Prepared (also subject responsible if other)		No.		
MÖ/EMW/FT/X Karlsson, Lundström, Westbergh				
Approved	Checked	Date	Rev	Reference

Ad Hoc Networks with Unattended Ground Sensors

Sensors & Information Networks
Ericsson Microwave Systems AB

Thesis for Bachelor of Science degree in Telecommunication

University of Kalmar
Department of Technology
2002

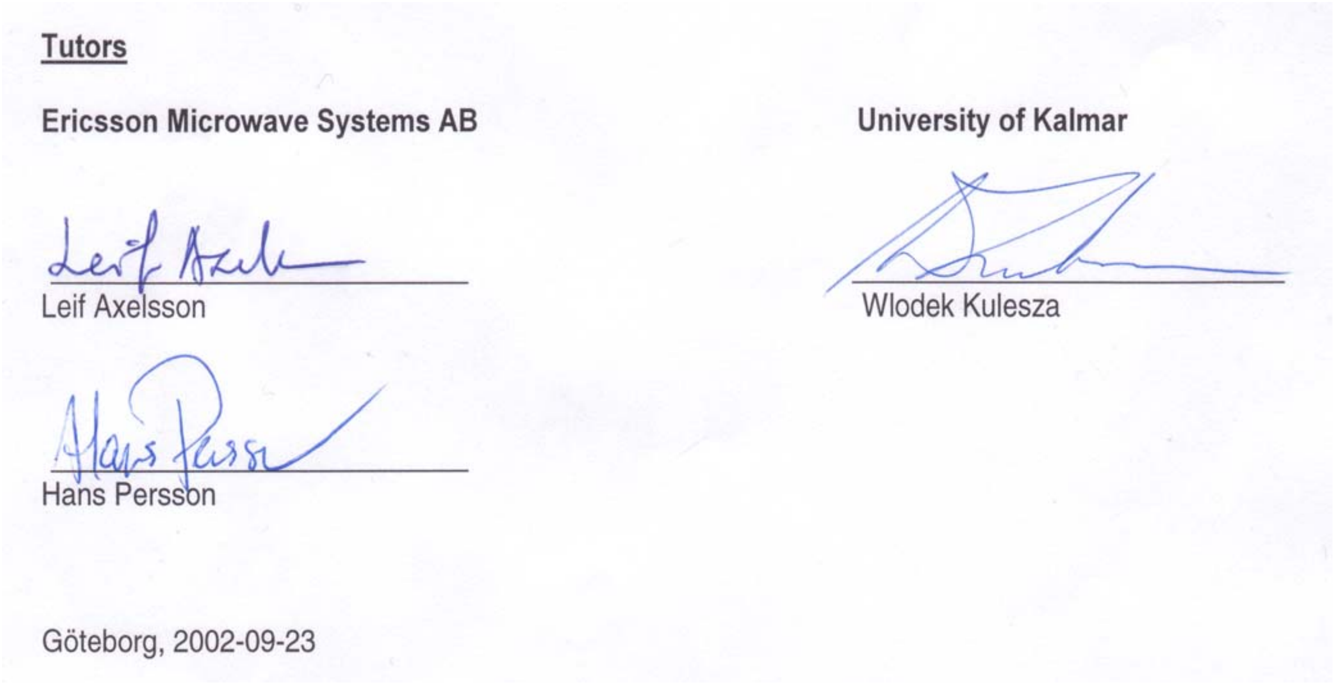
Björn Karlsson
Anders Lundström
Magnus Westberg

Supervisors:
Ericsson Microwave Systems AB: Leif Axelsson and Hans Persson
University of Kalmar: Wlodek Kulesza

Approval

This is to certify the Bachelor thesis ”Ad Hoc Networks Unattended Ground Sensors” done for Ericsson Microwave Systems AB in Göteborg by, Björn Karlsson, Anders Lundström and Magnus Westbergh from the University of Kalmar, has been approved.

The material in this report does not belong to any of the students stated above, has been clearly identified and has never before been included in an examination.



Tutors

Ericsson Microwave Systems AB

Leif Axelsson

Hans Persson

University of Kalmar

Wlodek Kulesza

Göteborg, 2002-09-23

Sammanfattning

Billiga, obevakade sensorer kan ge taktiska fördelar i militära situationer genom att användas för spaning och övervakning. Ett brett spektrum av sensorer lämpar sig för dessa obevakade sensorer. För att inte behöva placera ut sensorerna på förplanerat sätt krävs det att de själva kan bestämma sin position och etablera kommunikation sinsemellan. Låg energiförbrukning är ett krav för långvarig funktion, vilket i sin tur ställer krav på bland annat kommunikationssystemet. Därtill finns ett önskemål om låg sannolikhet för upptäckt av sensorena. En elegant lösning på dessa problem är så kallade ad hoc nätverk, som kan realisera infrastrukturlösa radiosystem där data förmedlas mellan noder på ett i förväg oplanerat sätt. Trots att sensorerna kan placeras ut slumpartat vill man i efterhand kunna kontrollera vilket täckningsområde man har. Därtill vill man kunna styra parametrar, kontrollera status, byta programvara och beordra tillfällig radiotystnad.

För att praktiskt testa och demonstrera denna teknik har vi i detta examensarbete på Ericsson Microwave Systems AB (EMW) byggt upp ett ad hoc nätverk bestående av några obevakade sensorer, vilka utrustats med GPS-mottagare. Nätverket visar samverkande funktionalitet mellan sensorer vilket bidrar till att man får en lokal lägesbild. Utgångspunkten för kommunikationssystemet har varit EMWs ad hoc testnätverk baserat på WLAN. Sensorvalet har styrts utifrån idag tillgängliga sensorer, såsom företaget Exensors UMRA och företaget WeSpots intelligenta kameror. För att öka räckvidden på ad hoc nätverket ingår även rena kommunikationsnoder vars enda uppgift är att förmedla information.

I denna rapport presenteras först bakomliggande teorier för obevakade marksensorer i nätverk, ad hoc routing samt framtida utveckling. Därefter följer en beskrivning av utvecklandet av en demonstrator för denna typ av nätverk.

En kunddemonstration av detta arbete har även färdigställts, där man kan se nyttan av dessa system vid övervakning av ett vägskäl. Demonstratorn kan även ses som en utvecklingsmiljö där man kan testa fundamentala byggklossar inom NBF (Nätverks Baserat Försvar) såsom central och distribuerad datafusion, flexibla kommunikationslösningar både internt och mot omvärld, olika typer av sensorer, säkerhetsaspekter, management- och styrningsproblematik samt användaråtkomst.

Abstract

Low cost, unattended sensors can give great tactical benefits in military situations by using them for searching, recognition and surveillance. A wide spectrum of sensors is suitable for these types of unattended applications. To be able to deploy these sensors without precision or planning, they need to be able to determine their own position and establish communication with each other and the surroundings. A requirement for long lasting operations is low energy consumption. This in turn, gives great emphasis to e.g. the communication system. In addition to this, there is a desire of low probability of recognition of these sensors against outsiders. An elegant solution to these problems is so called *ad hoc networks,* which can realize infrastructure less radio systems in which data can be forwarded through the nodes without planning and without using a centralized administration. Despite that the nodes can be randomly deployed, there is a desire to be able to control the area of coverage, change the parameters of the sensors, check the status of the sensors, change the software of the sensors and order radio silence.

In this final work, performed at EMW (Ericsson Microwave System AB), we have developed an *ad hoc network* with GPS-equipped unattended ground sensors that shows interactive functionality with each other, which results in the achievement of a local situation picture. The aim and purpose of this is to test and demonstrate the technology of UGS in ad hoc networks. The starting point has been a WLAN-based ad hoc testbed of EMW, and the choice of sensors has been choused based on what is available today. E.g. the UMRA (an acoustic, seismic and magnetic sensor) from Exensor AB and intelligent cameras from the company of WeSpot. To achieve greater range in the ad hoc network, we have also made us of pure communication nodes to forward the information.

The first section of this thesis provides theories of UGS networks, ad hoc routing and ideas for future development. The resuming sections describe the development of a conceptual testbed.

A presentation for customers over this project has also been completed, in which the advantages of the system can be observed in the surveillance of a fork. The testbed can also be seen at as an environment of development in which fundamental building blocks of NBF (Swedish: Network Based Defense) can be tested. These blocks could be for instance: i) data fusion, distributed or central, ii) flexible solutions of communication, internal and to the surroundings iii) different types of sensors iv) management of - and controlling of the system, and v) user availability.

Acknowledgement

We would like to give an acknowledgement to all the people who made this thesis and especially the testbed possible. It has been a great pleasure to work with the people at Ericsson Microwave System AB, Exensor Technology AB and WeSpot AB.

Thank you all!

Ad Hoc Networks with Unattended Ground Sensors.

1 Introduction

During the last decade, several applications using wireless communication, such as Bluetooth™ and WLAN (IEEE 802.11b), have been available for the public. The purpose of this technological revolution is to connect different types of devices and thereby allow data to be exchanged. One example of such a device is a Bluetooth™ equipped headset, which allows wireless voice communication towards a mobile phone.

The technological revolution is possible thanks to advances in Digital Signal Processing (DSP), which have paved the way for smaller, low power and faster microchips. These advances have for example made it possible to develop small and effective sensor systems, such as Unattended Ground Sensor (UGS) networks. An UGS network may consist of different types of sensors, depending on its purpose, and provide information about its immediate surroundings.

Ad hoc technology provides the possibility for devices to communicate with each other through each other. Devices that are not the final destination of the information simply acts as intermediaries. This allows reduction of the UGS's transmission output and thereby decreases battery consumption. Characteristic of an ad hoc network is that it does not need any preexisting infrastructure and that the topology may be dynamic. The networks ability of cell healing structure makes the communication less vulnerable for failing links. That is, nodes may fail, be removed or added to the network, but still the information will try to find its way through the network to its final destination.

An UGS network with ad hoc technology allows remote sensing in a hostile environment e.g. chemical industry, rescue mission after an earthquake etc. The network may be deployed quickly thanks to the ad hoc characteristics and the number of sensors and type of sensors may vary depending on the purpose of the network. Sensors may also be added to or removed from the network if required.

1.1 Problem Description

The main purpose of this thesis was to develop the base of a functional conceptual testbed consisting of a dynamic ground sensor network, based on ad hoc technology, to test and demonstrate the possibilities of UGS networks. Construction and choice of the integrated sensors have been made in cooperation between Ericsson Microwave Systems AB with focus on communication, presentation and data fusion in the network, and Exensor Technology AB, focusing on intelligent ground sensors and data fusion, and WeSpot AB dealing with intelligent cameras. The purpose of the testbed was to demonstrate the capability of connecting different types of sensors, equipped with positioning system, to the network. Data extraction and interaction with the sensors should lead to a situation picture of the monitored area.

1.2 Background

The field of networked UGS is a key technology for the future and has become a subject to a lot of interest. UGS is a part of the future military development, Network Centric Warfare (NCW). The vision is small, cheap and technical advanced devices with single or multiple on-board micro sensors networked trough wireless links such as Bluetooth™.

For military applications, the UGS technology provides numerous of advantages in areas such as reconnaissance and surveillance. Deployment of small, disposable sensors can take place whenever and wherever needed to detect threats such as vehicles, chemicals and personnel.

The technology also provides advantages in civilian applications such as autonomous intrusion alarms, chemical sensing in industrial environment etc.

Readers who want to learn more about NCW can visit the Swedish Armed Forces web page and download an introduction movie (in Swedish) [30].

1.3 Project Organization

The following persons have been involved in this thesis:

Thesis authors
Björn Karlsson
Anders Lundström
Magnus Westbergh

Supervisors at Ericsson Microwave Systems AB
Leif Axelsson, Ph. D.
Hans Persson, Ph. D.

Exensor Technology AB
Fredrik Frisk, Ph. D.
Christian Gravengaard, Man. Dir.
David Josefsson
Åke Mathiasson, Tech. Dir.
Kenneth Wester, M. Sc.

