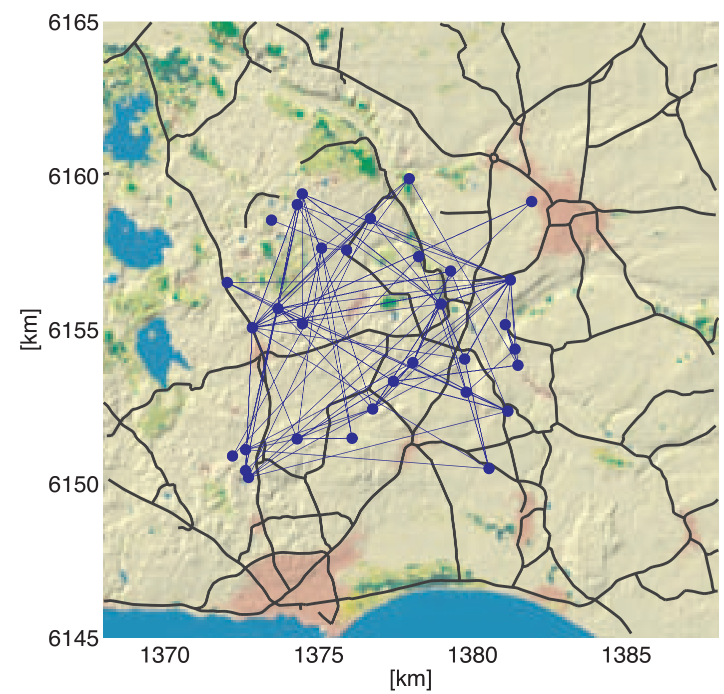
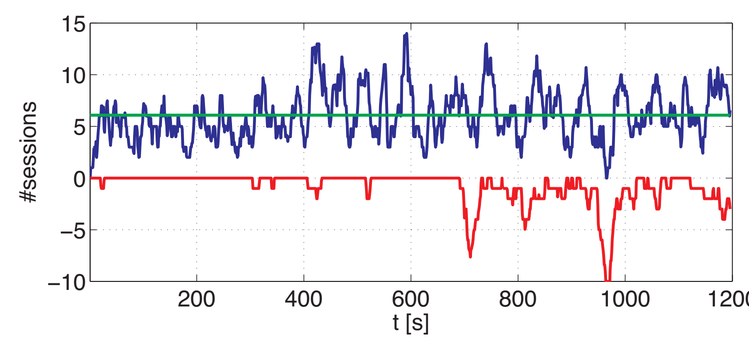
Ad Hoc Networks
- Routing and MAC design

EDITOR: JAN NILSSON





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 organization 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, IT-security an the potential of new sensors.

FOI
Swedish Defence Research Agency
P.O. Box 1165

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

www.foi.se

FOI-R--1801--SE Technical Report
ISSN 1650-1942 December 2005

Command and Control Systems

Ad Hoc Networks
- Routing and MAC design

FOI-R--1801–SE
ISSN 1650-1942

Command and Control Systems
Technical Report

December 2005

Issuing organization Swedish Defence Research Agency Command and Control Systems P.O. Box 1165 SE-581 11 LINKÖPING SWEDEN	**Report number, ISRN** FOI-R--1801–SE	**Report type** Technical Report
	Programme areas 4. C^4ISR	
	Month year December 2005	**Project No.** E7035
	Subcategories 41. C^4I	
	Subcategories 2	
Author/s Linda Farman, Jimmi Grönkvist, Anders Hansson, Erika Johansson, Jan Nilsson, Katarina Persson, Mattias Sköld, Ulf Sterner, Otto Tronarp and Pelle Zeijlon	**Project manager** Mattias Sköld	
	Approved by Sören Eriksson	
	Sponsoring agency Swedish Armed Forces	
	Scientifically and technically responsible Jan Nilsson	

Report title

Ad Hoc Networks - Routing and MAC design

Abstract

A tactical ad hoc network is an important component in future military communications. Such a network must be robust, self-forming, self-healing and be able to support different types of service requirements even in a high mobility scenario. To support Quality of Service (QoS) and high mobility careful design of the ad hoc network protocols is required.

This report investigates issues concerning the control of ad hoc networks and in particular the design of MAC and routing protocols. Furthermore, the routing and MAC protocols need to interact with each other and also with other protocol layers. When designing the network control, there is a need to understand what can be done on a single layer as well as through cross-layer interactions. Different protocols are preferred in different situations and this is one of the main topics for our investigation. Another topic has been the networking benefits of using variable data rates. This report summarizes the three-year long Heterogeneous Ad Hoc Network project.

Keywords

ad hoc networks, MAC, routing, variable data rate, quality of service, QoS

Further bibliographic information	**Language** English
ISSN 1650-1942	**Pages** 72 p.
	Price acc. to pricelist

<table>
<tr><td rowspan="6">Utgivare
Totalförsvarets Forskningsinstitut
Ledningssystem
Box 1165
SE-581 11 LINKÖPING</td><td>Rapportnummer, ISRN
FOI-R--1801–SE</td><td>Klassificering
Teknisk rapport</td></tr>
<tr><td colspan="2">Forskningsområde
4. Spaning och ledning</td></tr>
<tr><td>Månad, år
December 2005</td><td>Projektnummer
E7035</td></tr>
<tr><td colspan="2">Delområde
41. Ledning med samband och telekom och IT-system</td></tr>
<tr><td colspan="2">Delområde 2</td></tr>
</table>

<table>
<tr><td rowspan="4">Författare
Linda Farman, Jimmi Grönkvist, Anders Hansson, Erika Johansson, Jan Nilsson, Katarina Persson, Mattias Sköld, Ulf Sterner, Otto Tronarp och Pelle Zeijlon</td><td>Projektledare
Mattias Sköld</td></tr>
<tr><td>Godkänd av
Sören Eriksson</td></tr>
<tr><td>Uppdragsgivare/kundbeteckning
Försvarsmakten</td></tr>
<tr><td>Teknisk och/eller vetenskapligt ansvarig
Jan Nilsson</td></tr>
</table>

Rapportens titel

Ad hoc-nät - Routing och MAC design

Sammanfattning

Ett taktiskt ad hoc-nät är en viktig komponent i försvarets framtida kommunikationsarkitektur. Ett sådant nät måste vara robust, självkonfigurerande och självläkande och kunna tillhandahålla tillräcklig tjänstekvalitet för olika typer av tjänster också i ett scenario med hög mobilitet. För att uppnå tillräcklig tjänstekvalitet i ett scenario med hög mobilitet krävs en noggrann protokolldesign.

I rapporten undersöks frågeställningar avseende styrning av ad hoc-nät, speciellt undersöks design av MAC och routingprotokoll. Routing- och MAC-lagret behöver dessutom samverka, både inbördes och med andra protokolllager. Det finns ett behov av att både förstå vad som kan göras i ett specifikt lager och att förstå hur samverkan mellan lager kan ske effektivt. Olika protokoll är att föredra beroende på situation. Att undersöka detta närmare har varit ett av huvudämnena i rapporten. Ett annat ämne har varit att undersöka fördelarna, ur ett nätperspektiv, med att använda variabel datatakt. Denna rapport sammanfattar tre års verksamhet inom projektet heterogena ad hoc-nät.

Nyckelord

ad hoc nät, MAC, routing, variabel datatakt, tjänstekvalitet, QoS

Övriga bibliografiska uppgifter	**Språk** Engelska
ISSN 1650-1942	**Antal sidor:** 72 s.
Distribution enligt missiv	**Pris: Enligt prislista**

Foreword

This report summarizes the three-year long Heterogeneous Ad Hoc Network project. The report starts in Chapter 2 with an introductory part. The results from the research in the area of Medium Access Control (MAC) are presented in Chapter 2 and 3, where the latter chapter includes, previously unpublished, results from a comparision of two different MAC alternatives. Chapter 4 treats routing and Chapter 5 investigates the networking advantages of utilizing adaptive radio nodes. Finally, Chapter 6 contains the conclusions.

With the exception of Chapter 3, more information about our work and results can be found in the previously published project papers [1–29]. Furthermore, a summary of the project, in Swedish, discussing the topics from a user perspective can be found in [30].

Contents

Chapter 1

Introduction

An ad hoc network is a collection of wireless mobile nodes that dynamically form a temporary network without the need for any pre-existing network infrastructure or centralized administration. Due to the limited transmission range of radio interfaces, multiple network "hops" may be needed for one node to exchange data across the network with another. An ad hoc network is both self-forming and self-healing and it can thus be deployed with minimal or no need of network pre-planning. A tactical ad hoc network may be partitioned or fragmented into parts due to e.g., movements or terrain obstacles. It is therefore necessary that parts of the network can function autonomously, and this requires a distributed network control.

End users are usually not interested in networking details, the important thing is that the used application or service is delivered with sufficient quality. The Quality of Service (QoS) issue is complex and can be dealt with in many different ways. Furthermore, it involves all networking layers, from application to the physical layer. One method to obtain QoS, used in many wired networks, is over-provisioning of the bandwidth and "hoping for the best" while using standard network protocols. In wireless communication, where the bandwidth is scarce, some sort of QoS control and adaptation to the situation is necessary.

The network should be flexible to different types of command and control structures. Foreseeing future service requirements is difficult. Examples of some services are: situation awareness data, group calls, background traffic (e.g. e-mails), sensor data, and extremely urgent alarm messages. These services all have different QoS demands. However, one of the most important QoS demand to take into account is the delay requirement. We have restricted our investigations concerning different services to Best Effort (BE) traffic and delay sensitive traffic. For the latter the end-to-end delay requirement is chosen short enough (150 ms) so voice can be handled.

The main research direction in the project has been towards QoS oriented issues for ad hoc networks and distributed network control. The QoS issues are investigated by using the two services described above. The network is controlled at different layers. When controlling the network there is a need to both understand what can be done on a single layer and through cross-layer interactions.

In Chapter 2 and 3, Medium Access Control (MAC) is treated. MAC-protocols can be divided into reservation-based and contention-based. Chapter 2 investigates a reservation-based protocol called Spatial reuse TDMA (STDMA). Different assignment strategies and methods to facilitate a distributed version of STDMA is discussed. Traditionally, MAC protocols for ad hoc networks have been contention-based. One of the most frequently used protocols of this type is called Carrier Sense Multiple Access (CSMA). Chapter 3 carries out a comparison between STDMA and CSMA and discuss in which situations each protocol are preferable.

Chapter 4 treats routing. Two types of routing protocols, reactive and proactive, are studied. The Ad Hoc On-Demand Distance vector (AODV) routing protocol represents reactive routing, and the Fisheye State Routing (FSR) protocol represents proactive routing. These protocols are investigated and compared for different situations and networks.

Chapter 5 connects routing and MAC issues with low level adaptations that can be done in an adaptive radio node. In particular, the chapter focuses on assessing the benefits of using variable data rates and queuing systems. The study considers static and mobile networks as well as both best effort traffic and delay sensitive traffic.

Chapter 2

STDMA

Spatial reuse Time Division Multiple Access (STDMA) is a conflict-free time-slotted MAC protocol which achieves high capacity by letting the time slots be spatially reused, i.e., a time slot can be shared by radio units geographically separated if they cause sufficiently small interference to each other. This protocol was originally suggested in [31]. Since the radio units have assigned time slots, the protocol also have great advantages in terms of delay guarantees as compared to time conflict-based protocols such as Carrier Sense Multiple Access (CSMA).

An STDMA schedule describes the transmission rights for each time slot. The problem is to design STDMA schedules that fulfill required properties, e.g. minimizing delay or being able to update the schedules in a distributed fashion.

Before we go into more detail, we will show a small example of how we can go from a scenario to an STDMA schedule. In the top part of Figure 2.1 we see a group of nine tanks heading to the right.

Now, to represent this as a radio network each of these tanks will be shown as a node in the middle picture (circles), giving us a 9-node network. Furthermore, the ability to directly communicate between two of the tanks is represented by an line. Such lines are called links, and the lack of a link means that the two units cannot communicate directly, but must instead relay their message by intermediate nodes,i.e. use multihop. Direct communication is usually possible if the units are close to each other, but objects like a hill can prevent direct communication even when units are close, see for example nodes 6 and 8 in the figure.

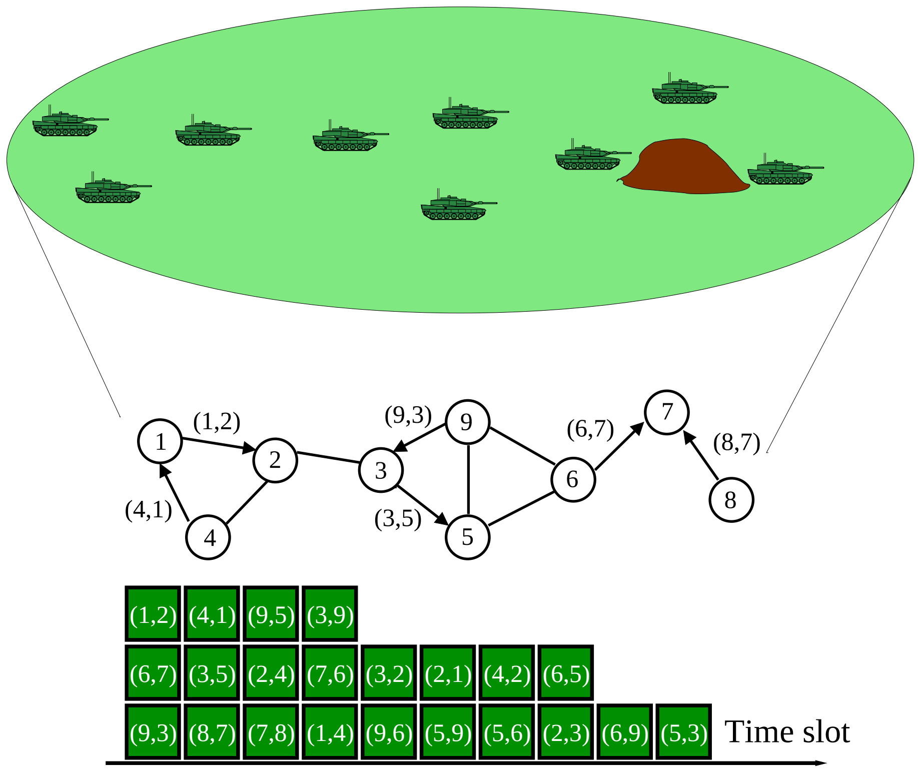


Figure 2.1: A unit of 9 tanks represented as a 9-node network.

