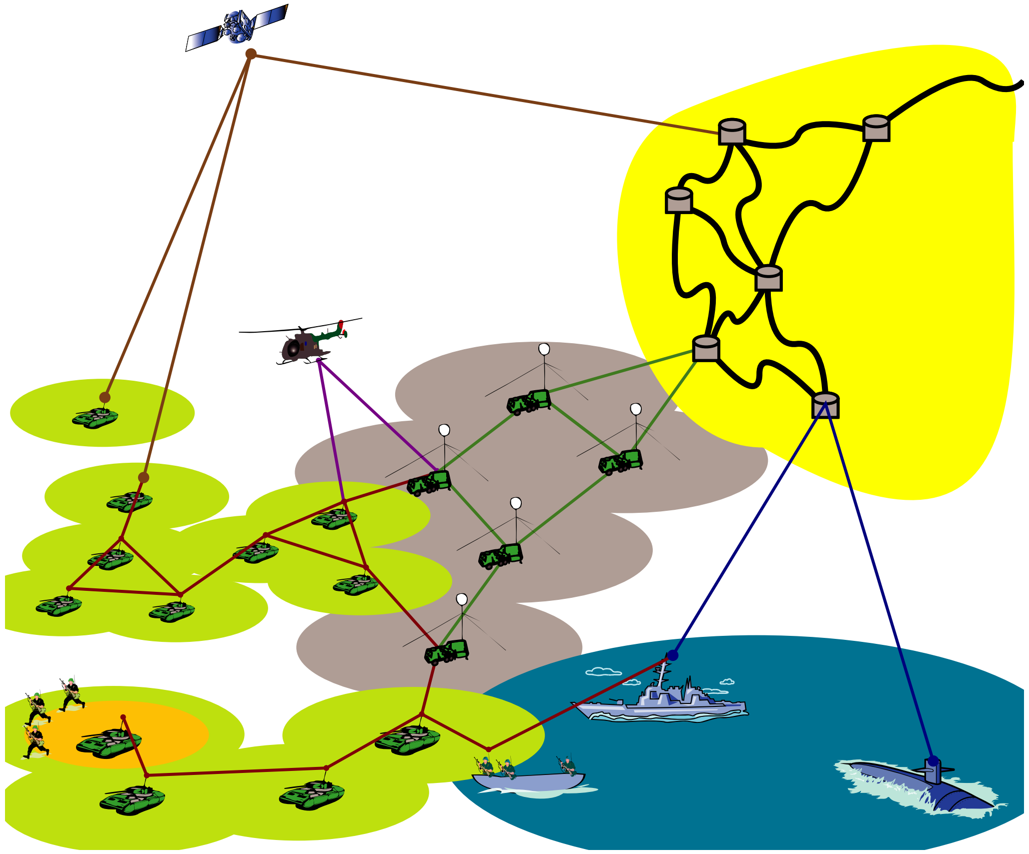
Connecting Mobile Ad Hoc Networks to Fixed Infrastructures – Address Assignment and Quality of Service

JIMMI GRÖNKVIST, ANDERS HANSSON, JAN NILSSON AND OTTO TRONARP



FOI-R--2265--SE
ISSN 1650-1942

Technical Report
April 2007

Command and Control Systems

FOI, Swedish Defence Research Agency, is a mainly assignment-funded agency under the Ministry of Defence. The core activities are research, method and technology development, as well as studies conducted in the interests of Swedish defence and the safety and security of society. The organisation employs approximately 1250 personnel of whom about 900 are scientists. This makes FOI Sweden's largest research institute. FOI gives its customers access to leading-edge expertise in a large number of fields such as security policy studies, defence and security related analyses, the assessment of various types of threat, systems for control and management of crises, protection against and management of hazardous substances, IT security and the potential offered by new sensors.

FOI
Swedish Defence Research Agency
Command and Control Systems
P.O. Box 1165
SE-58 111 LINKÖPING

Phone: +46 13 37 80 00
Fax: +46 13 37 81 00

www.foi.se

Connecting Mobile Ad Hoc Networks to Fixed Infrastructures – Address Assignment and Quality of Service

FOI-R- -2265- -SE
ISSN 1650-1942

Technical Report
April 2007

Command and Control Systems

Issuing organization Swedish Defence Research Agency Command and Control Systems P.O. Box 1165 SE-581 11 LINKÖPING SWEDEN	**Report number, ISRN** FOI-R- -2265- -SE	**Report type** Technical Report
	Research area code 7. C^4I and Human Factors	
	Month year April 2007	**Project No.** E75471
	Sub area code 71. Command, Control, Communications, Computers, Intelligence (C^4I)	
	Sub area code 2	
Author/s Jimmi Grönkvist, Anders Hansson, Jan Nilsson and Otto Tronarp	**Project manager** Jan Nilsson	
	Approved by Sören Eriksson	
	Sponsoring agency Swedish Defence Materiel Administration	
	Scientifically and technically responsible Jan Nilsson	

Report title

Connecting Mobile Ad Hoc Networks to Fixed Infrastructures – Address Assignment and Quality of Service

Abstract

In order to meet the requirements of secure, dynamic, robust and reliable information exchange over a heterogeneous network, including a mobile ad hoc network, several topics and problem areas have to be addressed, and a number of open issues have to find satisfactory solutions. In this report we have selected two topics of importance: mobility management, more specifically the problems of assigning IP addresses; and Quality of Service (QoS).
The aim of the report is to describe the problem areas, and discuss different solutions and their pros and cons. For IP address assignment there is the issue of dealing with node mobility, e.g. is variable IP addresses a good solution? The QoS issue is complex and involves all networking layers. However, at least some minimal control functionality is required to provide some sort of QoS.

Keywords

Heterogeneous networks, Ad Hoc Networks, Network mobility, IP-address assignment, Quality of service

Further bibliographic information	**Language** English
ISSN 1650-1942	**Pages** 54 p.
	Price acc. to pricelist

Utgivare Totalförsvarets Forskningsinstitut Ledningssystem Box 1165 SE-581 11 LINKÖPING	**Rapportnummer, ISRN** FOI-R- -2265- -SE	**Klassificering** Teknisk rapport
	Forskningsområde 7. Ledning med MSI	
	Månad, år April 2007	**Projektnummer** E75471
	Delområde 71. Ledning	
	Delområde 2	
Författare Jimmi Grönkvist, Anders Hansson, Jan Nilsson och Otto Tronarp	**Projektledare** Jan Nilsson	
	Godkänd av Sören Eriksson	
	Uppdragsgivare/kundbeteckning FMV	
	Teknisk och/eller vetenskapligt ansvarig Jan Nilsson	

Rapportens titel

Att koppla samman mobila ad hoc-nät med fast infrastruktur
– adresstilldelning och tjänstekvalitet

Sammanfattning

För att möta kraven på säker, robust och tillförlitlig informationsöverföring i ett heterogent nätverk, som inkluderar ett mobilt ad hoc nät, behöver ett antal problemområden adresseras. Det finns ett antal öppna frågeställningar som behöver finna tillfredställande lösningar. I rapporten väljer vi ut två viktiga problemområden, först mobilitetshanteringsproblemet med att tilldela IP-adresser, och sedan tjänstekvalitet.

Syftet med rapporten är att beskriva problemområdena, sedan att diskutera olika möjliga lösningar och deras fördelar respektive nackdelar. När det gäller IP-adresstilldelning är problemet hur man ska handskas med mobila noder, t.ex., är det lämpligt med variabla IP-adresser? Hur tjänstekvalitet ska hanteras är en komplex frågeställning och den berör alla nätverkslagren. Om man vill kunna erbjuda någon form av tjänstekvalitet behövs i all fall en viss funktionalitet som styr detta.

Nyckelord

Heterogena nät, Ad hoc-nät, mobilitetshantering, IP-adresstilldelning, tjänstekvalitet

Övriga bibliografiska uppgifter	**Språk** Engelska
ISSN 1650-1942	**Antal sidor:** 54 s.
Distribution enligt missiv	**Pris: Enligt prislista**

Contents

Chapter 1

Introduction

1.1 Background

Whenever different types of networks, e.g. ad hoc networks, fixed networks, and cellular networks are connected a so-called heterogeneous network is formed. An ad hoc network is a collection of wireless nodes that dynamically form a temporary network without any pre-existing network infrastructure or centralized administration. Due to limited transmission range, multiple hops may be needed to communicate between nodes.

Meeting the requirements on secure, dynamic, robust and reliable information exchange over the heterogeneous network's architecture is a challenge. Adequate Quality of Service (QoS) and security have to be provided for all the traffic flowing across the network boundaries. To accomplish that, in particular as soon mobile tactical ad hoc networks are involved, requires improved networking integration techniques.

Several important topics have to be addressed. The list is long, and includes: mobility management, auto-configuration, resource management, QoS and security. Other more general issues and topics are network planning and management. The last topic is very broad, and any discussion about it needs firstly to address the two basic questions: what exactly is it that's being managed, and what is "manageable" about it?