WeSpot AB
Fredrik Hederstierna, M. Sc.
Dan Hovang, M. Sc.
Henrik Larne, M. Sc.
Pontus Nobréus, M. Sc.

Examiner at the University of Kalmar, Department of Technology
Professor Wlodek Kulesza

1.4 Disposition

The work described in this thesis is structured as follows.

Firstly, a background of the subject and the theories of UGS networks are presented. Section 2.1 and further presents the vision, usage and problem definitions of Unattended Ground Sensors in an ad hoc network. Chapter 2.9 explains different techniques for ad hoc communication.

The second part of the thesis, starting in section 3, describes the development and implementation of a conceptual testbed. Issues dealt with in this section are presentation, interaction and user application for network data extraction. We describe a scenario for the testbed in chapter 3.1, followed by a presentation of the different consisting parts of the testbed. Communication interface among sensors, Global Positioning System (GPS) and network are found in chapter 3.4 and further. The implementation of the routing algorithm is found in chapter 3.5. The achievements and ideas for future development are given in chapter 3.6 and 3.7, respectively.

2 Ad Hoc Networks with Unattended Ground Sensors

2.1 Introduction to Unattended Ground Sensors

In the last years the development of Unattended Ground Sensors (UGS) has grown into a large research area. Progress has been made in areas like DSP, which has made it possible to create faster, smaller and less power-demanding chip for execution of advanced algorithms in real time. Advances in DSP and CPU technology have accelerated the development of UGS.

The main components of UGS are a battery, single or multiple sensors and a transmission utility for sending information to a remote monitoring location. These three components constitute a so-called node. The sensors can be magnetic, seismic, acoustic, chemical or any of a number of other types.

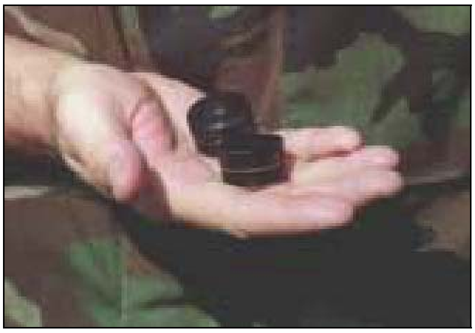


Figure 1. A small UGS. [23]

UGS nodes can be deployed by several means, detect different types of targets and send information about it to a monitoring area. The ideal UGS is a small, robust and inexpensive device. In order to keep the cost down Commercial Off The Shelf (COTS) products have to be used to a great extent. Possible modifications may be necessary to fulfill military applications. The UGS is expected to last for days after deployment. The system can be set to monitor the environment every second with a minimum power requirement, and once a target is detected the UGS power it self up for real time monitoring in order to save battery.

2.2 Vision

The objective of building a sensor network is to receive information and be able to act rapidly, come to a more correct decision and thereby achieve information advantage, which is essential for military applications. The sensor network is a surveillance system that may be a complement to land mines e.g. the enemy is spotted without him knowing about it.

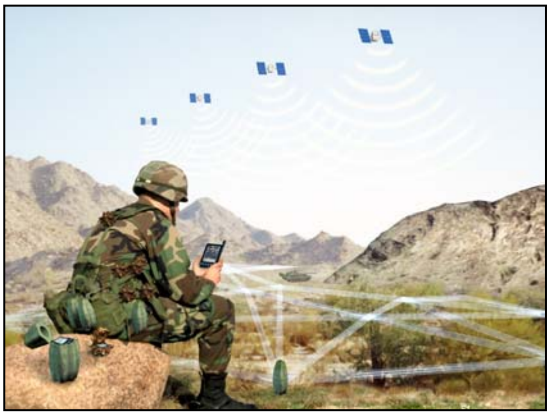


Figure 2. Soldier receiving a situation picture from UGS network.

From a network of sensors, target detection, tracking, localization, and recognition are vital information UGS can determine. Correlation of data from various sensors greatly enhances the target recognition capability. Such a system can achieve more advanced performance to a more sophisticated sensor at a fraction of the cost.

The UGS network has to be able to determine what it sees, where it occurs and when it happens and to supply this information to a decision-maker with an accurate basic data for decision-making. The decision-maker should also be able to interact with the network i.e. be able to give the network directions whether he or she wants supplementary information regarding a certain event.

The network should be designed so that false information is restrained; otherwise it is possible that the credibility of the network is damaged (compare to Aesop’s fable: “The shepherd’s boy and the wolf”). By restraining false information, less information has to be transmitted; thereby, less power is used by the sensors. It should also be designed so that an end user has to request rather than continuously receive information; however it should be possible to continuously receive information if needed.

2.3 UGS in Ad Hoc Networks

Sensor networks can be seen as puzzles where every sensor is a piece which contributes to the creation of a situation picture. The basic idea is to have a number of sensors cooperating and making a local decision both by themselves and with their adjacent sensors. Different types of sensors may be used to complement each other. For instance, an acoustic sensor hears something and an image sensor takes a snapshot.

UGS have a limited range of detection and identification. These limitations are generally due to background noise or weather and environmental changes. There is a need to deploy several UGS nodes in the vicinity to ensure continuous monitoring of detection. A network of nodes that uses multiple sensor technologies can accurately locate and identify target in the area. There are many different ways to communicate between the nodes but the most common way is by radio e.g. Bluetooth™ [1] or WLAN (IEEE 802.11b) [2]. A robust communication link is the key to a successful remote deployment of UGS.

By placing the sensors in a so-called ad hoc network many advantages will be achieved such as the possibility to a highly redundant network and possibility to add more sensors to the network if needed. A highly redundant network may take an advantage of synergism, however increases the complexity, cost, communication and thereby detection. Even many ad hoc benefits will be achieved such as rapid deployment, no pre-existing infrastructure is needed, and the dynamic topology of the ad hoc network i.e. if a sensor moves or a sensor failure the network adapts. Further the sensors may be distance deployed into enemy territory or even terrain where it is hard to get.

In a scenario where thousands of UGS are deployed from an airplane, the nodes will randomly land on the ground. Some of the nodes may not be able to reach each other. Thanks to ad hoc technology, sensors can interact with each other and construct, at deployment time, a wireless self-organized network. The network has the capacity to determine, on it is own, how to route the data to a randomly placed node. If the network changes e.g. a sensor fails, then the routing protocol finds a new way to its destination. If the coverage of the area is not good enough a new airborne deployment can be done and the system reconfigures its network.

The communication in the network can be divided into two parts. One internal part in which the UGS communicates, shares data and computational resources, and the second external part for humans to get access to the system, and get the situation picture.

Communication bandwidth and transmission power should be low to avoid the network for being discovered by enemies. Encryption and compression of data could also be used for increasing the security.

---

[1] http://www.bluetooth.org/
[2] http://standards.ieee.org/

2.4 Conceivable Usage Areas

There are many different conceivable usage areas for UGS. For example a police stake out, where a number of UGS could be placed out to send important information about what is going on in the monitored area and help out the police squad. UGS could even make a good home security system with the right configuration.

The largest usage area for UGS today is military. In different kinds of operations such as peacekeeping operation and border controls, where large areas are to be covered and people could risk infraction and violence, UGS can be very useful. UGS are suited to permanent, continuous monitoring. They can be deployed at important locations and sense movements, presence of vehicles, weapons, people etc. in its vicinity and signal an alert. When an alert occurs it is sent to the monitoring center where it is displayed and a patrol can be sent out to confront the intruders, try to stop them or document the infraction.

Figure 3. GIRAFFE S – Early Warning Surveillance Radar

Many works, e.g. [23], in the sensor network field describes the sensors as small and inexpensive, however a combination with larger and more expensive sensors is more likely needed. Imagine a radar station with surrounding ground sensor network. If an aircraft flies below radar coverage it may still be visible for e.g. acoustic sensors.

UGS provides maneuver commanders with continuous, near real time information on enemy movements. This improves the situation awareness and offers commanders a standoff strike capability. By recognizing the target sooner, units can maneuver and deploy large amount of resources more efficiently.

In the future we will see an extended use of UGS e.g. aircrafts deploying thousands of sensors over a hostile area and the sensors will work together in a large network and fusion its data to get a situation picture.

2.5 Different Types of Sensors

Generally there are two categories of sensors, active and passive sensors. Passive sensors use only available information for instance magnetism, whereas active sensors emit their own energy, which may reveal the sensor. The choice of sensors depends on the purpose of the task. A crisis situation for instance, such as a rescue mission for survivors after a natural disaster, does not necessarily need sophisticated sensors for fulfilling their task. For military use the sound of silence is more vital, therefore would passive sensors be the first choice.

There are many suitable sensors to use in sensor networks, which are listed below. [24]

Passive sensors:
acoustic: microphone, geophone, pressure, strain
optical: image sensors, PIR-sensors
field sensors: magnetic, field strength receivers, GPS, mm-wave
radiation detectors: neutron radiation, gamma radiation
chemical: gas sensors

Active sensors:
optical: laser radar, image sensor (illuminating), IR-sensor
microwave: radar
magnetic field sensor: magnetometers
multi point sensors: mechanical, optical

Different types of the sensors presented above may be used as complements to each other. For instance an optical sensor e.g. image sensor, is able to identify, classify and localize objects and their arming. A multi-sensor or a network of sensors can i.e. achieve the same objective. The network of sensors constitutes a multi sensor.

2.6 Data Fusion

The term data fusion means combining information from several sensors. Data fusion is a widely spread research area around the globe. The term data fusion means combining information from several sensors. Mainly there are two different methods of data fusion, central and distributed. Both fusion strategies aim to provide basic data for decision making to an end user. The distributed data fusion also aims to minimize transmissions within the network without loss of relevant information.

Data fusion is under development and many issues are unsolved, for instance fusion between different types of sensors with various characteristics, such as an image sensor and an acoustic sensor.

Another issue of great important is network queries. A query to the network does not necessarily involve the networks physical structure, for instance [26]:

Historical Queries:
Query historical data obtained from the sensor network. “Show the heavy traffic on road X for the past 10 days.”

Snapshot Queries:
Query the sensor network at a given point in time. “Provide IR image view of area Y.”

Future Queries:
Query the sensor network over a time interval. “For the next 12 hours, report any AUVs or divers that cross the perimeter Z.”

Almost every data network use logical addresses e.g. IP addresses. This is however not as important as the sensors geographical location. One solution is use so called spatial addressing e.g. using latitude, longitude and altitude. The spatial addressing provides the possibility to query a specific area about sensor information. Positioning system is an essential part of spatial addressing.

2.7 Existing UGS Networks

Today it exists various types of UGS around the world. The most systems are for military use. An example of such a system is HALO, developed in the UK, which is used to monitor artillery fire. A few acoustic and met sensors are deployed for long-range detection of a transient signal emanating from artillery fire. Bearing information is extracted from the various UGS and transmitted to estimate source location. [23]

2.8 Future UGS Networks

There are many unsolved issues regarding future UGS networks. CPU capacity and battery capacity for instance have to be improved in order to manage more complicated data fusion and wireless communication.

Security in a sensor network can be divided into two main categories, transmission and data management. A network should not be susceptible to countermeasures e.g. jamming and it should survive an electronic attack.

Management of data can be divided into three subcategories, integrity, authentication and availability.

Integrity implies that data has to be assured, no one should be able to modify or erase data that is being transferred. If data is altered while transmitted and accepted by the end user as correct information, this could result in devastating consequences. This could lead to an unusable and unreliable network in the long run.

It should not be possible to receive information or manage the network if the user has not been authenticated, that is confirmed his access level. The worst-case scenario is a hi-jacked sensor network that is being used against its owner.

It should not be possible to annihilate the network on TCP level, compare the Internet where many web servers have been subjects for attacks that have lead to malfunctioning servers.