Furthermore, we can see that communication between nodes 1 and 9 must be relayed by nodes 2 and 3. An STDMA schedule could assign link $(1, 2)$, $(9, 3)$, and $(6, 7)$ to transmit simultaneously, since they are sufficiently far from each other. Another set could be $(4, 1)$, $(3, 5)$ and $(8, 7)$, but not $(1, 2)$, $(3, 5)$, and $(8, 7)$ because, at least in this example, we are using omni-directional antennas, and the transmission of node 3 would interfere with the reception of node 2.

In the bottom part of the figure we show a possible schedule where each link is assigned one time slot each. As can be seen, at least ten time slots will be needed since the ten directed links between the four nodes in the center cannot share time slots.

Unfortunately, the nodes will be moving, and nodes that can transmit simultaneously without conflict at one moment will probably not be able to do so later. Returning to the above example we can see that if tank 3 moves slower than the three tanks to the left, the transmission by node 1 will eventually create sufficient interference when node 3 receives on link $(9, 3)$ so that links $(1, 2)$, $(9, 3)$, and $(6, 7)$ cannot share the same time slot anymore.

Therefore, the STDMA schedule must be updated whenever something changes in the network. This can be done in a centralized manner, i.e. all information is collected into a central node, which calculates a new schedule. This schedule is then propagated throughout the network. The schedules designed this way can be very efficient because the central node has all information about the network.

However, for a fast-moving network this is usually not possible. By the time the new schedule has been propagated it is already obsolete, due to node movements. Furthermore, it is not a robust solution, as the loss of the central node can be devastating for network communications.

Another way to create STDMA schedules is to do it in a distributed manner, i.e. when something changes in the network, only the nodes in the local neighborhood of the change will act on it and update their schedules without the need to collect information into a central unit.

Networks Models

In order to determine which nodes or links that can transmit at the same time when we do the scheduling, we need a description of the network. We usually refer to such a description as a network model. The network models used by STDMA algorithms have varied in complexity.

Most algorithms assume that transmission ranges are limited (usually circular), and beyond this no interference is caused. This allows the problem to be transformed into a graph-theoretical problem, i.e. the network is represented as a directed graph as we did in the previous section. In this graph, an edge between two nodes indicates that they can communicate with each other directly, and the lack of an edge indicates that they cannot affect each other even as interferences.

Using this model, scheduling can be transformed into the coloring of the nodes or links in a graph, which can be solved with the help of graph theory. This model thereby makes it simple to create STDMA algorithms. However, a disadvantage is that this model does not describe the wireless medium very well as it does not take capture into consideration or that the combination of interference from several nodes can cause comunications to fail.

A more realistic, although much more complex, model is the use of the signal-to-interference ratio, *Interference-Based Scheduling*. In this case, a node is assumed to be able to receive a packet without error if the received signal strength is sufficient compared with the noise and all interfering signals (from simultaneously transmitting nodes in the network).

The use of Interference-Based Scheduling have been the main focus of our STDMA research and all the following results assume the use of interference-based scheduling. Our main reason for this is that it gives a much better description of the network than a graph model.

2.1 Assignment Strategies

In the previous example, all transmission rights are assigned to the links, i.e. both transmitting and receiving nodes are determined in advance when the schedule is created. This is called *link assignment* or link activation. An alternative would be to assign transmission rights to the nodes instead. In this case only the node is scheduled to transmit in the time slot. Any of its neighbors, or all, can be chosen to be the receiving node. This is called *node assignment* or node activation.

Generally, node assignment is used for broadcast traffic, and link assignment is used for unicast traffic.

Link assignment behaves better for high traffic loads, achieving a higher throughput for unicast traffic. The reason to this is that in link assignment only the receiver needs to be guaranteed conflict free, in node assignment all neighboring nodes need to be conflict free, which gives less spatial reuse and thereby less throughput. However, the higher throughput comes at a cost of higher delay for link assignment than for node assignment for low traffic loads. There are more links than nodes so there will be a longer delay until a correct time slot will appear, thereby causing longer delay.

To avoid these drawbacks, we have suggested a novel assignment strategy. Our proposed strategy is based on a link schedule, but in which transmission rights are extended. When a link is supposed to transmit in a time slot, the node first checks whether there is a packet to transmit on that link. If there is

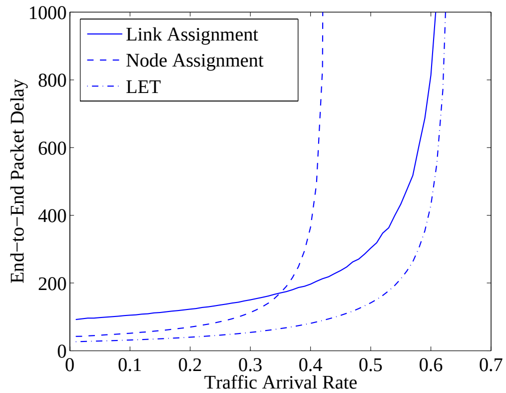


Figure 2.2: Network delay in a 30 node network.

no such packet, any other link with the same transmitting node might be used if the node has a packet to transmit. All transmissions will not be conflict-free anymore, though. We call this strategy Link assignment with Extended Transmission rights (LET).

In Figure 2.2, end-to-end delay for different traffic rates is shown for schedules using for the three different assignment strategies in a network of 30 nodes with unicast traffic. We see from this figure that in this network, link assignment is preferable to node assignment for high traffic loads. For low traffic loads, node assignment achieves a smaller delay. The LET method combines the advantages of the two methods and in this case achieves a smaller delay for all traffic loads.

Generally, for unicast traffic LET, is preferable to both link and node assignment except for networks of very high connectivity and low traffic. For broadcast traffic LET will outperform node assignment for low traffic arrival rates. However, this comes at a considerable cost in terms of maximum throughput. One exception is low connectivity, where LET can give the same maximum throughput as node assignment.

2.2 Frame Length

There is considerable variation of traffic over the different links of the network due to the relaying of traffic in multi-hop networks. An STDMA algorithm must adapt to this and give some nodes more capacity (time slots) in order to be efficient. Algorithms that do this are usually denoted as *traffic sensitive* or traffic controlled. The ability to give some nodes or links extra time slots related to the traffic loads on the links has been considered in several papers, see e.g. [32, 33]. Furthermore, studies on traffic sensitivity have also shown that it improves capacity considerably compared with giving each node or link a single time slot [34, 35].

The results regarding assignment strategies above assumes a variable frame length, chosen sufficiently long to obtain full traffic sensitivity. However, for large networks this is seldom possible, the overhead cost for handling the assignments of the time slots will simply be too high. Furthermore, a variable frame length will create problems for a distributed algorithm since the frame length may change for each update, which in turn will force a global update.

The choice of frame length thus have significant consequences for the result. It is an interplay between *traffic sensitivity* and *frame length*.

The required frame length is larger for link assignment than for node assignment, not just because there are more links than nodes, but there is also usually a higher variation of traffic over the links than nodes, requiring more time slots for compensation. Node assignment can therefore give good results with a shorter frame length than link assignment. This means that with a limited frame length, the results from the previous section changes. Only for longer frame lengths do we see these results from Section 2.1.

We therefore propose a novel assignment strategy–Joint Node and Link Assignment–that achieves the good results for all frame lengths. In this strategy each node is given a single node-assigned time slot, after this the links are assigned as link assignment [2].

For low connectivity, this strategy outperforms node and link assignment for all frame lengths, as can be seen in Figure 2.3. For medium connectivity and higher (not shown here), it cannot fully compete with node assignment for shorter frame lengths but it still considerably outperforms link assignment.

This new strategy will allow us to use shorter frame lengths, which will be an advantage in reducing unnecessary overhead for distributed algorithms that

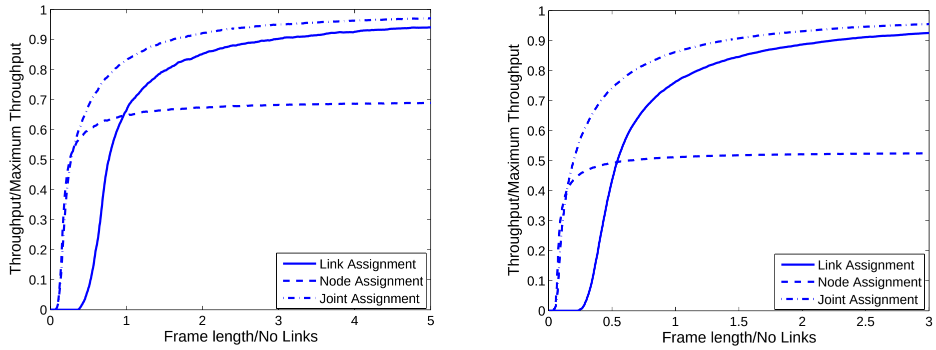


Figure 2.3: Average maximum throughput compared to the highest maximum throughput for different frame lengths for networks of low connectivity, using all three assignment strategies. 20-node networks to the left 60-node network to the right.

need to negotiate for each time slot.

2.3 Distributed Scheduling

We have also developed a novel interference-based distributed STDMA algorithm that can give results as good as those a centralized algorithm can. This algorithm can change the amount of overhead traffic, giving schedules with different capacity in order to be able to adapt to different scenarios. The development of this algorithm have been one part of an investigation on how to efficiently handle (use) distributed information, thereby giving an indication on how much network information should be transferred for different networks.

In addition, our algorithm can use two approaches, (refered to *states* and *no states*).

The first approach gives much more aggressive allocations, resulting in schedules with maximum throughput much closer to that of a centralized algorithm but at a cost of high overhead. Although, the overhead can probably be considerably decreased through further work.

The second approach gives an algorithm that is much more careful in assign-

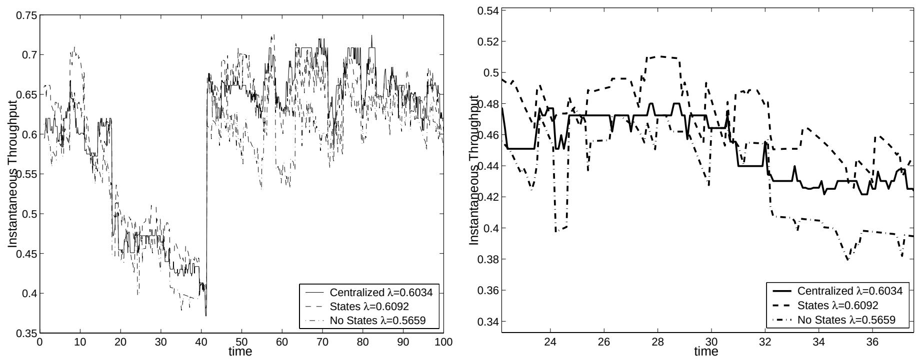


Figure 2.4: The figure shows the instantaneous maximum throughput during 100 seconds for the 32-node network.

ing time slots, resulting in lower throughput. The advantage is that the overhead requirements are then so much lower that the algorithm can possibly be used for the high mobility case for links with reasonable data rates.

In Figure 2.4 (left) we show the instantaneous maximum throughput for a mobile network for both the distributed algorithm as well as a centralized algorithm for a 100 second scenario. In addition, in 2.4 (right) we shows the instantaneous maximum throughput from 22 seconds to 38 seconds into the simulations.

As can be seen, the instantaneous maximum throughput when we are using the first approach is on average as high as the maximum throughput for centralized scheduling, despite the fact that information is limited. When we use the second approach, there is a 6% loss in maximum throughput compared to the centralized approach, but at half the overhead cost.

There is considerable variation of the instantaneous maximum throughput during the 100 seconds though, which is unwanted. However, this is the case for the centralized algorithm as well. This is a fundamental property of ad hoc networks and thus not a specific weakness in our distributed algorithm. The very quick changes taking place at for example 18 seconds in Figure 2.4 can however create a problem when trying to guarantee quality of service.

These results do not consider the cost of overhead, however, but in order to do so we would need to consider data rates on the channel as well. The overhead traffic requirements can be significant, especially if we need a large frame length and the network is very mobile. For slower networks the amount of generated overhead can easily be handled. For larger, highly mobile networks (which also requires larger frame lengths) high link data rates will be needed in order to keep the proportion of overhead traffic low while letting the algorithm generate good schedules. The algorithm will not fail if the capacity for the overhead is not sufficient, but the generated schedules will be less effective, with reducuced performance as result. The required overhead can probably be reduced in the future by adding the joint node and link assignment of Section 2.2 as this reduces the frame length requirement.

Chapter 3

Comparison of STDMA and CSMA

In this chapter we assess distributed STDMA and CSMA. We here evaluate how the choice of MAC protocol affects the users in a mobile ad hoc network under three different kinds of traffic loads. We also assess two mobility cases: a low mobility case where the radio nodes move at a velocity of 2 m/s and a high mobility case of 10 m/s.