In this report we have selected two of the topics above. In Chapter 2 we describe mobility management, more specifically the problems of assigning IP

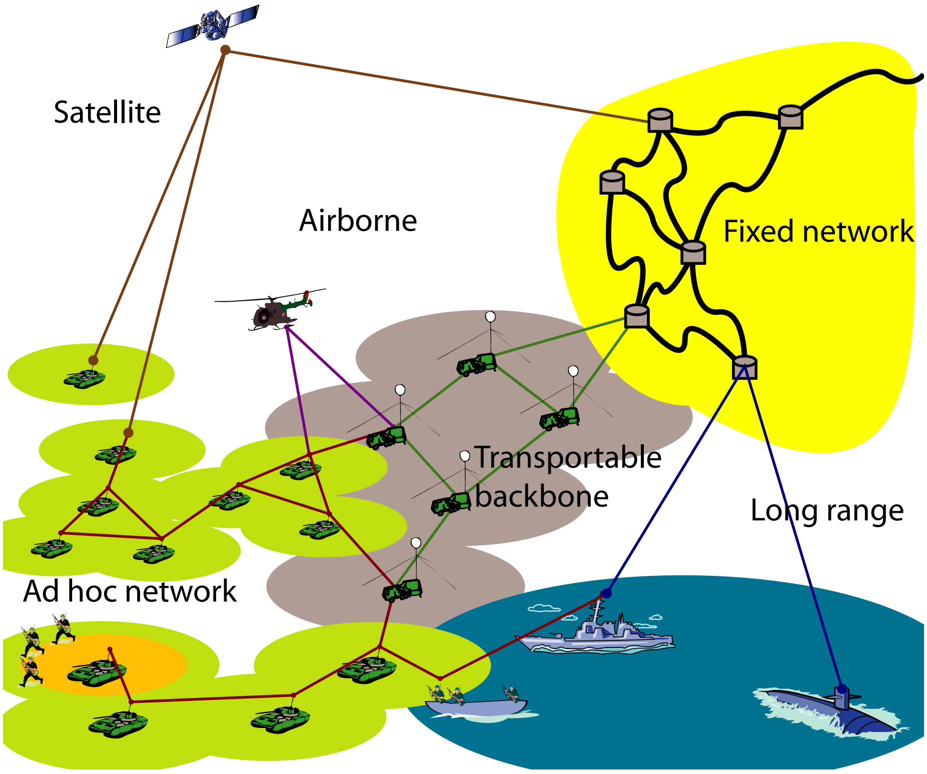


Figure 1.1: An example of a heterogeneous network.

addresses, and in Chapter 3 we focus on QoS. From a heterogeneous network perspective the topics are different in the sense of the necessity of complying with existing standards, where the first topic is more governed by such standards. The aim of the report is to describe the problem areas and to discuss possible solutions and their pros and cons. Furthermore, we try to assess what the anticipated outcome will be from the ongoing work within the research community (e.g. IETF) and point out suitable areas for more detailed research.

1.2 Heterogeneous Networks

In Figure 1.1 we give an example of a heterogeneous tactical network consisting of interconnected subnetworks. Note that there are other slightly different uses of the term *heterogeneous network*. For computer networks, a heterogeneous network can be a network of devices with different operating systems and protocols. Another common use of this expression is a heterogeneous "network of networks", for example the Internet with its worldwide collection of different computer networks, connected by copper wires, fibre-optic cables, wireless connections, etc. In the context of this report, however, we use the term for a network of networks with more extreme variations between the network properties. We give a short description of the components included in the example.

The main advantage of mobile ad hoc networks is that local transmissions are independent of a fixed infrastructure. Instead, nodes communicate with multi-hop connections through the network. Moreover, this type of network is dynamically self-configuring and can function autonomously when no contact with other networks is available. This leads naturally to large variations in the quality of the connections in this subnetwork. Two nodes that are close to each other can communicate with high capacity and low delay, although the connections may be quite poor, or even nonexistent, between parts of the ad hoc network that are more separated from each other.

A fixed network of connected routers built on the common Internet protocol stack can offer both high capacity and low delay. In tactically secure areas, transportable backbone networks can be deployed in a short time period ranging from hours to days, depending on the amount of planning needed. With high masts and directed antennas, these networks extend the connection to a fixed network with good transmission quality.

The networks can also be connected through long-range links, such as satellite communication. These links may have a large delay, and can offer both high and low capacity, depending on the budget for the mission. HF is another example of long-range communication, typically with low capacity and time-varying link characteristics. Temporary medium-range connections can also be obtained with airborne nodes, for example helicopters or UAVs.

The type of heterogeneous networks discussed offers connections over large areas and through different kinds of networks.

1.3 Network Scenarios

We are going to show problems regarding gateways (GW) through four different scenarios. They are all very closely related, and although some can be seen as sub-problems of others, they each simplify the description of specific problems that need to be addressed. The scenarios are mainly intended for the IP addressing part in Chapter 2, but they are also briefly discussed in the QoS part in Chapter 3.

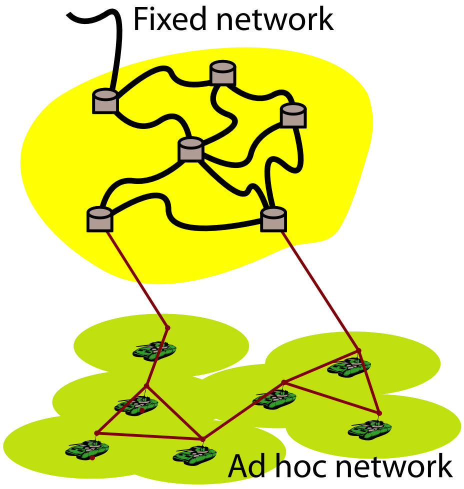


Figure 1.2: Multiple gateways.

1: Multiple gateways

A number of nodes in the ad hoc network have access (in some way) to a fixed high capacity network, see Figure 1.2. Not all nodes are GWs though and it may change over time.

The GWs could be several of the same type or they could be using completely different access technology (fixed connection, UMTS, satellite, etc). We can also see a specific case where a GW provides access to other wireless systems instead of a fixed network.

Main communication challenges: To decide which GW to use when communicating from the fixed network to an ad hoc network (and the reverse), and how to make it possible to shortcut through the fixed network.

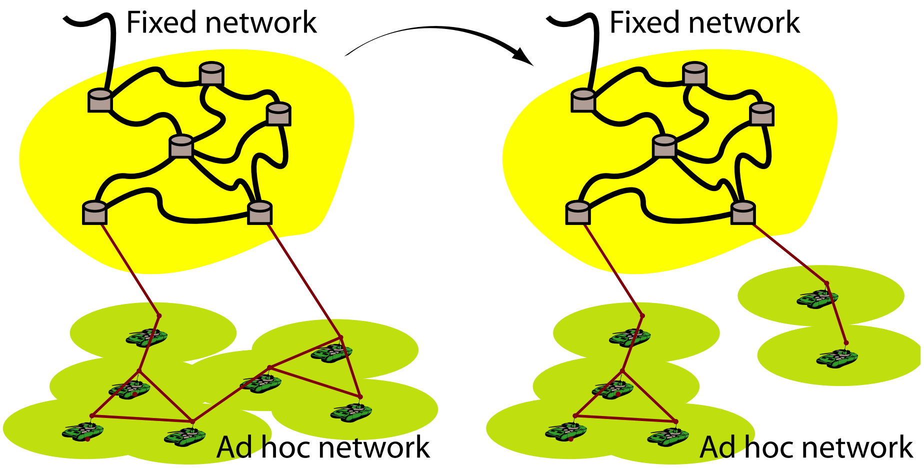


Figure 1.3: Network split.

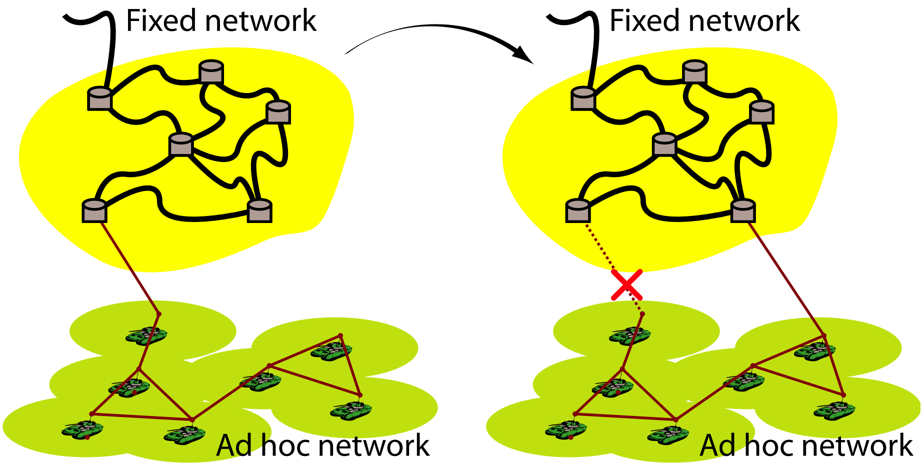


Figure 1.4: Changing gateway.

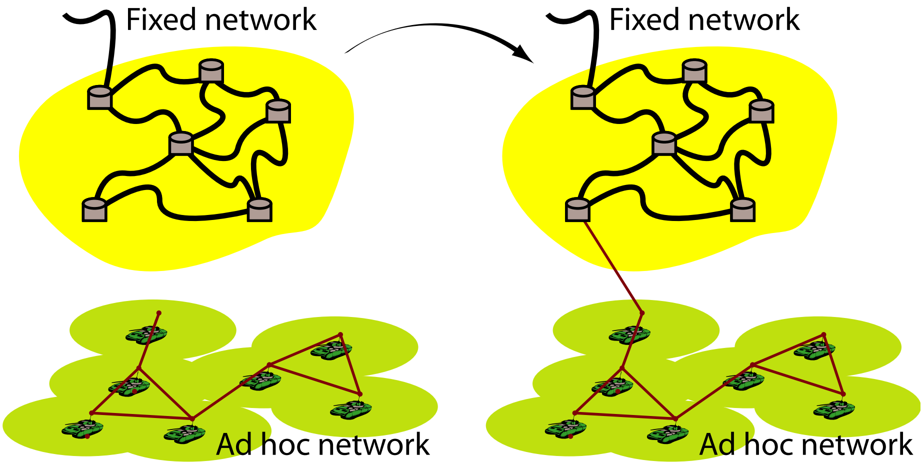


Figure 1.5: Connection to a gateway.

