong the lifetime of a sensor network. We will now briefly describe those schemes. After that studies about Bluetooth based sensor networks will be discussed.

In [7], the authors proposed a communication protocol for sensor networks, called LEACH, that improves the lifetime of a sensor network about eight times than conventional protocols. The idea is to reduce the number of messages sent from sensor nodes directly to a base station using a clustering approach. The sensor nodes in a sensor network are formed into clusters and in each cluster a cluster-head is selected to collect data from other cluster members and sent it to the base station. In order to distribute the load of transmitting packets to the base station, cluster-heads are changed periodically.

In [16], authors proposed an other protocol, called PEGASIS, that performs better than LEACH for about 100 to 300 %. They came to this result by reducing the number of nodes that directly communicate with the base station to one. The algorithm first forms a chain containing all nodes. One node in the chain is selected as the special node. The data is collected from other nodes towards this special node following the chain. So, the chain determines the path of the data packets. The special node then transmits the collected data to the base station. As LEACH, this scheme also assumes that the nodes are performing data aggregation. Data aggragation is the act of condensing several received data packets into one packet to be transmitted to the next node. Data aggregation may not be possible for all sensor network applications.

Authors of [31], propose two new algorithms, PEDAP and PEDAP-PA, that are near optimal minimum spanning based (MST) routing protocol. PEDAP performs better than LEACH for about 4x to 20x, and for about 3x than PEGASIS. Authors claim that to be able to prolong the lifetime of a network, the minimum energy must be consumed per round of communication with balancing the energy consumption among the nodes. In PEDAP, the last node achieves good lifetime since the energy consumed per round is the minimum. PEDAP-PA is the power aware version of PEDAP algorithm that balances the energy consumption among nodes by computing MST after each regular round. This protocol prolongs the first node lifetime, but the last node dies faster than the last node of PEDAP algorithm.

All these three algorithms mentioned above use the same first order radio model described in [7]. According to this model, the energy consumption while sending a packet from a transmitter to a receiver depends on the distance between the transmitter and the receiver in addition to the constant energy consumed at the electrical circuitries in the transmitter and receiver.

However, these algorithms are not designed for a specific wireless communication technology. Therefore, when technology requirements and constraints are considered, they may not be applicable for all technologies. For example, Bluetooth has a node degree constraint, and also the communication range is at most 10 meters for class 3 devices. Additionally, not all Bluetooth devices can apply power control. Therefore, either extra work has to be done in adapting these schemes for Bluetooth, or new algorithms have to be designed.

The algorithms developed for scatternet formation and mentioned briefly above are not very well suited for sensor networks as well. This is because sensor networks have unique features which require different optimized solutions. Objectives such as high bandwidth, fast running time, etc., are not as important as energy conservation in sensor networks as mentioned previously. The most related work among scatternet formation studies can be found in [26]. Their proposed algorithm aims at extending the lifetime of scatternet nodes. In that algorithm, transmission of messages are not always towards a single point but from any node to any other node; and in sensor networks the nodes far from base station have to forward their messages through nearer nodes. This is possible only when nearer nodes have master role. Thus, masters can not change their roles and the idea of master/slave role exchanging a piconet for energy conservation does not really work in sensor networks.

To the best of our knowledge, Bluetooth-based sensor network protocols are considered in three studies so far. In [17] (DCP), the network is formed with a clustering approach. The protocol is divided in two phases: set-up phase and steady-state phase. In set-up phase, each node learns its neighbors and at least one packet forward address (PFA). A node in a network can take one of the two roles: *cluster-member* or *cluster-head*. Cluster-heads are selected randomly with a given probability. In steady-state phase, PFA is used to forward the data to a base station. Cluster members in a cluster periodically forward sensed data to their cluster head, and the cluster head, after fusing or compressing data, forwards the data to the base station. If the cluster head is not in the communication range of the base station, it forwards the data through an other cluster head. In DCP, a node in the formed network is not necessarily a master or a slave. They allow more than seven nodes to connect to a single node. However, the authors do not describe how a node can get associated with more than seven nodes. This has to be explained well, since a Bluetooth node (a master) can have connection to at most seven other nodes (slaves) [19]. Moreover, simulation results show that for a given probability the number of unconnected nodes is high for the communication range of 10 meters.

In [15], A Bluetooth-based sensor network is formed using the Bluetree protocol mentioned in [35]. Since the main consideration in Bluetree algorithm is not energy consumption, it is not good choice for sensor networks.

In [18], the proposed scatternet forming algorithm for sensor networks is divided into two phases: knowledge discovery phase and connection setup phase. In knowledge discovery phase, some characteristics about the sensor nodes are gathered by the base station. In the connection setup phase, base station starts selecting one-hop apart nodes as slaves, those slaves select their neighbors as slaves, and this process is repeated until the leaves are reached. Since there can be only up to seven slaves in a piconet, they propose a new technique to select nodes as slaves according to some defined factors. They have used Simulated Annealing [12] for this purpose.

Chapter 3

Network Model and Problem Statement

In this chapter, we state the problem we solve. But‘ before that we will define the wireless sensor network environment our solution is developed for. We make some assumptions about the environment

3.1 Network Model

In our work, we assume the followings:

- Class 3 Bluetooth devices are used.
- Each node has at least one reachable neighbor so that it can get connected to the network. In this way we have a connected network.
- The power consumed to send a packet from one node to its neighbor is constant for all nodes. In other words, we assume that the devices do not have the capability of power control. When power control is possible, the power consumed to transmit a packet will be related to the distance to the

receiver. Without power control, however, power required is constant and independent from transmitter-receiver separation.

- Sensor nodes and base station are stationary.
- No data aggregation is used. And if a node receives k packets, it does not merge them into one packet before transmitting to the next node. It transmits each packet separately.
- The base station knows the distance or neighborhood matrix.[1] In other words, we assume that the base station knows the exact point location of each node so that it can compute which node can reach to which other nodes.
- Not all nodes must be in the communication range of each other and with the base station.
- All nodes are homogenous and use the same wireless communication technology, which is Bluetooth in this thesis.

A class 3 Bluetooth device is the one that consumes less energy compared to class 1 and class 2 devices. It has 1 mW (0 dBm) transmit power. But it has also a very limited range of communication which is at most 10 meters. So, in our network two devices can communicate with each other if and only if the distance between the devices is not greater than 10 m.

The reason of our second assumption is that we want to have all sensor nodes be able to communicate their data to the base station. Since we assume a multi-hop routing environment, which means a node may not be always reaching the base station directly, the network has to be connected in order all nodes to be able to send their data to the base station. In other words, we assume there will be no network partitioning initially if the network is established properly.

We assume that devices do not apply any power control. This implies that the energy consumed to transmit a packet between two nodes is constant and

---

[1] In [18], the way how the neighborhood matrix can be obtained by a base station is described.

independent of the distance. However, in the future this assumption can be relaxed as Bluetooth devices can be advanced enough to apply power control.

In some sensor networks, it is possible to have mobile nodes, and to have more than one base station. In our work, we assume that the network has a single base station and all nodes are stationary. This especially valid assumption for networks consisting of nodes which are primitive and small, since mobility requires more complex sensor node platforms.

3.2 Problem Statement

We can define our problem as, given a set of Bluetooth-enabled sensor nodes and a base station with distance information, constructing a Bluetooth scatternet spanning all the sensor nodes and the base station so that the scatternet will be the underlying network for sensor network applications. The scatternet has to be formed in such a way that the energy consumed per round of communication from sensor nodes to the base station will be kept as small as possible and the energy consumption of nodes will balanced as much as possible.

Since a sensor network contains many nodes that may not be in the range of each other, multihop communication is used to extend reachability and also to conserve energy. It is possible to conserve energy with multihop communication as opposed to a single hop communication due to the distance-power relationship observable in wireless communication. The power required at a transmitter is inversely proportional with the square of the distance between the transmitter and receiver.

Moreover, multihop forwarding brings also some problems. The nodes that cannot reach to the base station directly will forward their packets to nodes that are in the range and closer to the base station. This implies that the sensor nodes that are one-hop away from the base station (i.e. directly reachable from the base station) will take part in forwarding the data of all nodes. This will cause these nodes to drain their energy much quickly than other nodes that are away from the base station. The more data they will forward, the faster they will die. The situation will be worse if there is unbalance in the amount of traffic forwarded by these nodes that are one-hop away from the base station. The amount of data that a node forwards can be reduced by use of techniques like data aggregation or data fusion. This may be possible for some applications since the sensor nodes do also have processing capability required for data fusion or aggregation.