3.1 Carrier Sense Multiple Access

The access protocol used by the 802.11 family is called Carrier Sense Multiple Access (CSMA) [36]. The main idea of CSMA is that the sending nodes measure the signal level to check if the channel is idle before transmission. Collision avoidance can be used together with CSMA; i.e. the channel is reserved before sending takes place. Collision avoidance is desired in wireless environments with high bit error rate. It is also preferred where not all of the nodes are within transmission range from each other. Otherwise the hidden terminal problem may occur, which we will describe in this section.

In CSMA, a wireless node that wants to transmit a packet performs the following sequence:

1. Listen on the channel.

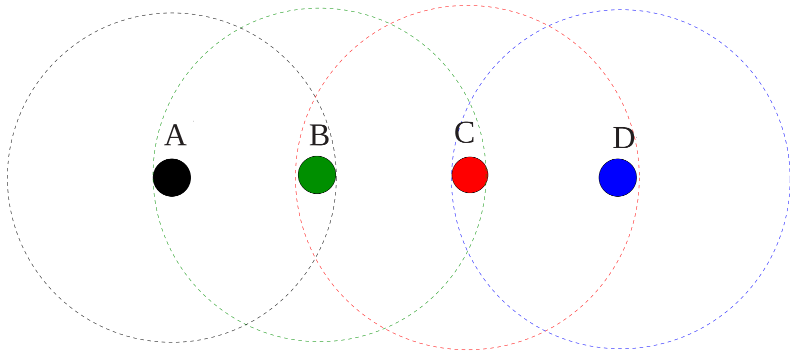


Figure 3.1: Principles of CSMA.

2. If the channel is idle, send the packet
3. If the channel is busy, wait until transmission stops and then for a further random time.
4. Listen on the channel.
5. If the channel is idle, send the packet. If not, repeat from step 3 until the packet is sent or a maximum number of allowed attempts is reached.

The hidden terminal problem occurs if node B and node D wants to transmit a message to node C, see Figure 3.1. Node C can hear both nodes but node B can not hear node D and vice versa, they are *hidden* from each other. In a CSMA environment nodes B and D would both properly listen to the channel and transmit, but node C would get corrupted data due to the collision.

The hidden terminal problem can be solved by use of RTS (Request To Send) and CTS (Clear To Send) packets. Before transmission, node B sends a small RTS packet. This is heard by node C, which sends a small CTS packet response.

The CTS packet is heard by both node B and D, and the collision can thus be avoided. The RTS/CTS packets contain information about how long the transmission will take, including the final ACK packet. All nodes receiving RTS/CTS packets consider the channel as busy for this time period, and keep this information in a Network Allocation Vector (NAV). Node A and D will in this example reserve the virtual channel and will not transmit, see Figure 3.2.

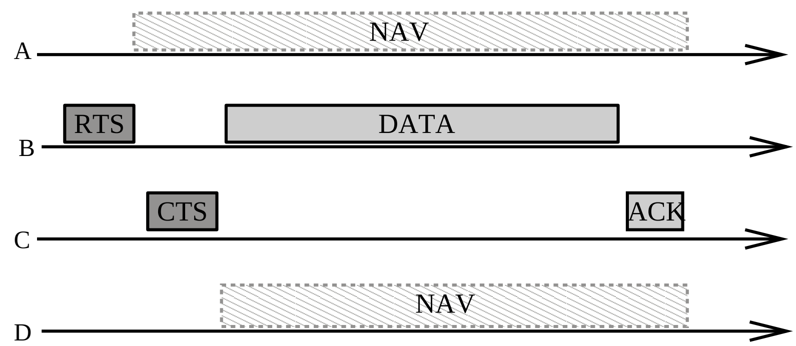


Figure 3.2: The use of virtual channel allocation in CSMA. The RTS/CTS packets contain sufficient information for node A and D to know how long the channel is reserved.

Small data packets will not use the RTS/CTS function when the probability of receiving an RTS correctly is equivalent to that of receiving the data packet. The threshold is set to 256 Bytes , which is the minimum packet size for sending RTS/CTS. Packets larger than 1024 Bytes are fragmented into smaller packets. Both of these two thresholds are default values from the 802.11 specifications. The fragmentation generates some overhead but in case a fragment is dropped, the damage is limited. An ACK is sent after each correctly received fragment and lost fragments are retransmitted.

3.2 Distributed STDMA

For the evaluation we use the interference-based distributed STDMA algorithm described in Chapter 2. The STDMA algorithm will generate a certain overhead traffic, the amount of generated overhead traffic will be dependent on the mobility and the rate at which sessions are generated. However, for simplicity we have chosen to set the maximum allowable overhead to be 25% of the channel capacity. The protocol only uses this given amount of link capacity or overhead traffic, which affects the efficiency of the resulting schedules. The remaining 75% of the capacity is available for user traffic. A more adaptive approach would be to use a percentage for overhead traffic that depends on the scenario. We will, however, ignore the possible gains of doing this in this comparison.

In addition, for all STDMA simulations we assume a framelength of 100 time-slots, which is sufficient to give a reasonable traffic sensitivity for the maximal traffic load in the simulations. However, for lower traffic loads, the framelength is longer than required and thereby results in unnecessary high overhead traffic.

In order to handle mobility and limited available information, our STDMA algorithm assumes that all used links have a margin for interferences. This means that the required signal-to-noise ratio is set higher than the minimum required to create a link. Better spatial reuse can then be obtained. This will have consequences for the connectivity, since the network will be less connected than if no interference margins would be used. A possible solution to this is to allow a few links with lower margin, i.e. near the minimum required SNR, just to keep the network connected. This can probably be done without any significant effect on capacity. For simplicity, this has not been done in the simulations presented here, which has some impact on the results presented in Section 3.4.

Even if this interference margin reduces the problems with link variations due to mobility, we still want to keep it small. There will be cases where the resulting Signal-to-Interference-and-Noise-Ratio (SINR) will be to low for the used data rate on the links. To handle such cases, our STDMA algorithm also requires that the link data rate can be lowered in time-slots where the SINR in the receiver otherwise would not be sufficient. In the simulations, the data rate can be lowered in steps of 5 percent until it reaches half the original rate. If yet a lower data rate is required, the time slot will be deallocated.

3.3 Traffic and Network Model

We use three different simulation cases for the user traffic load: voice, file and mixed traffic. Each traffic case is modeled as a number of running sessions with specified application packet sizes and end-to-end delay requirements. We vary the average total number of running sessions and evaluate their success rate with respect to the delay requirements. All traffic is modeled as point-to-point traffic with one way connections. New sessions start according to a Poisson process. Furthermore, we assume that the traffic is uniformly distributed over the nodes, i.e. each node is equally probable as the source node and each node (except the source node) is equally probable as the destination node. During a session, the source transmits packets to the destination with a constant bit rate, 12.2 kbps. Thus the average amount of data transferred during one session is 144 kb.

Voice Traffic

In this traffic case we model voice sessions, i.e. sessions with small packets and hard delay requirements. The call durations are exponentially distributed with a mean of 12 seconds. To model a delay sensitive traffic, we have a maximum acceptable delay on the packets, chosen to 150 ms as recommended by ITU [37], which is small enough to carry voice. Further, the session is considered failed if more than 5 % of the received packets are delayed more than 150 ms during the session. The size of each voice packet is 256 bits. On average, 576 voice packets are sent during a session.

File Traffic

We model file traffic as sessions that uses large packets with relaxed delay requirements. In order to compare the network efficiency for both file and voice traffic, we specify the file application so that on average the same amount of data is transferred during one session as with the voice application. The size of a file packet is 12 kb, so the mean number of packets sent during a session is 12. A session is considered successful if each of the received packets is delayed less than 2 s.

Mixed Traffic

We also evaluate the network when the user traffic is a mix of both voice and file sessions. The average number of started sessions is the same for both traffic types. We do not use any priorities to differentiate between the two services.

In the nodes, all packets are queued and treated equally on a first come, first served basis. The maximal number of packets in queue in the nodes is 166 for the file traffic case and $166 \times 48 = 7968$ for voice traffic, each case corresponding to the same amount of data, 2 Mb. In the mixed traffic case, the maximal queue length is 4067 packets (the mean of 166 and 7968). On average the maximal queue lengths are the same in all three traffic cases, measured in number of bits.

Link Model

An essential part of modeling an on-ground or near-ground radio network is the electromagnetic propagation characteristics due to the terrain variation. A common approach is to use the basic path-loss, L_b, between two nodes (radio units). To estimate the basic path-loss between the nodes, we use a Uniform geometrical Theory of Diffraction (UTD) model by Holm [38]. To model the terrain profile, we use a digital terrain database. All our calculations of the basic path-loss are carried out using the wave propagation library DetVag-90® [39].

For a sending and a transmitting node with isotropic antennas (antenna gain equal to one), we define the signal-to-noise ratio (SNR) in the receiving node as follows:

$$\frac{P}{N_R\, L_b\, R}, \tag{3.1}$$

where P denotes the power of the transmitting node (equal for all nodes), N_R is the receiver noise power, R is the data rate, and L_b is the basic path-loss between the nodes. In the simulations, we assume that the required link data rate R in absence of interference from other nodes is 1 Mbps.

Mobility

In our scenario, we have 32 nodes moving randomly in an area of 4x4 km. They move independently of each other at a constant velocity. The nodes randomly change direction at certain intervals. When a node reaches the area border, it turns and proceeds in a new direction. In the two mobility cases used in the evaluation, the velocity is either 2 m/s or 10 m/s. The scenario is running during 2000 seconds for CSMA and 1000 seconds for STDMA, depending on the required simulation time for each protocol. This means that the variance of the results are slightly larger for STDMA.

Generated Networks

Due to the mobility, the amount of node pairs that have single- or multi-hop connections will vary. In the generated networks, approximately 98% of all pairs of nodes were connected. The average number of hops in the existing routes was nearly 2 (1.8 for CSMA and 2.0 for STDMA). The same node movements are

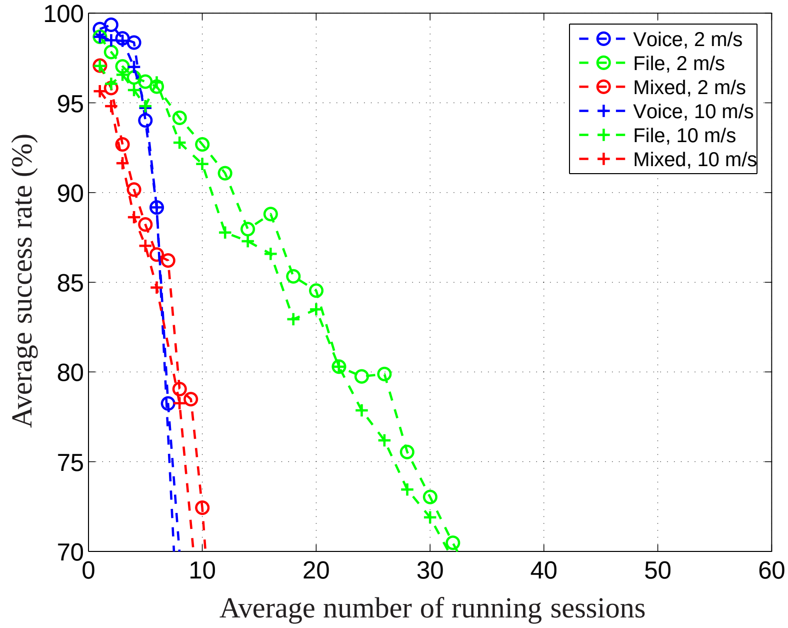


Figure 3.3: Average session success rate for CSMA, evaluated for voice, file and mixed traffic along with high and low mobility.

used for both STDMA and CSMA, but the resulting networks used by STDMA are slightly less connected due to its interference margin mentioned above.

3.4 Simulation Results

We compare the success rate (fraction of successful sessions) for our three kinds of traffic sessions. The acceptable success rate level may vary depending on the situation. We here assume that the sessions must be very reliable to be used in a tactical network, thus having a success rate of at least above 90% if the user should trust the system. As already mentioned, a voice session is successful if 95% of the sent packets arrive with an end-to-end delay less than 150 ms. A file session is successful if all of the sent packets arrive with an end-to-end delay less than 2 s.

In Figure 3.3 we can see that the results for CSMA are not affected by the

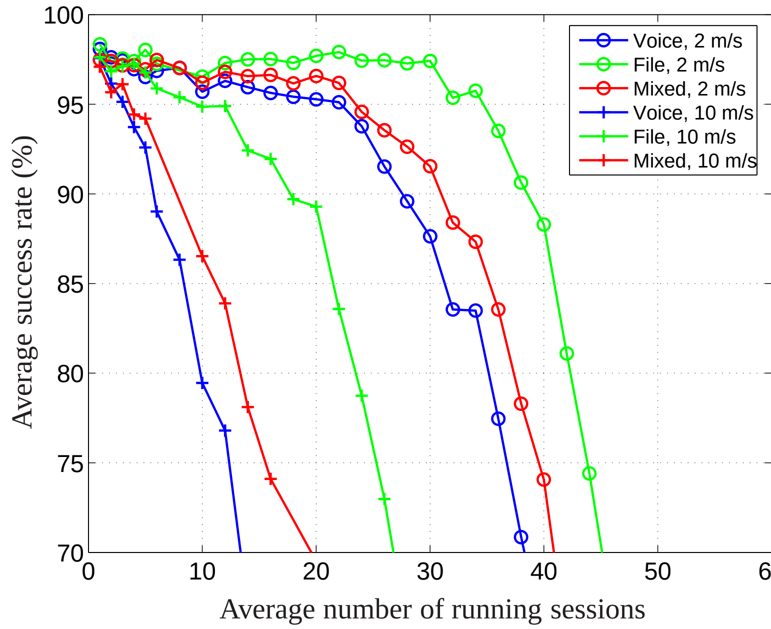


Figure 3.4: Average session success rate for STDMA, evaluated for voice, file and mixed traffic along with high and low mobility.

mobility. The reason for this is that CSMA does not rely on administrative traffic. The CSMA protocol, however, does not handle small packet sizes well. For small packets of size near the size of RTS/CTS packets, channel allocation with RTS/CTS would be inefficient and is thus not used. Another reason for the low success rate for traffic that contains voice sessions, is that the CSMA overhead becomes propotionally large. Even without counting the acknowledgements, less than a third of the transmission time for each sent packet is used for the voice data. Data overhead and guard time for each packet are independent of the total packet size.