To be able to determine the location of the sensors in the network some kind of positioning system is needed. In accordance to a presidential decision in May 2000 the Selective Availability (SA) in the GPS system was removed, which provides more accurate positioning. The SA may however be turned on, which could result in inaccurate positioning in the sensor network. For a live sensor network the European Global Navigation Satellite System (GNSS), informally known as Galileo[3], is to be considered because the SA functionality is not implemented. The Galileo system will be fully operational in 2008.

Other possibilities for positioning are to use existing infrastructure for mobile communications such as Global System for Mobile communication (GSM) and Universal Mobile Telecommunications Service (UMTS). These systems do not however provide the same accuracy as the positioning systems described above.

---

[3] http://europa.eu.int/comm/energy_transport/en/gal_en.html

2.9 Ad Hoc Networking

An ad hoc network, with mobile nodes that are likely to move relative to each other, are often referred to as Mobile Ad hoc Network (MANET). MANET communication is in a way similar to the humans' approach to relay information to each other. If you were standing in a crowded place, and would like to say something to a friend standing far away, you don't need to raise your voice unless necessary. You notice the people you have around you, and may ask another person to relay your message. The message will now walk its way through the crowd and finally reach your friend. In the same moment, the crowd has suddenly become a human variant of an ad hoc network.

The abbreviation of MANET is also associated with the IETF MANET Working Group (IETF MANET WG), which is a working group dedicated to MANET research in deployment of open standards for ad hoc routing protocols. Their work is further described in chapter 2.9.6.

In a regular hardwired network of today data packets are forwarded between source and destination by routers with relative settled rules. The network consists of mostly fixed infrastructure. Each node and router in such a network has to be manually configured to its present address and location. All communication depends on centralized administrative infrastructure.

On the other hand a MANET consists of more or less mobile devices / platforms. These, referred to as “nodes”, are functioning as routers, one or more hosts and wireless communication units of receivers and transmitters. That is, the routing functionality is incorporated into the mobile nodes. The network does not rely on any pre-existing infrastructure. The topology of the network may change rapidly and unpredictable. In Latin, ad hoc literally means "for this", and is often applied in terms of networks were devices may be quickly added and integrated using wireless technology. One advantage of MANET among many is their ability to be rapidly deployed. In addition to that, the users are relieved of a lot administrative job, setting up those “spontaneous” networks. Nodes can dynamically, without warning, join and leave the network without possibly harming other nodes communication. Each node may as well move physically relative to each other and still remain connected.

2.9.1 MANET Characteristics

A MANET differs from a wired static network in many ways. There are some key challenges related to an ad hoc environment, and some specific characteristics. In this section, we will try to identify some of these [4], [5].

Dynamic topologies: Nodes within network may move arbitrary and randomly. The routing between nodes, often consisting of multi-hop, therefore may change rapidly and unpredictably. Due to the nodes' movement, their routing information will be updated more often than its hardwired counterpart. This results in more overhead information, which in turn increases use of the radio medium resources. This puts great emphasize to the chosen routing protocol.

Multi-hop: Because the communication between nodes consists of wireless transmission where nodes often do not have direct radio contact with each other, connectivity between nodes is not guaranteed. Data packets often have to be forwarded between multiple middle nodes on their way to the destination. In a MANET, there is a higher risk of topological changes meanwhile data packets are routed, compared to its hardwired counterpart. Routes can be out-of-date (i.e. old and unusable), while a packet is routed with the current route.

Bandwidth-constrained, Variable link capacity: Compared to hardwired networks, communication over wireless medium will often have less bandwidth capacity. At the same time, signal interference, noise and fading, will most likely be higher, resulting in even less useful bandwidth. Different mobile nodes may use different network interfaces with different power capacity, with the result of nodes hearing other nodes, without ability to answer bi-directional.

No pre-configuration or planning: Because a MANET is to be set up anywhere, at anytime and with any participants, the network configuration information that can be pre-configured is limited. Each node does not know anything about other nodes services, addresses or other capabilities prior connection. This affects security, which usually relies on availability of key management infrastructure [3]. In conventional networks, logical and administrative boundaries suggest where services should be hosted and replicated for highest efficiency. This is not the case in infrastructure-less networks. Security is one such area. Firewalls are usually used to create security boundaries. In an ad hoc network, it is a complex task for a node to create a network of trusted partners.

2.9.2 MANET Applications

This thesis main part deals with UGS using ad hoc technology. In this chapter, we will try to give the reader a wider perspective of MANET and its different capabilities. There are numerous of different applications, both civilian and military, where the ad hoc technology can be used with great benefit. Those include spontaneous networks at meetings and conferences, substitute or support of dynamic communication for emergency/rescue operations and disaster relief efforts.

One scenario of using ad hoc communication with multi-hop is a user of a mobile device, out of reach from an access point. This could be for instance in an office where the user has lost connection with the nearest WLAN access point, or a user of a cellular phone who is too far away from a base station. These problems could be solved if there were other intermediate users of the same system between the end user and the access point/ base station. With ad hoc technology implemented, the “out of reach-users” could get connection through multi-hop of their data through one or more “middle nodes”. The middle nodes could be just other users of the same system, mobile as well as hardwired. This would solve tremendously many situations of today when you cannot use your cellular phone due to bad coverage.

Of course, the technical implementation would be one thing, but there would also be other tough questions to be answered. One would be “how are those middle node users, whose devices are used for relaying data, compensated due to the use of their own resources like the battery?”

2.9.3 Network Reference Model OSI

The Network Reference Model reviewed here is to provide background information and theory relevant to the ad hoc networking.

In the 1980's ISO developed the seven layered reference model. The seven layers describe different stages that information has to pass on its way from one computer to another. In each layer the data packets is prepared for the next layer. The IEEE uses the OSI framework as a guideline when establishing network protocol standards. In many network protocols, one or more of the OSI layers are combined to one layer.

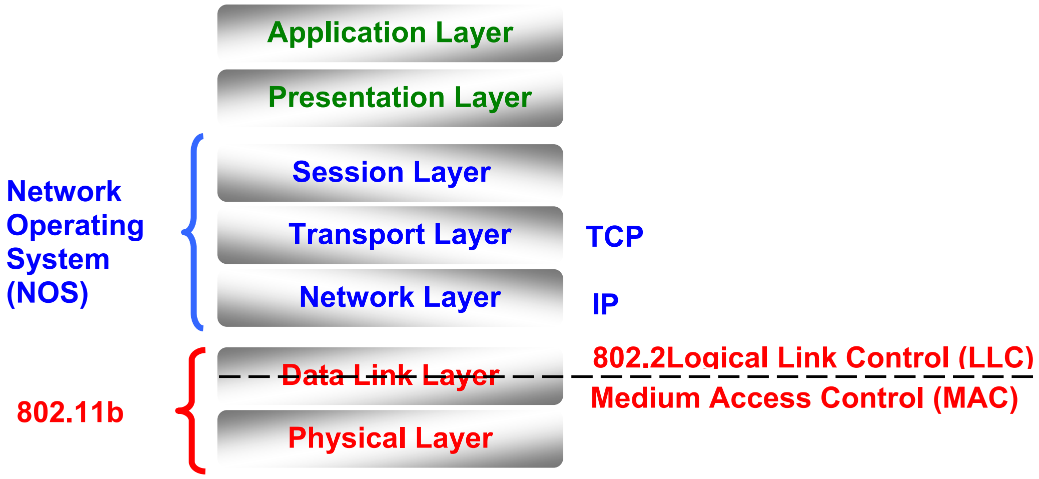


Figure 4. The OSI Reference model.

The seven layers are

- **Layer 1 - Physical**: The lowest layer of the model defines electrical and mechanical characteristics such as the connections, voltage and timing for the medium transmission.
- **Layer 2 – Data Link:** This layer controls the transmission of blocks of data over a physical link. The layer assigns the appropriate physical protocol and defines the type of network. It consists of the two sub layers *Logical Link Control* (LLC) and *Medium Access Control* (MAC). The task of the MAC layer is to communicate with the network interface card, while LLC works as an interface to the network layer.
- **Layer 3 - Network:** The network layer routes data and handles logical protocols and addressing such as IP.
- **Layer 4 - Transport:** This layer ensures through error checking and flow control that data from the source arrives at the destination correctly and in proper sequence.

- **Layer 5 - Session:** The session layer handles the communication with the receiving device through communication establishing, maintaining and ending.

- **Layer 6 - Presentation**: This provides services that interpret the meaning of the information exchanged with the application layer.

- **Layer 7 - Application:** This layer directly serves the end user, and interacts with the operating system or application.

2.9.4 Existing Commercial Products with Ad Hoc Technology

The field of wireless ad hoc commercial products is still in its inception. The last decade, the number of wireless products has literally exploded on the market. But only a few are based on ad hoc technology. Those technologies described below both operate in the ISM band of 2.402 to 2.480 GHz. We are not aiming to describe these very deeply in this thesis, but rather mentioning them and describe their basic concepts.

2.9.4.1 Bluetooth™

Bluetooth™ could be called “a cable-replacement technology” [8]. Its main area of usage and capability today would probably be to replace cables to devices like keyboards, mouse, printers or cameras, cellular phones and other things with need of short-range communications. This functionality is accomplished through cheap radio chips, which are integrated in the products, communicating with each other according to the Bluetooth specifications,[9] beside its hardware standard. Bit rate of current Bluetooth version is 720 Kbps user transfer rate with approximately 10 meters distance. Both these values are supposed to increase in forthcoming version Bluetooth v.2.0; with between 2 to 10 Mbps bit rate and distances up to 100 meter. The Bluetooth protocol stack makes the ad hoc functionality possible, hierarchical however. Bluetooth devices can form “piconets “with up to 8 members, including one Master. The current version of Bluetooth does not yet specify the formation of “scatternets” which consist of several piconets linked together. A piconet only uses single hop communication.

2.9.4.2 IEEE 802.11b

802.11b is a communication standard usually used for WLAN. The Bit Rate capacity holds between 1 to 11 Mbps and distances of approximately 15 to 500 meters, depending on the surroundings. In the 802.11 standard, only the two bottom layers of the OSI reference model are defined, see figure 4. These are the physical layer and the Data Link Layer (Medium Access Control, MAC sub-layer). This, in conjunction with the ad hoc mode (see 2.9.4.4), opens up the possibility to route data packets above the MAC layer, from and to higher layers, which provides the capability to realize a multi-hop network. In Part two of the Thesis, when developing a testbed, we make use of this to achieve multi-hop ad hoc routing. The signaling mechanism of the 802.11b MAC-protocol to transmit between two nodes relies on request-to-send (RTS), clear-to-send (CTS) and data-packet-acknowledges (DATA-ACK) to avoid transmission collision in the medium. This procedure makes the 802.11b protocol not supporting unidirectional links between nodes. Some modifications would be required on the 802.11b MAC-layer to be able to transmit outgoing packets for unidirectional links without the RTS/CTS procedure.

The 802.11 standard supports WLAN equipped devices to operate in two different modes: *Infrastructure mode* and *ad hoc mode*.

2.9.4.3 Infrastructure Mode

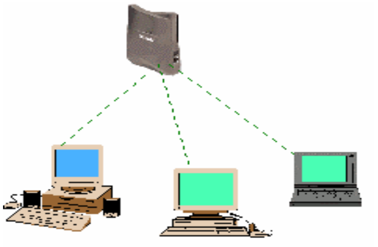


Figure 5. Illustration of infrastructure mode of three computers and an access point.

Each wireless station that is operating in this mode has connection to an access point, through which all communication passes. The access point may route or bridge the traffic to an Ethernet network or the Internet. This is the most common mode and usually the default.

2.9.4.4 Ad Hoc Mode

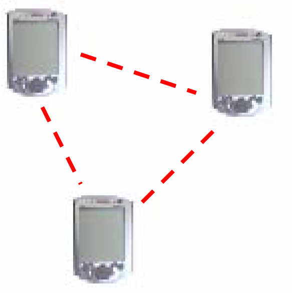


Figure 6. Illustration of ad hoc mode of three handheld computers.