However, there exist applications which require every sensed data to be sent to the base station. For this type of applications, sensor nodes cannot aggregate data. So, in these kinds of applications, data aggregation and fusion techniques developed for the purpose of reducing the amount of data forwarded by a node are not helpful to conserve energy (our fifth assumption). Therefore, some other kinds of methods must be used.

If sensor nodes are equipped specially, we may not need always to construct a connected scatternet which spans all the nodes. Another possibility is application of a clustering approach. In such an approach, each cluster may contain one cluster head and seven cluster members connected to the cluster head directly. Assuming Bluetooth is used for these connections, the cluster head can assume the role of master and the cluster members can be slaves. Each master collects data from its slaves and transmits it directly to base station. Hence, each sensor node can reach to the base station in two hops. This is different than forming and using a scatternet for transporting data to the base station. However, the cluster head may not be always in the range of base station, and therefore, may require use of another long-range wireless technology, such as GPRS or 3G, to reach to the base station directly. This is why we said initially that this approach is feasible only if sensor nodes are equipped specially. In our work, we will not focus on this type of approach. We assume that all nodes have only a single communication technology, which is Bluetooth, and therefore they are homogenous. Constructing a scatternet in this case is a must for having each node be able to send data to the base station (our eighth assumption).

Chapter 4

Solution Approach

In this chapter, we describe our proposed algorithm for the network model described in the previous chapter. To be able to describe our algorithm clearly, we have used some new terminology that will be defined in the next section.

4.1 Definitions

Before describing our solution, let us define a few terms. The one *round of communication* is the activity in which each node senses a data and all nodes forward their sensed data to base station. *Degree* of a node is the number of neighbors of that node. *Parent* of a node X is the node that is connected to node X and that has one less hops to the base station. *Possible parent* of node X is the node which is in the communication range of X and has one or more less hops to the base station. *Possible brother* of a node X is the node which is the child of the parent of X. *Possible sibling* of a node X is the node which is at the same level with X. *Level* of a node is the number of hops between the node and the base station. *Grandparent* of a node X is the first level node which is on the path between node X and the base station. Note that our definition for grandparent is different than the common definition which states that the grandparent of a node is the parent of the parent of that node.

4.2 Scatternet Construction Algorithm

As it is claimed in [31], to prolong the lifetime of a network the power consumption per round has to be close to minimum and the energy consumption must be balanced among the nodes. Using this idea, our scatternet construction algorithm can be divided into two parts. In the first part, our algorithm constructs a shortest path tree rooted at the base station and spanning all the nodes. After constructing the shortest path tree, it makes arrangements in the connections between nodes so that the degree of a node is not greater than seven (a Bluetooth master can have at most seven slaves) In second part, our algorithm tries to balance the energy consumption of the first level nodes in the tree so that the lifetime of the earliest dying first-level node is prolonged. Both parts of the algorithm are run at the base station.

Our goal in the first part of the algorithm is to form such a scatternet so that the power consumed in a round of communication is reduced. While traveling from a sensor node to the base station, the less the number of hops that is passed through by a packet, the less will be the amount of energy consumed. Therefore, minimum energy will be consumed while transporting a packet from a sensor node to a base station when the packet is routed through the shortest path from that node to the base station. And in order to minimize the energy consumed per round of communication, all packets sent from all nodes have to be routed over shortest paths, i.e., over a shortest path tree rooted at the base station. Here, with shortest path we mean the path with the minimum number of hops, since the cost of every link is the same. But after forming the scatternet as a shortest path tree, we have to make arrangements so that the degree of each node does not exceed seven.

The minimum energy that is consumed per round of communication, for a given number of nodes, can be approximated with the energy spent in a round in a tree that is formed in such a way that each node except root has six children. The root, which corresponds to the base station, can have seven children. We can call this a 6-ary tree. The total energy consumption (E_{total}) per round of communication in such a tree can be expressed as follows:

$$E_{total} = 7 \times \sum_{i=1}^{\lfloor \log_6 N \rfloor} 6^{i-1} \times i + (N - 1 - 7 \times \sum_{i=0}^{\lfloor \log_6 N \rfloor - 1} 6^{i}) \times \lceil \log_6 N \rceil) \times \alpha \quad (4.1)$$

where N is the number of nodes and α is a constant value of energy consumed to transmit and receive a packet between neighboring nodes. It is an approximate value because the tree is not an exact 6-ary tree since the base station can have seven slaves. This is a lower bound on energy consumption per round of communication. We can use this lower bound in our simulations to compare it with the results of our algorithm.

The energy consumption of each node of 6-ary tree will be

$$E_{l,i=1..N} = (\sum_{j=0}^{\log_6 N - l} 6^{j} + 1) \times \alpha \quad (4.2)$$

where l is the level of that node.

If we use data fusion or aggregation, then, according our third assumption, every Spanning Tree (ST) with node degree ≤ 7 can be a solution to our problem. The degree constrained ST can be formed by the existing algorithms [27]. Total energy consumption of ST per round will be

$$E_{total} = (N - 1) \times \alpha \quad (4.3)$$

Data fusion combines several packets into a single packet [7]. As seen from the equation 4.3, in addition to bandwidth consumption, data fusion or aggregation reduces energy consumption for communication leading to conserving a significant amount of energy.

The first part of the algorithm (see Algorithm 1) works as follows. In this part, first a shortest path tree spanning all nodes and rooted at the base station is formed using Dijkstra's single-source shortest paths algorithm [4]. Lets call the tree formed in this way a SPT. The SPT can have nodes whose degree is greater than seven. Therefore, after forming the SPT, the algorithm, starting

Algorithm 1 Scatternet Construction Algorithm

```
Input: Distance matrix or neighborhood matrix
Output: Balanced Degree Constrained Tree (BDC Tree)
Form Shortest Path Tree using Dijkstra's Algorithm
for each level $k$=numberOfLevels-1 to 1 do
    for each node $n$ of level $k$ do
        if $n.numberOfChildren > 6$ then
            for each child $ch$ of $n$ do
                for each possible parent $pP$ of $ch$ do
                    if $pP.numberOfChildren < 6$ then
                        disconnect $ch$ from $n$
                        connect $ch$ to $pP$
                        break
                    end if
                end for
                if $n.numberOfChildren \leq 6$ then
                    break
                end if
            end for
        end if
        if $n.numberOfChildren > 6$ then
            while $n.numberOfChildren \geq 7$ do
                Reconnect($n.child$ whose number of descendants is the minimum)
            end while
        end if
    end for
end for
if $root.numberOfChildren > 7$ then
    while $root.numberOfChildren \geq 8$ do
        Reconnect($root.child$ whose number of descendants is the minimum)
    end while
end if
Balance()
```

from the leaves upto the root, checks all nodes if there exists a node that has more than six children, except the base station. Base station can have seven children. If it finds such a node X, then the children of node X is tried to be connected to some other possible parent, whose number of children is less than six. If possible, this is repeated until the number of children of node X becomes at most six. If we cannot reduce the number of children of X to six in this way (that means there is no alternative parent), then, starting from the child of X with minimum number of descendants, each child of X is tried to be connected to possible brothers or possible siblings. If possible, this is repeated until the number of children of X becomes at most six. If, after this process, the number of children of X still exceeds six, then the child A of X with minimum number of descendants is connected to the child B of X where B has minimum number of descendants after A. After getting connected to B, A is disconnected from X. In this way the number of children of X is reduced by one. Then, if B's degree exceeds six, it is tried to be reduced using the same approach applied to X. Hence, a recursive algorithm is used here. Notice that, since the algorithm starts from the bottom, B had already solved its degree problem. So, B had to have at most six children before A is connected to it.