2: Network split

The ad hoc network splits into two different subnets as shown in Figure 1.3. Any combination of nodes may be in each of the subnets, which complicates planning.

Main communication challenges: Now the only possibility of communication between the subnets is by using the fixed network. Furthermore, communication from the fixed network to the ad hoc network will not work if we use an incorrect GW (at least not without cooperation between the GWs).

3: Changing gateway

The entire network switches GW as illustrated in Figure 1.4. This is usually the case because one GW is lost and another takes over. Alternatively, a much better GW takes over. However, the latter is a much simpler case as the old GW usually is still functioning.

Main communication challenges: The GW information needs to be updated in the ad hoc and in the fixed network.

4: Connecting to a gateway

The ad hoc network connects to a fixed network after being autonomous for a while, as shown in Figure 1.5. Preferably, this should not happen often, but may be more common for smaller subgroups.

Main communication challenges: Some nodes in the fixed network may need to know that the ad hoc network now is accessible again, also the ad hoc nodes (and users) should now again be given access to applications of the fixed network.

1.4 Internet Standardization and IETF

The Internet Engineering Task Force (IETF) is a loosely self- organizing group that contributes to the development and evolution of the Internet. It is the main body handling the development of new Internet standards specifications.

The IETF defines standards that are often adopted by Internet users, but it does not control, or even patrol, the Internet. The IETF does not in any way ”run the Internet”, but if anything is to be widely adopted by the Internet community, it will need to be standardized by the IETF, see [1] for more information,

Working Groups

The main work of the IETF is done within the many Working Groups (WG). In the latter part of 2006 there were about 115 such groups. Each WG follows a charter that states the scope of discussion for the working group, a discussion that is mostly held through a mailing list.

The WG that has been most relevant for the development of ad hoc networks is the Mobile Ad-hoc Networks (MANET) WG. ”The purpose of the MANET working group is to standardize IP routing protocol functionality suitable for wireless routing application within both static and dynamic topologies with increased dynamics due to node motion or other factors.”

The work done by the MANET WG has mostly been concerned with the internal routing problems within the ad hoc network, but it also states that solutions should function when attached to a fixed IP infrastructure.

A more recently started WG for ad hoc networks is the Ad-Hoc Network Auto configuration (AUTOCONF) WG. ”The main purpose of the AUTOCONF WG is to standardize mechanisms to be used by ad hoc nodes for configuring unique local and/or globally routable IPv6 addresses.” At the time of writing no documents have yet been released by this WG besides the charter.

Documentation and Standards

The main form of written documents describing standards is published as Request for Comments (RFC), although not all RFCs describe standards. In order to standardize an Internet protocol it is first published as an Internet Draft. Internet drafts have no formal status, and are subject to change or removal at any time.

After an Internet draft has been commented on by other people and revisions based on the comments have been made, usually an updated version of the draft is published. This will usually be iterated a few times before the document is taken to the IETF for publication as an RFC (IETF may, however, also have opinions on the document).

Once the document is published as an RFC, it can be one of 6 classes:

1. Proposed standard
2. Draft Standard
3. Internet Standard
4. Informational documents
5. Experimental Documents
6. Historic Documents

Only the first three are standards within the IETF. Experimental RFCs describe specifications that may be interesting but for which it might be unclear whether there is an interest in implementing them, or whether they would work once implemented. If the specification becomes sufficiently popular and/or can be shown to work in practice, it may be re-issued as one of the first types.

At the moment all published ad hoc network protocols are described in documents of the experimental type. This means that there are still no formal standards that are specific to mobile ad hoc networks. Furthermore, it is still uncertain when (or whether) this will happen. We might still see significant changes to the existing protocols, especially in terms of security, the existing RFCs currently do not handle this issue at all. How this will change existing protocols is uncertain.

Chapter 2

IP-addressing

To be able to communicate at all in an IP network, a node requires an IP address. In general every interface on an Internet is given a unique IP address that is used both for determining the identity of a node and its geographic position in the network. There are exceptions to the requirement that the address is globally unique, but then the address cannot be used outside a local network in which the address still must be unique.

2.1 Traditionally Fixed Networks - The Internet

Two problems related to IP addresses must be solved before two nodes can communicate.

First, the initiating node needs to determine the IP address of the destination. This can be solved in several ways. If we know in advance which nodes will communicate, addresses can be manually set. However, this is seldom the case and, moreover, humans do not like to use IP addresses directly. For portable nodes, a node may be attached to any point on an Internet, and the different local servers will not necessarily have the same IP address. This means that the IP address of the destination has to be figured out before communication is possible. The most common method is DNS, translation of URLs (Universal Resource Locators) to IP addresses, but specific service location protocols for different applications are also common.

The second problem is to find a path to the destination now that we have its IP address. As there are millions of nodes in the Internet and the destination could be anywhere (for some applications at least, hopefully, the local printer is closer), this is not a trivial task. To solve this problem, the Internet IP address architecture is highly hierarchical and requires considerable pre planning of addresses to function well.

A central part of IP routing is aggregation of routes. In order to create a hierarchy of addresses, the IP address is divided into two parts. The first part describes which network the node is part of and the second part describes which specific node it is in that network. The bits that are used to describe network size are variable and information about this is provided with a subnet mask. This enables routes to be aggregated. Thus, instead of a router being required to tell the world which specific nodes that it has a route to, it can advertise only the networks that can be reached (with a subnet mask). Furthermore, information about several networks can be aggregated by making the subnet mask shorter (as long as network addresses are properly chosen). By aggregating networks by planning network addresses well, routing to very large number of nodes is possible.

2.2 Portable Nodes - Single Nodes that Attach Anywhere

Portable nodes are not fixed to a single physical position in the network, but instead move from one attached point to another. We distinguish them from being mobile by assuming that communication does not take place during transfer. The problem with these nodes is that to communicate in the hierarchical (regarding address at least) Internet, a new IP address is needed at each attachment. This is normally solved with a DHCP server that gives a node an IP address when it connects to a network. In most cases this is sufficient for communication *from* the node. However, it creates problem when another node attempts communication *to* the portable node, as its IP address is not constant. Either the DNS server needs to be updated or the old IP address still needs to be useful.

The second solution is usually the simplest for portable nodes. In Mobile IP, a node uses a fixed IP address that belongs to a home link where a “home agent” receives packets for the portable node and tunnels them to the present location. Each time a node moves, it updates its home agent about its present

IP address. This might seem good, but it has some drawbacks that decrease its usefulness in some cases. First, the home agent must be reachable at all times both from the portable node and the node that wants to communicate with it, which may be a problem in some scenarios. Second, when a node changes attachment point, existing communication sessions will not work until the home agent has been updated, which can take time. Firewalls can also create problems as the portable node will use an IP address that is not hierarchically correct, and firewalls seldom allow such packets to pass.

If we make the nodes mobile, i.e. can move during a communication session, the scenario becomes even more complicated.

But how important is it to initiate communication with a portable or mobile node? Most systems today are based on a server-client architecture. In such a case the client will initiate the communication with a server, and as long as the server is static, the reachability of a portable client is not an issue. For example, e-mail is one application for which communication between portable/mobile clients is solved by using static servers. Keeping the server static will not necessarily solve all possible applications, however. For example, setting up real-time applications, such as voice applications, can be more difficult unless all traffic goes through a static centralized server, which is undesirable.

2.3 Applications - Do They Care About IP-addresses?

Unfortunately, an IP address is used not only for the topographic description of a node's position in the network but also as a identifier. Transport protocols (TCP/UDP) use IP addresses and specific port numbers to connect two applications on different nodes. A change in either IP address often breaks the session. In some cases a change in IP address can be hidden by tunnelling the old packet with the old IP address into a new packet with the new address. This, however, not only increases the overhead, but both sides also need to be able to do this, which is not always possible. Other solutions can be used for greater separation of identity and location, see for example the Host Identity Protocol (HIP) [2], in which case the identity has been separated from the IP address. However, HIP requires a significant change in the Internet architecture and cannot currently handle multicast traffic, which is very common in military networks.

Another alternative is to let the applications be aware of mobility and handle changes in IP addresses on their own. One protocol that can be used is the Session Initiation Protocol (SIP), which can be used to negotiate (and renegotiate) the setup of sessions including IP addresses. But such a solution requires all implemented applications to be able to do this.

2.4 Ad hoc Networks

Unlike traditional fixed IP networks, an ad hoc network is not pre- planned. Nodes may move around, and the network can have any shape. Therefore specific routing algorithms for ad hoc networks have been developed, see for example [3, 4]. Most of these are completely flat, i.e. no hierarchy at all. In addition they are often optimized to transmit as little information as possible because link capacity is limited and not just shared between different traffic flows, but also over the hops in a single flow.

Routing for ad hoc networks can be divided into two types, *proactive* and *reactive*. Proactive routing attempts to keep up routes to all nodes all the time, which means that when a message arrives at a node the next hop on the route will be known and the packet can be transmitted immediately. Reactive routing, on the other hand, only finds routes whenever someone wants to use them. When a packet arrives at a node, it must first initiate a route search (and get a response) before it can be forwarded.

Generally, the type of routing preferred depends on traffic scenarios. If only a few nodes at a time are interested in communicating with a few other nodes, reactive routing has an advantage as we do not need to waste overhead traffic setting up routes that are not needed. On the other hand, if many users often need to send messages to a large portion of the network, most of the possible routes will be needed anyway. Therefore, the systematic approach of proactive routing may be more efficient, especially because routes exist directly when they are needed, and the extra time to do the route search for reactive routing will not be needed.

In military tactical networks where multicast traffic and position information that is regularly sent to a large portion of the network are valuable services, proactive routing does have advantages in many cases.