In this mode, the wireless stations make connectivity directly, i.e. bi-directional, assumed that they are within range of each other. There is no need of access points.

2.9.5 Ad Hoc Routing Characteristics

The concept of routing can be described as the process of finding a path, for the data packets, to traverse the network from the source to the destination.

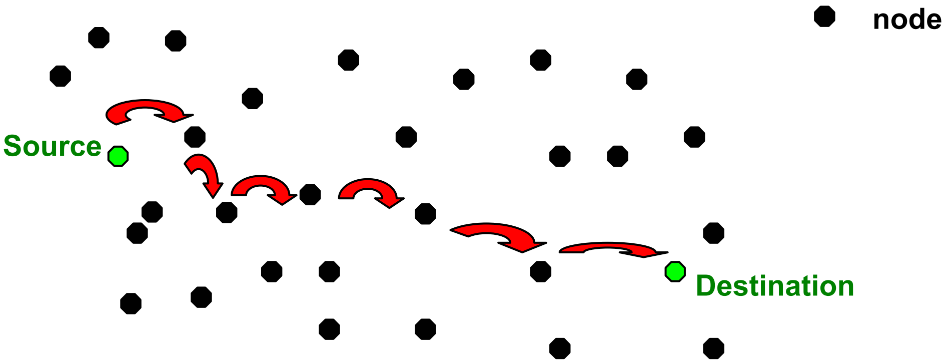


Figure 7. Illustration of the multi-hop ad hoc routing concept.

As pointed out in chapter 2.9.1 an ad hoc network has some very specific characteristics compared to the hard-wired counterpart. The available routing algorithms are designed to be optimal under different circumstances. During the development, the originators have focused on different abilities for the network, which may often be in conflict with each other. Some constraints could be:

Battery life vs. network topology update frequency

Bandwidth vs. routing overhead data

Delay vs. network capacity usage.

In this chapter, we will look at some types of classification and differential features for ad hoc routing: *proactive*, *reactive*, *hybrid, clustering and symmetric / asymmetric routing.*

2.9.5.1 Proactive Routing Algorithms

Proactive routing also referred to as “table driven” routing, acts a bit similar to the traditional fixed networks routing. Each node keeps an updated route table over the entire network. This is done by advertising the route table periodically over the entire network. Proactive routing is fast in finding a path to a destination because the nodes just have to get the route from its route table. However, this method is not very resource saving in terms of batteries, processor usage, power and bandwidth.

2.9.5.2 Reactive Routing Algorithms

Reactive routing is also referred to as “on-demand” routing. That is because such algorithms do not have any knowledge of the network topology before it needs to. The node obtains a suitable route, when a route is needed. Hence, this methodology makes the overhead data much less, which saves battery and other resources. The radio transmission will also be kept low, which is desirable considering ElectroMagnetic Compatibility (EMC) in civilian applications, and discovery avoidance and electronic surveillance in military use. Delays, due to the route discovery mechanism are larger in reactive algorithms, than the proactive ones.

2.9.5.3 Hybrid Routing Algorithms

A hybrid proactive/ reactive routing algorithm makes use of both the techniques above. This could be for example, proactive activity within a limited range from each node, while reactive route discovery over wider range is restricted to a few numbers of nodes over the global network. That is, only a few (one or more), nodes in each “proactive zone” need to participate in the global route discoveries. Those nodes already know the network state of their neighbors. More resource is devoted to the topology update of nearby, and more frequently used, parts of the network. This reduces the cost of resources, and gives more scalability to the network, but brings more complexity to the algorithm.

2.9.5.4 Clustering

In larger ad hoc networks it can be beneficial to let the network dynamically divide its nodes into clusters. This can be made in an attempt for e.g. more effective transmission management, backbone formation or higher routing efficiency. Clustering can be beneficial for networks in applications using distributed data fusion as described in chapter 2.6. The common goal of these is to achieve some kind of *control structure* for the network to accommodate to property changes in the network.

To achieve better transmission management with less risk of interference, clusters of nodes could be coordinated to separate transmission in time, frequency, space or spreading code.

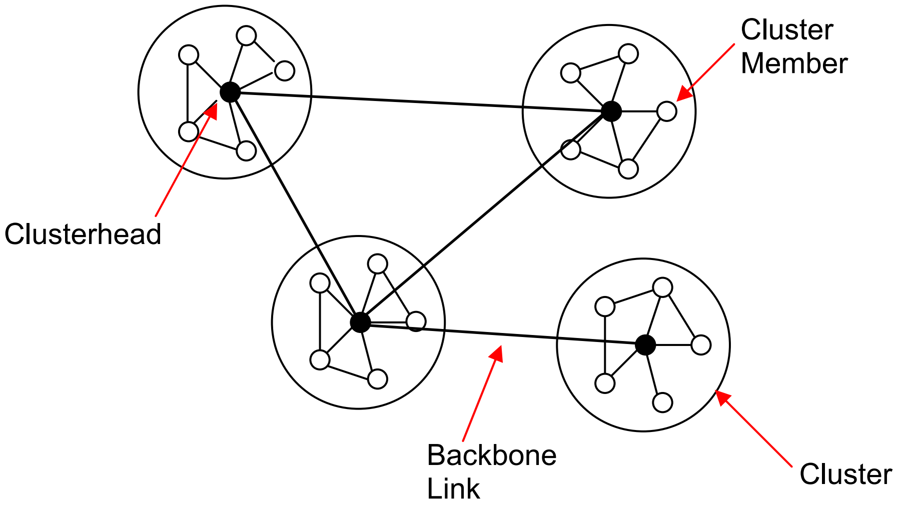


Figure 7. Backbone Cluster Network formation.

The cluster technology could also be used to make the ad hoc routing more efficient. Dynamically backbone formations could be used to reduce the number of hops, and furthermore the end-to-end delays of network routing. Each cluster would use one selected node (*clusterhead*), optionally with increased transmission power, to relay data to other clusters. This could reduce the number of hops in long distance communication in the ad hoc network.

In the ad hoc network, events that change the network state, as the structure or the loading of the network, would more likely affect nodes in close vicinity of the direct affected node, than nodes on the network's distance parts. Unconditional widespread dissemination of information about each change in the network state may be wasteful and ineffective. Local changes within a cluster would primarily be advertised to the cluster members. Each cluster could use a dynamically chosen *clusterhead* to administrate the routing and use of resources and *gateway* nodes to connect with other clusters.

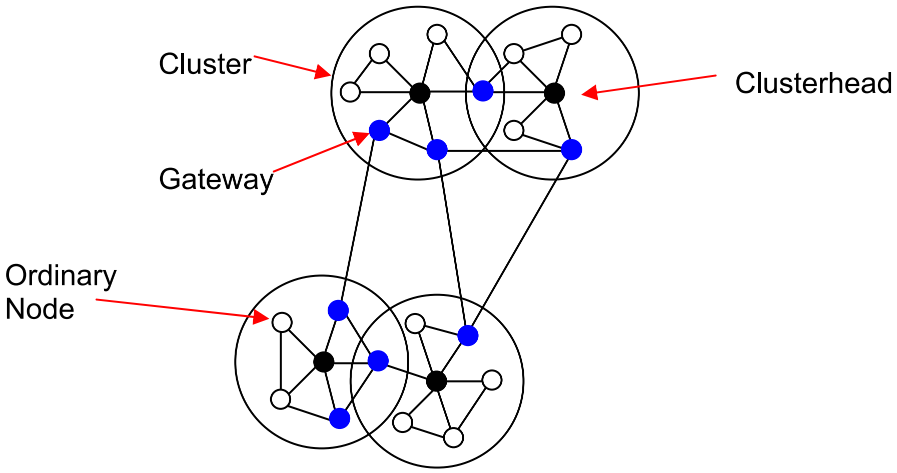


Figure 8. Cluster Network with Clusterheads and Gateways.

2.9.5.5 Symmetric / Asymmetric Routing

This characteristic is independent whether using reactive or proactive routing. Symmetric routing means bi-directional link communication. A route between the source and destination is the same in both directions, meaning that data exchange between two neighbor nodes works equally in both directions. Routing protocols with support of uni-directional links offers the possibility of asymmetric routing. This can be beneficial when connectivity between two nodes differs, e.g. due to differing antenna, sources of interface or transmission power.

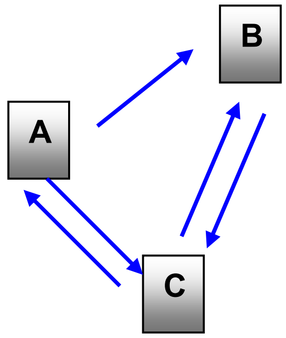


Figure 9. Uni-directional link between the nodes A and B.

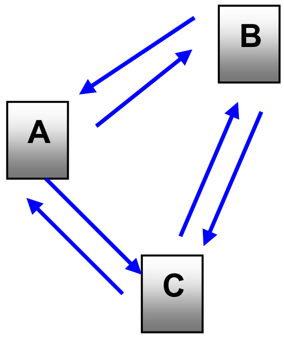


Figure 10. Bi-directional link between all the nodes.

In real applications it is highly likely that there are some nodes in the network whose radio transceivers have higher power capacity than the other nodes. These could for instance be devices mounted on people or vehicles. Those could be used for transmissions over larger areas of the network and to more distant nodes. If the receiving nodes do not have the same transmit power, they will not be able to send packets back directly to the source. With asymmetric routing this would not be a problem. In an application of UGS there could be some high-powered nodes implemented in the network, which would be able to transmit the fused data of several sensor nodes to some kind of situation picture node.

2.9.6 Routing Protocol Overview

In this chapter, we will describe a few of the existing routing algorithms. We will concentrate on some of the protocols that the IETF MANET Working Group (IETF MANET WG) is currently working with, because those are likely to be part of the MANET research of tomorrow, in some way or another.

IETF MANET WG is dedicated to MANET research in deployment of open standards for ad hoc routing protocols. Their official charter states it, as “The primary focus of the working group is to develop and evolve MANET routing specification and introduce them to the Internet Standards track. The goal is to support networks scaling up to hundreds of routers. If this proves successful, future work may include development of other protocols to support additional routing functionality.” [14]. Their objective is to “standardize an interdomain unicast routing protocol which provides one or more modes of operation, each mode specialized for efficient operation in a given mobile networking” [14].

One routing protocol will probably not be able solve all desired functionality. Forthcoming lager ad hoc networks will most likely consist of clusters with different types of routing algorithms and capabilities, between and within the clusters.

In this thesis, focus will be on the DSR protocol, because this was our chosen algorithm when we developed our testbed, as described in part 2. In part 1 we point out specific important characteristic for ad hoc with UGS, and in the testbed part we will justify our choice of the DSR protocol.

2.9.6.1 Comments on the Past of Ad Hoc Routing Protocols

The earlier developed routing algorithms for ad hoc networking was often based on the Distributed Bellman-Ford (DBF), which is a traditional distance-vector (DV) algorithm. The term DV comes from that the route table entry for a destination contains, among other things, the number of hops to the destination (the *distance*), and the next hop (or vector) towards the destination. DBF works, according to Charles E. Perkins [16], as follows “In distance-vector algorithms, every node *i* maintains, for each destination *x*, a set of distances $\{d_{ij}(x)\}$ for each node *j* that is a neighbour of *i*. Node *i* treats neighbour *k* as a next hop for a packet destined for *x* if $d_{ik}(x)$ equals $min_j \{d_{ij}(x)\}$. The succession of next hops chosen in this manner leads to *x* along the shortest path. To keep the distance estimates up to date, each node monitors the cost of its outgoing links and periodically broadcasts, to all of its neighbors, its current estimate of the shortest distance to every other node in the network.”
The main problems of this algorithm are convergence and excessive control traffic [5].