4.3 Balancing Algorithm

In this part of the algorithm, first level nodes are balanced according to their number of descendants. Since the nodes that are one hop apart from the base station will drain more energy due to having more descendants than the other nodes, they will die first. These first level nodes have to forward their descendants data in addition to their own sensed data. The situation will be worse if they are formed in an unbalanced manner, in other words, the number of descendants will differ a lot. The nodes with more descendants will die quicker than the nodes with less descendants. Furthermore, if the children of that node do not have any other possible parents, these children cannot forward their data to the base station when that node dies. Figure 4.1 shows a network that is unbalanced at the first level. In this figure, node B has six descendants while node A has only

Algorithm 2 Reconnect(node)

```
boolean cont=true
tempParent=node.parent
for each node.possibleParents pP do
    if pP.numberOfChildren < 6 then
        disconnect node from tempParent
        connect node to pP
        cont=false
        break
    end if
end for
if cont then
    for each node.possibleSiblings pS do
        if pS.numberOfChildren < 6 then
            disconnect node from tempParent
            connect node to pS
            cont=false
            break
        end if
    end for
end if
if cont and number of possible brothers ≥ 1 then
    brother=child of tempParent whose number descendants is the minimum after node
    disconnect node from tempParent
    connect node to brother
    Reconnect(brother.child whose number of descendants is the minimum)
end if
```

one. The other first level nodes do not have any descendants. The dashed lines show the reachibility information. If there is a dashed line between two nodes, the nodes are not connected with a Bluetooth link at the moment, but can be connected with a Bluetooth link if required. The balance of this tree can be improved at the first level, because the nodes D, F, and G can be connected to the nodes A, C, and F, respectively. When this re-arrangement is done, the tree will be more balanced at the first level. Note that we are only concerned with balancing at the first level of the tree, since this is the level that will have nodes to die first. If we do not balance the tree, node B can die very fast. After the death of node B, nodes D, F, and G can be connected to other parents, namely to A, C, and F, respectively. But node E does not have any other parent to connect to. Node E and its descendants can only connect to node G. The new of the tree (i.e. scatternet) after such a balancing is shown in Figure 4.2.

The balancing should be done in a way so that the degree constraints of the nodes are not violated. Additionaly, our balancing algorithm balances the descendants of first level nodes in such a way that the energy consumption in one round of communication is not increased in the resulting topology. In fact the energy consumption may even decrease. Although our algorithm is only concerned with balancing at the first level at the moment, if needed, it can be easily modified to balance other levels as well. We just have to call it recursively to balance other levels.

The idea of the algorithm can be illustrated using the Figure 4.3(a). In the figure, a number beside a node shows the number of descendants of that node. For the sake of simplicity, we will label nodes in the network with those numbers. The bold lines in the figure show the current connections between nodes, and the dashed lines show that the nodes connected with dashed lines are within communication range of each other. Nodes connected with dashed lines are not connected at the moment with a Bluetooth link, but can be connected if desired. Balancing the descendants of nodes at the same level causes also balancing the energy consumption at those nodes, since the energy consumption at a node depends linearly on the number of descendants of that node. This is because a node has to forward the data of its descendants as well. The equation 4.5

expresses the amount of energy consumption at a node X $(E(X))$ as a function of its descendant nodes. The number of descendants of a node X $(D(X))$, on the other hand, can be expressed depending on the descendants of its set of children (ς).

$$E(X) = (D(X) + 1) \times \alpha \ . \tag{4.4}$$

$$D(X) = \mid \varsigma \mid + \sum_{i=1}^{|\varsigma|} D(\varsigma_i), \quad \varsigma_i \in \varsigma \ , \ 1 \leq i \leq \mid \varsigma \mid \ . \tag{4.5}$$

Although it is not explicit in the equation, the amount of energy consumed at a node depends on both the shape and size of the sub-tree rooted at that node. The size of the sub-tree minus one is the number of descendants of that node.

In order to balance the energy consumption, we have to make the number of descendants of first level nodes as equal as possible. To achieve that, we look to the nodes at the second level (these nodes are the children of first level nodes) and find the one that has the maximum number of descendants. We then try to reconnect it to another parent in the first level. We are starting from the second level node that has the maximum number of descendants because if we dont, we may not reconnect that node after some changes. The idea can be best described by an example shown in Figure 4.3. In this example, we start from node labeled with 15 because it is the maximum, indicating that this node has the maximum number of descendants (15 descendants). Since there is no other possible parent of 15 other than 16, we leave it as it is. Second maximum number is 11. We look all the possible parents of 11 and see which one has the least number of descendants other than descandants due to node labeled with 11. Node labeled initially with 32 has 20 descendants (32-11-1) and node labeled with 10 has 10 descendants. Therefore, we choose the node labeled with 10 in the first level as the new parent of the node labeled with 11 in the second level. We disconnect node 11 from 32 and connect it to node 10. New values of parents will be 20 and 22, whereas they were 32 and 10 earlier. So we achieve a better balance at the first level.

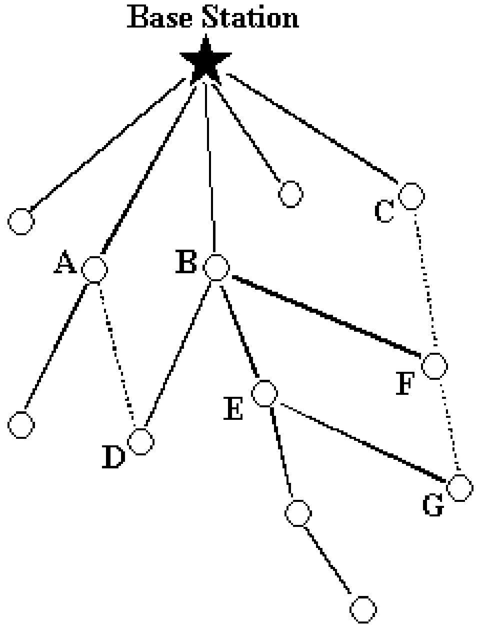

Figure 4.1: Unbalanced tree.

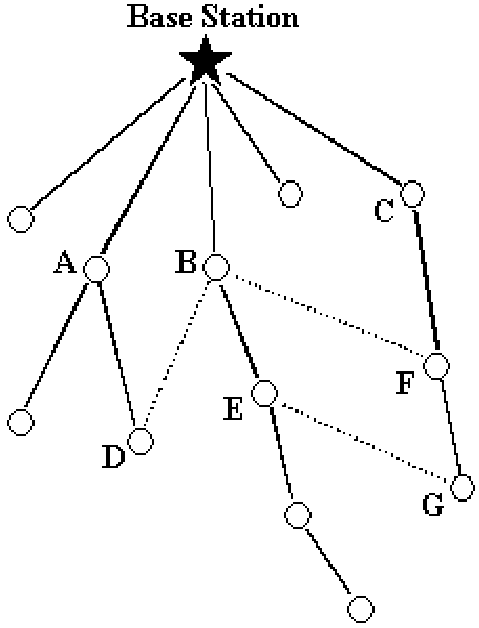

Figure 4.2: Balanced tree.

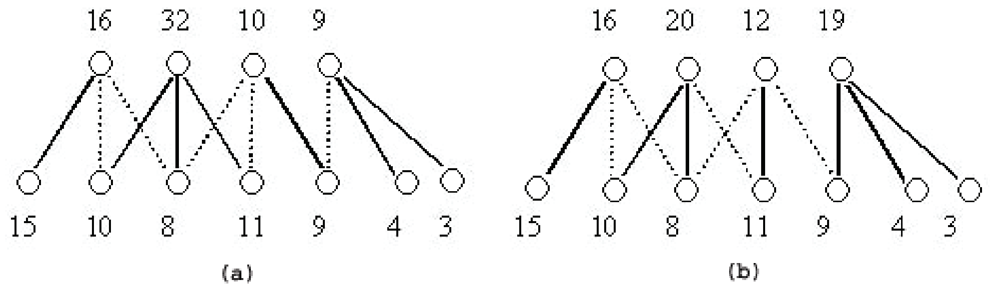

Figure 4.3: a) Unbalanced nodes; b) Balanced nodes

We continue doing the same procedure until all nodes at the second level are checked in the sorted order of their labels. Figure 4.3(b) shows the balanced configuration of nodes. As it can be seen in the figure, the node that was consuming the maximum energy in the unbalanced configuration, is still the node that is consuming the maximum energy, but its energy consumption is reduced by about 30 %.

We next describe how our balancing algorithm (Algorithm 3), acting as shown in the example above, is working. The algorithm starts from the second level nodes. Starting from the node that has the maximum number of descendants, each node is checked if it can be connected to another possible parent, whose grandparent has fewer descendants than its current grandparents descendants minus its own descendants. If it finds a new parent, then that node will be connected to a new parent, and will be disconnected from its current parent. Then the algorithm continues doing the same procedure further with other second level nodes at each time with the node with less number of descendants. After finishing the second level nodes, it continues doing with the third level, fourth level, and so forth.

