FFI-rapport 2012/01884

NATO Narrowband Waveform (NBWF) – multilevel precedence and preemption for IP traffic

Tore J. Berg

Norwegian Defence Research Establishment (FFI)

22 October 2012

FFI-rapport 2012/01884

1175

P: ISBN 978-82-464-2177-3
E: ISBN 978-82-464-2178-0

Emneord

Modellering og simulering

Datanett

Radionett

Trådløse nett

Approved by

Torunn Øvreås	Project Manager
Anders Eggen	Director

English summary

NATO has an ongoing activity with the objective to develop a narrowband waveform (NBWF) standard. This is a single-channel mobile ad-hoc network (MANET) which shall serve voice traffic and data traffic over a 25 kHz radio channel. A Multi-Level Precedence and Preemption (MLPP) service is regarded as a mandatory service in military MANETs that operate in the tactical domain.

MLPP is a network service supporting priority labelling of the IP packets according to their importance as determined by the IP clients. During network congestion, the NBWF core protocols shall first serve the packets with the highest priority. IP Quality of Service (QoS) classes and MLPP are two different subjects. The former specifies a method to differentiate between application types which demand different communication channel characteristics (e.g., low jitter, short delay). This in contrast to the MLPP priority level that shall reflect the importance of the information content.

A narrow band network has low throughput capacity, and to experience network congestion during usage must be regarded as an ordinary event. Therefore it is important to have a robust MLPP service which increases the passability of the highest priority traffic during periods of network congestion. Many tactical communications scenarios may take benefit of the MLPP service: 1) Unforeseen external events imply need for more network capacity than predicted; or 2) The radio environment became more difficult than expected (e.g., higher pathloss, network jamming or network interference from friendly forces); or 3) Higher user mobility than planned introduces more routing traffic and more radio hops.

The MLPP service is based on a cross layer design where each layer implements priority handling. Each node has a priority aware buffer system for storing the input traffic from the IP clients. A node internal flow control function between the adjacent layers provides for service of the highest priority traffic.

This document addresses also adaptive MAC scheduling. The purpose of adaptive MAC scheduling is to maintain an optimum throughput/delay-performance under changing traffic load in a fully meshed network. NBWF uses a radio with a large coverage area and a typical network has good connectivity. The benefit of adaptive MAC scheduling increases as the network size (number of nodes) increases.

The adaptive MAC scheduling works in consort with the MLPP but is not a prerequisite for the MLPP service. The MLPP service shall be a mandatory service for NBWF, while the adaptive MAC scheduling may be specified as optional since this function has less importance for the network users than the MLPP service. However, some of the simulation experiments indicate increased network performance when the adaptive MAC scheduling is implemented in addition to MLPP.

Sammendrag

Nato har en pågående aktivitet for standardisering av en smalbånds bølgeform (NBWF) for bruk i VHF området. FFI har en modellerings- og simuleringsaktivitet som skal bidra til Nato aktiviteten ved å vurdere alternative protokollfunksjoner for betjening av tale og datatrafikk i distribuerte mobile nett. Dette dokumentet omhandler flernivåprioritet (MLPP) i NBWF. En MLPP tjeneste skal sikre god fremkommelighet for IP-trafikk med høy prioritet i de perioder der nettet ikke har kapasitet til å betjene all trafikk.

Et smalbåndnett har lav trafikkapasitet og metningssituasjoner kan lett oppstå. I et IP-nett uten MLPP har ikke IP-klientene et verktøy som lar dem bestemme hvilken trafikk som skal avvikles i nettet. En robust MLPP tjeneste er derfor påkrevd når nettet skal betjene militær trafikk i taktisk sone. Her kan uforutsette eksterne hendelser medføre at trafikkbehovet blir større en planlagt. Samtidig medfører bruk av radio med rundtstrålende antenner at transmisjonskapasiteten er vanskelig og predikere. Jamming og interferens fra andre radio systemer er ofte også en utfordring i taktisk sone.