The traffic scenario inside the ad hoc network should usually have precedence when deciding which type of routing protocol is preferable. However, that choice will affect the interconnection between a fixed network and an ad hoc network. In order to communicate with networks outside the ad hoc network, a node needs to discover a GW (or at least be able to send packets that reach a GW). In proactive routing this can easily be incorporated into the routing protocol itself. For reactive routing the node either needs to specifically search for a GW, or other mechanisms are needed to find a GW.

The problem here is twofold. First, is a desired destination inside or outside the ad hoc network? Second, how do we relay packets to the destination if it is outside? In reactive routing, routes are only searched for if a route is needed, which means that the GW does not know which nodes that are inside its part of the network. If the destination is inside the ad hoc network, it will reply to the route search; if it is not inside the network, who will answer? We might have several GWs, and we might have a split ad hoc network so that we also need to relay traffic over the fixed network to reach the destination. One possible solution is that all GWs respond to the route search, and the source responds to one of them unless a direct response from the destination is given. One GW in each separate part of the ad hoc network can then send out a separate route search to find the destination. (All could send out the route search message, but for efficiency reasons it would be good if each node only retransmits this message once independently if it originates from the same GW, so we will use the closest GW to the destination.) This solution can have scalability problems when the network gets large as there is no simple way to exploit the higher capacity of the fixed network between the GW to decrease traffic within the ad hoc network. For a large ad hoc network it may be more efficient to take a shortcut through the fixed network to reach another node. In proactive routing the GW always knows which nodes it can reach directly through the ad hoc network. Hence the main problem to solve here is coordination between GWs, which is also required for reactive protocols.

Another problem is how external nodes find the node inside the network. If we ignore for now how a node finds which IP address the destination uses, the essential problem is to know which GW to send the packet to if there are several. Some form of interaction between GWs is probably necessary here, especially for reactive routing, where each GW may have to do a route search to find the ad hoc node.

2.5 Assumption 1: Fixed IP-address

We can divide the possible solutions to IP addressing into two different categories. Either the ad hoc node keeps its IP address despite mobility or we change it to reflect the effects that mobility has on network topology. We will start the discussion with the first case, keeping the IP address even if the ad hoc node changes attachment to the fixed network (and/or gateway).

It should be noted that we do not mean a permanent IP address that the node will always keep, but rather we assume a fixed IP address during a certain operation (hours, at most days). For example, it might be difficult to change DNS to handle mobility, but it is possible, over a longer time scale, to handle planned movement. For example, if a battalion moves to a foreign country, changing the IP address in DNS servers to point to an appropriate point in the fixed network is still simpler than other problems.

Keeping the IP address will have benefits with respect to applications because then we won't have any broken sessions. However, the main drawback of fixed IP addresses is that the hierarchical structure of inter-networking will not be upheld. To realise the consequence of this, it is important to discuss the concept of an "autonomous system" (AS) [5]. "An autonomous system is a set of routers under a single technical administration, using an interior gateway protocol and common metrics to route packets within the AS, and using an exterior gateway protocol to route packets to other ASs." An AS must appear to other ASs to have a single coherent interior routing plan and presents a consistent picture of what networks are reachable through it. The number of ASs is not that great but normally they do not have routing policies that allow nodes to attach anywhere without changing IP address (except with specific protocols as mobile IP), which means that mobility of this kind can only be allowed in the ad hoc network's "own" AS.

This means that all GWs that the ad hoc network is attached to must be part of the same AS. If not, we must make it so in practice with the help of virtual links. That is, if we have an external link using a service provider outside our control, we must tunnel traffic to a point where we control the internal IP routing, preferably physically close to the ad hoc network.

Further, non-hierarchical IP routing must take place in all routers that will be affected by the different possible placements of the ad hoc network. So what does that mean? Normal routing tries to match as long a part of the IP address as possible when routing. The longest match determines which direction to send the packet to. In order to minimise the number of routing entries that needs to be updated, sets of IP addresses are aggregated using a common prefix, i.e. the first part of the IP address is identical, so only this prefix needs to be sent. This technique is then hierarchically used in several steps, giving us an increasingly shorter prefix.

We cannot break this hierarchical aggregation at the top levels, as this would mean that we would be going outside the present AS. But within our controlled AS, the lowest aggregation levels can be changed.

As long as the next hop for a packet with destination within the ad hoc network is equal, independent of which GW the ad hoc node is closest too (or at least using), the routers do not need full-length address routes to each node. However, if we are so close to the GWs (topologically) that, depending on which GW the node is presently attached to, the router needs to make different decisions, it will need to handle full length addresses for all ad hoc nodes that are attached. That is, this set of routers is not only the GWs but also the routers that split traffic flows to the different GWs the ad hoc nodes can be attached to.

All such routers need to be updated with information on which GW is to be used at a given time for each ad hoc node. Notice, this does not mean that the GW used must be the best GW for a specific node. Rather, it is up to the routing protocols to determine when to update routers in the fixed network about new routes. But if the node is not available through the chosen GW (or there is a better one), it is up to the GW to handle the final delivery of the packet (for example through tunnelling to a more appropriate GW). It is even possible to aggregate all traffic to a single GW and let this one handle all traffic to where the ad hoc nodes are that that time, although this might not be an efficient solution. In such a case only the main GW needs to be updated on where all nodes are. However, changes of gateways and new or lost GWs must constantly be sent to the main GW (which after a while might not even have a connection of its own to the ad hoc network).

So what problems does this lead to? IP routing protocols used in the fixed network need to update all the routers in the area affected by the mobility of the ad hoc network (in terms of attachment to the GWs). Normally, updates to the routing tables do not need to occur especially often, especially if links between servers are constant and those are online most of the time. In the ad hoc network case, updates and changes are much more common, and many of these need to be spread to the entire affected area.

Let us now specifically discuss the four different scenarios and see what requirements they have on the case in which we retain the IP address.

2.5.1 Multiple Gateways

In this case we have a single ad hoc network with access to the wired infrastructure at several points. Since we are using a fixed IP address it is up to the GWs to determine which of these advertise the node's presence out to the rest of the network. As previously stated it does not need to be the closest, but that might be most efficient, because otherwise packets need to take a detour via this GW (inside or outside the ad hoc network depending on routing strategy). Routing inside the ad hoc network can use all traditional methods for ad hoc network routing. In fact this is the scenario which makes such routing the most simple, due to the fact that the nodes know which addresses are network-local and which need to be sent through a GW.

Due to node movement, the GW that is closest changes. However, updates of this information are not really critical. As previously stated, it can at worst lead to unnecessarily long routes. This also means that the fixed network that interconnects the GW need not be updated as quickly as is required inside the ad hoc network regarding the positions of nodes (or rather which GWs they are reachable through). Notice also that this fixed network can be virtual if, for example, the GWs are using different access technology.

2.5.2 Network Split

Now things are getting much more difficult. In the previous scenario, the distance to the GW would increase, but a path would still exist. In this case the path to the advertising GW can break, which means that until a new GW takes over that role, no communication from the outside is possible, not even such initiated by the ad hoc node.

Consequently, in this case reaction needs to be swifter. However, if both the GWs involved have detected the situation, the old GW can reroute packets through the new one until the rest of the fixed network is updated on the situation.

2.5.3 Changing Gateway

In the next scenario the available GW is changed from one to another. The most difficult scenario is if the old GW was of higher capacity than the new GW (specifically if the difference was so large that the new GW was not used at all previously) so that if it is lost for some reason, there is no time for a planned hand-over. In this case the new GW must be able to set up connections to the rest of the fixed network on the fly. This can probably be done in several ways but if we want to avoid long disconnections and limit overhead in the fixed network, we might need to tunnel network traffic to and from the parts of the fixed network where the network was previously attached.

2.5.4 Connecting to a Gateway

If the network has been disconnected for a while, and a GW becomes available, that GW needs to advertise the presence of the ad hoc network to the fixed network in a similar way as above. As the network has been disconnected there are no sessions in progress, which means that the time requirements are easier to handle than in the case of changing GWs.

2.5.5 Concluding Remarks on Fixed IP Address

By keeping the IP address fixed we concentrate all problems on the actual routing layer. The IP address will represent identity (or at least a specific interface) of a node and does not really help much for the routing in itself.

It does, however, make routing much more difficult, especially by creating scalability problems, as the routing information that needs to be transmitted in the fixed network will not only increase with the number of mobile nodes, but also with the number of potential GWs that can be used. GWs that are connected to other networks, i.e. networks outside our direct control, cannot be used directly. Instead they need to be brought into our network with tunnelling techniques, thereby making them virtually part of our controlled network. However, this will most likely need manual interaction to be possible, and generate additional overhead.

It will also require considerable pre-planning and possibly new software and hardware in the routers of the fixed network. For large networks, we may even need a new routing protocol for the fixed network, or at least an upgrade of the existing protocol, in order to make it efficient.

2.5.6 Mobile IP - More or Less Fixed IP Address

In the above description we assumed that a node would always keep its IP address and the network would be updated to handle this. On a global scale this is not really possible due to scalability issues, although pre-planning can bring it into a local scale and thereby make it solvable (although that does not necessarily mean efficient). But as a general solution it has not been seen as feasible by the Internet community. The protocol that has been developed and standardized by IETF is Mobile IP [6]. In Mobile IP, the mobile node retains its IP address with the help of two (for IPv4 at least) agents that keep track of the position of a node – the home agent and the foreign agent.

The home agent is situated in the mobile node's home network, or more specifically, in the network in which the mobile node's IP address should be. At this position it intercepts packet for the node and tunnels them to the existing foreign agent, which is an entity in the network where the mobile is currently situated. The foreign agent then sends the packet to the mobile node. The existence of a foreign agent is not completely necessary because this function could be handled by the mobile node if it is capable of obtaining a new IP address, something that is assumed in IPv6.