For success rates above 90 %, the mixed traffic case gives the lowest result values. The cause for this behaviour needs to be evaluated further.

STDMA make use of received information about the surrounding part of the network. For low mobility, the success rate is good up to a certain session load and then quickly decreases. For higher mobility, however, the overhead traffic

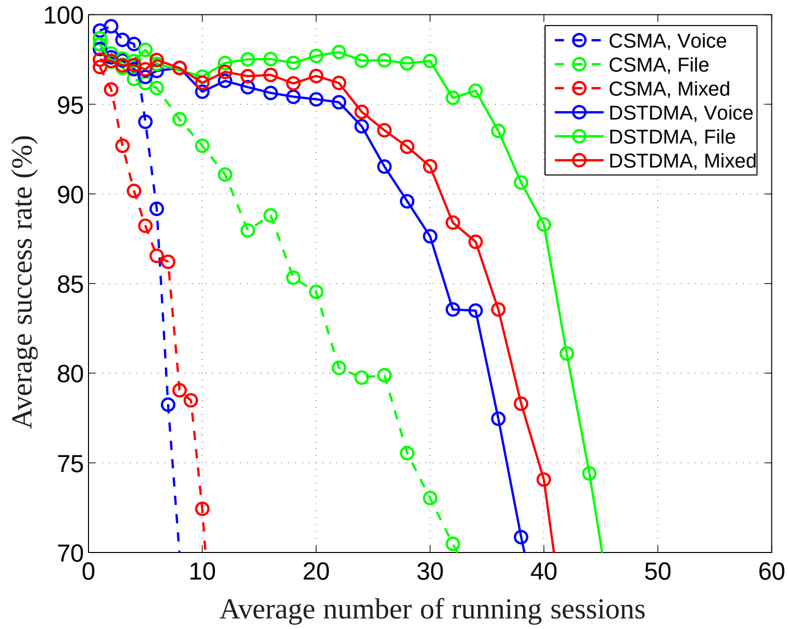


Figure 3.5: Comparison of CSMA and STDMA for low mobility,2 m/s.

does not give enough information, which results in less efficient schedules. This is illustrated in Figure 3.4.

In Figure 3.5 and 3.6 we compare the session success rate for CSMA and STDMA in each mobility case. Although the performance of STDMA degrades with higher mobility, it still performs better than CSMA in the high mobility case. A drawback of dynamic allocation protocols, such as CSMA, is that the packet delay varies a lot depending on the amount of collisions in the network. When collisions occur, packets may be deleted because a node has too many packets in queue. Another consequence is that the total delay of the packet is to large to satisfy the delay requirements. Because of its maintained transmission schedule, STDMA avoids more collisions and does not suffer as much from delay variations as CSMA.

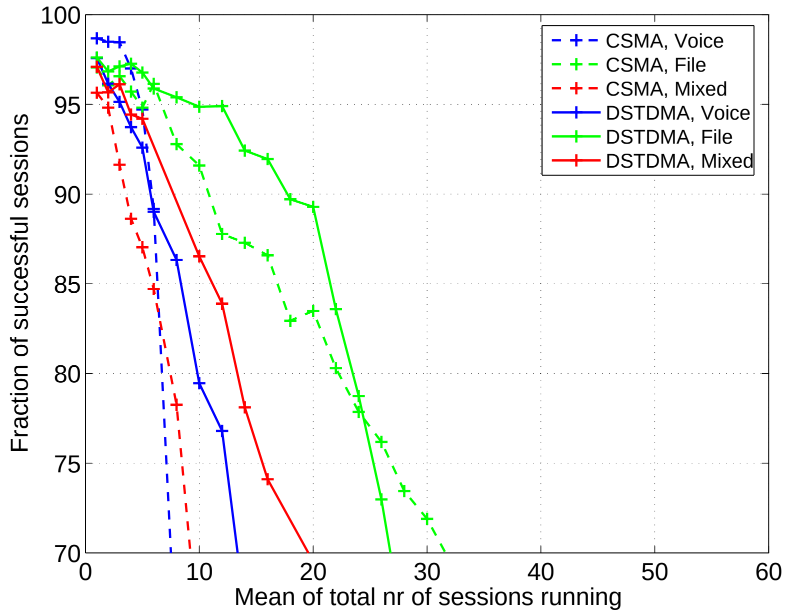


Figure 3.6: Comparison of CSMA and STDMA, in the upper plot for low mobility (2 m/s) and in the lower plot high mobility, 10 m/s.

3.5 Summary

We have evaluated the two multiple access protocols STDMA and CSMA for wireless networks of 32 nodes moving randomly in a 4x4 km terrain area. The simulations were performed for two choices of node speed and three types of traffic loads. For both protocols, file traffic gives the highest success rate. For CSMA, the lowest success rate is obtained for a mix of voice and file traffic. For STDMA, voice traffic gives the lowest success rate.

Mobility

- For node speeds up to 10 m/s, STDMA can handle a higher load of running voice and file sessions than CSMA.

- Mobility affects the two evaluated protocols differently. While CSMA is almost unaffected, STDMA deteriorates with increasing mobility. For increasing node mobility, STDMA needs a larger proportion of administrative traffic, or its computed schedules will be less efficient with mobility.

Small packet sizes

- The simulations show that voice traffic can not be handled well together with CSMA as specified in the 802.11 standard. Since voice traffic packets are small, RTS/CTS reservations are not used. Moreover, because packets are sent one by one, proportionally more CSMA traffic overhead is required for small packets. The same amount of channel resources, due to address data and guard time, is occupied for each transmitted packet regardless of its size.
- The STDMA protocol is not affected in the same way by small packet sizes, since the traffic packets are managed together in the nodes.

Chapter 4

Routing

There are two categorys of routing protocols for ad hoc networks; reactive and proactive protocols. A reactive routing protocol does not search for a route until it is needed. Once a route has been established, it is maintained by a route maintenance procedure until the route is no longer desired. Proactive routing protocols on the other hand attempt to maintain, at all times, up-to-date routing information from each node to every other node by distributing and exchanging routing table information (e.g., distance vectors).

In this project we have studied a reactive routing protocol, Ad Hoc On-Demand Distance Vector (AODV) and a proactive one, Fisheye State Routing (FSR). In Section 4.1, we present the AODV algorithm and discuss some results when modifying it to be able to use a generic cost metric. FSR is presented in Section 4.2 and the choice of parameter settings is discussed. Finally, in Section 4.3, we compare the protocols and make some concluding remarks.

4.1 AODV

The Ad Hoc On-Demand Distance Vector (AODV) routing protocol [40, 41] is a reactive routing algorithm designed for mobile ad hoc networks. It is loop-free, scalable to large networks, and distributed. Furthermore, AODV uses sequence numbers to avoid old routes and the propagation of old information.

When a node has data to send to another node and it does not already have a route to that node, the source node broadcasts a route request (RREQ), see

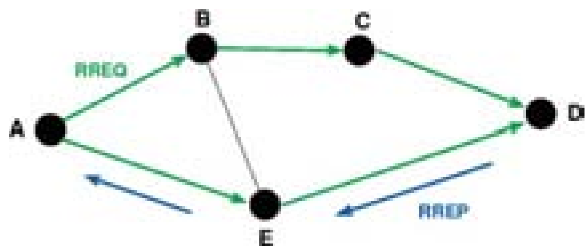


Figure 4.1: Principles of AODV routing. Node A sends out route requests to find a route to node D. Node D then sends a route reply.

Figure 4.1. The RREQ packet contains the address and sequence number for both the source and the destination node. Nodes that receive the RREQ update their information regarding the source and set up backwards pointers towards the source node as a preparation for the transmission of a route reply (RREP).

A RREP is transmitted either by the destination node when it has received a RREQ or by an intermediate node, if it has a valid route to the destination. A node that receives a RREQ and does not return a RREP either re-broadcasts the RREQ or, if it already has processed a packet with the same RREQ, discards the RREQ.

While the RREP is unicasted to the source node, the intermediate nodes set up forward pointers towards the destination. This means that once the RREP reaches the source, the route is ready to use. Sometimes the source node receives additional RREP packets. If these packets contain newer information regarding the destination, i.e. has a higher destination sequence number, or if it contains a route with fewer hops, the source may change its route.

A route is maintained as long as it is needed, i.e. as long as packets are transmitted on it. If no packet is transmitted for a certain period of time, the links time out and are deleted from the routing table.

Metrics

Most routing protocols for mobile ad-hoc networks tries to find routes with as few hops as possible [42, 43], i.e. uses a minimum hop metric. In a heterogeneous network, there are often other factors than the number of hops that determine how good a route is, e.g., the data rates of the included links, the traffic load, or the power consumption. A more complex metric must therefore be used, since the minimum hop metric does not take any other factors into account. We here choose to use a data rate sensitive metric, but it is interchangeable with any other type of metric. This type of routing is refered to as $1/R$ routing. We here define the metric, C_U, as a function of the link data rates:

$$C_U = \sum_{\forall (i\,j) \in U} \frac{1}{R_{i\,j}}, \tag{4.1}$$

where U is the set of links in the route and $R_{i\,j}$ is the data rate used on the link (i, j). When choosing between several possible routes, the best route (according to this metric) is found when the cost is as small as possible. A minimized metric means that a minimum of network capacity is used and that the throughput is as high as possible.

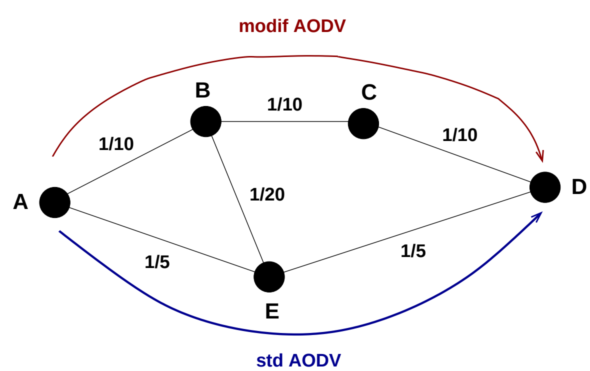


Figure 4.2: A network with multiple data rates. Routes found by standard and modified AODV when using the metric defined above.

An example of how this works can be seen in Figure 4.2, where node A wants to transmit a packet to node D. We have different data rates (5, 10 and 20 Mbit/s) for different links, and four possible routes. The cost of using these are shown i Table 4.1. According to this, route U_3 would be chosen if standard AODV with minimum hop routing was used. Modified AODV, using our metric, instead chooses route U_1.

	route	# hops	C_U
U_1	$\{A, B, C, D\}$	3	0.3
U_2	$\{A, B, E, D\}$	3	0.35
U_3	$\{A, E, D\}$	2	0.4
U_4	$\{A, E, B, C, D\}$	4	0.45

Table 4.1: Number of hops and cost, C_U, for the routes in Figure 4.2.

Modifications of AODV

A node using AODV starts sending data on the route from the first arrived RREP. When AODV uses the “first” found route, this might not be the most efficient route, especially when variable data rates are used. Even though it is the fastest route for a route request, it is not necessarily the fastest route to send data traffic on.

We hence modify AODV to use the metric presented in Equation 4.1, instead of the minimum hop metric, when determining if an additional RREP contains a better route. Furthermore, we allow intermediate nodes to forward additional RREQs if the new RREQ packet contains a route that is better than the routes of previously transmitted RREQ. In accordance to this, both destination and intermediate nodes may also transmit multiple RREPs if it is an improvement to the previous route.

Results

In our simulations, the networks are stationary and exists of 32 nodes, who are fully connected when a link data rate of 256 kbit/s is used. On each link, there are 7 available data rates, each a factor of 2 higher than the previous; 256 kbit/s, 512 kbit/s, ... , 16 Mbit/s. During the simulation all nodes initiate and maintain routes to all other nodes in the network. Once a route is found, we transmit the minimum amount of traffic needed to keep the route active for the rest of the simulation run.

Cost of Using a Route

In Figure 4.3 we compare the mean cost (according to the metric in Section 4.1) of using routes found by both the modified and the standard AODV with the

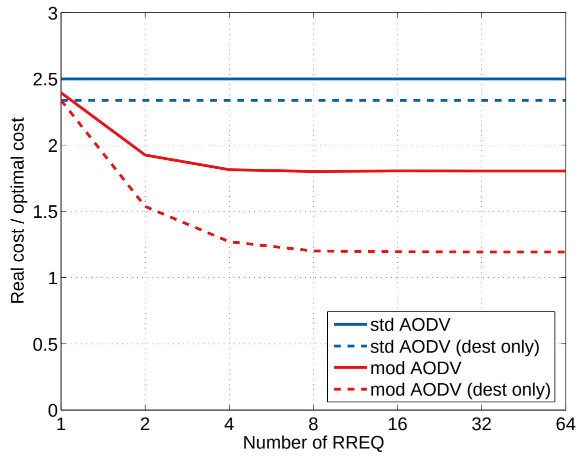


Figure 4.3: Mean cost of using a route for standard and modified AODV (32 node networks). The result is normalized with the optimal route costs.

cost for optimal $1/R$ routing. The cost is shown as a function of the allowed number of transmitted RREQs/RREPs. The results are normalized by the mean cost of optimal $1/R$ routing, i.e. we desire a result as close to **one** as possible.