2.9.6.2 Destination-Sequenced Distance Vector (DSDV)

One protocol based on DBF is the DSDV protocol [16]. DSDV is as well as DBF a proactive protocol, but with the divergence of enhanced qualities of convergence speed, reduced routing overhead and elimination of route looping. Each node's route table contains a next-hop destination to every node in the network, distance to the nodes and a sequence number of each route ("the age of the route"). The network topology is updated periodically through broadcasting, as well as event-driven incremental updating when changes occur in the network. The route table is updated if an incoming route has larger sequence number (i.e. newer route) or equal sequence number with shorter distance.

2.9.6.3 Ad Hoc On-Demand Distance-Vector (AODV)

AODV is a reactive routing algorithm. Its primary goal was to reduce the routing overhead in the network as much as possible, and was based on the earlier DSDV protocol. The nodes in an AODV network do only maintain routes to destinations, which the nodes need to keep connectivity. When a source does not have a route to a given destination, it makes a route discovery by broadcasting a *route request*. The *route request* cross over the network until it reaches the network boundary or its Time-To-Live (TTL) value has reached zero. The first *route request* is set with a short TTL, which will be increased in new route requests if the destination is not found at first. This is made, to try avoiding unnecessary broadcasting in the network. For each node a *route request* traverse, the node will add a *reverse route* entry with the source address and next-hop towards the source in its route table entry. That can be used to forward a *route reply* later if one is received. When the destination or a node with a valid route to the destination receives a *route request*, it will send back a *route reply* to the source by just using the route of the *route request*. When the *route reply* reaches back to the source, the route will be cached in the source table entry. Each sent *route request* from a source also contains an ID to prevent nodes from handling already seen or old *route requests*. Since a node gets the route for a *route reply* by turning the *route request*s route, AODV does not utilize asymmetric routing. When a route fails, the source is notified, and a new *route request* can be initialized if necessary.

2.9.6.4 Dynamic Source Routing (DSR)

The sought of the originators of DSR was to create a routing protocol with very low overhead, yet able to react quickly to topological changes in the network and providing highly reactive service to help ensure successful delivery of data packets [17]. The DSR algorithm has its greatest benefit under the consumption that the number of hops for a data packet to travel is usually relative low (perhaps 5 or 10 hops) [6 (p.140:5.1)] and the speed of nodes movement is moderate with respect to the packet transmission latency and OSI Layer1 transmission range.

Route Discovery

When a source need a route to a destination, which is not it its *route cache*, the node initiate a route request for the destination. The *route requests* broadcasts locally. Each *route request* contains a Time-To-Live (TTL) and a unique ID. Each node within range of this broadcast handles it. If the node has seen the *route request* before (identified the packet-ID and source address), it silently discards it. Otherwise, if the node is not the destination and does not know a route to it, the same *route request* will be re-broadcasted, after the node has added its own address in a *source route* record of the packet.

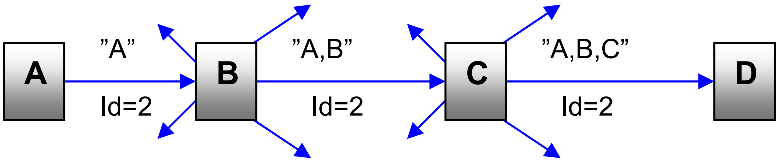


Figure 11. Example of Route Discovery Node A as the source and initiator of a route request, and node D as the target.

Else, if this node would be the destination, or has a route to the destination cached in its route cache, it sends a *route reply* back to the source node. The node can either use an own-cached route back to the source, or use the reversed hops of the *route request*'s *source route*. This implies that DSR supports asymmetric routing. If the MAC layer requires bi-directionality for unidirectional transmissions, the *route reply* must use the reversed *route request* hop sequence. The route cache table of each node contains all hops to the destination for each route, and not just the next-hop as in the AODV protocol.

Route Maintenance

Each sent data packet in DSR requires some sort of confirmation of receipt. This acknowledgement can be done either by MAC-acknowledgement if supported (i.e. link level ack.), passive ack. through *promiscuous mode* (see chapter “Additional Features” below) or by specific DSR-ack.-packets sent by the recipient. If a node is unable to deliver a packet with a given route, it sends back a *route error* to the source, which removes the specific route from its *route cache.* The undelivered data packet is lost, and it is left to upper-layer protocols such as TCP to retransmit the data. Every node receiving packets such as *route request*, *route reply* or *route error* can use the overhead information to update its own *route cache*.

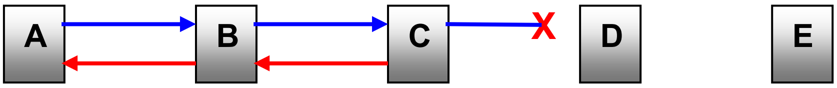


Figure 12. Example of Route Error Node C is unable to forward a packet from source A over its link to the next hop D, and sends a route error back to node A.

Additional Features

If setting the network interface into promiscuous mode, a node can listen to all data packets within reach. In this way, a node can catch and examine all DSR specific packets, even if the target address of the packets is neither broadcast nor its own IP. The network can save a lot of overhead packets by using such information. A node can get a *passive ack* of a transmitted packet by overhearing the next node forward the same packet. Another feature of promiscuous mode is *route shortening*. When a node is overhearing a packet with a source route where this node is used later in the route, but not the next hop, it can send back a *gratuitous route reply* to the original sender of the packet, telling it there is a shorter route to the destination.

A node that is receiving a *route error* can, in addition to update its route cache, also resend the *route error* to its neighbors with the next *route request*. By caching recently *route error*s, the node can also assure that an incoming *route reply* does not deliver an old previously broken link.

2.9.6.5 Zone Routing Protocol (ZRP)

ZRP is a hybrid of reactive/proactive routing protocol. The main features of ZRP are the capability of routing with a large number of nodes, fast nodal mobility, frequent topological changes and adaptive behavior, based on the mobility and calling patterns of the mobile users. Each node proactively advertises its link-state to neighboring nodes in its *routing Zone*, which are nodes within a defined maximum number of hops from the node itself. A node only needs to make a route inquiry when the target node does not lie within the routing zone. When a route inquiry is made, the node uses a service called *bordercasting,* multicasting the inquiry to all nodes of its own zone. If a receiving node of such inquiry does not either has the target node in its routing zone, it forwards the route inquiry to *its* zone's nodes. If a node has the target node in its own zone, a *route reply* will be sent back to the source. This way, a route inquiry does only need to reach a node with the target node within its routing zone. Used multi-hop routes within a zone can be maintained and enhanced by re-routing of failed links or route shortening.

3 Testbed Development

The main motivation was to build a flexible UGS network for testing and developing fundamental Network Centric Warfare (NCW) building blocks, using a combination of commercial ground sensors. Our vision building UGS in an ad hoc network was to show that it would work in practice.

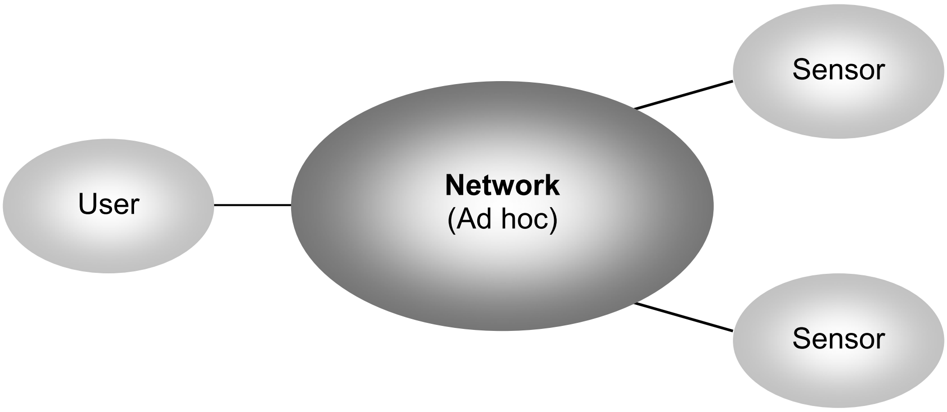


Figure 13. Testbed layout.

The picture 13 shows schematically the UGS network layout, where a number of sensors and users can be connected, depending on the needs.

3.1 Scenario

The testbed scenario takes place in an UGS network monitored crossroads which prior task is to give an on-the-spot report about vehicles heading for an object of special interest.

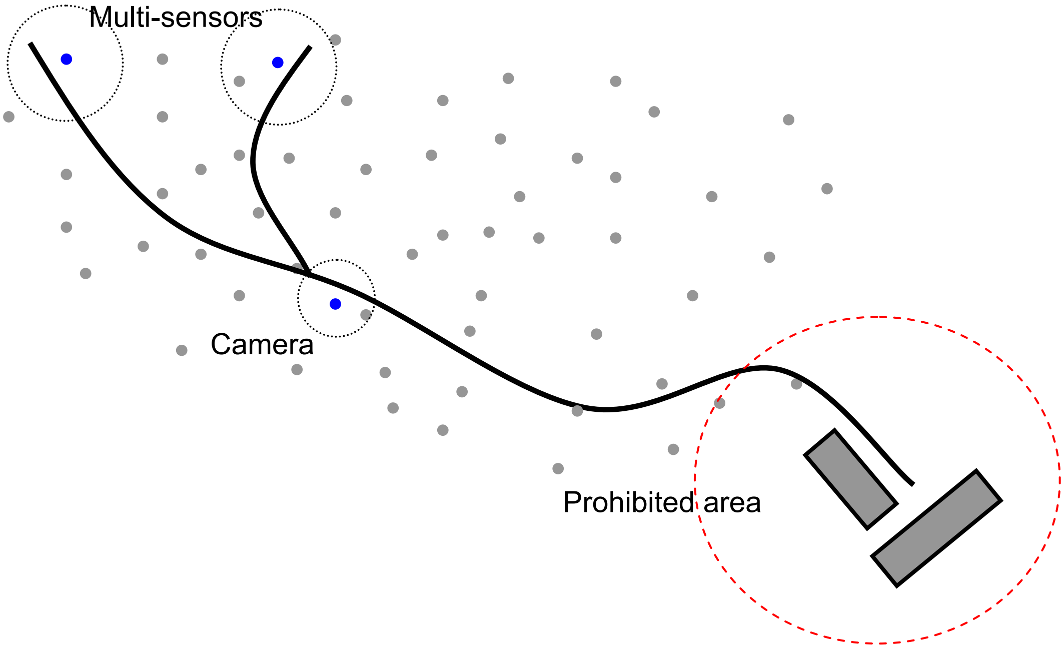


Figure 14. Crossroads – The testbed scenario.

Above is a schematic picture showing a hypothetical scenario where the small dots represent ground sensors and relay nodes. The network will detect movements in the specific area and report this so that actions can be taken.

3.2 Used Products

In order to develop the testbed many things had to be investigated, such as type of platform, operating system, communication interface, sensors and how the integration of the parts should be performed. In this chapter we will describe the used parts in the UGS network and the partners we cooperated with.

3.2.1 IPAQ

Figure 15. Compaq iPAQ.

In the choice of platform for the nodes we wanted to have a small handheld device with great battery power and good communication possibilities. The iPAQ proved to meet our requirements even though the battery and Central Processing Unit (CPU) power could have been better, however; it was the best Personal Digital Assistant (PDA) available on the market at the time of our work. Another issue of decisive importance for our choice was the possibility to use Linux as Operating System (OS).

The iPAQ is a handheld device developed by Compaq [4]. The model we used was HP3760 [5]. The iPAQ is equipped with a 32-bit 206MHz Intel StrongARM RISC processor to fulfill our needs, along with the CPU, 32MB ROM for storing data. The device is also operational with a color display, which functions as a touchpad to enable interacting with the computer.

On the iPAQ we are using Familiar v0.5.2 [6] that is a Linux [7] based operating system (see 3.5.4). Familiar is a free OS using Itsy Package Management System (IPKG), which allows dynamic installation and removal of packages on a running system. Thanks to this we choose to use Familiar. To be able to execute JAVA software on the iPAQ we installed Blackdown [8] JRE v1.3.1 package.