Denne rapporten beskriver også en adaptive MAC (Medium Access Control) protokoll der en rekke MAC-parametre reguleres dynamisk etter påtrykt IP trafikk. Hensikten med dette er å gi optimal IP-ytelse (minimum forsinkelse og minimum pakketap) under alle lastnivåer.

Rapporten inneholder en rekke simuleringseksperimenter som kvantifiserer gevinsten av MLPP og adaptiv MAC. Rapporten konkluderer med at MLPP er et nødvendig verktøy for IP-klientene, mens det å utelate adaptive MAC ikke er så kritisk.

Contents

1 Introduction

The NATO NC3B CaP/1 LOS CaT has an ongoing activity with the objective to develop a narrowband waveform (NBWF) standard [1]. This is a single-channel mobile ad-hoc network (MANET), which shall serve voice and data traffic over a 25 kHz radio channel.

In the past, Multi-Level Precedence and Preemption (MLPP) has been a mandatory service in connection oriented military networks. With the introduction of datagram oriented IP services, the situation is much the same but an MLPP service must now operate on packet streams and not connections. The purpose of this document is to establish a common understanding of how an MLPP service shall behave in NBWF as seen from the IP clients' point of view. MLPP shall specify the importance of the information content and not the application type [9], and the priority level shall be set by the users[1] of the network. In the NBWF simulator [4,5,6], an MLPP value is assigned per call basis, or per packet basis at OSI layer 7, and four levels are supported. The MLPP function handles the traffic strictly after rank and not by the application type. An MLPP function must be based on a cross layer design to have good performance, and this document analyses the performance of the MLPP service outlined in [4, section 7.1], focusing mostly on layer 2a issues; the medium access (MAC) layer.

In this document, many throughput plots have been produced to communicate the performance of the MLPP service. Figure 1.1 illustrates a typical throughput plot versus offered traffic. As long as a network operates in the stationary state, no IP packets are lost since the throughput is identical to the offered traffic. When the saturation point is reached, the task of the MLPP is to sort the traffic according to priority and serve the highest priority traffic first.

We define the *throughput capacity* of a network as the point where the network starts to discard packets, which are 860 bytes/s in this example[2]. Only *loss tolerant* IP applications can continue to operate above this load level. To decide how many nodes that shall operate on the same radio channel, is an important outcome of traffic planning and the figure illustrates that these network users require 600 bytes/s. This gives a spare capacity of 260 bytes/s. The spare capacity must be dimensioned according to the consequences of not fulfilling throughput requirement, the uncertainty of the radio coverage predictions, etc.

The task of an MLPP function is to execute admission control on the traffic from the IP clients such that the high priority traffic is served at the sacrifice of the lower priority traffic. Chapter 2 explains the behaviour of an ideal MLPP service, and this will be the design goal for an MLPP service for the unicast IP transport service in NBWF. Multicast IP traffic is expected to have a higher traffic volume than unicast traffic. The reason why this document does not address multicast is that the NBWF-project has not yet designed the multicast protocols. However, the MLPP service specified herein is also applicable for multicast traffic.

---

[1] A user is a person sending a message or a computer process.

[2] This network, using mode N1 (20kbps), has allocated 2 TDMA slots for voice relays and only 4 of 9 slots are available for data traffic. Then 8.9kbps gross rate is available for IP traffic.