As we can see in Figure 4.3, we get closer to achieving optimal throughput if only the destination is allowed to reply to route requests, i.e. generate RREPs. This is due to the fact that intermediate nodes, when they may reply, can reply and suggest non-optimal routes. Also, when the intermediate nodes reply, they do not forward the RREQ and hence the destination may never actually be reached by the request. Furthermore, Figure 4.3 shows that allowing additional RREQs initially improves the routes but only up to a certain, network dependent point.

Routing overhead traffic

Here we present results for the amount of routing overhead traffic generated, both when we use modified and standard AODV. The routing overhead traffic is here calculated as the mean number of routing overhead packets that are transmitted when creating a route.

In Figure 4.4, the mean number of routing packets necessary to create a route is depicted as a function of how many route requests each node may forward.

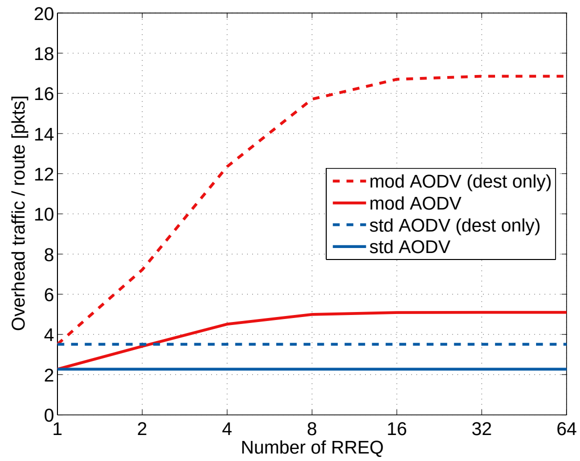


Figure 4.4: Mean traffic needed to create a route for standard and modified AODV.

From Figure 4.4, we can see that more routing overhead traffic is generated by AODV when only the destination may reply to the route request. The results for the standard AODV algorithm with minimum hop routing equals the results for the modified AODV algorithm with a allowed number of transmitted RREQ set to one and, as could be expected, the use of a data rate sensitive metric along with allowing additional RREQs results in a larger amount of routing overhead traffic.

Discussion

We can conclude that the use of a data rate sensitive metric along with allowing the forwarding of additional route requests, indeed yields routes with better throughput than if standard AODV routing is used, particularly for routes when only the destination may generate route replies. Use of the modified AODV algorithm, also results in an increased amount of overhead traffic. It is thus crucial to consider both the total cost, e.g., overhead traffic, and the resulting capacity gain when evaluating if an algorithm modification is worth while.

Furthermore, AODV utilizes routing information gathered from other nodes and sometimes this result in non-optimal routes. However, reactive routing protocols are not designed for networks where all nodes simultaneously create and

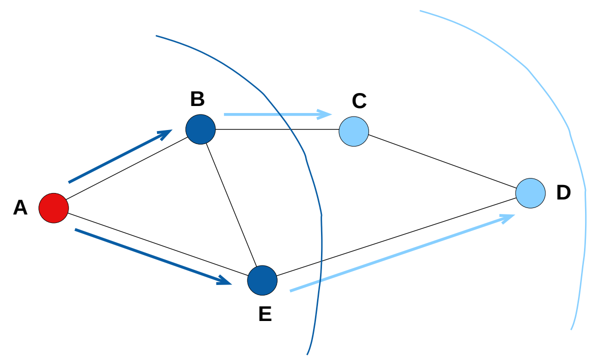


Figure 4.5: An example of the FSR Technique. Information about node A is distributed with a higher frequency to B than to C, i.e. information updating frequency decreases when the distance to A increases.

use routes to all other nodes, hence we believe that better results would have been achieved if only a couple of node pairs were active at a time.

4.2 The Fisheye Technique

The Fisheye State Routing protocol [9, 27, 43] for wireless ad hoc networks is a proactive link state protocol where the objective is to keep the control traffic low and still be able to provide accurate information about the routes. A node's perception of its surroundings is similar to that of a fisheye, where the level of detail is high near the "focal point" and decreases with the distance from the focal point. This means that when a user packet is sent, the intermediate nodes will have increasingly better routing information available as the packet approaches its destination and will use this to gradually improve the route.

Each node running FSR stores information about all destinations in the network. This information is used when creating routes and is continuously updated through update messages sent from neighbors.

Each node divides the network into a number of *scopes*, defined as the set of destinations that a node can reach within a given interval of hops. An example where two scopes are seen is shown in Figure 4.5. Here, the scopes are defined so that they contain nodes one hop away, and two hops away, respectively. To obtain lower levels of overhead traffic in a mobile network, the generation of update messages is periodic instead of event-driven. A node *must* update and transmit information regarding itself with a periodicity

$$T_u = n \cdot \delta,$$

where n is a parameter we can change to improve the performance of the algorithm and δ is the minimum time between two updates, see [9]. This is done whether any of the node's neighbors have changed or not. A node *may* transmit information about the nodes in scope i to its neighbors with the periodicity T_s^i if the information has changed,

$$T_s^i = \delta \cdot \textit{scope update factor} = \delta \cdot \text{round}(h^\alpha).$$

Here h is the distance in number of hops to the nodes in this scope, and α determines the grade of attenuation of overhead traffic in the network. This means that the distribution of overhead traffic is kept low and this makes the protocol scalable.

Performance Measures

Minimum route length can often be achieved if we accept a large amount of overhead traffic. We here define a measure that takes into account both the route length and the overhead traffic of finding good routes. The total network capacity is G Bytes/s. If all routes were optimal and we ignore the overhead traffic of finding them, the users could transmit $\lambda_{max} = G/h_o$ Bytes/s through the network, where h_o is the average length of an optimal route. In a more realistic network, user capacity is lost due to routing overhead traffic, imperfect routes, and incorrect routes, see Figure 4.6 i.e.,

$$\begin{aligned}\text{Network Capacity} &= \text{User traffic} + \text{Routing (overhead) traffic} \\ &\quad + \text{Imperfect routes traffic} + \text{Incorrect routes traffic}\end{aligned}$$

We define S as the traffic generated in the network by the users. If ϵ is the fraction of incorrect routes in the network, $S\epsilon$ is the part of the user traffic that will be lost, and $U = S\,(1-\epsilon)$ is the part of the user traffic that is successfully delivered to its destinations. The extra traffic caused by longer, imperfect routes

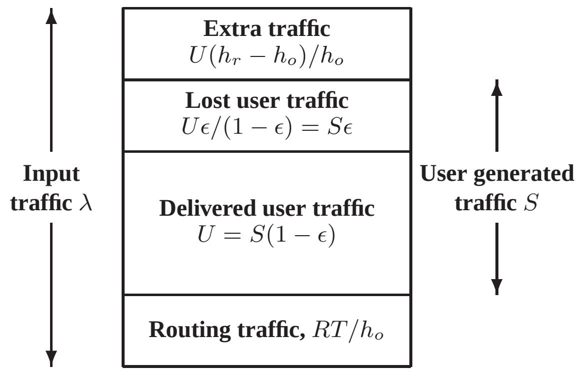


Figure 4.6: Illustration of the partitioning of the total network capacity into different kinds of traffic.

is defined as $U\,(h_r - h_o)$ and the traffic caused by incorrect routes is estimated as $Sh_o\epsilon$. Using these assumptions, we get

$$\begin{aligned} G &= \lambda_{max} \cdot h_o \\ &= Uh_o + RT + (h_r - h_o)\,U + \frac{\epsilon}{1-\epsilon} h_o U. \end{aligned} \tag{4.2}$$

To find optimum parameter settings for different network capacities G, we can then maximize the fraction of user traffic, i.e.

$$\frac{Uh_o}{G} = \frac{U}{\lambda_{max}}. \tag{4.3}$$

This measure expresses how efficiently the total network capacity is utilized when using a certain routing algorithm in the network.

Results

The networks used in our simulations consist of 64 nodes that move at 70 km/h, resulting in a highly mobile scenario. The connectivity ϕ , i.e. the amount of node pairs that are connected by single-hop or by multi-hop, is alternated between 95% and 90%.

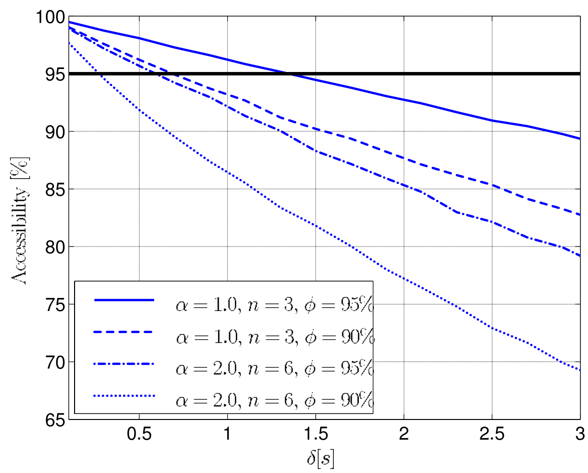


Figure 4.7: Accessibility as a function of the minimum time between updates, δ, for a few combinations of α, n, and network connectivity.

Accessibility

Due to delays and inaccuracy in routing tables, FSR will not find all routes. In Figure 4.7, examples of *accessibility*, the percentage of routes that FSR found, compared to those found by an optimal routing algorithm, are shown. We can see from this figure that accessibility decreases faster for a network with 90% connectivity than for a network with 95% connectivity. One reason for this is that the routes are longer in a network with low connectivity and hence more difficult to find. We can also see that when α, n and δ are increased, the accessability demand is harder to meet, since the attenuation of routing overhead traffic in the network increases.

Utilization of Network Capacity

As could be expected, different optimum values for the FSR parameter settings were found when minimizing the overhead traffic and the route length respectively. If we optimize the routing algoritm for minimimum overhead traffic, we get infrequent updates and large attenuation (i.e. high values for δ, α, and n), thus generating rather bad routes. If we instead minimizing the route length, low values will be choosen, resulting in much overhead traffic. Neither of the two suggestions will thus give the routing algorithm a good overall

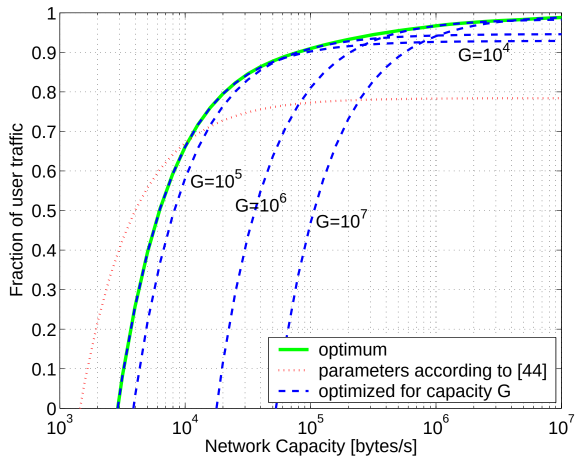


Figure 4.8: Fraction of the maximum user capacity available for user traffic in the network.

performance [27].

In Figure 4.8 we can see the fraction of the network capacity available for user traffic as a function of maximum network capacity, see Equation (4.3), for a few different cases. We can see that the choice of parameter settings is important for obtaining good performance. Four examples of fixed parameter settings are shown in Figure 4.8 as dashed curves, each one chosen to maximize the performance for a specific network capacity, G. The dotted curve shown in Figure 4.8 shows a parameter proposal for FSR from [44]. We can see that if bad parameter settings are chosen, the loss in user capacity can be considerable.

The solid curve in Figure 4.8 shows the optimum performance for our scenario when using adaptive parameter settings, i.e. when the optimum FSR parameter setting is chosen for each network capacity. For the capacities shown here, the optimum user capacity is achieved for $\alpha \approx 1.5$.

However, if the total network capacity further increases, the values for α, δ, and n decreases. This is due to the fact that for very high capacities, it is important to keep the route length close to optimum, while the routing overhead traffic becomes insignificant.

Discussion
From our simulations, we conclude that the Fisheye technique is efficient in the mobile ad hoc networks studied, since the maximum user capacity is achived when the routing overhead traffic attenuation is prominent. However, the choice of parameter settings is important for obtaining good performance. We can also gather from our simulations that FSR will suffice even in a highly mobile network.

Combining Distribution of Position Information with the Routing Overhead Traffic
In many cases, it is important to have certain information regarding other nodes in the network, either for the user/application or for the network protocols. In [3] we show that it is possible to combine the distribution of position information with the routing overhead traffic generated by the Fisheye State Routing protocol. Furthermore, given user demands on position accuracy can be met for large, mobile networks moving through real terrain. The technique used can easily be generalized to distribution of other types of information, e.g., battery power or traffic load.

4.3 Summary

We can conclude that proactive and reactive routing protocols are to be preferred for different situations and networks.

- In heterogeneous networks, where links e.g., can have different possible data rates, a routing protocol should efficiently be able to take the heterogenity (e.g., the data rate of each link) into account when determining which route should be used. For a reactive routing protocol, it is quite difficult to change the metric since we do not use or uphold as much information regarding the network topology as we do in proactive protocols.

- A proactive routing protocol has continous information about the network and consequently better chances of succeeding in providing QoS.

- The amount of overhead traffic that is generated in the network can be of great importance. In a proactive routing protocol overhead traffic is continously generated and can be crucial to the network capacity. In a reactive protocol, overhead traffic is only sent when a node searches for a route that is needed. However, the route search delays the start of sending data traffic and this delay can be crucial to some applications.

- From our studies, we can see that the quality of the found routes are slightly better when using proactive routing protocols. However, there is a trade-off between the generated overhead traffic and the route quality.