The time complexity of balancing algorithm is $O(n \log^2 n)$.

We do the same procedure starting with the leaves towards the second level, from bottom to top. And achieved the result that is a little bit worse than the top to bottom approach.

Algorithm 3 Balance()

```
for each level $k$=2 to numberOfLevels do
    Sort level $k$ nodes in descending order according to their labels expressing the number of descendants
    for each node $n$ of level $k$ do
        $newParent$=$Min$(possible parents of $n$)
        if $n$.parent $\neq$ $newParent$ then
            disconnect $n$ from $n$.parent
            connect $n$ to $newParent$
        end if
    end for
end for
```

Algorithm 4 Min(array)

```
return the node whose grandparent's energy consumption is minimum and the number of children $< 6$
```

4.4 Correctness of the Algorithm

Our algorithm, after solving the degree constraint of the tree formed as a shortest-path tree, balances the traffic load of the first level nodes. In this section, we will prove the correctness of the part of the algorithm that is assuring degree constraint of the nodes.

In [35], authors show that if a node has more than five neighbors, then at least two of these neighbors are neighbors themselves. This means that if a node has more than five neighbors then the node and all its neighbors can be connected using Bluetooth. To connect them using Bluetooth, we select the node with several neighbors as the master. If the master node has more than seven neighbors, then, we select seven of these neighbors as slaves. The remaining neighbors are not connected to the master, but are connected to one or more of the slaves of the master. These slaves of the master, that are connected to other neighbors, will be functioning as bridges.

The shortest-path tree that is initially formed in our algorithm is constructed using Dijkstra's well-known single-source shortest path algorithm (actually bread-first search could also be used to construct this tree). In the second part of our algorithm, i.e. in the balancing part, a node will not be connected to another parent if it does not have any possible parent (see Algorithm 3). That means, the second part of our algorithm can not violate the degree constraints of the nodes. And the degree constraints are satisfied with the Reconnect() procedure executed in the first part of our algorithm. Therefore, it is enough to prove only the correctness of the procedure Reconnect(). In other words, we want to prove that procedure Reconnect() ensures that every node in the tree has a degree less than or equal to seven. That means every node except the root will have link to exactly one parent and links to at most six children.

Proposition 1 *Procedure Reconnect ensures that every node in a tree has a degree no more than seven.*

Proof of Proposition 1:
We will prove by induction.
Basis step: Since algorithm starts checking for the degree from the pre-last level nodes up to root, by the observation in [35], we can connect all nodes satisfying the degree constraint since algorithm runs until the degree of a node becomes less than seven.
Inductive hypothesis: Assume that algorithm has solved all level nodes until level n, including level n, for degree constraint.
Statement to be proven: The algorithm solves the degree constraint problem of level *n-1*.
Proof of inductive step: If there is no node at level *n-1* that has a degree more than seven, then we are done. If a node X has more than seven degree, then, starting from the child whose number of descendants is minimum, each of its child tries to connect to possible parent until the degree of X is reduced to seven. If the degree of X is still more than seven, then, again starting from the child whose number of descendants is minimum, each child tries to connect to possible sibling. If X has still a degree more than seven, then the child Y of X with the minimum number of descendants connects itself to child Z of X with the minimum number of descendants after Y. By our inductive hypothesis Z has no more than seven degree. If the degree of Z is smaller than seven, then another

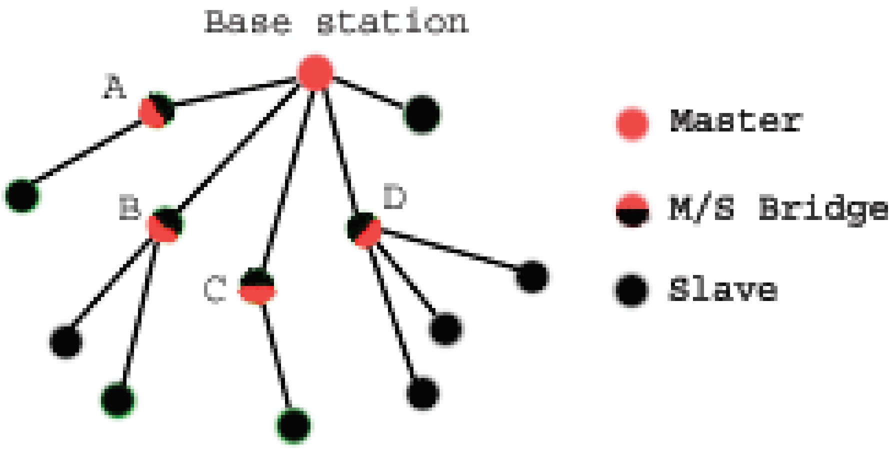

Figure 4.4: A scatternet after role assignment.

new child $Y2$ is selected and the same procedure will be done until the degree of X becomes less than or equal seven. If Z has degree seven, new degree becomes eight, then the algorithm continues doing the same procedure with Z as with X, recursively. This all procedure continue until X's degree becomes less than or equal to seven. Thus, algorithm solves the degree constraints of level *n-1*. ■

4.5 Role Assignment

After forming a Bluetooth scatternet, the nodes must be assigned to master or/and slave roles to be able to communicate. There is only one way that each node can take the role assignment. The role assignment will be done as follows. The base station, the root of a tree, will be a master, the leaves will be slaves, and the other internal nodes will be a M/S bridge, master of its children and slave of its parent. Figure 4.4 shows the example of role assignmnet, where the base station is a master and the internal nodes A, B, C, and D are M/S bridges. The other nodes are all slaves. This way of role assignment eases the routing.

4.6 Routing

Routing in a BDC Tree scatternet is simple. Since only one path exists between each node and a base station, no routing tables or other node's address have to be kept in the memory of nodes, except the address of a master node and children. Each node will forward its data to its master. Master, then forwards it to its master and so on, until the data reaches the base station. However, it becomes a complication if one of the nodes on the path will fail.

4.7 Node Failures

Sensor nodes may fail because of the complete depletion of the node's energy or from physical destructions. In the case of node failures, the children of that failed node will try to connect to another possible parent if the failed node is master. If there is more than one possible parent, then the nodes will connect to the one that has the less number of descendants. If there is no possible parent that has less than six children, then they try to connect to the nodes that are the nearest to a base station. If the failed node is a leaf, then nothing should be done to the network. Notice that all these reconnection steps are done at base station.

4.8 Scheduling

Scheduling is also important aspect since good scheduled communication links reduces delay and increases throughput. The scheduling of communication links in BDC Tree can be done as follows. At a given time, the children of a given master of each nonconsecutive level transmit data to its master. After the completion of transmission of the data, each master switches to slave role and transmits the collected data to its respective master. If the time that takes for a master M to collect data from its children is T_s, and the time that takes for M (now slave) to transmit its collected data to its master is T_m, then children of a master M can transmit their data at each T_m. Master node M, therefore, should switch to different piconets after each T_s and T_m alternately. While master node M of leaves is acting as a slave in another piconet, leaves can switch to low power mode for T_m time to conserve energy.

Chapter 5

Simulation and Results

In this chapter we describe our simulation environment and the results obtained from our simulations. We have implemented our simulation using Java programming language. Our simulation is static without a time axis.

In our simulation experiments, we compare performance results for various topologies: unbalanced degree constrained tree (UDC Tree), balanced degree constrained tree (BDC Tree), shortest path tree (SPT), and 6-ary tree (a totally balanced and degree constrained tree). 6-ary tree gives the lower bound for energy consumption in balanced tree satisfying Bluetooth node degree constraint. SPT gives the lower bound for energy consumption in a tree that is not a feasible topology for Bluetooth technology since some of the nodes in an SPT tree can have more than seven slaves.

In our simulation model, different number of nodes, ranging from 75 to 500, are deployed randomly on an area of 50 m by 50 m. Since some of the nodes may not have any neighboring nodes after random deployment, we get rid those nodes and consider only nodes that have at least one neighbor as the members of a sensor network. For each simulation experiment we repeat running the simulation 100 times and we take the average of 100 measurements while finding the results for that experiment.

Figures 5.1, 5.2, 5.3, and 5.4 show how 150 nodes are scattered randomly over the region that is 50 m x 50 m, and how various topologies look like: shortest path tree, unbalanced degree constrained scatternet, and balanced degree constrained scatternet. As it can be seen in the figure, shortest path tree can have nodes that have node degree greater than seven. But, both UDC and BDC based scatternets satisfy the degree constraint: the number of children of each node is no more than six, except the root node (the root can have seven children). Notice that some nodes in unbalanced tree have different parents in the balanced version.