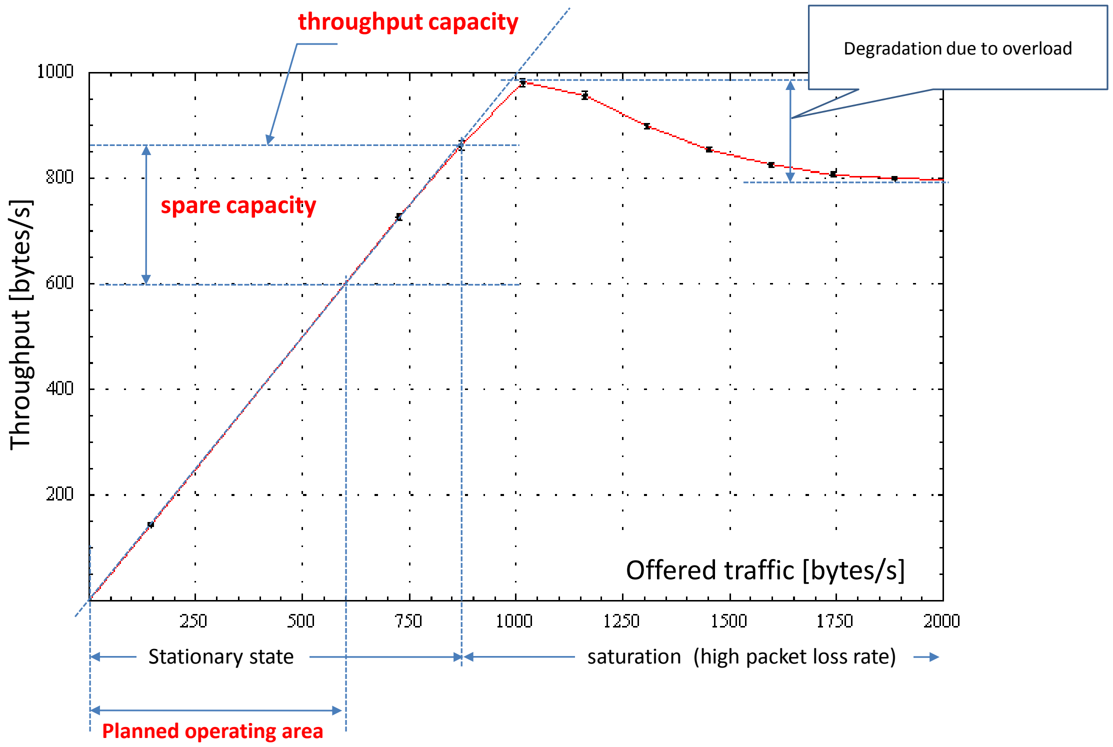

Figure 1.1 Illustration of throughput capacity vs. offered traffic.

At the first glance, chapter 3 “Adaptive MAC Scheduling” may be regarded as a misplaced topic in this report. However, the NBWF MAC protocol may require a function which regulates the random access delay [7, section 4.6.1] according to the traffic load level (number of active nodes), and the MAC layer component of the MLPP service is affected directly by the adaptive MAC scheduling. These two functions, MLPP and adaptive MAC, must therefore be treated as cumulative functions. Chapter 3 outlines the principles of adaptive MAC scheduling while chapter 4 analyses its performance.

We have implemented an MLPP service in the NBWF simulator and the purpose of chapter 5 is to make more specific statements on how a real network behaves. The simulation experiments presented here are based on preliminary MAC parameters but they suffice to illustrate the performance of the MLPP service; how much do the performance plots deviate from the idealised performance plots in the chapter 2?

Implementing MLPP support in the MAC protocol is basically to delay low priority packets such that the high priority packets improve their chance to win. This has an implementation cost in form of lost transmission capacity as well as increased protocol/software complexity. So, why implement priority in the MAC layer? Why is it not sufficient that the layers above MAC implement MLPP? Chapter 6 deals with these questions by analysing the performance of MLPP networks with and without MLPP support at the MAC level.

An important part of the NBWF protocol design is to identify external events that stress the NBWF core protocols. Chapter 7 identifies some challenging operating conditions and analyses some of them.