For mobile IP to work the home address must always be reachable; if it cannot, the nodes cannot communicate. This home address must be hierarchically correct, which means that this address node needs to be in the fixed network. This can create certain problems when two nodes want to communicate inside the ad hoc network as the nodes need to be able to determine whether the other node is inside the present subnet or not.

The static address used by an ad hoc node will point towards the home location for the node, which will be part of the fixed network as seen from the other nodes. This can be handled by letting the ad hoc network prefer routing internally. However, this is not so good either because we cannot take shortcuts through the fixed network, which could be preferred for large networks.

Another problem is that all nodes update separately from each other (if we follow the present standard), and this can mean very many required updates, especially if we switch GWs for a large part of the network. Further, where should the foreign agent be put? The GW is probably the most obvious solution, but how do we handle changes of GW. For IPv6 we have no foreign agent, but then the node needs to acquire a new IP address that describes its current location, which really does not simplify the problem. In this case, many problems are similar to those encountered for variable IP addresses.

2.6 Assumption 2: Variable IP-address

The second solution to investigate is to change the node's IP address according to where it is attached to the fixed Internet. In this case, IP routing is simple because ordinary methods can be used everywhere. All ad hoc nodes will be able to attach anywhere (at least from IP routing point of view) as they follow the hierarchical address structure of the Internet. Virtual tunnels between the GWs are not needed, and cooperation between all GWs are not absolutely necessary (They would still be useful, though. We will discuss this later). It also would require less network planning because if the network can handle changes of IP address, it can most likely handle initialization of the network with similar techniques.

However, changing the IP address does not really solve the major problems; it simply moves them to other layers, mainly upward in the protocol stack. In addition, these problems are both inside and outside the ad hoc network, whereas fixed addresses mainly caused problems at GWs and outside the ad hoc network. We now need to solve some internal problems as well, which may increase overhead traffic.

First, how does a node get a new IP address? As it needs to be hierarchically correct, it must correspond to a network prefix of a preferred GW. Network addresses could potentially be supplied by the GW, but due to mobility and split networks, it would be difficult to track the addresses that are already in use. If we use IPv6, it would be possible to self-configure an IP address, but existing suggestions for doing this have security issues that need to be resolved [7].

If we have several GWs (and we would like that in case of network splits), which GW should a node connect to? The nearest? Should we then change GW when another GW is closer, resulting in a potentially expensive address change. If we do not change it, external network traffic may arrive at a GW further away, which will give longer routes in the network and therefore lower network capacity.

Then we have the problem of how other nodes find out which IP address is currently being used by the ad hoc network node. The unwanted pre planning is not even possible anymore. Traditional methods of using DNS are problematic because such protocols are not at all designed for the rapid changes that can occur. Dynamic DNS allows updates of the DNS data base automatically, although whether or not this could be done quickly enough is unclear. Even if it is possible, session mobility is not solved because applications still use IP addresses as identification. The total size of the network will also have an impact on overhead.

Further, to make it even more complicated, we sometimes also have stand-alone ad hoc networks, where no GW at all is available. Which IP address should be used here and, perhaps more important, how does a node know which nodes it can communicate with and which services are available at a given time.

Below, we discuss the specific scenarios.

2.6.1 Multiple Gateways

In the case of multiple GWs into the ad hoc network, the ad hoc node need to get an IP address corresponding to the preferred GW, either by self-configuration or by receiving one from the GW. Depending on the method we have different problems; self-configuration is very difficult in IPv4 due to the limited amount of addresses (for example, just testing one to see if it works involves a great risk of collisions). It is more viable for IPv6 because we have many more addresses, but it can give rise to security issues. No one owns an IP address in such a case which can be exploited by an intruding node as such a node can claim to own all addresses.

Letting the GW handle the assignment of addresses simplifies the above problem but instead introduces the problem of how to return the IP addresses to the GW. An ad hoc network node can disappear from the network very suddenly, and it would be up to the GW to determine when an IP address can be reused for another node. This might not be a major problem if the number of available addresses is large compared with the expected number of nodes, however. But with variable addresses, a GW would need to keep track of which nodes can be reached at any moment. However, the military scenarios we envision for this would need similar information anyway, e.g. positioning information, so it can probably be dealt with.

2.6.2 Network Split

If a network splits into two parts, an IP address from a new GW needs to be acquired quickly (assuming that the previously used address belongs to a GW in the wrong part of the network) in the same way as described above. In this case it is unlikely that the old address will be returned to the old GW in a controlled fashion so this must be handled somehow, e.g. through a timeout.

2.6.3 Changing Gateway

If we change GW unexpectedly, it will be similar to the case we described for network split.

2.6.4 Connecting to a Gateway

In this case the network is stand-alone and one or several nodes get access to a GW (or one becomes a GW, depending on situation). The first issue we have previously mentioned is which IP address a node should have if it is not connected to a GW. It has to have some IP address for communication, but, for a stand-alone ad hoc network this is just one issue regarding higher levels; service discovery, DNS and other problems need to be solved. Any subset of the nodes can form the stand-alone network, so in order for these to function well the problems need to be solved based on that fact. One functional solution might be a pre-assigned fixed address that the nodes use only when they do not have access to the fixed network.

When the network gets a GW, this node now needs to advertise a prefix to the network and possibly also assign IP addresses.

2.6.5 Concluding Remarks on Variable IP Address

By changing IP addresses we retain the hierarchical structure of addressing in the Internet, which means that IP routing will generally function as it does today. The scalability of the network is primarily dependent on the number of mobile nodes and not on where they attach to the fixed network. Planning should be much less of an issue than if we retain IP addresses.

However, this solution does not in any way solve the fundamental problem of how we know where a mobile node is at a particular time. We only move it to another layer; with a variable IP address it is easy to route packets to a node if we have the correct IP address, but it is up to higher layers to know which one should be used at a specific moment. Present implementations of DNS are probably not fast enough to handle this.

In addition, as previously mentioned, changing IP address causes problems for all applications that use IP addresses as their identity, viz. other solutions might be developed which use URLs instead, and every time an IP address is changed the session is renegotiated, e.g. by using SIP [8]. However, this would either require a substantial change in how the Internet is built or all applications would have to be specifically designed.

2.7 Fixed or Variable IP Address?

So which solutions should we choose? At the moment too much is undecided in the Internet and ad hoc networking world to give a definite answer.

Using fixed addresses is only possible for a limited number of nodes whose position in the network is fairly predictable. Such a solution might require a large amount of pre planning, and it might prevent us from quickly reacting to new events if the pre planned set of GWs is not enough. On the other hand, such planning is generally needed in military campaigns today, anyway, and might still be needed in the foreseeable future. Further, it does not limit the self-configurative properties inside the ad hoc network, only in its interaction with other networks. Already today it could be a practical solution in small scenarios with limited mobility. Nevertheless, it is a solution that, in the long term, will limit the full potential of ad hoc networks.

Chapter 3

Quality of Service

Quality of Service (QoS) is another important subject for military networks. Discussed in many situations, people generally state that it is important or even necessary for military networks. However, but although the term QoS is often used, it is more seldom properly defined. One definition that is usual (in the few cases it is actually defined) is that it is a contract between network and application that states how the applications packets will be handled by the network. Usually this means that the application guarantees to deliver a determined (or maximum) amount of packets within a specified time. Note here that this sets bounds on the application as well, not only on the network. If the application inserts too many packets, guarantees cannot be honoured, and in such cases packets are usually treated as "best effort".

It is also important to note that applications might need QoS for several reasons. In general, we can divide them into two categories: technical and importance. Importance is probably the easiest to explain, especially in military networks. It is probably more important that warnings and direct orders reach the destination than a Word document with today's menu, although the latter may be considered important as well. For civilian network providers, importance can be connected to how much people have paid. Although this is still not commonly used, it would be a simple way of differentiating the service to different customers.

Technical reasons are somewhat more vague. Even a voice session with low priority will need some delay guarantees on packets due to the fact that humans dislike long (and different) sound delays when speaking to each other, whereas a few extra seconds delay on a written order may not create a significant problem. For this reason packets with information of lower importance may sometimes need to be sent before packets of higher importance.

One thing to notice here is that QoS becomes an especially significant issue when we have different kinds of services sharing the same network, especially if the traffic to be transported is of different degrees of importance. As we will see later, some of the specific properties of ad hoc networks make this scenario even more complicated.

3.1 QoS in the Internet and Other Fixed Networks

Guaranteeing anything in a network is not that simple. Applications are independent of each other, and the network can usually only offer statistical predictions about incoming traffic flows. If users are going to view a network as useful, it must work for most of the time they try to use it. However, failure to function will usually affect services that require some guarantees first, such as voice, as they are disturbed more easily if the network becomes heavily loaded. Less sensitive services have a tendency to work anyway; if an e-mail is delayed some extra minutes due to queuing, it matters less than a minute's delay of voice packets.

Fixed networks have at least two positive properties in terms of QoS that wireless networks lack. First, capacity is cheap. The expensive part in wired networking is the deployment of the wire, not the cost of the wire itself. It does not cost much to double the link capacity if it is done when the network is built.

Second, capacity is fixed on the links. A node knows how many packets it can transmit on each of its outgoing links. It is not dependent on the traffic load in other parts of the network (at least this is normally the case).

For these reasons, over-provisioning is the solution used in fixed networks. It it is cheaper to add so much capacity that all applications will work for "most" of the time statistically than to add complex mechanisms that attempt to determine which services are less important. In addition it keeps users happy (because all applications usually work).