In figure 5.5, energy consumption per round of communication versus number of nodes in the network is shown. The lower bound of energy consumption for a given set of nodes is achieved if routing is done according to a shortest path tree. As it is seen in the figure, the energy consumption in sparsely deployed networks is almost equal as the lower bound. However, in densely deployed networks, the energy consumption is more than the lower bound. This is because some of the nodes have more than six children in a dense network and forcing the tree to be degree constrained make the tree non-optimal for routing. The children of nodes with degree greater than seven have to connect to other possible brothers or siblings, thus, the number of hops between those nodes and the base station increase, which also increases the total energy consumption per round of communication.

Figure 5.6 shows the energy consumption of the maximum energy consuming node versus the number of nodes in the network for different topologies: SPT, UDC tree, and BDC tree. BDC tree has better energy consumption values compared to UDC tree. Balancing algorithm reduces the energy consumption of the maximum energy consuming node by about 30 % to 50 %. Thus, the lifetime of the first dying node is increased by about 40 % to 100 % (assuming that all nodes have equal amount of initial energy).

Figure 5.7 shows the average number of slaves per node in a network. The number of nodes in the network is varied on the x-axis. The figure also shows the average number of hops between a node and the base station. The average number of slaves increases slightly as the number of nodes increases, as expected.

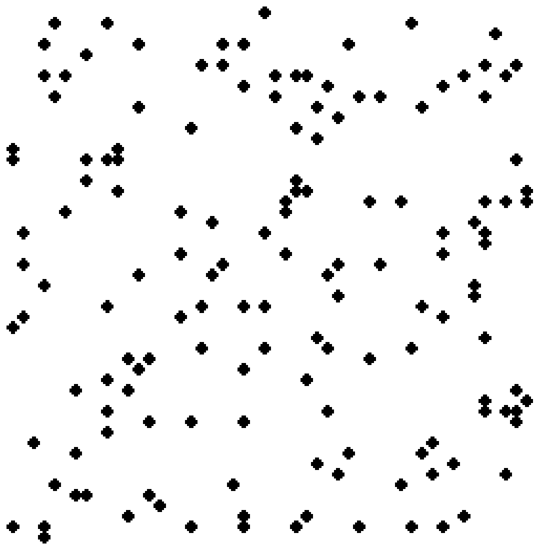

Figure 5.1: Randomly deployed sensor nodes.

The hop number is between 4 and 4.5. There is no significant change on the number of hops as a function of number of nodes. We think this is because as the network becomes denser, both the number of nodes which are nearer to the base station (small hop count) and the number of nodes which are further (large hop count) increases with the same ratio not changing the average value of hop count.

Figure 5.8 compares the total energy consumption per round in a BDC tree topology and in a 6-ary tree topology. 6-ary topology is an optimal configuration to consume minimum energy in a balanced tree satisfying Bluetooth constraints. We can see that energy consumed in a BDC tree is a little more than the lower bound, i.e. energy consumed in a 6-ary tree.

Figure 5.9 shows an histogram which counts the number of nodes consuming energy in some interval. The network size used for this experiment is 200 nodes. We can see that the number of nodes consuming between 20 and 40 units of energy per round is more in the BDC (balanced) tree topology than in a UDC (unbalanced) tree topology. On the other hand, the number of nodes consuming more than 40 units of energy per round is less in the BDC tree topology than in the UDC tree topology. This also indicates that a network with BDC tree topology is more energy efficient and will have longer lifetime compared to a network with UDC tree topology.

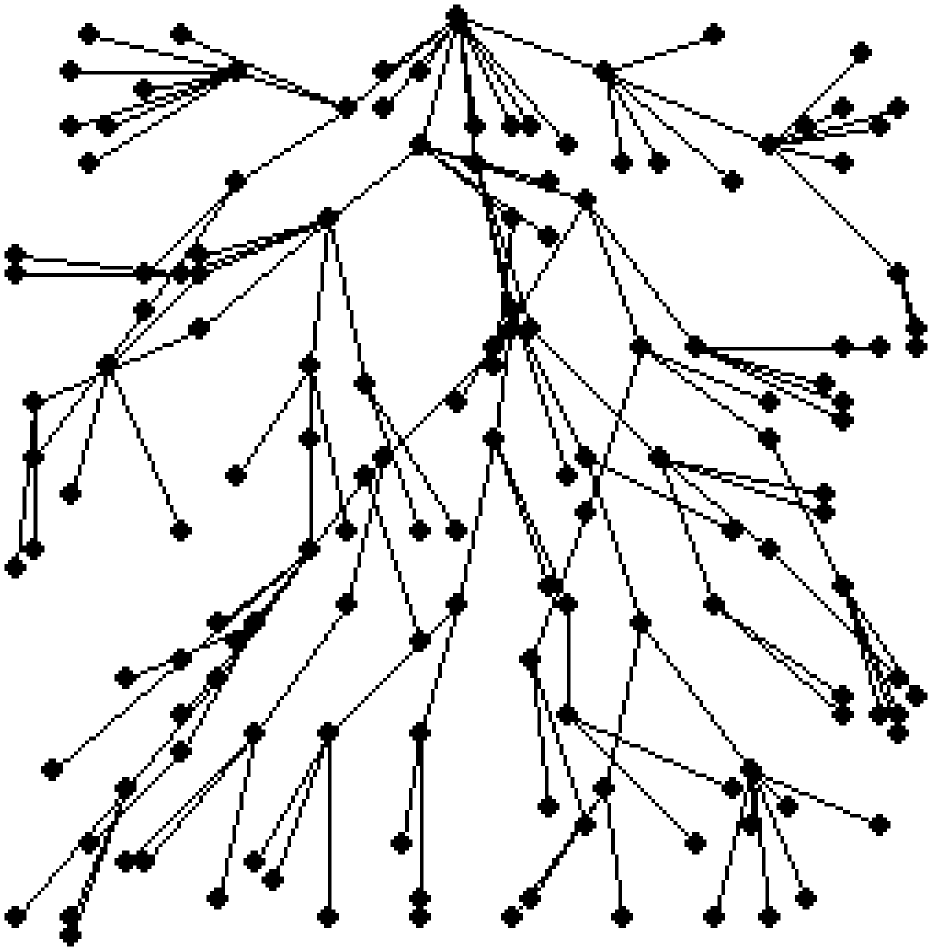

Figure 5.2: Shortest Path Tree formed from randomly deployed nodes.

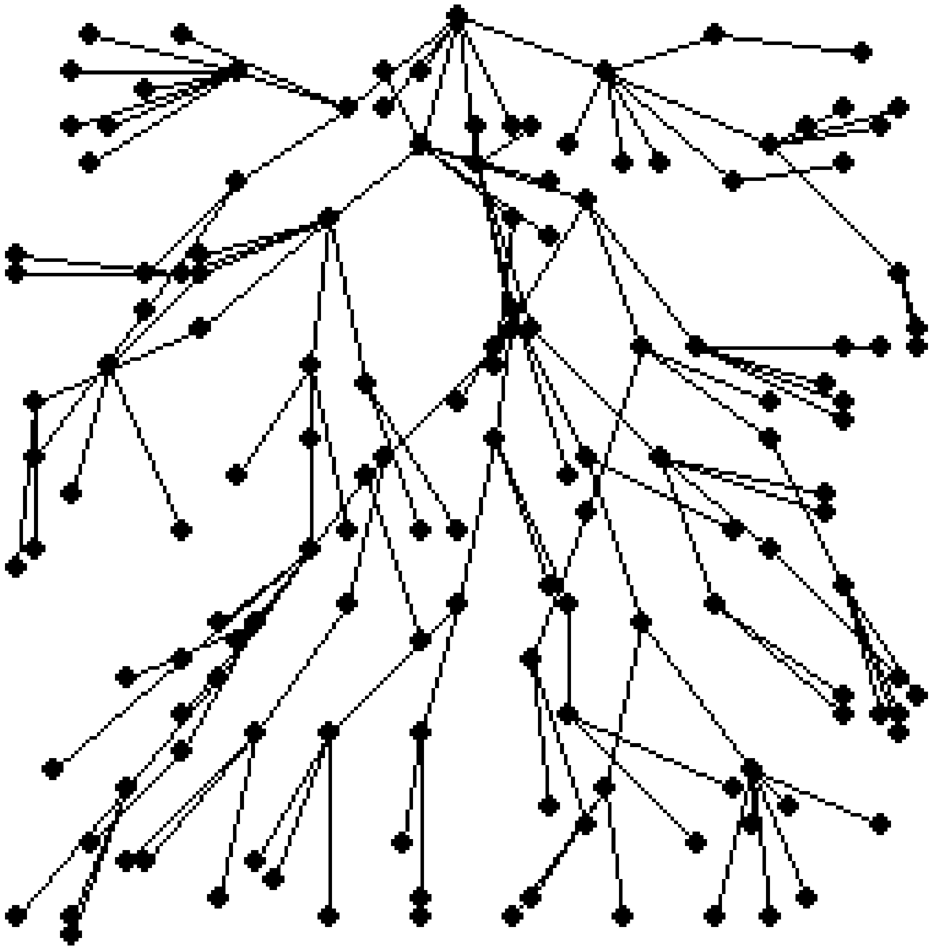

Figure 5.3: Unbalaced Degree Constrained Tree (UDC Tree).