---

[4] http://www.compaq.com/
[5] http://www.compaq.com/products/handhelds/pocketpc/h3760.html
[6] http://familiar.handhelds.org/releases/v0.5.2/
[7] http://www.linux.org/
[8] http://www.blackdown.com/

3.2.2 WLAN PC-card

Using AVAYA's [9] waveLAN PC-card, former LUCENT ORINOCO waveLAN PC-card, for the 802.11b communication standard from Lucent ORINOCO [10] solved the communication between the iPAQ's and the communication to the UMRA (see 3.5.2). AVAYA is a cheap and reliable hardware.

Figure 16. ORINOCO waveLAN PC-card.

3.2.3 GPS receiver

In order to find the position where the nodes are located we used a Global Positioning System (GPS), Marco Polo PC-card, manufactured by Peak [11].

3.2.4 Sensors

In our testbed we choose two different types of sensors. One mutli-sensor to detect and classify vehicles, and one smart camera to take snap-shots of the detected vehicles.

3.2.4.1 UMRA Sensor

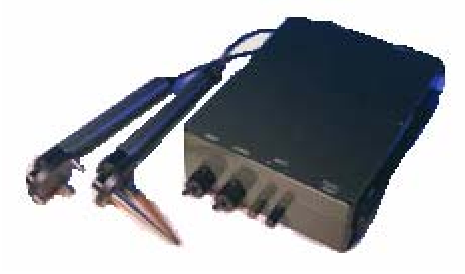


Figure 17. UMRA multi-sensor.

The UMRA (Intelligence Multisensor RAdio) is a sophisticated ground multi-sensor developed by Exensor Technology AB [12]. The sensor identifies objects passing through using a signature database. It can identify everything from soldiers to helicopters.

---

[9] http://www.avaya.com/
[10] http://www.orinocowireless.com/
[11] http://www.peak-uk.com/
[12] http://www.exensor.com/

The UMRA is working in two zones, close range and distant range. In close zone, information about the objects identity, passage distance, speed and direction can be obtained. In the distant zone only acoustic sensor data is employed, which permits identifying.

The sensor consists of a field-computer and two sensor probes attached to the computer. Each probe contains one acoustic, one seismic and one magnetic sensor. The UMRA should be placed along a road and information from the probes is collected when an object passes by. When information is received it is analyzed and compared to the sensors existing signature database. When an object is identified a report is presented.

In order to integrate the UMRA in the UGS ad hoc network, Exensor provided their help. Together with Exensor we developed a communication protocol to be able to give the sensor commands and receive information about alerts. This protocol is built on a technique similar to XML. The communication between the UMRA node and the UMRA is based on WLAN, and peer-to-peer connection.

3.2.4.2 Camera Sensor

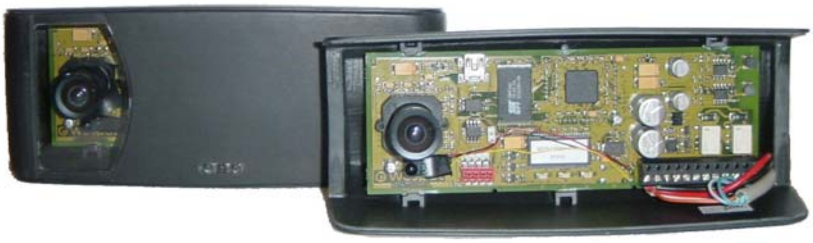


Figure 18. Camera sensor.

The second sensor is a smart camera provided by WeSpot AB [13]. The smart camera comprises optics, image sensor, a powerful ASIC-processor and a communication unit.

This sensor can work as a smart door opener. A detection zone is initiated, where the door sensor will sense movement and presence of a person. The sensor will in an intelligent way decide if the door will be opened or not. This will save energy and increase the security.

The communication with the camera is handled by a JAVA interface implemented in the camera. The camera communicates through an RS232 cord.

---

[13] http://www.wespot.com/

3.3 Decision Support Node (DSN)

The Decision Support Node (DSN) is the user interface application with control and management possibilities. The application consists of a map over a given geographical area where connected sensors will appear (with the help of GPS receivers on the sensors). By touching a sensor on the touch screen specific sensor information will appear. This information consists of the sensors battery status, position (longitude and latitude), possible sensor error codes and mode status (online or offline).

Since a network in general needs to know IP addresses for TCP/IP communication the DSN has to initiate communication towards the sensors using a sensor inquiry (see 3.3.1).

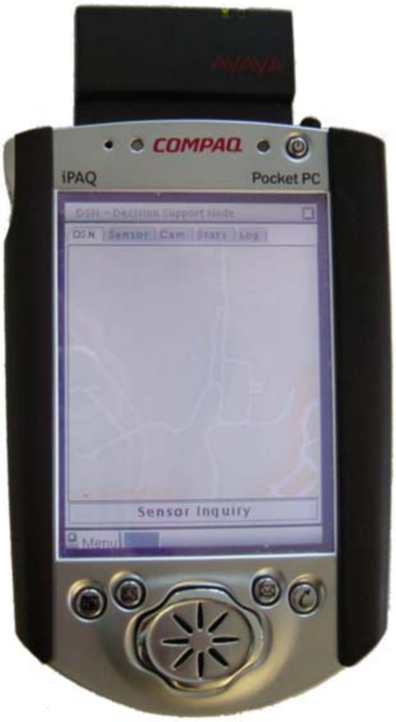


Figure 19. Decision Support Node.

One could resemble the applications in the sensor network and the routing protocol as mail and postal services where the applications act as mail. If a mail is to be sent, the application fills in the address to the receiver and posts it. The routing protocol then delivers the mail to the receiver.

When the user has affiliated himself to the network all sensors will report alerts to him when or if they occur. The DSN handles only alerts from two different sensors, UMRA from Exensor and an intelligent camera from WeSpot. The number of sensors that can be handled are almost unlimited, however only two UMRA and two cameras have been used in the testbed.

When a sensor alert from an UMRA arrives to the DSN a tab starts blinking to get the users attention. The information provided by the UMRA is which kind of vehicle that caused the alert, when it happened, the direction and velocity of the vehicle and identification probability. The user may also see the sensors current battery status, position and possible error codes.

Figure 20 below shows an alert from an UMRA. This tab also provides the possibility for the user to interact with the sensors. Synchronization of time can be made if there's any sensors connected. The sensors can also be turned on or off if needed.

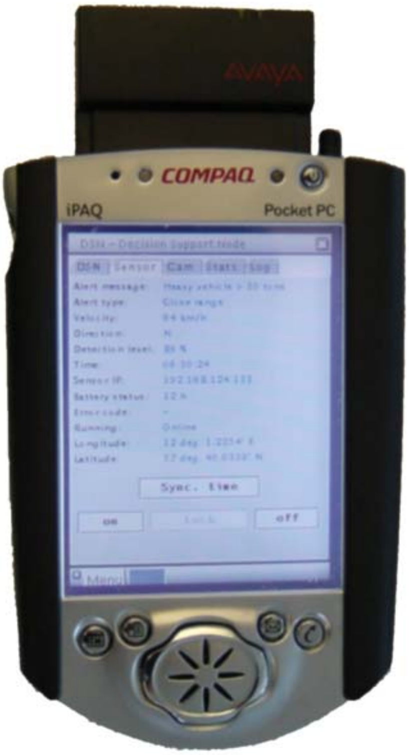


Figure 20. DSN showing sensor alert message.

When an alert has been reported from the UMRA the camera is trigged to take a snapshot. This image will be sent to the DSN, which treats the image as an alert and notifies the user that an alert has occurred. The user has the possibility to manually take snapshots from the connected cameras.

For evaluating purpose some statistics features have been implemented which shows how many sensors that are connected, their type, how many messages that have been sent and received. There is also a possibility to view the actual routing information from the routing protocol.

3.3.1 Topological Sensor Structure Inquiry (TSSI)

One of the first challenges we stood in front of was the limitation in the TCP protocol, which is that you have to know the IP address to whom you are about to communicate with.

The solution was to create a Topological Sensor Structure Inquiry (TSSI). The TSSI is a message, which is being broadcasted over the entire sensor network, containing the senders' IP address and port. The message is sent as an eXtensible Markup Language (XML) like message using User Datagram Protocol (UDP) for communication. Every sensor in the network that receives this message responds automatically with a TCP communication towards the specified IP address and port, which enables further communication between the DSN and the sensors.

The implementation of TSSI in the DSN was made so that a user must manually send it (by pressing a Sensor Inquiry button).

3.3.2 Sensor Communication

To demonstrate control and management of the network we implemented the possibility to order the sensors to synchronize their time. The time that is used for synchronization is received from the GPS receivers mounted on the sensors. This may be of importance if a sensor reports an alert. There's also a possibility for the user to demand an image from connected cameras.

3.3.3 Robustness

When we developed the applications in testbed we had to consider that the communication might go down for while. To prevent any loss of information during this time we implemented resend- and reconnect-functionality in both sensor and DSN nodes. Before a connection is treated as lost every message has to be resent at least ten times.

3.4 Internal Communication within the Sensor Nodes

The implementation of the sensors in the ad hoc network resulted in two different types of sensor nodes. The first type, an UMRA link connected to the UMRA and the second type, a camera link connected to the camera. To get the position from the sensors Global Positioning System (GPS) software was developed.

3.4.1 Global Positioning System (GPS)

To determine the geographical position of the sensors we had to integrate Global Positioning System (GPS) receivers. The GPS software provides the possibility for a sensor, or in our case the sensor gateway, to pinpoint its location and determine the time. The time received by the GPS is in the form Coordinated Universal Time (abbreviated as UTC), which does not consider daylight time. The developed GPS software adjusts the time zone and considers daylight time for usage in Sweden.

3.4.2 UMRA Node

The UMRA link takes care of the communication to the UMRA, figure 21. It works as a gateway between DSN and the UMRA. The link handles incoming data from the DSN, gives commands to the UMRA, and receives alerts from the UMRA. When an alert is received it is immediately sent to all the connected DSN nodes.

A connection to the GPS software is also made. The GPS software provides the UMRA link with information about position and time continuously. This is received and stored in the UMRA link, this so that the DSN will know where the sensor is located. When a message is sent to the DSN the GPS information is attached to the message to keep the DSN up to date.

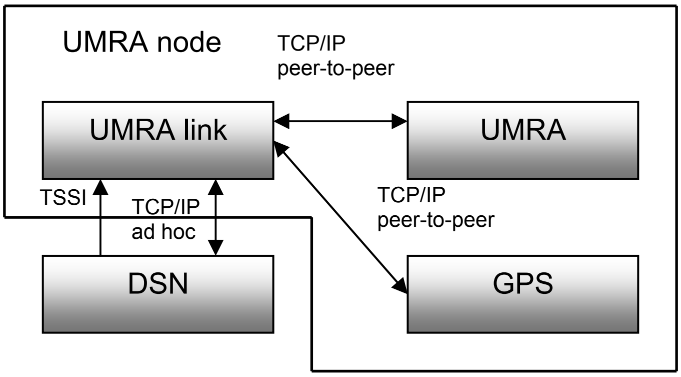


Figure 21. Communication overview of the UMRA node.

3.4.2.1 UMRA Communication

The connection to the UMRA is based on TCP/IP communication over WLAN. When the UMRA link is started it automatically tries to connect to the UMRA. It will continue to try until a connection is established. The communication is a peer-to-peer connection. If the connection fails after establishment the node will repeatedly try to reconnect till it is established again.

3.4.2.2 GPS Software Communication

To receive information from the GPS the UMRA node also has to make a connection to the GPS software. This communication is based on the same technique as the communication to the UMRA (see 3.4.2.1). When the UMRA node is started it tries to connect to the GPS software, and tries until a connection is obtained. When the connection is established the GPS software sends continuously updated information to the UMRA node. This connection will also be reestablished if it is lost.