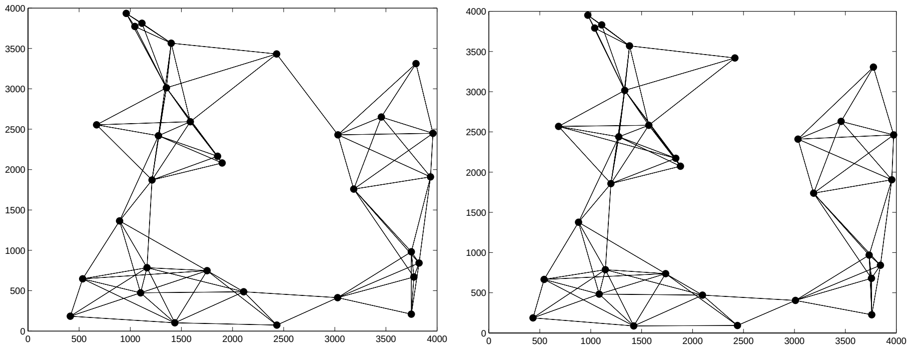


Figure 3.1: The figure shows the network at time instance 17 seconds (left) and 18 seconds (right).

3.2 Wireless Networks and Ad hoc Networks

In wireless networks the first of these assumptions - that capacity is cheap - is no longer true (and the second only sometimes). Adding more capacity to links requires either more complex systems and/or more frequency bandwidth. Although the radio spectrum might not be very heavily utilized everywhere, more bandwidth can only be obtained to a limited extent. Similarly, more complex systems can only give limited gains. For cellular systems, capacity can be increased by adding more base stations, but that is expensive, and pure overprovisioning requires considerable additional capacity if it is to work. In cellular networks, only a single hop is wireless, which means that link capacity is at least predictable. The capacity for a new session can be centrally controlled by the base station, and new sessions, e.g. phone calls, can be admitted or denied based on information in the base station.

For ad hoc networks, the equivalent to adding more base stations is not really comparable, because adding more units means that there are also more units to share the available bandwidth. An additional problem with respect to QoS in ad hoc networks is capacity estimation, which can be illustrated in Figure 3.1. These are two time instances for a mobile network. At one time, the network is circle shaped; then the upper part splits and acquires the shape of a "U". The consequence of this split is that all traffic between the top left and the top right side of the network needs to be relayed much further than before, which significantly decreases the capacity of the network.

Several aspects affect how much traffic an ad hoc network can handle. Resources can be reused when the network becomes large, but if traffic is not local, it must also be relayed more times, which consumes resources. It has been shown, see e.g. [9], that ad hoc networks are not scalable. The available capacity per user decreases with network size, which suggests that large ad hoc networks probably need the support of fixed networks for them to function. The size of "large" is, however, more unclear and depends on the applications, traffic patters and algorithms used.

The consequences of this is that not only is over-provisioning difficult in an ad hoc-network, but techniques for QoS are also more difficult to implement than for fixed networks and other wireless networks. Even if delay guarantees can be given for a service at a particular time, this may be impossible to uphold later due to changes in topology, longer routes or because the network may even become disconnected. Note that this does not mean that QoS is impossible, merely difficult. A common way to provide a semblance of QoS for ad hoc networks is a relative priority setting, such as DiffServ. That is, the network does not guarantee that all traffic of a service will be delivered, only that certain packets will be prioritized more highly than others.

Although the sharing of bandwidth over the nodes in the network is the main reason why capacity is limited and over-provisioning is difficult and expensive, it also makes an ad hoc network adaptive in the sense that resources can be moved from one point to another when needed. In a time-slotted system, more transmission time can be given to bottleneck nodes to compensate them for carrying more traffic. When mobility changes which nodes are bottlenecks, more time slots can quickly be given to the nodes that would otherwise be the new bottlenecks. However, this requires both knowledge of traffic flows and the capacity to control network resources efficiently, which is still at research level in the more general setting.

3.3 QoS Mechanisms

The QoS control issue is complex and involves all networking layers from application to the physical layer. At the upper layers, context aware applications, or adaptive applications, could be applied. Moreover, the ability to store non-delay-sensitive data in intermediate nodes is an attractive feature in many situations, see disruptive tolerant networks [10]. Thereafter the traffic management protocols become important. To avoid overloading, some sort of network admission control has to be implemented. Furthermore, it might be necessary to pre-empt lower priority connections when traffic with very high priority, e.g., nuclear, biological or chemical alerts, is present.

Related to the services is the source coding. Adaptive and efficient source coding for wireless networks should, for example, have the ability to deliver graceful degradation. An example of this is a video transfer: as long as the connection is good, it has high resolution, but when the connection degrades, only the most important parts in the video picture are transmitted in high resolution. Graceful degradation can to some extent be provided by clever source and channel coding independently of each other, but many cases, such as the video example, require interaction between non-adjacent network layers because the service needs to know the quality of the connection.

Seamless communication means primarily that IP addressing needs to be supported. However, assuming that the IP QoS protocols are used, then how well they perform over wireless links will determine how our heterogeneous network handles QoS. QoS IP protocols exist, e.g., Differentiated Services and Integrated Services, and others are under considerations. Furthermore, the connection control protocols, such as TCP, may need to be modified to better allow for QoS.

QoS-aware routing can have a great impact on overall performance. Large heterogeneous networks will have a hierarchical structure. The global routing is handled by the standard IP routing protocols (OSPF, etc.). However, for a particular tactical ad hoc network, subnetwork or network domain, it may be clever to apply other and more local routing protocols. That is, we will have routing at different levels, and it is important to set up a "good" overall route between the end users. Good means that we should set up a route that takes the QoS requirements into account.

Much can be done at the link layer. The queuing system is important, and letting the MAC protocol be aware of QoS parameters makes it possible to be able to control the crucial traffic flows more locally. The MAC layer controls the channel resource and determines which node and which traffic flow should have access to the channel at a given time. Clearly, also at the lowest levels, much can be done to improve performance. Here, we control the quality of particular links. By selecting frequency band, adaptive modulation, smart antennas, power control and multi-user detection we have the ability to adapt the links to the service requirements. Furthermore, we can control the overall network connectivity.

Next we will further describe some of these QoS mechanisms. We focus on queuing systems and traffic management, especially admission control and traffic estimation. For routing and the link layer, see [11, 12]. Furthermore, the IP QoS protocols have been analyzed in [13]. Adaptive services have been studied in [14].

3.4 Queuing Systems

When a node generates traffic, or receives relay traffic, at a higher rate than it can transmit, a *queue* is formed. The packets in the queue might originate from different traffic flows, so there is clearly a need for a mechanism that distributes the node´s capacity between the traffic flows.

One of the commonest and most intuitive ways to deal with those queues is the *first-come-first-serve* (FCFS) discipline, by which the packets are served in the order of their arrival. However, with this system a bursty flow can take up most of the resources during long periods and in effect block out other flows from using the network.

Another common scheme is the *packet based round robin* (PBRR). Here, one packet from each flow is transmitted in turn if it has a packet in queue. This system works fairly well if all flows have equally sized packets. However, if one flow uses longer packets than the others, it will take more of the resources from the other flows.

It is clear that we need more elaborate queuing schemes, with better control over the queue's behaviour. An entire family of queuing disciplines is the priority queues, in which the packets are given differentiated treatment according to which service class they belong to. In priority queues the packets are assigned a priority that is a function of service class and possibly other parameters, e.g. number of hops left, time-to-live, etc. Then the packets are served in decreasing order of priority.

The *fixed priority queuing* (FPQ) discipline is probably the simplest form of priority queues. As the name suggests, the packets are assigned a fixed priority according to service class membership, i.e. if the packet belongs to service class c, it is assigned the priority q_c. There is no sense of fairness in this strategy because packets that belong to the service class with the highest priority are always served first. Hence, packets that do not belong to that service class are not guaranteed any service at all; they are merely given what is left after the highest priority class has been served.

A theoretical system that gives perfect fairness is *generalized-processor sharing scheme* (GPS) [15]. The GPS queuing scheme has a very attractive property. One can allocate a specific percentage of the total system capacity to a service class. Further, if some service classes do not utilize their full share, the excess capacity is fairly shared between those classes that need it. Thus, every service class is guaranteed a minimum service rate, but they may experience a better service rate if the system is not fully utilized. GPS is a flow-based scheme that serves multiple service classes synchronously at a fixed rate r so it cannot be implemented in a packet based system. However, there exists packet based approximations of GPS, e.g. *packet-by-packet generalized-processor sharing* (PGPS) [15] and *Weighted fair queuing* (WFQ) [16].

Priority queues only solve the problem of differentiating between traffic flows locally within a node. But this is not only a local problem, as shown in Figure 3.2. In this case, we have a high priority flow that is relayed through node A and a low priority flow that is relayed through node B. In this situation, even with priority queuing, the low priority flow will get as good service as the high priority flow if nodes are assigned channel resources proportional to the traffic load on the node.

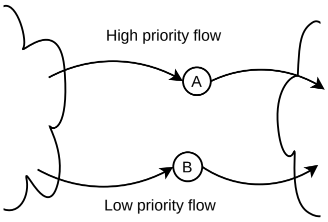


Figure 3.2: The problem of differentiating between flows is not a local problem.

3.5 Traffic Management

By traffic management we mean policy and admission control, and the support mechanism to get admission control to work properly. Policy control is the application of rules to determine whether or not access to a particular resource, in this case the network, should be granted, see [17]. This is the first step to decide whether a user is at all allowed to access the network with a given request. The criteria for policy control include identifying users and applications or identifying traffic based on how, when, and where it enters the network. Next, admission control decides whether a permitted traffic session that tries to access the network can be served, i.e., a mechanism is required that can estimate the level of QoS that a new user session will need and whether there is enough bandwidth available to service that session. If bandwidth is available, the session is admitted. Thus, there is a need to estimate the present traffic and the capacity in the network. Whether to reserve bandwidth to the admitted session, and in that case how to do it, are other issues to consider.