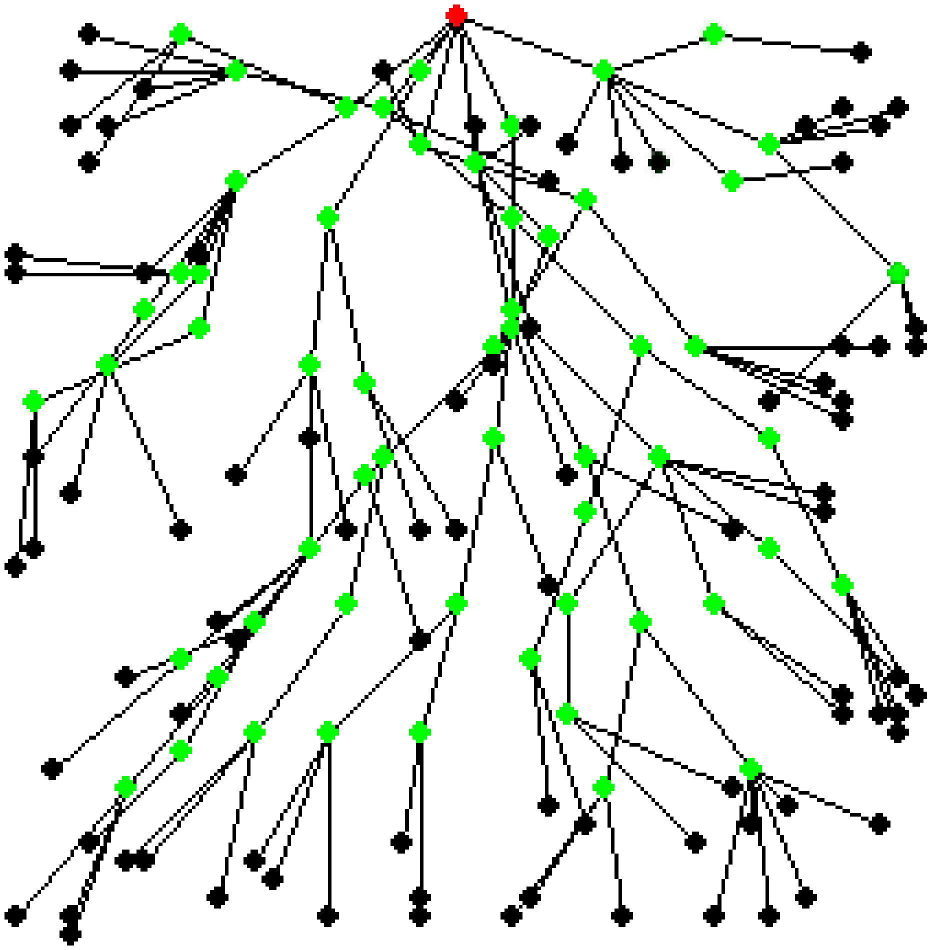

Figure 5.4: Balanced Degree Constrained Tree (BDC Tree). Light-color nodes are the M/S bridges, dark-color nodes are the slaves, and base station is a master.

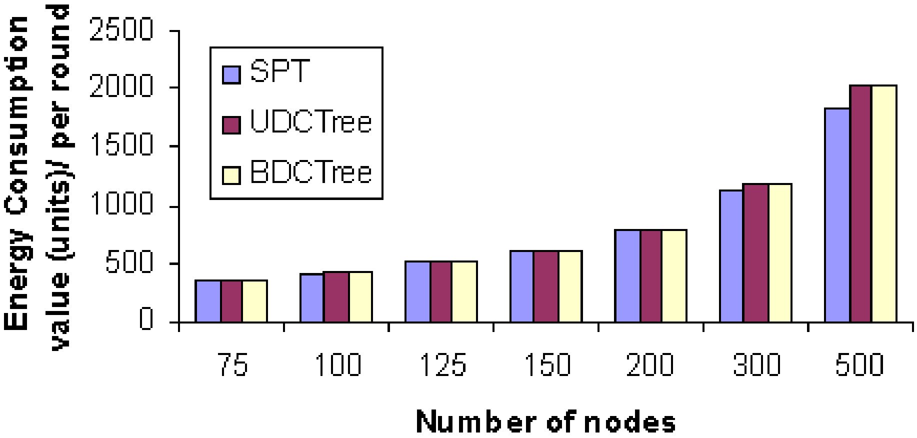

Figure 5.5: Average energy consumptions of SPT, UDC Tree, and BDC Tree per round.

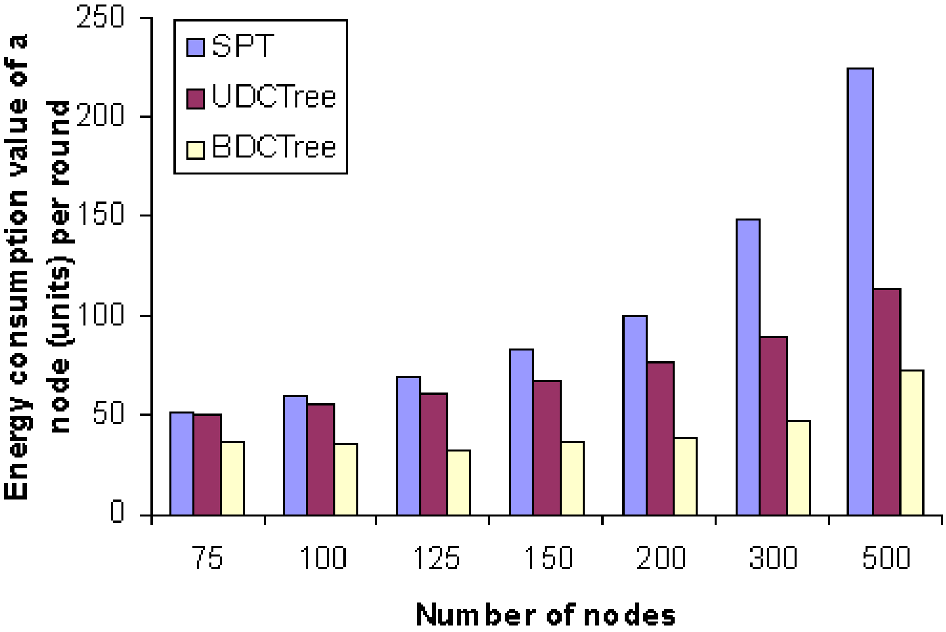

Figure 5.6: Average maximum energy consumptions of a node in SPT, UDC Tree, and BDC Tree per round.