All the experiments in this document are conducted on single-hop networks (a fully meshed topology) operating in an excellent radio environment. The networks consist of either 25 or 50 nodes, and the IP clients use either short packets (10 bytes) or long packets (1500bytes). In the scenarios simulated, the throughput capacity will be low since the networks are configured to have two voice relays [7, section 4.4] and use mode N1 (20kbps) [6, table 4.1]. Then we have a gross transmission capacity of 1111 bytes/s for IP traffic, see Figure 1.2.

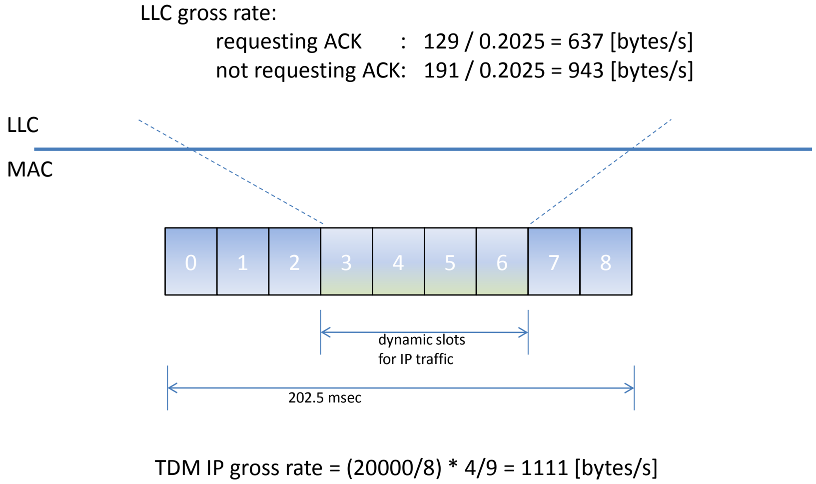

Figure 1.2 IP traffic can use the slots numbered 3 to 6 since the TDMA allocation scheme reserves 3 slots for multicast voice and 2 slots for other application such as network management and routing. The LLC gross rate is calculated from the overhead we currently have in the NBWF simulator.

Our choice of using mode N1 (20kbps) doesn't affect the conclusions since the focus is the shape and the relative magnitude of the performance plots, and not the absolute network throughput capacity.

A real network must guarantee a maximum packet lifetime. For NBWF, the maximum packet lifetime is set to 60 seconds[3]. Below we give an overview of the other scenario parameters.

Network parameters:

Number of nodes: 25 or 50
Number of voice relays: 2
Pathloss: Fixed 10dB (low loss since the network shall operate under excellent SNR conditions).

---

[3] The NBWF core protocols demand a limit for reusing unique identifiers (e.g., the packet duplicate filtering function implemented in the network layer).

Traffic parameters:

Traffic pattern: Uniformly distributed over all nodes
Payload length: Fixed size 10 bytes or 1500 bytes
Priority distribution: {0.1,0.4,0.4,0.1}, see chapter 2
ARQ: Enabled

Other parameters of interest for the scenarios simulated are shown in Table 1.1.

Parameter		Value	Unit
$f_{payload}$	Payload transmission rate	20000	kbits/s
$t_{v,pcas}$	PCAS detection delay	1.75	msec
$t_{v,cas}$	CAS detection delay	2.50	msec
t_{on}	Receive to transmit switching delay	1.00	msec
t_{off}	Transmit to receive switching delay	0.00	msec
t_{som}	Length of the SOM field	2.1	msec
t_{par}	Length of the PAR field	1.6	msec
t_{pci}	PHY PCI (radio overhead: preamble,SOM,PAR)	5.20	msec
n_t	Thermal noise of the receiver	-119.85	dBm

Table 1.1 Physical layer parameter values. Figure 1.3 specifies the meaning of these parameters.