3.5.1 Admission Control

In order to use the resources efficiently it is important to drop packets that cannot be delivered in time as early as possible in a path. It is a serious waste of resources if a packet is dropped at the last hop; all the network resources used at previous hops have been wasted. Preferably, dropping would be done even before the first hop when a packet enters the network. The mechanism to handle this and to prevent congestion is called *admission control*, and is a process that determines whether there are sufficient resources in the network to allow the traffic flow. If there is not, the traffic will be dropped or downgraded to a lower (or no) QoS grade depending on the situation and traffic scenario. Admission control is especially important if we have interconnected networks as described in the four scenarios in Chapter 1.

Setting up the rules for whether at all, and secondly how, to serve different traffic flows can be a very tricky task. Many issues have to be considered. How many QoS classes do we need? Both the importance of the traffic and technical requirements have to be taken into account. Next we need to decide on the rules for strict precedence (pre-emption), i.e., when an important traffic flow cannot be served will it then be allowed to abort other flows of lower importance, and how should the rules for that be set up? Fairness, which was briefly mentioned earlier, is also a difficult issue. Because the cost in terms of network resources for a traffic flow depends on its location in the network, and can vary considerably, there is trade-off between serving many flows at good locations and few flows at bad locations. Furthermore, is it more important to serve one flow at a given QoS class than two flows at the next lower QoS class? A more general issue is what the admission control mechanism should be allowed to control, e.g., should it be able to redirect network resources by interactions with the MAC protocol so it assigns more time slots to a crucial link in order to admit a high prioritized flow? Finally, let us point out that admission control is needed to deal with QoS, but it will also increase system complexity, introduce delay and add to overhead traffic. The question is how sophisticated feasible admission control mechanisms can really be. For admission control to work at all, however, we need at least some rough estimates of the traffic and the capacity in the network.

3.5.2 Measurements and Evaluation of Traffic and Capacity

As the capacity of ad hoc networks is highly variable over time, it is important to be able to determine existing capacity and preferably also to estimate possible future values so that the admission control does not allow too much of a certain priority grade into the network. Due to the changing available capacity, only a part of the currently available capacity should normally be used for QoS traffic. The rest of the available capacity can be used to send best effort traffic.

This leaves us with the problem of estimating how much traffic the network can handle. So far we have used the term "capacity" to describe this, which from an information theory point is highly incorrect. The problem is that, unlike a link, we cannot directly put a value on how much traffic a network can handle. We are interested in how much end-to-end traffic the network can handle, not really how many bits the network internally sends over its links. If the arriving traffic flows needs to be relayed many times in the network, available end-to-end throughput will be much lower than if two neighbouring nodes communicate. A consequence of this is that the link data rates and which links exist are not sufficient for estimating the available capacity in the network; the input traffic pattern is also relevant. However, this traffic pattern is also dependent on how well the network can handle application demands, which thereby further complicates the problem. Estimation of available capacity in an ad hoc network is generally very complex. However, a simplified problem is to determine whether it is possible to handle a new traffic flow between two nodes simultaneously with the existing flows. The real problem is then to predict how likely it is that the network can handle the service while it still lasts despite the effects mobility will have. New traffic flows can be handled in the same manner or be reduced to best-effort traffic flows by admission control. More highly prioritized services arriving later may be more of a problem, though.

At present, there is no really good way of measuring the capacity of a network. A commonly used measure is the *uniform capacity*, which is the highest rate any node can send with if all nodes send an equal amount of packets to all other nodes in the network. Uniform capacity has certain advantages as a result of which it is commonly used. First, it gives a single value for the network capacity, not complex functions or plots that are difficult to evaluate. Second, it is the fairest measure; all nodes get exactly the same end-to-end capacity, which fits well with the statement that there should be no special nodes in an ad hoc network. All nodes should be equal. In reality this will not be the case, of course. If a node fails to communicate, the network should still be able to function. However, traffic patterns will be different in the nodes, and uniform capacity only gives us one picture. It can still be useful in some cases when we are comparing two different solutions, but as we will see in the next section, when introducing QoS, it can give very strange results that are not normally desirable.

3.5.3 Variable Link Capacity Makes the Problem Even More Difficult

From the above it is easy to see that estimation of capacity (and traffic loads) is not simple. But even if it is achieved, the division of capacity over the different links in the network is not always obvious. Giving more highly prioritized traffic more available capacity (mainly through additional time slots) is easy to suggest. The question that is more difficult to answer is how much more capacity. Say that a network wants to handle two different services, a highly prioritized service and a rather highly (but lower) prioritized service. In this case also the less highly prioritized service is relevant; we cannot merely ignore it. A simple (at least in theory) attempt is to give the more highly prioritized service sufficient capacity for it to work in all nodes. The other service will get the rest of the available capacity, which probably means that it will fail in many cases. If we have a fixed (and equal) capacity on all links, this is usually a plausible solution. One problem might be long routes. A traffic flow whose packets are relayed many times is more expensive to uphold (in terms of network resources) than a one hop service. This means that we can sometimes choose between servicing one multi-hop traffic flow and servicing several single-hop flows. From a fairness point of view, users should not be put at a disadvantage due to their position; on the other hand the total performance of the network might sometimes be better if the total number of network flows increases.

To get perspective, we will study a more extreme version of this problem that occurs if we have variable rates on the links in the network. Due to different terrain and distance between nodes in an ad hoc network, we can potentially obtain very different data rates on the links. Two nodes in line of sight close to each other can uphold much higher data rates than can two nodes at long distance shaded by a hill. The ability to handle variable rates is a technique that has become more interesting lately due not only to being able to increase the network capacity but also being able to keep the network connected when nodes get separated. The downside is not only complexity; it can also make QoS more difficult. The problem is that the range of data rates can be large. Sometimes the difference between the highest and lowest data rate can be tens, or even hundreds, of times the size. The cost of transmitting traffic over the low-rate links will simply be much higher, in terms of network resources, than transmitting traffic over the high-rate links. This is specifically evident in terms of uniform capacity, where everybody is given equal capacity. Assume we have a single node that due to terrain only has a very poor link into the rest of the network, which we assume is very well connected (with high link data rates). Now, the cost of giving this node the same share of the capacity as the others means that we may have to give this node a very large portion of the time slots to compensate for its lower data rate, thereby considerably decreasing the capacity of the rest of the network.

Now returning to the QoS problem, what if this node had a highly prioritized service, and the rest of the nodes only had the less highly prioritized services. In this kind of scenario, we might have to determine whether the prioritized service of this badly positioned node is actually more important than all the other nodes' less prioritized services. Its not one service versus another, which is easier to deal with. Rather we might end up prioritizing between one service versus many, or even one service versus all others.

Tools for making more advanced priorities are lacking and their development is not at all self-evident.

3.5.4 Architectural Issues

A main architectural issue is where to locate the traffic management functionality, and how different units with specific tasks should interact. It is desirable to distribute the traffic management functionality to all nodes so they can make decisions locally without time-consuming interaction with other nodes, but this may be difficult to realize in practice. Let us call a node that carries out the traffic management functionality a QoS manager.

In general terms, a QoS manager monitors and estimates the total aggregated traffic in the network, or domain, for which it is responsible. Based on that estimate and what total network resources (capacity) are available, it derives a new estimate of the remaining network capacity that can be used for additional traffic. The manager then uses this estimate when it negotiates with other managers about how to set up new data flows. Nevertheless, the management software could be available in all nodes, so that in principle all nodes could be QoS managers. However, they are only called QoS managers when they carry out the QoS management tasks. This means that local admission control functionality can be implemented in a node without it being a QoS manager.

In a heterogeneous network one choice could be to place the QoS manger in the boundaries between different networks, or network domains. Thus, in the scenarios depicted in Section 1.3 the gateways would also be the QoS mangers. However, due to mobility the gateways may change, and a whole network may be difficult to monitor by one manager. Also, is it better to locate a manager in the centre of the domain than in the boundary? There are many other issues to consider when selecting the manager. Dependent on nodes properties such as location, mobility, connectivity, power supply, transmission and processing capabilities, etc., some nodes will be more suitable in the role as manager than others.

When the network grows, it can be divided into sub-nets or domains with a QoS manager for each domain. The nodes forming a domain are preferably chosen so that it is likely that the domain stays connected for some time. For example, when fast-moving nodes are passing a domain consisting of stationary nodes, they should not be integrated into that domain. How large a domain should be is an open issue.

To estimate capacity the first approach is to use routing information. A proactive routing protocol tries to have an updated table about the network, including the links between nodes and, in some cases, also the link data rates. Note, however, that this does not mean that the capacity can be calculated. To do that, link layer information is needed, e.g. how the MAC has assigned time slots. However, from the routing protocol's information about the link data rates we may be able to get a rough estimate about the capacity in different parts of the network by making assumptions about the link layer and input traffic.

3.6 Concluding Remarks

For fixed networks, existing traffic management and admission control solutions can be applied. For example, an industry initiative lead by CISCO has developed a set of technologies and solutions [18]. Furthermore, the IETF has published a number of RFCs on the topic [17]. Considering the network scenarios in Chapter 1, the main problem therefore becomes the wireless domain, i.e. when traffic enters an ad hoc network from a fixed network or when traffic originating in an ad hoc network needs to be admitted.

Today there is no agreed approach to admission control and bandwidth reservation in multi-hop ad hoc networks, [19]. There are two different approaches that can be used; either to implement it as separate signalling scheme on top of the routing, such as INSIGNIA [20], or to integrate it with QoS routing and MAC.