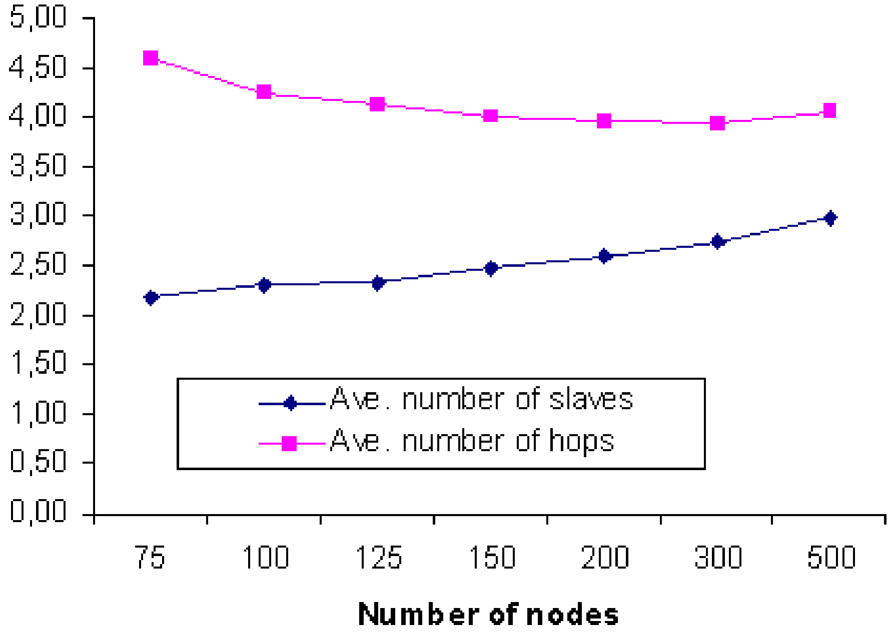

Figure 5.7: Average number of hops of BDC Tree as a function of node numbers.

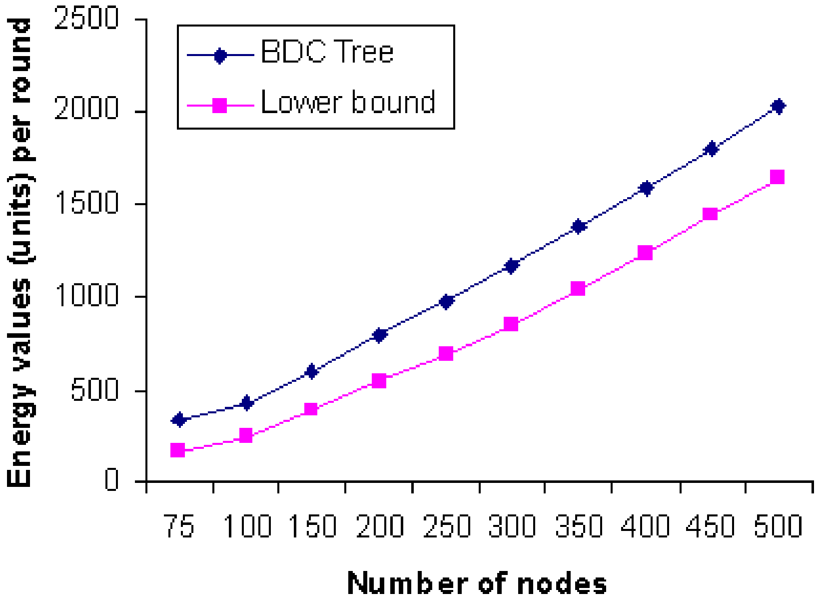

Figure 5.8: Comparison of energy consumptions of BDC Tree with lower bound.

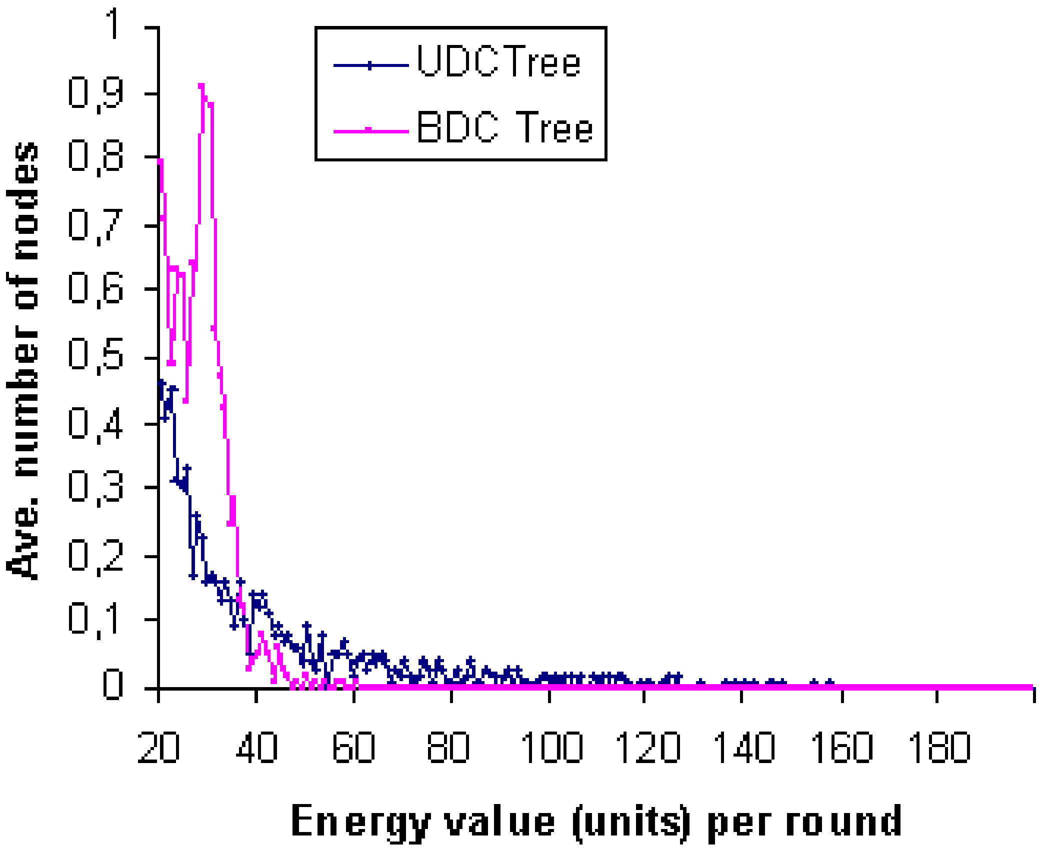

Figure 5.9: Average number of nodes as a function of energy consumption value in a network size of 200 nodes.

Chapter 6

Conclusions and Future Work

Bluetooth is one of the wireless communication technologies that can be used for wireless sensor network applications. Its reducing cost over time, low power, and small size are the main features that should be met for sensor network requirements, while the delay of connection and the up to seven active slaves per piconet at a given time are the drawbacks.

In this thesis work, we proposed an algorithm about how to form an energy efficient Bluetooth scatternet for wireless sensor network applications to prolong the lifetime of the network. Our aim was to form a scatternet that consumes less energy per round of communication and to balance the energy consumption among the nodes. After forming an initial shortest path tree topology spanning all the nodes and rooted at the base station by using Dijkstra's SPT algorithm, our algorithm then solves the degree constraints of the nodes in the tree and then balances the first level nodes.

Simulation results show that our algorithm consumes little bit more than the lower bound. Lowest energy consumption in a network of nodes using Bluetooth can be achieved when the network is formed in a 6-ary manner. Our balancing algorithm prolongs the lifetime of the first dying node up to 100%.

Since our first part of the algorithm does not guarantee that the formed tree is shortest path tree, one of the future works can be done on finding degree constrained shortest path tree before balancing the energy consumption among the nodes. However, it may not be still more balanced than our algorithm since balancing algorithm does not depend on the formed links in a tree.

In one of our assumptions, we have assumed that all the links have the same weights (see Section 3.1, Third assumption). This is because we assumed that Bluetooth devices have no power control unit. So, another work can be done with the devices with power control unit, in other words, the links have not equal weights. So, in this future work, the minimum number of hops may not give us the minimum energy consumption per round of communication.

Bibliography

[1] N. Bambos. Toward power-sensitive network architectures in wireless communications: Concepts, issues, and design aspects. *IEEE Personal Comm.*, pages 50 – 59, June 1998.

[2] P. Bhagwat. Bluetooth: technology for short-range wireless apps. *IEEE Internet Computing*, 5:96 – 103, May/June 2001.

[3] C. Chong and S. P. Kumar. Sensor networks: Evolution, opportunities, and challenges. In *Proceedings of the IEEE*, volume 91, Aug. 2003.

[4] E. W. Dijkstra. A note on two problems in connection with graphs. *Numerische Mathematik 1*, pages 269 – 271, 1959.

[5] D. Estrin, L. Girod, G. Pottie, and M. Srivastava. Instrumenting the world with wireless sensor networks. In *Proceedings of the International Conference on Acoustics, Speech andSignal Processing(ICASSP 2001)*, May 2001.