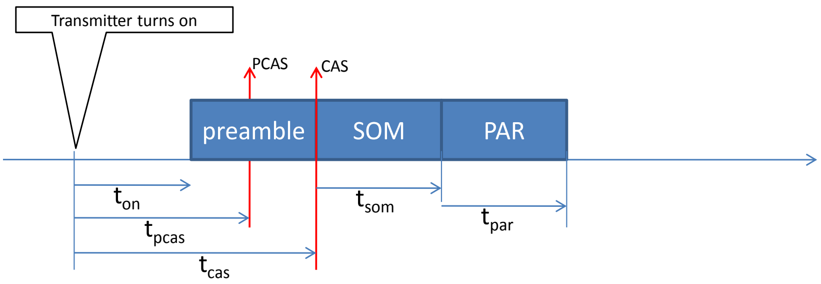

Figure 1.3 Illustration of physical layer parameters.

1.1 Terminology and probes

The first part of this section defines the most important terms used in this report while the second part specifies the probes used and what they measure. A network is a stochastic process and a probe is the tool for observing the network behaviour. In the simulator, a probe is a software component/object which collects data (e.g., end-to-end packet delays) and produces an estimate of the first order moment of the underlying distribution.

Busy/active/idle node

A busy (or active) node has outgoing traffic while an idle node has not.

MAC entity

The active process in a radio node which executes MAC layer functions. For example, it is the MAC entity that executes the MAC protocol.

AHAnN

All-hearing-all (AHA) refers to a network topology where all the nodes have overlapping radio coverage areas. n*N* specifies an AHA network containing *N*-nodes (e.g. AHAn25).

Throughput capacity

When the IP traffic requests use of ARQ, the offered traffic and the throughput shall follow a straight line up to the point where the radio channel becomes congested, see Figure 1.4. The throughput capacity is defined as the point on the curve where the deviation between the offered traffic and the throughput becomes higher than approximately 1%.

Maximum throughput

The highest point on a throughput plot, see Figure 1.4. Only loss tolerant IP applications can operate at this load level.

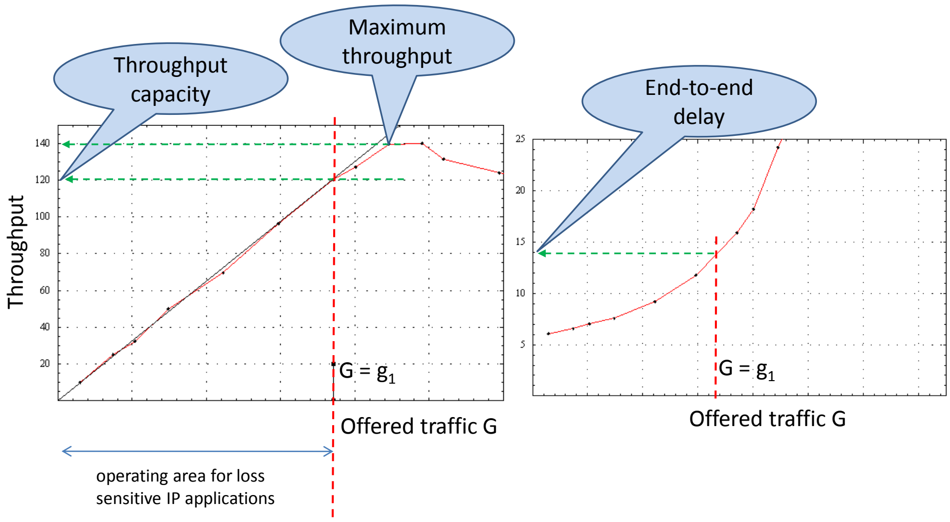

Figure 1.4 Throughput and delay plot examples.

Below we specify the probes used in this report.

Link delay [sec]

End-to-end delay and link delay are the same delays since all the simulation experiments deal with single-hop networks only. The link delay is the IP packet delay from the source IP client to the sink IP client, that is, the delay includes the queuing delays within the network layer.

MAC connection setup delay [sec]
The latency time measured between the **first** *MAC-Connect.request* and the corresponding *MAC-Connect.confirm* which results in a MAC connection. Generally, the LLC entity must issue a number of requests before it wins the channel, see Figure 1.5.