Reactive and proactive routing protocols are well suited for different situations. There is probably a need for hybrid protocols that can adapt to different scenarios. In some cases it is important to keep the overhead traffic low or even to have silent nodes. In other cases, a low delay is of critical importance.

We can also see that for both reactive and proactive routing protocols it is of highest importance to choose the right parameter settings. In a network where it, for practical purposes, is neccessary to use a fixed parameter setting, there can be heavy losses in user capacity if badly chosen parameter settings are used. It can also be noted that in a real network, finding suitable parameter settings for all occations will be very difficult for the user. It would hence be advantagous if the algorithm could automaticly adapt its parameter settings to changes in the network environment. Furthermore, by adapting the parameter settings, higher user capacity can be achieved.

Chapter 5

Adaptive Radio Nodes

Efficient methods for QoS provision are crucial in military wireless networks. More or less all the networking layers are involved and there is a need to both understand cross-layer interactions and what can be done on a single layer. When controlling the network resources for QoS it is important to assign them where and when they are needed. At the lowest level, the adaptive radio node level, adaptive rate, power and antennas can be used. However, priority queuing, i.e., how packets are treated and given priority in a node, will also have a great importance and is a fairly simple QoS method to implement. No interaction with other nodes, and limited interaction with other layers, are needed.

The research focus has been on assessing the benefits from using variable data rate and queuing systems. We have considered static and mobile networks as well as two traffic types, best effort traffic and delay sensitive traffic.

The four first sections each summarizes one of four reports previously published within the project [20–22, 29]. The first section investigates possible throughput gains that can be obtained by using variable data rates for best effort traffic in a static network. The second section investigates how good some queuing systems are in treating traffic classes according to their priorities. In the third section four queuing systems are employed, and for those the throughputs for best effort traffic, delay sensitive traffic, and a mix of those, are estimated. The last section also consider variable data rates, but in this case for delay sensitive traffic in a mobile scenario. Finally, in the last section we combine the results into some more general conclusions.

5.1 Throughput Gains Through Using Variable Data Rates

To be able to vary the data rate is clearly an advantage. The use of variable data rates both enables the network to increase its throughput and to adapt to changes in the environment, which is crucial for the ability to guarantee QoS in a mobile network. This section summarizes the results from [20], where it is investigated how to use variable data rates in ad hoc TDMA networks, and the possible gains in throughput. The study also includes the impact of the routing metric and traffic adaptivity on the network's ability to utilize variable data rates. Also included is how the number of data rates to chose from improve the throughput.

Pre-requisites

The radio network is modeled as a set of nodes and the basic path loss between any two nodes, see Section 3.3. Nodes are placed randomly within a given terrain area. The path losses between the nodes are calculated using a terrain data base and a ground wave propagation library [39]. Based on the path loss, the SNR value on a link can be obtained.

Depending on the SNR on the link, the data rate is choses as high as possible, while still meeting bit error rate budget and the goal is to achieve the highest possible throughput. In this work, we have used six different data rates, starting with 100 kbit/s as Level 1 and ending with 20 Mbit/s as Level 6. The SNR and data rates used in our model correspond to an information block size of 256 bits at a packet error probability of 10^{-4}, with a bandwidth of 10 MHz. To determine which data rates that can be used we use results from [45].

As MAC protocol we investigate TDMA without and with traffic adaptivity, i.e., either the nodes are allocated only one time slot each, or each node is allocated time slots corresponding to the traffic load that the node is exposed to.

The transmission time for a packet is denoted T_p, and the duration of a time slot T_s. Depending on the data rate on the link, T_p varies and a node can transmit different number of packets during a time slot. To optimize the use of each time slot, as many packets as possible are sent in each time slot. In the end of each time slot, a guard time, T_g, is inserted to avoid collisions on the channel, see Figure 5.1.

We use minimum cost routing with two different cost metrics. In the first

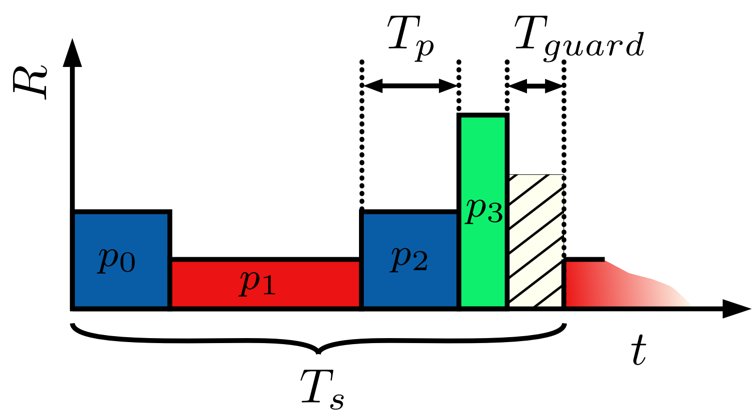


Figure 5.1: Example of transmission of multiple packets in one time slot.

	No traffic adaption	Traffic adaption
Minimum hop	$S_{1,1}$	$S_{1,2}$
$1/R$	$S_{2,1}$	$S_{2,2}$

Table 5.1: The four systems in form of combinations of routing protocol and MAC protocol.

case, the cost for all links are equal to one (minimum-hop routing). In the second case, we use a metric where the cost for using a link is inversely proportional to the data rate ($1/R$) on the link. This creates a routing table that minimizes the channel utilisation needed to transport a packet to its destination node.

Furthermore, we consider unicast traffic, i.e., a packet has a single source and a single destination. The four combinations investigated are shown in Table 5.1

Results

In [20] we investigate relative throughput gains, throughput-delay characteristics and possible gains by adding additional lower data rates. Here we only present results for the possible throughput gains. Figure 5.2 illustrates the ratio between the throughput for modulation group $M_{1,1}$ to $M_{1,6}$, and the first modulation group $M_{1,1}$ (100 kbit/s). For example, $M_{1,4}$ denotes that the data rate

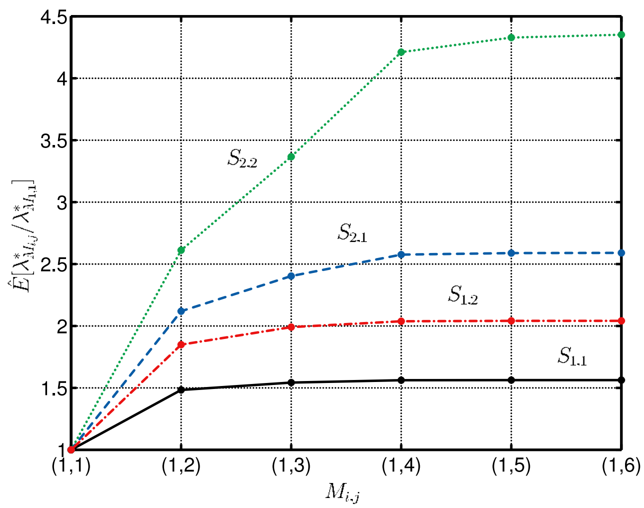


Figure 5.2: Relative throughput gain, $\hat{E}[\lambda^*_{M_{i,j}}/\lambda^*_{M_{1,1}}]$, for the four systems when we increase the number of available data rates.

Level 1 to 4 is used. We start with modulation group $M_{1,1}$, which also means that the network is fully connected at the lowest data rate. Higher data rates are then added gradually to the system, but the same network topology is retained.

For systems $S_{1,1}$, $S_{1,2}$, and $S_{2,1}$ we can see a clear improvement when adding one additional data rate to the first one, but the improvements from additional data rates are not so pronounced. The reason why the ratio for $S_{1,1}$ increases when the second data rate is added is probably because a bottleneck link gets a higher data rate. However, when additional higher data rates was added, the links that did not get higher data rates when $M_{1,2}$ was introduced, will certainly not get higher data rates when $M_{1,3}$ is introduced. These links will therefore probably become bottleneck links. The small improvement is likely the result of some local redistribution of the nodes resources when adding higher data rates. System $S_{1,2}$ has a higher ratio than $S_{1,1}$, due to the traffic adaptivity.

The increase in performance for system $S_{2,1}$ is due to that the routing will choose links with higher data rates. However, since the system has no traffic adaptivity, nodes with low data rate links will soon become the bottlenecks of the network.

Finally, for system $S_{2,2}$, i.e., for traffic adaptivity and $1/R$ routing, the improvement of having a high number of data rates available is much greater. The improvement of going from one data rate to four data rates is approximately a factor 4.3. The reason for the significant increase, resulting from adding the first data rates is partly due to the use of links with higher data rates and partly due to the traffic adaptivity, which adjusts the resources. However, having five or six data rates, $M_{1,5}$ and $M_{1,6}$, compared to four, $M_{1,4}$, gives a very minor increase in throughput. The reason why the ratio does not increase for adding the last data rates is probably that the traffic adaptivity cannot handle this big a difference in data rate in a proper way. Due to the limited frame length of the TDMA schedule we get a cource resolution when distributing the capacity among the nodes. This leads to difficulties in dealing with nodes having a great dynamic in data rates. On the other hand, a very long frame length leads to other problems. Furthermore, the number of links that have channel condition necessary to meet the SNR requirements is limited for these high data rates.

Even if it is not shown here, let us mentioned another of our findings. When lower data rates are added to an already fully connected network, only minor improvements are possible.

Conclusion

The results show that it is of paramount importance to take data rate into consideration when routing, especially when a variable data rate is used, to be able to add long range low data rate links. We also note that traffic adaptivity plays an important role for the utilization of variable data rate at the network level. Furthermore, the gain of having great dynamic in data rates, i.e. having many data rates to choose from, is not obvious. The results also show a substantial improvement in throughput when variable data rate is used.

5.2 Priority Queuing

This section summarizes some of the results from [29], where the possibility of providing a QoS mechanism in ad hoc networks by using priority queuing is examined.

Queuing Systems

In priority queues, the packets are assigned a priority as a function of the traffic, or service class, and then the packets are served in decreasing order of priority.

Fixed Priority Queuing

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 since packets that belong to the service class with the highest priority are always served first. So packets that do not belong to that service class are thus not guaranteed any service at all; they are merely given what is left after the highest priority class has been served.

Weighted Fair Queuing

Weighted Fair Queuing (WFQ) was first introduced in [46]. It was also developed in parallel under the name *Packet-by-packet Generalized-Processor Sharing* (PGPS) in [47], and is a packet approximation of the *Generalized-Processor Sharing scheme* (GPS) [47]. WFQ allows a minimal percentage of the total capacity to be allocated to each service class and uses proportionally fair sharing of any excess capacity. In other words each service class is guaranteed a minimum service rate, and any excess capacity is shared fairly between active service classes.

Results

To illustrate the behaviour of the queuing systems, we show results from one of the scenarios in [29]. The scenario is a static network consisting of 40 nodes. Three service classes are considered, class 1, class 2 and class BE. Each service class has an average arrival rate of $\lambda_N/3$ packets/time slot. The resource allocation parameters in WFQ are set to $\phi_1 = 0.7$ and $\phi_2 = 0.3$. Thus, class 1 with the highest priority is assigned 70 percent of the resources, and class 2 with the second highest priority is assigned 30 percent of the resources. The best effort class (class BE) gets the remaining resources whenever something is left.

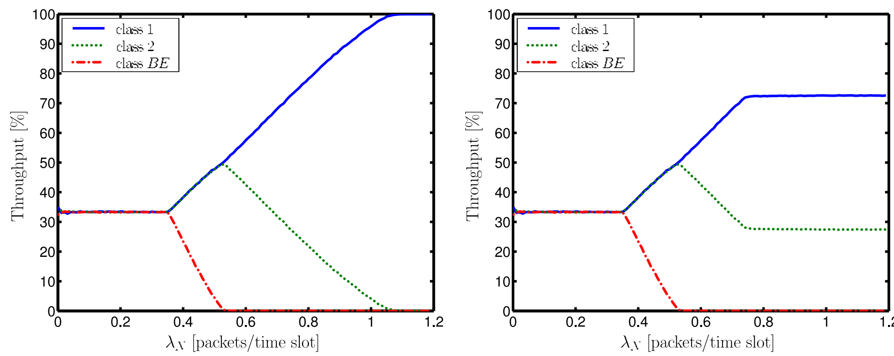


Figure 5.3: Throughput for the three service classes as percent of total throughput in a generalized TDMA network with FPQ to the left and WFQ to the right.

An interesting measure to look at is the throughput, which we define as the average number of packets per time slot delivered to their final destination. The throughput for the three service classes is shown in Fig. 5.3 as a percentage of total throughput in the network for both FPQ and WFQ.

We see that the behavior of the BE class is essentially equal for the two queuing systems. This is expected, since from the BE point of view, the two queuing systems work as an FPQ with 2 service classes, the low-priority BE class and the high-priority class consisting of the original class 1 and class 2. The mutual ordering between classes 1 and 2 in the high-priority class is done with the corresponding queuing system.

Furthermore, we see that, in FPQ, the performance of class 2 is suppressed in favor of class 1, whereas in WFQ, the resources are shared between the two service classes according to the resource allocation. For WPQ service class 1 is suppressed, compared with FPQ, to give service class 2 its fair share of the resources. With the resource allocation parameter, ϕ_i, we can adjust allocation and, in the limiting case, when $\phi_1 \rightarrow 1$ and $\phi_2 \rightarrow 0$, WFQ will behave like a FPQ scheme.