3.4.2.3 DSN Communication

The UMRA link acts as a UDP server and is always listening for incoming TSSI packages. When a proper TSSI arrives, it tries to connect to the DSN that sent the message, by extracting the address from the TSSI package. This connection is TCP/IP based and if its lost the UMRA node will try to reconnect ten (10) times before it throws away the destination address. If the DSN sends a new TSSI the connection will be reestablished again. The UMRA node has no maximum limit of DSN connections.

3.4.3 Camera Node

The camera link handles the communication towards the WeSpot camera, figure 22. This link is divided into two separate applications with one application located on the iPAQ and one located on a laptop.

The laptop application handles the camera connection, and takes snapshots on demand from DSN.

The application located on the iPAQ is similar to a gateway; it sends information from the connected DSN to the laptop application.

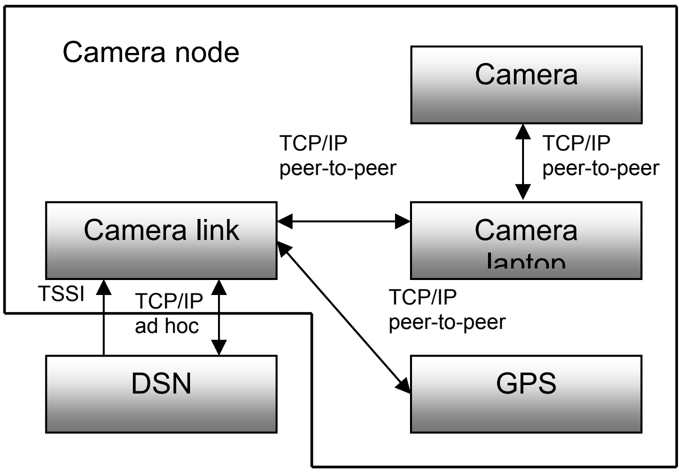


Figure 22. Communication overview of the camera node.

3.4.3.1 Camera Node Communication

When the camera link is started, it automatically tries to make a connection to the software running on the laptop. This connection is based on a TCP/IP peer-to-peer technology. The link between the two applications can be seen as a tunnel where information from the DSN is forwarded to the laptop, and the information extracted from the camera is send back to the iPAQ application. If the connection is lost the iPAQ software will continue to reestablish the link.

3.4.3.2 Camera Communication

The connection to the camera is created from the software running on the laptop. This connection is TCP based and created when the application is started.

3.4.3.3 GPS Software Communication

This connection is created from the iPAQ application when it is started. Based on peer-to-peer TCP/IP communication. The GPS software continuously sends updated information about time and position, which is stored in the camera link software and provided to the DSN when a package is to be transmitted. If the connection is lost it is automatically reestablished.

3.4.3.4 DSN Communication

This communication link is like the link between DSN and the UMRA node based on TCP/IP technology. The camera node functions as a UDP server and listens for incoming TSSI packages. When a correct package arrives, a connection is made to the DSN from which the package arrived. This connection is vital and has to be alive if any images are to be sent. If it is lost the camera node will try to reconnect ten (10) times. If it still cannot reach DSN it assumes that the node has left the monitored area or has been turned off.

3.5 Ad Hoc Routing Implementation

3.5.1 Choice of Routing Protocol – Dynamic Source Routing

In each different type of ad hoc application, there are different needs and requirements of the routing algorithm. Some of these were lined up in chapter 2.9.1. When we were about to decide which routing protocol to work with, we tried to take the desirable requirements of a full scale UGS network in consideration. Although, probably a whole lot of different protocols would have been sufficient to make our testbed work, it was beneficial and logical to use one that can be used in further development. In this thesis, we made no measurements of parameters like battery life, radio transmission medium usage, processor usage etc. even though we had those constraints in mind. Our primary goal of this testbed was to present a conceptual and functional network with different kinds of sensors, data fusion, presentation of a situation picture serving as a decision support tool.

Choosing the DSR algorithm was beneficial in many ways. The DSR protocol is one of those protocols, which IETF is currently working on to outline Internet Standards of MANET routing algorithms. Furthermore, DSR is an on-demand algorithm, using less of the network resources than a proactive one. When all nodes in the network are approximately stationary, which would be the case in an UGS network after initiation, the number of overhead packets due to DSR would scale down to zero. To be a reactive algorithm, DSR is fast in its route update mechanism, partly because of the ‘all-hops’ route caching. In the future, if using a network interface protocol that is not requiring bidirectionality for unidirectional transmissions, DSR also supports asymmetric routing.

3.5.2 Choice of Network Interface

In an early stage of our testbed implementation, we decided to use the 802.11b standard. This network interface, described in chapter 2.9.4.2, is a widespread wireless network technology. It offers a good range of transmission, high bit rate capacity and there are many 802.11b products on the market. The 802.11b standard, using the “Ad Hoc mode”, allows us to route IP packets in OSI layer 3, which is needed to use the DSR protocol.

3.5.3 DSR in OSI

The DSR algorithm is implemented at the network layer (OSI layer 3) to achieve ability to support nodes with multiple network interfaces of different types at lower layers [13] (originally, the originators of the DSR algorithm aimed to use MAC-layer routing). Using the network layer means modifying the IP packets [28 (IP tutorial)]. A special header with DSR specific data is inserted in the IP packets between the IP header and any following header such as TCP or UDP. The IP header protocol field (indicating the following protocol header) is set to a DSR-specific number, indicating that a DSR packet is following the IP header. The IP’s original protocol field info is set in a Next Header field in the DSR header.

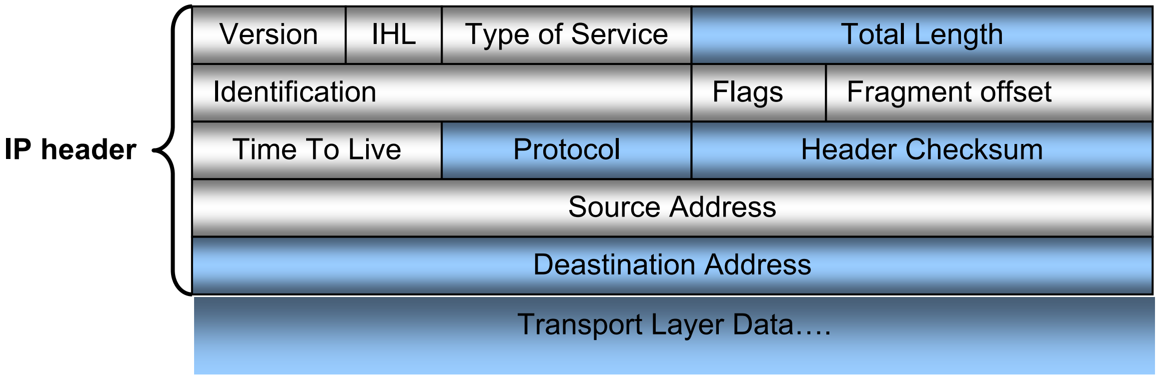


Figure 23. IP packet overview.

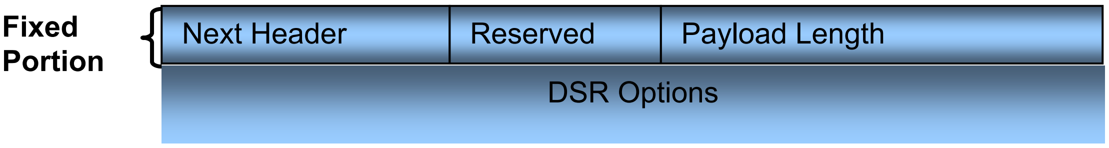


Figure 24. DSR header with fixed portion and DSR options.

The fields of Next Header, Reserved and Payload Length are set to fixed lengths. The options field holds one or more different DSR options. Such options can be:

- **Route Request Option**
  Indicating a DSR specific packet sent by a node when looking for route. The option contains among other things, the source address, destination address and an address field of the traversed nodes.

- **Route Reply Option**
  A reply sent by the destination node back to the source of route request.

- **Route Error Option**
  Information from a node indicating a broken link between two nodes. A node receiving such an option can then optionally remove routes from its route cache that is using that link.

- **Acknowledgement Request Option**
  Indicates that a data packet original source node wants acknowledge from the destination node that the packet has reached its destination.

- **Acknowledgement Option**
  A response from a data packets destination node to the source node, due to an *Acknowledgement Request* option

- **Source Route Option**
  Contains the routing information such as addresses needed for middle nodes in the network to route a packet from the original source to the destination node.

3.5.3.1 Packet Traversal through the OSI Network Layer

The DSR header is added to all IP packets addressed to within a specified DSR routing address range. When the packet finally reaches the destination node, the DSR header is removed from the IP packet before it is handled as an ordinary IP packet and sent up to higher layers.

3.5.4 Implementation of DSR

Our implementation and coding of the DSR protocol is based on a work called picoNet II [14], by Alex Song [2]. Alex's work consists of a DSR-implementation for Linux, written in the C programming language. The original implementation provided the basic features of the DSR protocol, such as route discovery and route maintenance, but not additional features such as promiscuous mode, route shortening or advertising of route errors.

---

[14] picoNet II is free software under the terms of the GNU General Public License.

3.5.4.1 Linux and Netfilter

The Netfilter is a framework for packet mangling provided in Linux 2.4. Netfilter can be used to "grab" IP packets at well-defined points when traversing the IP protocol stack in Linux. It provides the possibility to filter in-/outgoing IP packets and alters them according to the DSR protocol.

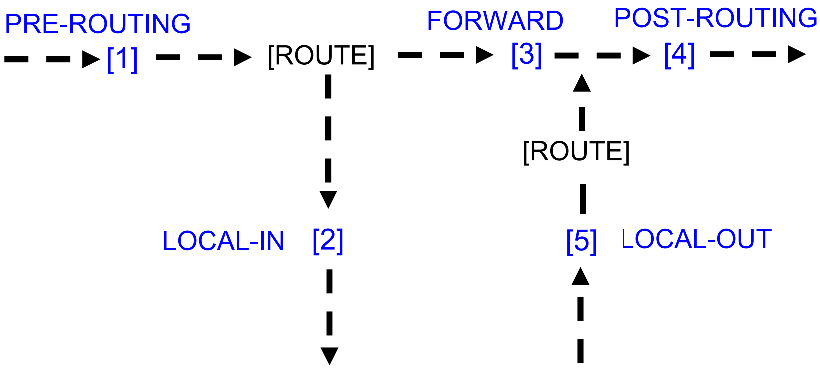


Figure 25. Defined 'hooks' for IPv4 in Netfilter.

By register one or more of the Netfilter defined 'hooks' for a kernel module, the protocol stack will call the Netfilter framework with the packet and hook number for each packet traversing the hook. Outgoing IP packets from uses space can therefore be catch at the LOCAL_OUT hook. If the packet is destined to a DSR network address, a DSR header with appropriate options is added. Likewise, the incoming packets can be grabbed at the PRE_ROUTE hook. If this node is the packets final destination, the DSR header is removed, and packet in sent up to user space. If the packet is addressed to another destination, the packet can be rerouted and then sent again.

3.5.4.2 Modifications in Used DSR Algorithm

Salvaging

When a forwarding node fails in delivering a data packet to the next node of the route, a route error is sent back to its source, indicating a failing link. The undelivered packet is normally discarded. But through salvaging, the middle node can try to forward the packet to its destination by using another cached route. The route error is still sent back to inform about the failing link. If the middle node success in salvaging the packet, it relieves the source from retransmission. Unless the source node wasn't the previous node, the retransmission from the source node will take place in the upper OSI layer (i.e. TCP-layer). That is because a node does not buffer sent data packets after they have received acknowledge from the next node.