P(receive CC), p_{CC}
This measurement is taken by each node that has sent a CR PDU and expects to receive a CC PDU. When a CC PDU is received, the value sampled is one. Otherwise, zero. Two or more CR PDUs may be sent simultaneously[4] (i.e., a packet collision event) but then the probability to get a CC PDU is very low since NBWF demands a positive signal-to-noise ratio (SNR) to generate a CAS, see [6, table B.4].

MAC load level
Each node takes traffic load level measurements as described in chapter 3, and the load state is either high (sample value 1) or low (sample value 0). Samples are collected at the time instances where the MAC entity draws a random delay.

MAC busy CR nodes, N_{busy} (N_{Gbusy})
With “busy CR nodes” we mean the **number** of nodes that have outgoing CR PDU(s). The MAC scheduling process operates on CR PDUs and not DT PDUs. Therefore it is more interesting to take measurements directly from the CR PDUs when we study MAC scheduling. Each node tracks the number of busy nodes in their neighbourhood by using the process described in chapter 3. Samples are collected at the time instances where the MAC entity draws a random delay.
The simulator has implemented an additional version of this probe (N_{Gbusy}) giving improved accuracy. This probe does not rely on signalling across the radio channel but is implemented by using a global object in the simulator. Both probes are important since the adaptive MAC scheduling also rely on samples from the “MAC busy CR nodes”-distribution. However, the last probe can only be implemented in a virtual world and the benefit is a more correct view of the network state. For example, lost messages due to background noise do not affect its accuracy.

Queue time fresh traffic [sec], QD_{L3a}
This probe measures the queuing delay in the fresh traffic input queues at layer 3a. Only packets which are taken under service by the LLC service provider are included. For example, packets deleted due to lifetime expiry while residing in the fresh traffic input buffers are excluded.

Queue size of fresh traffic input buffer, QS_{L3a}
This probe measures the number of packets in the layer 3a input buffer as seen by an arriving IP packet. A sample is taken at each packet arrival event given that the packet is not rejected due to buffer overflow[5]. If all priority queues are empty, the sample value is zero.

---

[4] Two transmissions give 0 dB SNR at the receivers.

[5] We should change the probe to make a registration also for packets that are rejected.

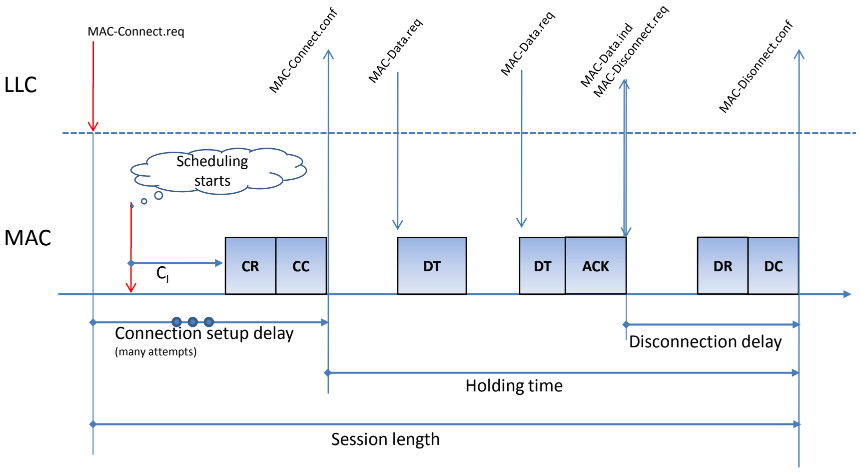

Figure 1.5 MAC delivery cycle when ARQ is enabled. A node must generally take part in a number of access cycles before it wins the channel access.