[6] T. He, S. Krishnamurthy, J. A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. Krogh. An energy-efficient surveillance system using wireles sensor networks. *MobySys'04*, June 2004.

[7] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *33rd Annual Hawaii International Conference on System Sciences,*, pages 3005 – 3014, 2000.

[8] J. Hill, M. Horton, R. Kling, and L. Krishnamurthy. The platforms enabling wireless sensor networks. *Communications of the ACM*, 47:41 – 46, June 2004.

[9] A. Iwata, C.-C. Chiang, G. Pei, and T.-W. Chen. Scalable routing strategies for ad hoc wireless networks. *IEEE Journal on Selected Areas in Communications*, pages 1369 – 1379, Aug. 1999.

[10] M. Joa-Ng and I.-T. Lu. A peer-to-peer zone based two-level link state routing for mobile ad hoc netwroks. *IEEE Journal on Selected Areas in Communications*, pages 1415 – 1425, Aug. 1999.

[11] D. Johnson and D. Maltz. Dynamic source routing in ad hoc networks. In *Mobile Computing*, 1996.

[12] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecci. Optimization by simulated annealing. *Science*, 220:671 – 680, 1983.

[13] C. Law, A. K. Mehta, and K.-Y. Siu. A bluetooth scatternet formation algorithm. In *Proceedings 2001 ACM International Symposium on Mobile ad hoc networking computing*, pages 183 – 192, Oct. 2001.

[14] C. Law and K.-Y. Siu. A bluetooth scatternet formation algorithm. In *Proceedings Global Telecommuniations Conference 2001 GLOBECOM'01*, volume 5, pages 2864 – 2869, Nov. 2001.

[15] M. Leopold, M. Dydensborg, and P. Bonnet. Bluetooth and sensor networks: A realty check. In *Proceedings of SenSys 2003*, Nov. 2003.

[16] S. Lindsey and C. S. Raghavendra. Pegasis: Power-efficient gathering in sensor information systems. In *IEEE Aerospace Conference*, March 2002.

[17] H. Mathias, D. Jan, and T. Dirk. Energy-efficient data collection for bluetooth-based sensor networks. *IEEE Instrumentation and Measurement Technology Conference*, May 2004.

[18] V. Mehta and M. E. Zarki. Fixed sensor networks from civil infrastructure monitoring - an initial study. In *First Annual Mediterranean Ad Hoc Networking Workshop(Med-hoc-Net 2002)*, Sept. 2002.

[19] B. A. Miller and C. Bisdikian. *Bluetooth Revealed: The Insider's Guide to an Open Specification for Global Wireless Communications.* Prentice Hall, 2001.

[20] S. Murthy and J. J. Garcia-Luna-Aceves. An efficient routing protocol for wireless networks. *ACM Mobile Networks and Applications Journal*, pages 183 – 197, Oct. 1996.

[21] V. Park and M. S. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In *INFOCOM'97*, Apr. 1997.

[22] C. E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing(dsdv) for mobile computers. *Computer Communications Review*, pages 234 – 244, Oct. 1994.

[23] C. E. Perkins, E. M. Royer, and S. R. Das. Ad hoc on-demand distance vector routing, Oct. 1999.

[24] C. Petroli, S. Basagni, and I. Chlamtac. Configuring bluestars: Multihop scatternet formation for bluetooth networks. *IEEE Transactions on Computers*, 52:779 – 790, June 2003.

[25] R. Powers. Batteries for low power electronics. In *Proceedings of the IEEE*, pages 687 – 693, April 1995.

[26] B. J. Prabhu and A. Chockalingam. A routing protocol and energy efficiency techniques in bluetooth scatternets. *Design and Test of Computers*, Apr./May 2002.

[27] G. R. Raidl. An efficient evolutionary algorithm for the degree-constrained minimum spanning tree problem. In *Proceedings of the 2000 IEEE Congress on Evolutionary Computation*, volume 1, pages 104 – 111, 2000.

[28] T. Salonidis, P. Bhagwat, L. Tassiulas, and R.LaMaire. Distributed topology construction of bluetooth personal area networks. *Proceedings Twentieth Annual Joint Conference of the IEEE Computer and Communication Societies IEEE INFOCOM 2001*, 3:1577 – 1586, Apr. 2001.

[29] K. Sivalingam, J. Chen, P. Agrawal, and M. Srivastava. Design and analysis of low-power access protocols for wireless and mobile atm networks. *ACM/Baltzer Wireless Networks*, 6:73 – 87, Feb. 2000.

[30] S. Slijepcevic, M. Potkonjak, V. Tsiatis, S. Zimbeck, and M. B. Srivastava. On communication security in wireless ad-hoc sensor networks. In *WET-ICE'02*, 2002.

[31] H. O. Tan and I. Korpeoglu. Power-efficient data gathering and aggregation in wireless sensor networks. *ACM SIGMOD Record*, 32:66 – 71, Dec. 2003.

[32] S. Tilak, N. B. Abu-Ghazaleh, and W. Heinzelman. A taxonomy of wireless micro-sensor network models. *ACM Mobile Computing Communications Review(MC2R'02)*, 2002.

[33] P. Trakdas, T. Zahariadis, S. Voliotis, and C. Manasis. Efficient routing in pan and sensor networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 8(1), 2004.

[34] P. F. Tsuchiya. The landmark hierarchy: a new hierarchy for routing in very large networks. *ACM SIGCOMM Computer Communication Review*, 18(4):35 – 42, Aug. 1988.

[35] G. V. Zaruba, S. Basagni, and I. Chlamtac. Bluetrees-scatternet formation to enable bluetooth-based ad hoc networks. In *Proceedings of IEEE International Conference on Communications*, volume 1, pages 273–277, June 2001.

Appendix A

List of Acronyms

ACL	Asynchronous Connection-Less
BDC	Balanced Degree Constrained
FEC	Forward Error Correction
FHSS	Frequency Hopping Spread Spectrum
ISM	Industrial Scientific and Medical band
MST	Minimum Spanning Tree
PAN	Personal Area Network
PDA	Personal Digital Assistant
PFA	Packet Forward Address
QoS	Quality of Service
RF	Radio Frequency
SCO	Synchronous Connection Oriented
SPT	Shortest Path Tree
TDD	Time Division Duplex
UDC	Unbalanced Degree Constrained
WSN	Wireless Sensor Network

Glossary

PAN - Personal Area Network

A Personal Area Network is the interconnection of information technology devices within the range of an individual person, typically within a range of 10 meters.

PDA - Personal Digital Assistant

A personal Digital Assistant can be generally described as a small handheld computer holding such information as dairies, address books etc.

RF - Radio Frequency

Any frequency within the electromagnetic spectrum normally associated with radio wave propagation.

FHSS - Frequency Hopping Spread Spectrum

In FHSS the total frequency band is split into a number of channels. The broadcast data is spread across the entire frequency band by hopping between the channels in a pseudo random fashion.

TDD - Time Division Duplex

In a Time Division Duplex system common carrier is shared between the uplink and downlink, the resource being switched in time. Users are allocated one or more timeslots for uplink and downlink transmission. The main advantage of TDD operation is that it allows asymmetric flow which is more suited to data transmission.

Data Aggregation

Data aggregation is any process in which information is gathered and expressed in a summary form, for purposes such as statistical analysis.

Data Fusion

Data fusion is the combining of data from different complementary sources to form a coherent information.

Data Compression

Reducing the representation of the information, but not the information itself. Compression is accomplished by running a data set through an algorithm that reduces the space required to store, or bandwidth required to transmit the data set.

FEC - Forward Error Correction

A method of communicating data that corrects errors in transmission on the receiving end. Prior to transmission, the data is put through a predetermined algorithm that adds extra bits specifically for error correction to any character or code block. If the transmission is received in error, the correction bits are used to check and repair the data.

Shortest Path

Given nodes n_1 and n_2, the shortest path from n_1 to n_2 is a path P such that $\sum_{e \in P} w(e)$ is minimum.

SPT - Shortest Path Tree

Given a weighted graph (G,w) and a node n_1, a shortest path tree rooted at n_1 is a tree T such that, for any other node $n_2 \in G$, the path between n_1 and n_2 in T is a shortest path between the nodes.

Spanning Tree

A connected, acyclic subgraph containing all the vertices of a graph

MST - Minimum Spanning Tree

A minimum-weight tree in a weighted graph which contains all of the graph's vertices.