Conclusions

The evaluation of FPQ shows that it gives a very distinct delay differentiation [29], i.e. there is a very distinct difference in network delay between packets in the high-priority classes packets in and classes with lower priority. The high-priority class can in fact dominate so much that no other traffic can pass through the network. This is because in FPQ, high-priority classes always take precedence over low-priority classes. Is this a desirable property? It certainly has its applications in a military context where, for example, *priority messages*[1] should always take precedence over all other traffic. However, it might not be the best way to differentiate between traffic that has different priorities for technical reasons, because in this case the priorities do not indicate the importance of the traffic and therefore it is no longer obvious that the prioritized traffic always should take precedence over other traffic.

WFQ, on the other hand, provides the means to control how much of the resources that are dedicated to a specific service class. Consequently no service class can totally dominate the network. One interesting property with WFQ is that if it is combined with an *admission control policy* that controls how much traffic that is allowed to enter the network, we can give absolute end-to-end guarantees if the arrival process fulfills certain constraints [47]. This scheme might thus be better suited for creating QoS for technical reasons.

We conclude that both of the evaluated queuing schemes have their advantages and disadvantages, and none of them alone is likely to be the full answer to providing QoS. Instead, a combination of them could be used. For example; there could be a FPQ with three service classes on top: class 1 for *priority messages* and the like; class 2 for traffic that is prioritized for technical reasons; and class 3, a best effort class. Class 2 could then be divided into subclasses, and a WFQ scheme could be used to determine the mutual ordering within that class.

5.3 QoS and Throughput Tradeoffs

To which extent QoS can be provided at a reasonable cost is a key issue in ad hoc networking. The paper [21] investigates the tradeoffs between QoS and the cost in bandwidth for some sample tactical networks. In particular, we gain insights into the issue of the cost of transmitting delay sensitive traffic. This cost is measured in terms of reduced throughput when delay sensitive traffic is served, compared to that when there is only best effort traffic. It is also of interest to investigate what happens in the normal case when all packets are treated equally on a First Come, First Served (FCFS) basis, to the case when some simple

[1]Military term for messages that are allowed to interrupt all other messages.

queuing systems are applied. We have not applied advanced queuing systems, where the issue of fairness also needs to be considered.

Pre-requisites

The simulated networks are static and they are generated in a way similar as described in Section 5.1, also see Section 3.3. Traffic adaptive TDMA and minimum-hop routing are employed.

We consider two traffic classes, the first class is Best Effort (BE) traffic and it is modeled as unicast traffic, i.e. a packet entering the network has only one destination. The packet size is equal to 256 bits. The second traffic class is called Constant Bit Rate (CBR) and models unicast voice calls between nodes. However, when the delay constraint is imposed in the measurement we call the traffic Delay sensitive Constant Bit Rate (D-CBR). The parameters for a voice call are taken from the GSM standard (Enhanced Full Rate vocoder) [48]. The vocoder produces bits at a rate of 12.2 kbps, and we assume the packets are of size 256 bits. The maximal acceptable delay between sender and receiver is selected, as recommended by ITU [37], to 150 ms. The mean call duration time is set to 12 s. A call is succesful if at most 5 percent of the packets are delayed more than 150 ms.

Queuing Systems

The maximum queue length in a node is here set to 500 packets. The four different methods of serving packets in the nodes are:

First-come-first-serve (FCFS): The most common way to deal with packets, the packets are served in the order of arrival.

Delete packets only: Requires that a time stamp for when the packet first is transmitted is stored in the CBR packets. The time stamps are checked in the nodes and a packet is deleted when the delay timer of 150 ms has expired. The advantage of this method is that obsolete packets are removed and do not occupy network resources.

Priority queue 1: The CBR packets are stored in an separate queue and served on a separate first-come-first basis. BE packets are only served once the CBR packet queue is empty.

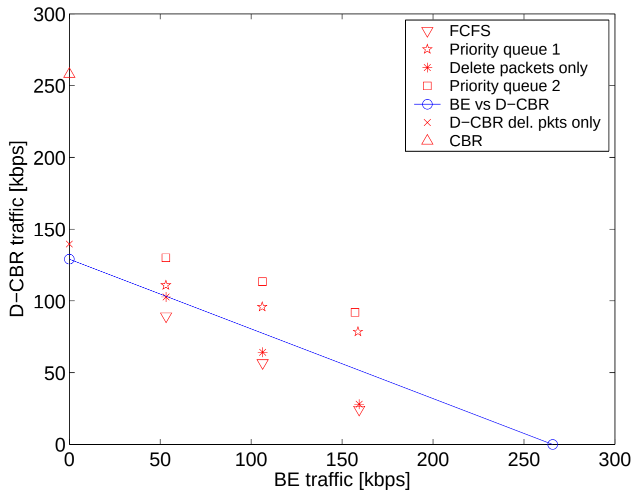


Figure 5.4: Results for a 40-node network: 1.8 number of hops in average

Priority queue 2: As above, with the addition that obsolete CBR packets are deleted.

Results

To get a picture of the QoS and throughput tradeoffs we first consider a reference system (FCFS) without any QoS adaption, i.e., all packets are treated equally in the nodes. The reference system is then compared to systems with priority queuing.

Figure 5.4 presents results for one of the networks considered, a 40-node network with an average number of hops of 1.8. The throughput of BE traffic is shown on the x-axis and the throughput of D-CBR traffic on the y-axis. The maximal throughput for CBR (without delay requirement) is marked on the y-axis. The total throughput for a mixed traffic is simply obtained by summing up the throughputs on the x- and y-axis.

The straight line (D-CBR vs BE), connects the points for only D-CBR traffic and only BE traffic. When the traffic mix is scalable the results should follow this line if the traffic mix is scalable. However, as can be seen for *FCFS* the total

	20-node Networks			40-node
	No.1	No.2	No.3	No.4
D-CBR	0.58	0.49	0.36	0.48
Mixed traffic/ Normal (FCFS)	0.67	0.61	0.47	0.57
Delete packets only	0.73	0.67	0.51	0.62
Priority queue 1	0.91	0.86	0.68	0.74
Priority queue 2	0.97	0.96	0.78	0.84

Table 5.2: Reduced throughputs for D-CBR and mixed traffic as compared to BE traffic only: see the text for further explanations.

throughput falls below the line. This means that an increasing amount of BE traffic affects the D-CBR traffic negatively. On the other hand, for *Delete packets only*, *Priority queue 1*, and *Priority queue 2* we can see an improvement in throughput for the mixed traffic. In particular, we can see a substantial gain with *Priority queue 2*, i.e., when packets older than 150 ms are deleted, and the queue system prioritizes D-CBR traffic. Furthermore, it can be noticed that the small measure of deleting packets gives an improvement in throughput for the case when we only have D-CBR traffic (the second point ($\times$) on the y-axis).

In [21] we tested the queuing systems for four different type of networks. The results are summarized in Table 5.2 for pure D-CBR traffic and for one of the possible traffic mixes (50% BE and 50% D-CBR). The No.1 network is a dense 20 node network (1.3 number of hops in average), No.2 a medium dense 20 node network (1.7 number of hops in average) and No.3 is a sparse 20 node network (2.7 number of hops in average). The No.4 network is the 40 node network used to get the results in 5.4.

The numbers presented in the table are normalized throughputs, or reduced throughputs when compared to BE traffic. For mixed traffic we obtain the numbers, by summing up the throughputs of the two traffic classes when they are equal, and dividing with the maximum BE throughput.

In the table (first row) we can for example see, that the normalized throughput for network No.3 is 0.36, i.e., the cost of sending D-CBR traffic is 2.8 times higher than the cost of sending BE traffic. Noticeable is also that sending D-CBR traffic is more costly in the sparse Network (No.3), compared to the denser networks (No.1 and No.2), with fewer hops per route. Furthermore, for the mixed traffic, the improvement in throughput for *Priority queue 2* compared to using FCFS is substantial (e.g., from 0.47 to 0.78 for No.3).

Conclusion
The cost is here measured in terms of reduced throughput when delay sensitive traffic is served, compared to best effort traffic only. The results show a substantial cost increase for serving delay sensitive traffic. The cost increases more than two times in some of the cases. Of particular interest is the much better througputs that can be obtained for mixed traffic by fairly simple methods, such as deleting old packets and using priority queues. Nearly two times improvements can be seen in the examples tested. Using those methods becomes essential when a mix of traffic is served, in particular when a large portion of the traffic is of best effort type.

5.4 Variable Data Rate and Delay Sensitive Traffic

Efficient resource management is crucial in order to deal with delay sensitive traffic. The paper [22] presents results from studying such traffic in a mobile scenario and focuses on the gains that can be obtained by using variable data rates. With respect to this, we have studied the effect of adding lower and higher data rates, the importance of the routing metric, and the effects of delayed routing update.

Pre-requisites

The tested networks are generated in a way similar to that in Section 5.1, also see 3.3. However, in this case we consider mobile networks. In the scenario the nodes move around randomly for 2000 seconds in a given area, employing a random walk mobility model, whith the speed of 20 m/s.

The delay sensitive traffic is modelled as unicast sessions between nodes with a delay constraint of 150 ms, i.e., the traffic described in Section 5.3 and called D-CBR. The packets are 256 bits and the mean session duration time is set to 12 s.

Six different data rates are considered, starting with 0.25 Mbps as Level 1 and ending with 8 Mbps as Level 6. However, different rates as compared to level 1 to 6 in Section 5.2. In this case, the MAC protocol needs to adapt to mobility as well as traffic changes. To include also mobility adaption, we have used a centralized and near optimal method to decide which node may use a certain time slot in a TDMA based protocol [22].

Depending on the data rate on a link, a node can transmitt a different number of packets in a time slot, see Figure 5.1. A packet can be fragmented down to block sizes of 64 bits. The length of a time slot is adjusted so that a packet of 256 bits precisely fits into the time slot at 1 Mbps.

We investigate minimum cost routing with two different cost metrics. In the first case, the cost for all links are equal to one (minimum-hop routing). In the second case, called *data rate based routing*, we use a metric where the cost for using a link is inversely proportional to the data rate on that link ($1/R$ routing).

Results

An example of the results from [22] is shown in Figure 5.5. The network consists of 40 nodes and has a connectivity of 0.98 when only 1 Mbps links are used. Connectivity is here measured as the fraction of existing point-to-point connections and averaged over a simulation run. The success rate is defined as the percentage of successful CBR sessions during a simulation run. A session fails if more than 5 % of the packets are delayed more than 150 ms during the session. The same scenario is repeated for different traffic loads, i.e. for diffrent numbers of CBR flows or sessions.

By also allowing high data rate links, 2 Mps, 4 Mps and 8 Mps, we can see that almost twice the number of sessions can be supported as compared to only using data rates up to 1 Mps. In 5.5 we can also see a gain by allowing low data rate links with 0.25 Mps and 0.5 Mps, a closer study however revealed that almost all the gain is due to increased connectivity. Using a data rate based metric is important, corresponding simulations with a minimum hop metric showed that most of the gains with allowing high data rates disappeard.

Conclusions

From our studies, included in [22], we can draw the following conclusions. When comparing results for the data rate based metric to that of the minimum-hop metric, the latter never gives a better result in terms of success rate at a

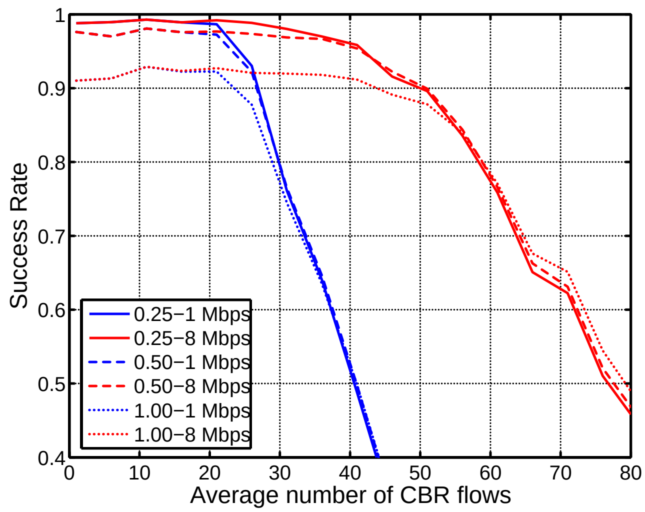


Figure 5.5: Mean success rate for a 40 node network with data rate based routing. The range of allowed data rates is varied according to the legend.

given traffic load (even at very light traffic loads). On the other hand, minimum-hop routing is more tolerant towards long delays between routing information updates, while data rate based routing needs much more accurate routing information.

When considering to add low data rate links, it is adding a data rate that gives a fully connected network that is important. Using links with even lower data rates does not give any further noticeable improvement. The exception is when we have long delays between routing updates, then some improvements can be seen.

Adding very high data rate does not necessarily contribute to a better overall performance. It is likely that the time slot, for such a link, is partly empty. This does not occur for maximum load (full queues), but in that case we cannot support delay sensitive traffic.

It is clear that having the possibility to use variable data rates is an advantage. Moreover, to fully utilize the potential of a large range of data rates the routing and MAC interactions becomes very important. For example, by aggregating traffic over high data rate links, partly empty time slots can be avoided.

5.5 Summary

The studies described in this chapter have provided insight into how to use variable data rates and queuing systems in different situations. Some general conclusions are:

- To be able to vary the data rate is clearly an advantage since it enables the network to increase its throughput and to adapt to changes in the environment.
- The gain from having a great dynamic in data rates, i.e. having many data rates to chose from, is not obvious. For example, a MAC protocol that can handle the dynamic is required.
- The greatest benefit with adding low data rate links, is the increased connectivity in the network.
- For MAC protocols like TDMA with a fixed slot size there is the issue of how to fill the slot efficiently with packets, to avoid that part of the time slot is empty.
- The choice of routing metric is important in order to be able to fully utilize a large range of data rates. The data rate based metric is not the best in all situations but it works satisfactory. For example, the data rate based metric does not consider the number of hops at all. Furthermore, as described in Chapter 4, it may not always be straightforward to include such a metric into the routing protocol.
- Supporting QoS involves a cost in throughput. We have investigated delay sensitive traffic and this sometimes cause a significant reduction in throughput, compared to only supporting best effort traffic. There are different methods that can mitigate this reduction but such methods also involve a cost in terms of increased complexity. A relative simple way to mitigate some of the reduction is to use queuing systems.
- Simple queuing systems, which do not involve any additional interaction with other nodes, provide fairly good service class differentiation in average in ad hoc networks. In average, this means that we can not be sure that

a particular session with high priority will be satisfactory handled. Thus, simple queuing systems alone are not sufficient. To provide satisfactory QoS further mechanisms are needed.