In our DSR-implementation a packet may be solved a fixed number of times. The unused part of the source route is replaced with another route from the nodes route cache. We may optionally also perform a check that the original source node is not a part of the new suffix. In that case, maybe it is better to let the source retransmit the packet, rather than letting it traverse the network all way back to the source again.

Promiscuous Mode

Promiscuity is implemented, and is optionally used. In promiscuous mode, we are promiscuous listening to DSR specific packets not broadcasted or destined for this node. The promiscuity is currently utilized to getting passive acknowledgements; offer route shortening and snooping route errors to delete broken links from the route cache.

Multicasting Support

DSR does not currently support true multicast routing. In our UGS network we had a need of broadcasting data packets across the entire network. As described in chapter 3.3.1 we wanted our Topological Sensor Structure Inquiry (TSSI) to reach all sensor nodes. The authors of DSR suggest piggybacking the data on a route request as a solution of controlled flooding of data through out the network. Our solution to the problem is similar in using the route request mechanism. We defined an own new DSR option called TSSI, which acts approximately as a route request. When an IP packet is destined to a pre-defined TSSI address e.g. .254, the TSSI option is added to the DSR header. Optionally, the node will also advertise its latest route errors with the TSSI packet. The packet is thereafter sent to the network interface destined to the broadcast address .255. Every sent TSSI packet contains a unique TSSI ID number. When a node receives a DSR packet containing a TSSI option, it will first check that is has not seen this TSSI before, to avoid looping in the network. The receiving node extracts any enclosed route errors. Thereafter it examines the packets address field, which is increased with each traversed node, and updates its own route cache. Thereafter the TSSI packet is updated with fresh route errors and re-broadcasted. All nodes receiving a TSSI packet send the enclosed data up to user space (OSI layer 6-7). This way, all nodes receive the data attached with TSSI, and DSR lets the application layer filter the data. Only the sensor nodes make use of the TSSI packets.

3.5.5 Additional Changes

In the testbed, we used the same network interfaces to communicate with the sensors at the sensor nodes, as to communicate with the DSR network. Because the WLAN cards used in the testbed only communicate within the same IP subnet we divided the subnet in two parts at the network layer. The nodes belonging to the DST network have IP addresses in the range 0 and 128 belonging and the range 129 to 253 belonged to local communication not using DSR routing.

Due to low processor and memory capacity on the iPAQ, there is a heavy load on the iPAQ nodes, using our application specific JAVA implementations. This lead to long processing time of arrived packets, making the DSR protocol’s retransmit queue time out. To reduce the side effects of this, the queue size and time constants are modified, compared to the IETF DRS draft [13].

3.5.6 Routing Implementation Considerations

Making use of promiscuous mode will lower the amount of over head data packets such as acknowledge packets for received packets, which is desirable considering the radio medium usage. Battery conservation is another important issue for mobile devices. In a simulation study for a Master Thesis at Ericsson [29], it was stated that operating in promiscuous mode would consume roughly 50% more energy than non-promiscuous mode. When nodes are receiving promiscuous, much resource is spent on actively decoding the received packets. Considering this, promiscuous mode should maybe not be used in our testbed; to reduce the problems of DSR acknowledge packets time out mentioned in chapter 3.5.5.

3.6 Future Development and Improvements

During the development of the testbed we encountered some issues that we would like to modify.

The map implemented in the DSN is given from development, which would not last in a live sensor network. A more sophisticated solution would be to automatically download a map showing the area in which the user is present at. To implement this feature a GPS receiver is needed in the DSN application. Further more, some sort of map storage must be developed to provide this service.

Optimization of software is another very important issue since the processing capabilities in the iPAQs are limited. One optimization would be to convert all software, not the routing algorithm, to C++, which would enhance processing of data.

A general communication solution for sensors would allow many different types of sensors in the network. This would however require further research in data fusion among other things.

3.7 Achievements

The final test of the testbed was a demonstration. For future demonstration there are some issues to think about.

Firstly, the demonstration should not be held on public roads. This led to many alerts during our demonstration. However, the audience had themselves some laughs since alerts came all the time.

Secondly, functionaries should be equipped with communication possibilities other than mobile phones, since the area we demonstrated the network in had almost none mobile network coverage.

Overall the demonstration went very well and we proved that it is possible with an UGS network to surveillance a geographical area.

4 Conclusions

The concept of Unattended Ground Sensors (UGS) is a growing technology with worldwide interests. The technology will provide unprecedented possibilities. *The one with the information is the one with the power* and that is what a UGS network may provide.

The conclusion we have come to during this thesis is that the technology is ready for development of technically advanced ad hoc based UGS networks. There are still many unsolved issues regarding management of great amounts of information (data fusion), routing and development of low cost sensors with great battery power and radio capabilities. Further work must also be done to ensure compatibility between different systems such as allied countries UGS networks.

The results of our work have proven the functionality of existing sensor systems in an ad hoc environment by the developed testbed. We have shown that information from randomly deployed sensors has been received by a remote end-user thanks to a sophisticated routing algorithm. Interaction and management of the network can also be done to provide required information.

The research of UGS in an ad hoc is still in its inception and breakthroughs in the field are regular made. Unprecedented applications will be evolved in the future, a future not so distant.

5 Abbreviations

ad hoc	Literally means *for this* in latin.
AODV	Ad Hoc On-Demand Distance Vector
AUV	Autonomous Underwater Vehicle
CTS	Clear To Send
COTS	Commercial Off The Shelf
CPU	Central Processing Unit
DBF	Distributed Bellman-Ford
DSDV	Destination-Sequenced Distance Vector
DSN	Decision Support Node
DSP	Digital Signal Processing
DSR	Dynamic Source Routing
DV	Distance Vector
EMC	Electro Magnetic Compatibility
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GSM	Global System for Mobile communication
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IP	Internet Protocol
ISM	Industrial Scientific Medical
ISO	International Organization for Standardization
Kbps	Kilobits per second
LAN	Local Area Network
LLC	Logical Link Control
MAC	Media Access Control
MANET	Mobile Ad Hoc Network
Mbps	Megabits per second
NCW	Network Centric Warfare
NIST	National Institute of Standards and Technology
NOS	Network Operating System
OS	Operating System
OSI	Open Systems Interconnection
PDA	Personal Digital Assistant

RTS	Request To Send
SA	Selective Availability
TCP	Transmission Control Protocol
TSSI	Topological Sensor Structure Inquiry
TTL	Time To Live
UDP	User Datagram Protocol
UGS	Unattended Ground Sensors
UMRA	Intelligence Multisensor RAdio (sw. Underrättelse Multisensor RAdio)
UMTS	Universal Mobile Telecommunications Service
XML	eXtensible Markup Language
ZRP	Zone Routing Protocol
WLAN	Wireless Local Area Network

6 References

[1] ”Adhoc Mobile Networks”
Hridesh Rajan, Bell Labs India
http://www.cs.virginia.edu/~hr2j/work/MANET.html

[2] “picoNet II : A Wireless Ad Hoc Network for Mobile Handheld Devices”
Alex Strong
October 2001

[3] ”Spontaneous Networking: An Application-oriented Approach to Ad Hoc Networking”
Laura Marie Feeney
http://www.sics.se/cna

[4] ”Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations”
S. Corson and J. Macker
http://xml.resource.org/public/rfc/html/rfc2501.html – June 2002

[5] “Wireless Ad Hoc Networks”
Zygmunt J. Haas, Jing Deng, Ben Liang, Panagiotis Papadimitratos and S. Sajama
http://www.edu/~haas/wnl/html

[6] “Ad Hoc Mobile Wireless Networks: Protocols and Systems”
C-K Toh
Prentice Hall PTR – 2002
ISBN: 0-13-007817-4

[7] “Mobile Ad Hoc Networks (MANETs)”
NIST
http://w3.antd.nist.gov/wctg/manet/manet.html

[8] “What Is Bluetooth?”
PaloWireless - Bluetooth Resource Center
http://www.palowireless.com/infotooth/whatis.asp

[9] “Bluetooth SIG, Inc. - Public Specifications”
Bluetooth SIG
http://www.bluetooth.org/specifications.htm

[10] “WLAN standard ieee Std 802.11b-1999”
IEEE
http://www.ics.uci.edu/~support/wireless/802_11b-1999.pdf

[11] ” IEEE 802.11b ’High Rate’ Wireless Local Area Networks”
Kanoksri Sarinnapakorn
March 2001
http://alpha.fdu.edu/~kanoksri/IEEE80211b.html

[12] ”The OSI Seven-Layer Model”
Northwest educational Technology Consortium
http://www.netc.org/network_guide/c.html

[13] “The dynamic Source Routing Protocol for Ad Hoc Networks”
IETF MANET Working Group
http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-07.txt – July 2002. Work in progress.

[14] ”Mobile Ad-hoc Networks (manet) charter”
IETF MANET Working Group Charter
July 2002
http://www.ietf.org/html.charters/manet-charter.html

[15] ”Mobile Ad Hoc Networking and the IETF”
Joseph P. Macker & M. Scott Corson
http://www.acm.org/sigmobile/MC2R/articles/manet_v3n2.pdf

[16] “Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers” p.2
Charles E. Perkins and Pravin B.

[17] “Dynamic Source Routing in Ad Hoc Wireless Networks”
David B. Johnson and David A. Maltz
http://www.cs.brown.edu/courses/cs194-5/docs/adhoc_wireless_ntwk.pdf

[18] “Ad Hoc Networking”
Charles E. Perkins
Addison-Wesley – December 2000
ISBN: 0-201-30976-9

[19] “The Netfilter framework in Linux 2.4”
Harald Welte
September 2000
http://www.gnumonks.org/papers/netfiltr-lk2000/presentation.html

[20] “Handling Unidirectional Links in Ad-Hoc Wireless Networks”
Sharad Agarwal
December 2000
http://www.cs.berkeley.edu/~sagarwal/research/cs294-1/report.pdf

[21] “Providing Ad-Hoc Connectivity with Reconfigurable Wireless Networks”
Zygmunt J. H. and Marc R. P. School of Electrical Engineering, Cornell University Ithaca, NY
http://www.ee.cornell.edu/~haas/wnl.html

[22] “Unattended Ground Sensors Stop and Analyze the Roses”
http://www.spie.org/app/publications/magazines/oerarchive/april/apr00/cover2.html

[23] “Unattended Ground Sensors A Prospective for Operational Needs and Requirements”
Nino Srour, U.S. Army Research Laboratory Sensor and Electron Devices Directorates
October 1999
http://www.arl.army.mil/acoustics/UGS for NATO Land Panel 6.pdf

[24] “Decision Making and Data Fusion in an Interactive Adaptive UGS Network”
Erland Jungert, Christian Jönsson, Per Klöör, Stan Zyra
FOA (Swedish Defence Research Establishment)

[25] “Traditionella underrättelsefrågor vid val av sensorkombinationer”
Major Folke Sundqvist, Swedish National Defence College

[26] “Sensor Networks for Network-Centric Warfare”
Planning Systems Inc.
October 2000
http://www.plansys.com/Content/NavigationMenu/Products/Sensor_Network_and_Data_Acquisition_Products_White_Papers/Sensor_Networks_for_Network_Centric_Warfare_NCW00.pdf

[27] “Distributed Sensor Processing over an Ad Hoc Wireless Network: Simulation Framework and performance criteria”
Robert E. Van Dyck & Leonard E. Miller
Wireless Communications Technologies Group National Institute of Standards and Technology

[28] “A TCP/IP Tutorial”
T. Socolofsky & C. Kale
Spider Systems Limited
January 1991
http://www.ietf.org/rfc/rfc1180.txt

[29] “Routing Protocols in Wireless Networks – A Simulation Study”
Tony Larsson & Nicklas Hedman
March 1999
http://www.ietf.org/proceedings/99mar/slides/manet-thesis-99mar.pdf

[30] “Swedish Armed Forces – Introduction of Network Centric Warfare”
MPEG Movie (in Swedish)
http://www.mil.se/download/NBF.zip