2 MLPP in a Perfect Network

This chapter describes how an ideal MLPP function shall operate in a perfect network with **finite** capacity. A perfect network refers to a network which is not restricted by other practical matters. An MLPP service shall let the IP clients (users) signal the priority level on a per packet basis. We consider a multilevel MLPP service supporting four priority levels, referred to as P0, P1, P2 and P3. P0 is the lowest priority level and P3 is the highest level. How an MLPP throughput plot shall look like depends on the offered traffic per priority level. Firstly, we consider the priority distribution $\alpha_1 = \{0.4, 0.3, 0.2, 0.1\}$. This set says that 40% of the offered traffic volume uses the lowest priority P0 while the highest priority P3 amounts to 10% of the offered traffic.

Consider the throughput plot in Figure 2.1. The upper curve depicts a perfect shape for the network throughput. This figure assumes a network capacity of 1000 bytes/s. When the offered traffic increases beyond 1000 bytes/s, the capacity limit is reached and any network with limited capacity must discard packets. A perfect shape is a straight horizontal line from this point since a perfect network shall not drop any packet until the capacity limit is reached.

The task of the MLPP network service is to execute admission control on lower priority traffic when the network becomes overloaded. In our example, the MLPP function must start to reject P0-traffic when the offered traffic (Λ) reaches 1000 bytes/s. Therefore the maximum P0-throughput is 400 bytes/s and from this point the P0-throughput decreases gradually at the same rate as the (P1+P2+P3)-offered traffic increases.

The ideal shape of the end-to-end IP packet delay plot is shown at the bottom of Figure 2.1. An arriving IP packet is served immediately at low load levels and the typical situation is that only one node competes for channel access. As the offered traffic increases, the input queues increase in size, more nodes become busy and take part in the competition for the network resources. The major components of the end-to-end delay are the following delay components: 1) Input buffer queuing delay; 2) channel access delay; and 3) transmission time. In high load states, 1) is the largest component while 3) is the smallest. Note the P3-delay plot, which is drawn constant and independent of the other priority levels. This is difficult to achieve in a real network because low priority traffic will affect high priority traffic as long as they share the same network resources.