An adaptive radio node may have more features then just to be able to vary the data rate, for example by also adapting the power, additional improvements are possible. This type of features, together with priority queuing are reasonably straightforward to integrate into ad hoc networking. To go a step further, smart antennas, or MIMO type of systems, could be used. Such systems have been shown to give significant improvements, but it is far from straightforward to integrate them into ad hoc networking. The MAC protocols in particular, but also the routing, need to be modified to take full advantage of e.g., MIMO systems.

Chapter 6

Conclusions

This report investigates issues concerning network control for ad hoc networks and in particular the design of MAC and routing protocols. One type of protocol seldom suits all situations. Instead different protocols are preferred for different situations and networks. Some important factors that determines the protocol design are, node mobility, number of nodes and types of traffic. Is one traffic type dominating or is it a mix of many traffic types? Clearly, the type of radio node is also important to take into account, e.g., is it an adaptive radio node, and in that case, what link data rates can it support?

We have focused much of the MAC research on STDMA, since it offers a high maximum throughput. This makes STDMA very competitive in a static or semi-static scenario. However, since coordination among nodes is necessary and the schedule needs to be maintained, overhead traffic is generated. This overhead grows with increases in node mobility, traffic changes and number of nodes in the local neighborhood. Thus, in a chaotic scenario with high mobility and a lot of traffic fluctuations the STDMA overhead is large. However, STDMA may still be feasible if we have a large bandwidth. On the other hand, the prefered protocol in such a scenario is normally CSMA since its overhead is almost unaffected by node mobility. Anyway, there are a need for both reservation based protocols like STDMA and contention based protocols like CSMA.

A very large number of routing protocols have been proposed in the literature. They have been designed with different scenarios in mind and they are therefore also suitable in different situations. Exactly which protocol to choose may be of less importance, but there is need for at least one efficient protocol working according to the reactive and proactive principle, respectively. An alternative is a hybrid protocol that can work in both ways. How to choose the parameter settings, or finding ways so the protocol can adapt to the situation is an interesting topic. Furthermore, due to the broadcast or multicast nature of many services, designing efficient routing protocols for such services is another important topic. This topic has however not been considered in this report.

For QoS-handling, the ability to adapt quickly to a new situation is crucial. The fastest response can be obtained at the low layers through an adaptive radio node. By being able to select frequency band, variable data rates, smart antennas, power control and multi-user detection, we then have the ability to adapt the links to the service requirements. A main issue is how tight the coupling between the layers should be. Service parameters could be used to guide the setting in an adaptive radio node. Then, by using the service information, the task of the adaptive radio node is to make its links stable and predictable for the higher layers. For example, particular links carrying vital data can be stabilized by using adaptive radio node features. If the link conditions are changed, the MAC and routing protocols are also affected. The MAC protocol may want to re-assign time slots and the routing protocol may want to send packets on another route with a lower cost metric. However, changes on the MAC and routing layer requires interaction with other nodes and takes time.

In this report we have studied benefits from using variable data rates, and considerable gains are possible. However, how the routing and MAC should interact with the adaptive radio node, that can vary its data rate, depends on the situation and the traffic types. We have also tested diffrent queuing systems, where no additional interaction with other nodes is required. Such systems help in the QoS support but they alone are not enough, additional QoS mechanisms are needed.

Bibliography

[1] J. Grönkvist, “Novel assignment strategies for spatial reuse TDMA in wireless ad hoc networks,” *Wireless Networks*, 2005, accepted for publication.

[2] A. Dhamdhere and J. Grönkvist, “Joint node and link assignment in an STDMA network,” in *Proc. of ICC*, 2006, submitted for publication.

[3] E. Johansson, K. Persson, M. Sköld, and U. Sterner, “Combining distribution of position information with fisheye routing traffic for mobile ad hoc networks,” in *RVK’05*, June 2005.

[4] L. Farman, U. Sterner, and O. Tronarp, “Analysis of capacity in ad hoc networks with variable data rate,” in *IEEE VTC04-Spring*, 2004, reg: FOI-S–1469–SE.

[5] L. Farman, J. Nilsson, and O. Tronarp, “On QoS and throughput tradeoffs for tactical ad hoc networks,” in *Proc. of NATO-RTO-MP-IST-054 Symposium*, Rome, Italy, April 2005.

[6] T. Giles, P. Karlsson, J. Lind, G. Malmgren, J. Markendahl, J. Nilsson, J. Zander, and P. Zetterberg, “Cost drivers and deployment scenarios for future broadband wireless networks - key research problems and directions for research,” in *IEEE VTC2004-Spring*, 2004, reg: FOI-S–1466–SE.

[7] J. Grönkvist, J. Nilsson, and D. Yuan, “Throughput of optimal spatial reuse TDMA for wireless ad-hoc networks,” in *IEEE VTC04-Spring*, 2004, reg: FOI-S–1467–SE.

[8] J. Grönkvist, “Distributed scheduling for mobile ad hoc networks – a novel approach,” in *Proc. of PIMRC*, 2004, reg: FOI-S–1470–SE.

[9] E. Johansson, K. Persson, M. Sköld, and U. Sterner, “An analysis of the fisheye routing technique in highly mobile ad hoc networks,” in *IEEE VTC2004-Spring*, 2004, reg: FOI-S–1468–SE.

[10] ——, “Routing in tactical ad hoc networks with heterogeneous links,” in *Proc. of 5th Scandinavian workshop on wireless ad-hoc networks*, Stockholm, Sweden, May 2005.

[11] ——, “Reactive routing in ad hoc networks with various link costs,” in *Proc. of RVK 2005*, Linköping, Sweden, June 2005.

[12] K. Persson, U. Sterner, M. Sköld, and E. Johansson, “Distribution of situation awareness data in mobile tactical ad hoc networks using the fisheye routing algorithm,” in *Proc. of NATO-RTO-MP-IST-054*, Rome, Italy, April 2005.

[13] K. Persson, M. Sköld, E. Johansson, and U. Sterner, “Combining distribution of position information with fisheye routing traffic for mobile ad hoc networks,” in *Proc. of RVK 2005*, Linköping, Sweden, June 2005.

[14] M. Sköld, Y. Choi, and J. Nilsson, “An analysis of mobile radio ad hoc networks using clustered architectures,” in *Proc. of VTC03-Spring*, 2003, reg: FOI-S–1248–SE.

[15] O. Tronarp, “Quality of service in tactical ad hoc networks by priority queuing,” in *3rd Swedish Workshop on Wireless Ad-hoc Networks*, 2003.

[16] ——, “Quality of service in tactical ad hoc networks by priority queuing,” in *In Proc. 5th European wireless conf.: Mobile and wireless systems beyond 3G*, February 2004, reg: FOI-S--1259–SE.

[17] J. Grönkvist, “Distributed STDMA algorithms for ad hoc networks,” Swedish Defence Research Agency., Div. of Command and Control. Linköping, Sweden, Scientific Report FOA-R--1059--SE, December 2003.

[18] ——, “Overhead traffic for distributed STDMA algorithms,” Swedish Defence Research Agency., Div. of Command and Control. Linköping, Sweden, Technical Report FOA-R--1338--SE, September 2004.

[19] ——, “Interference-based scheduling in spatial reuse TDMA,” KTH, Stockholm, Sweden, Doctoral Thesis TRITA-S3-RST-0515, September 2005.

[20] L. Farman, U. Sterner, and O. Tronarp, “Analysis of capacity in ad hoc networks with variable data rates,” Div. of Command and Control Systems, FOI, Swedish Defence Research Agency, Technical Report FOI-R--0928–SE, June 2003.

[21] L. Farman, J. Nilsson, and O. Tronarp, “On QoS and throughput tradeoffs for tactical ad hoc networks,” Div. of Command and Control Systems, FOI, Swedish Defence Research Agency, Technical Report FOI-R–1425–SE, December 2004.

[22] ——, “Using variable data rates in mobile ad hoc networks supporting delay sensitive traffic,” Div. of Command and Control Systems, FOI, Swedish Defence Research Agency, Technical Report FOI-R–1725–SE, October 2005.

[23] E. Johansson, K. Persson, M. Sköld, and U. Sterner, “AODV routing in ad hoc networks with variable data rates,” Div. of Command and Control Systems, FOI, Swedish Defence Research Agency, Technical Report FOI-R–1430–SE, December 2004.

[24] J. Nilsson, “4th Scandinavian workshop on wireless ad hoc networks,” FOI, Swedish Defence Research Agency, Tech. Rep. FOI-Memo-910, June 2004, in swedish.

[25] ——, “5th Scandinavian workshop on wireless ad hoc networks,” FOI, Swedish Defence Research Agency, Tech. Rep. FOI-Memo-1359, June 2005, in swedish.

[26] K. Persson, “Quality of service routing in tactical mobile ad hoc networks,” Div. of Command and Control Systems, FOI, Swedish Defence Research Agency, Technical Report FOI-R–0886–SE, June 2003, in swedish.

[27] K. Persson, E. Johansson, U. Sterner, and M. Sköld, “The fisheye technique in highly mobile ad hoc networks,” Div. of Command and Control Systems, FOI, Swedish Defence Research Agency, Methodology Report FOI-R–1058–SE, December 2003.

[28] M. Sköld, J.Nilsson, and H. Tullberg, “Diskussionseminarium kring tjänster i ett framtida mobilt radionätverk,” FOI, Swedish Defence Research Agency, Tech. Rep. FOI-Memo-853, March 2005, in Swedish.

[29] O. Tronarp, “Quality of service in ad hoc networks by priority queuing,” Div. of Command and Control Systems, FOI, Swedish Defence Research Agency, Scientific Report FOI-R–1156–SE, January 2004.

[30] M. Sköld, et.al., “Mobile ad hoc networks - challanges and possibilities,” Div. of Command and Control Systems, FOI, Swedish Defence Research Agency, User Report FOI-R–1799–SE, December 2005, in Swedish.

[31] R. Nelson and L. Kleinrock, “Spatial-TDMA: A collision-free multihop channel access protocol,” *IEEE Trans. Commun.*, vol. 33, no. 9, pp. 934–944, Sept. 1985.

[32] I. Chlamtac and A. Lerner, “A link allocation protocol for mobile multihop radio networks,” in *GLOBECOM ‘85, IEEE Global Telecommunications Conference, Conference Record*, vol. 1, 1985, pp. 238–242.

[33] J. Shor and T. Robertazzi, “Traffic sensitive algorithms and performance measures for the generation of self-organizing radio network schedules,” *IEEE Trans. Commun.*, vol. 41, no. 1, pp. 16–21, Jan. 1993.

[34] O. Somarriba, “Multihop packet radio systems in rough terrain,” Tech.Lic. Thesis, Radio Communication Systems, Department of Signals, Sensors and Systems, Royal Institute of Technology, SE-100 44 Stockholm, Sweden, Oct. 1995.

[35] J. Grönkvist, “Traffic controlled spatial reuse TDMA for multihop radio networks,” in *Proc. of IEEE PIMRC’98*, vol. 3, 1998, pp. 1203–1207.

[36] ANSI/IEEE Std 802.11, 1999 Edition [ISO/IEC DIS 8802-11], “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY)

Specifications," http://standards.ieee.org/getieee802/download/802.11-1999.pdf.

[37] "One-way transmission time," International Telecommunications Union ITU-T G.114, 2003.

[38] P. Holm, "UTD-diffraction coefficients for higher order wedge diffracted fields," *IEEE Trans. Antennas Propagat.*, vol. AP-44, no. 6, pp. 879–888, June 1996.

[39] B. Asp, G. Eriksson, and P. Holm, "Detvag-90® — Final Report," Defence Research Est., Div. of Command and Control Warfare Technology, Linköping, Sweden, Scientific Report FOA-R–97-00566-504–SE, Sept. 1997.

[40] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-demand Distance Vector (AODV) routing," ftp://ftp.rfc-editor.org/in-notes/rfc3561.txt, July 2003.

[41] "AODV - Ad hoc On-Demand Distance Vector Routing," "http://moment.cs.ucsb.edu/AODV/aodv.html.txt", 2004.

[42] C. Perkins and P. Bhagwat, "Highly dynamic destination- sequenced distance-vector routing (DSDV) for mobile computers,," in *ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, 1994.

[43] M. G. G. Pei and T.-W. Chen, "Fisheye state routing in mobile ad hoc networks," in *Workshop on Wireless Networks and Mobile Computing*, 2000, pp. 71–78.

[44] M. Gerla, X. Hong, and G. Pei, "Fisheye State Routing protocol (FSR) for ad hoc networks," IETF Internet Draft (work in progress), June 2002, draft-ietf-manet-fsr-03.txt.

[45] D. D. S. Dolinar and F. Pollar, "Code performance as a function of block size," TMO Progress Report 42-133, May 1998.

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

[47] A. Parekh and R. 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.

[48] “Digital cellular telecommunications system (Phase 2+); Enhanced Full Rate (EFR) speech transcoding; (GSM 06.60 version 8.0.1 Release 1999),” European Telecommunications Standards Institute, ETSI EN 300 726 v8.0.1, nov 2000.