In [19] the authors point out high overhead, slow response and inefficiency as main concerns with the first approach. Inefficiency is due to not taking advantage of underlying routing and link information. The latter approach is therefore more attractive in the research community. Notice that QoS routing with admission control can be made independent of the underlying network technology, which is an advantage in heterogeneous environments, by using generic QoS measures (e.g., minimum bandwidth, maximum end-to-end delay, etc.). On the other hand, QoS routing tailored to a particular underlying access technology may perform substantially better for that technology. In TDMA-based networks, QoS routing with admission control and resource reservation mainly focuses on scheduling to make a multi-hop end-to-end route possible [21]. Nevertheless, IEEE 802.11 is the most common link and access technology today, and most research papers on QoS routing with admission control consider networks based on IEEE 802.11, see [22]. In these papers, estimation of available bandwidths and load balancing are the main topics.

In summary, the main research challenge in the traffic management area is the support mechanism to get admission control to work properly in ad hoc networks. Firstly, how do we acquire enough information to be able to make "good" admission control decisions? And secondly, if a session is admitted, how do we, with high probability, complete the session with the requested quality?

Chapter 4

Summary and Discussion

A heterogeneous network may comprise many different types of networks. Providing seamless communication in such a heterogeneous environment is a challenge.

In this report we focus on an ad hoc network connected to a fixed network, see Chapter 1. Furthermore, we have selected mobility management and the problem of assigning IP addresses and QoS management as the main topics. The problems related to these areas are discussed, and briefly analyzed and structured. Nevertheless, when making a structure of the problem area, there are many aspects to take into account.

Clearly, the relevance of the problem is important. Is it a critical problem (show-stopper) or can it be overlooked? For example, if the system does not fulfil the basic requirement, it may be useless. However, it may be acceptable that the system does not work exactly as we want in a particular situation. The magnitude of the problem and the consequences of dealing with it are also important. For example, is a new architecture needed or is a new protocol (or modification) enough to solve the problem? One slightly different approach is to take a functional perspective on the problems. That is, what is needed to get the system to work (be useful) at all? Thereafter, what is needed to get a satisfactory performance, and, finally, what fine-tuning can be added to achieve full performance?

However, we can also divide problems according to: areas where we can contribute and areas that are largely governed by de facto standards we have to adapt to. The first is a potential research area, in particular when the area is confined and well defined, whereas the second area is more suited for a study approach. This does not mean that the second problem area is less important, but it may have to be approached differently when considering a heterogeneous environment; on a very detailed level, when a particular protocol or two protocols inter-working are studied in detail (e.g., in a simulation) or on a high conceptual level. The latter involves the study of existing solutions and protocols and trying to analyze (rather than simulate) how they would work in a particular heterogeneous network setting. The problem, in most cases, is that simulating a heterogeneous network that includes all protocol layers is too complicated.

Next let us consider our two problem areas within the aforementioned context, in particular whether the area is suitable for research or whether a study approach that follows ongoing developments is more adequate. Firstly, a general conclusion is that research and development in the area of heterogeneous networks is dependent on standardizations and the de facto standards that are used. Clearly, it is the nature of heterogeneous networks that standards are needed to be able to communicate over different types of networks. However, different networks may still internally use different protocol designs as long as they can be connected through standardized interfaces and gateways.

A central problem in mobility management is the assignment of addresses to nodes that move in the network (and address) hierarchy. Using fixed IP addresses (used today) works, but requires extensive network planning and limits "free" mobility. What would be desired is variable IP addresses in order to retain the address hierarchy. However, this moves the mobility management problem to another level above routing, with uncertain consequences as a result, and it is unclear whether it would be possible to use variable IP addresses and how it would be solved. None the less, mobility management and address assignment is an important area for future studies, especially because the development of a viable variable IP address solution would substantially increase flexibility in a heterogeneous network. On the other hand, variable IP address assignment is a less suitable area for research as the solutions will be determined by the standard that evolves. In fact, address assignment is one of the areas most closely tied to the standards because the address solution needs to work everywhere in the heterogeneous network.

The QoS issue is complex and involves all networking layers, from application to the physical layer. A basic solution can be obtained by using queuing systems and traffic management, i.e. policy and admission control and their support mechanisms. However, rules on how to serve different traffic flows are also needed. Defining QoS classes, and in particular how to map applications onto QoS classes, is far from straightforward. This is only partly a networking technology issue. It also involves application developers and users. QoS solutions exist for fixed networks, and the main problem therefore becomes dealing with QoS on the boundary of, and within the ad hoc network. Today there is no agreed approach to how to handle QoS in ad hoc networks, and it is unlikely that any such approach will emerge in the near future. Instead it is more plausible that different types of ad hoc networks will have different internal QoS solutions, but that they can adapt, through boundary nodes (gateways), to the QoS standards that will be used for communications between networks.

The QoS area has to be followed in future studies, but the area also includes several suitable research topics, of which traffic management is one such topic. Traffic management includes several parts. In particular we see traffic and capacity estimation as parts that can be suitable areas for research. These parts can be well defined and confined, and can be addressed in several ways. Traffic estimation can be made generic, i.e., independent of underlying routing and access technology, or tied to a particular access technology. Furthermore, traffic estimation is required for admission control and QoS. It is likely that traffic estimation will be done differently for different types of wireless networks, but even if we consider traffic estimation for multi-hop ad hoc networks, the estimation principles for those networks could be useful in other types of wireless networks.

Finally, many exciting research challenges need to be addressed, and a range of problems have to find satisfactory solutions before true seamless communication in a heterogeneous network environment will be feasible. Traffic management and QoS are two of those challenges, but several others exist, including security.

Bibliography

[1] http://www.ietf.org.

[2] R. Moskowitz and P. Nikander, “Host identity protocol (HIP) architecture,” *RFC 4423*, 2006.

[3] C. Perkins *et al.*, “On-demand distance vector (AODV) routing,” *RFC 3561*, 2003.

[4] T. Clausen and P. Jacquet, “Optimised link state routing protocol (OLSR),” *RFC 3626*, 2003.

[5] J. Hawkinson and T. Bates, “Guidelines for creation, selection, and registration of an autonomous system (AS),” *RFC 1930*, 1996.

[6] C. Perkins, Editor, “IP mobility support,” *RFC 2002*, 1996.

[7] E. Hansson, J. Grönkvist, and J. Nilsson, “Intrångsdetektering i mobila ad hoc-nät,” Swedish Defence Research Agency., Div. of Command and Control. Linköping, Sweden, Technical Report FOI-R--1375--SE, 2005, (In Swedish).

[8] J. Rosenberg *et al.*, “SIP: Session initiation protocol,” *RFC 3261*, 2002.

[9] P. Gupta and P. R. Kumar, “The capacity of wireless networks,” *Transactions on Information Theory*, vol. 46, no. 2, pp. 308–404, mar 2000.

[10] S. Farrell *et al.*, “When TCP breaks: Delay- and disruption- tolerant networking,” *IEEE INTERNET Computing*, pp. 72–78, Aug. 2006.

[11] J. Nilsson *et al.*, “Mobile ad hoc networks - a project summary,” Command and control systems, FOI, Swedish Defence Research Agency, Tech. Rep. FOI-R–0705–SE, December 2002.

[12] ——, “Ad hoc networks - routing and MAC design,” Command and control systems, FOI, Swedish Defence Research Agency, Tech. Rep. FOI-R–1801–SE, December 2005.

[13] K. Persson, J. Grönkvist, and A. Hansson, “Guaranteed quality of service in tactical IP networks,” Command and control systems, FOI, Swedish Defence Research Agency, Tech. Rep. FOI-R–0641–SE, November 2002, (In Swedish).

[14] Adaptive & context-aware services. [Online]. Available: http://www.wireless.kth.se/AWSI/ACAS/publications.php

[15] A. K. Parekh and R. G. Gallager, “A generalized processor sharing approach to flow control in integrated services networks: The single-node case,” *IEEE/ACM Transactions on Networking*, vol. 1, no. 3, pp. 344–357, June 1993.

[16] A. Demers, S. Keshav, and S. Shenker, “Analysis and simulation of a fair queueing algorithm,” in *SIGCOMM*. ACM, 1989, pp. 3–12.

[17] A framework for policy-based admission control. [Online]. Available: http://www.ietf.org/rfc/rfc2753.txt

[18] Network admission control - CISCO systems. [Online]. Available: www.cisco.com/en/US/netsol/ns466/networking_solutions_package.html

[19] H. Zhu and I. Chlamtac, “Admission control and bandwidth reservation in multi-hop ad hoc networks,” *Computer Networks*, vol. 50, pp. 1653–1674, 2006.

[20] S.-B. Lee *et al.*, “INSIGNIA: an IP-based quality of service framework for mobile ad hoc networks,” *Journal of Parallel and Distributed Computing*, vol. 60, pp. 374–406, 2000.

[21] C. Zhu and M. S. Corson, “QoS routing for mobile ad hoc networks,” in *IEEE Infocom*, 2002, pp. 958–967, NY, USA, vol.2, June 2002.

[22] Y. Yang and R. Kravets, “Contention-aware admission control for ad hoc networks,” *IEEE Transactions on Mobile Computing*, vol. 4, no. 4, pp. 363–377, 2005.

FOI is an assignment-based authority under the Ministry of Defence. The core activities are research, method and technology development, as well as studies for the use of defence and security. The organisation employs around 1350 people of whom around 950 are researchers. This makes FOI the largest research institute in Sweden. FOI provides its customers with leading expertise in a large number of fields such as security-policy studies and analyses in defence and security, assessment of different types of threats, systems for control and management of crises, protection against and management of hazardous substances, ITsecurity and the potential of new sensors.

FOI
Swedish Defence Research Agency
P.O. Box 1165
SE-581 11 LINKÖPING

Phone: +46 13 37 80 00
Fax: +46 13 37 81 00

www.foi.se