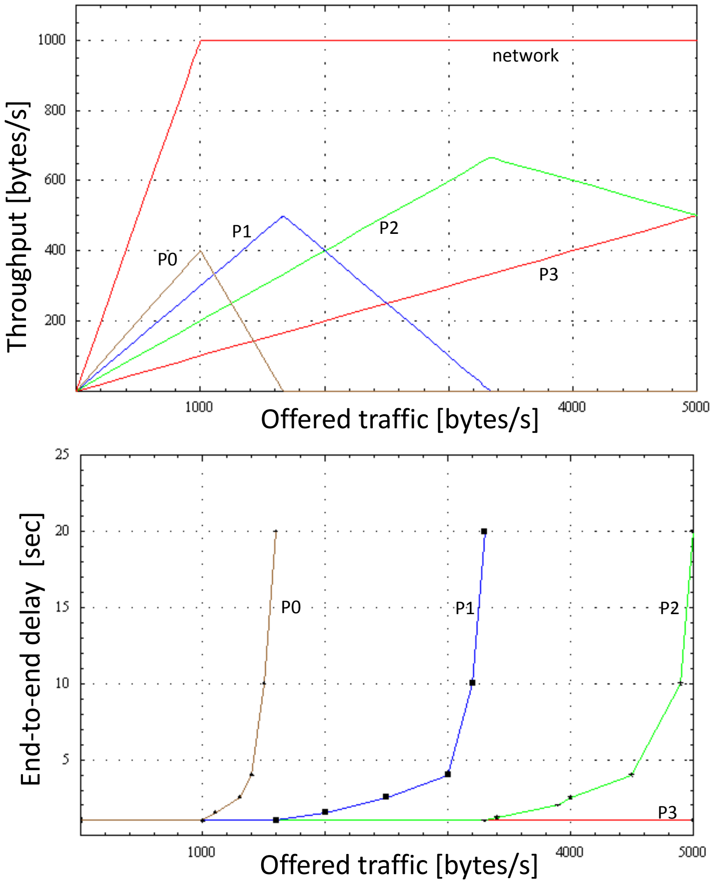

Figure 2.1 Optimum performance plots for $\alpha_1 = \{0.4, 0.3, 0.2, 0.1\}$

The priority distribution α_1 assumes that the (P0+P1)-offered traffic amounts to 70% of the total traffic volume. Consider a network which supports a low priority application that requires a low capacity (e.g. a logistic application). A preferred traffic profile might be $\alpha_2 = \{0.1, 0.4, 0.4, 0.1\}$. Here 80% of the traffic volume is (P1+P2)-traffic. The idealised throughput plot for this priority distribution is shown in Figure 2.2.

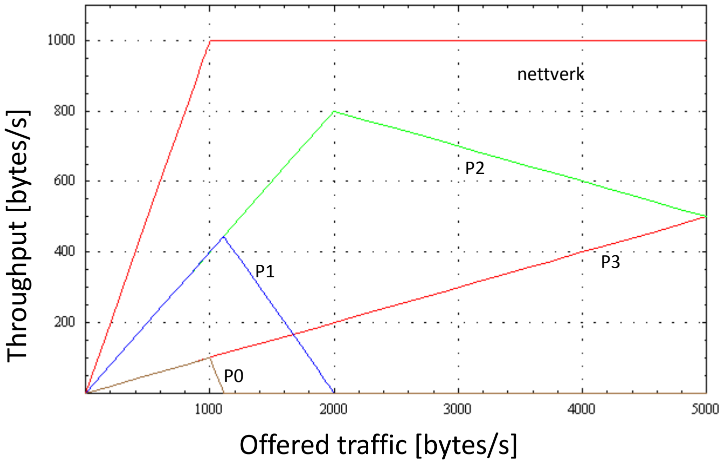

Figure 2.2 Optimum throughput plots for $\alpha_2 = \{0.1, 0.4, 0.4, 0.1\}$

2.1 The priority level for routing traffic

All the traffic over the air interface passes the MAC layer and also the routing traffic must set a priority level. Presumably, most readers argue that routing should apply level P3 since all multihop traffic, regardless of its priority level, must have a correct route to the end-destination to be delivered. However, a NBWF radio has a large coverage area which gives a well connected network. Most traffic is single-hop and by sending routing on priority P2, the P3-traffic will see a very fast network. An important condition for this option is that the P3-traffic from the IP clients never exceeds approximately 10% of the traffic volume.

We also have the option to allow the IP clients to use only three levels (P0+P1+P2)[6] and reserve the highest priority for routing. The benefit is a lower risk of having displaced routing traffic in periods with much user traffic.

We conclude here that the priority level assigned to routing traffic needs further study.

3 Adaptive MAC Scheduling

The NBWF MAC protocol entity applies a random access protocol when serving the CR PDUs (see Figure 3.1) and the scheduling function has the form:

$$D_{scheduling} = RandUniform[0, t_u] \tag{3.1}$$

By using a short t_u, we get a shorter time delay till the first CR PDU is sent on the radio channel but the probability of having a collision may be high. Let p_{coll} denote the latter while $E[C_I]$ denotes the expectation of the channel idle period. Given that *n*-nodes compete for channel access, these terms can be expressed as

---

[6] The implementation cost (software complexity, bandwidth usage, etc) of a four level MLPP service is high enough and we advice against introduction of more levels.

$$p_{coll} = 1-(1-t_v/t_u)^n \qquad \text{and} \qquad E[C_I] = t_u/(n+1) \;\; \text{for} \;\; n \geq 2 \tag{3.2}$$

where t_v is the ”carrier sense” detection delay (vulnerable period). The purpose of adaptive MAC scheduling is to maintain an optimum balance between the collision rate and the channel idle period by adjusting t_u to the number of busy nodes.

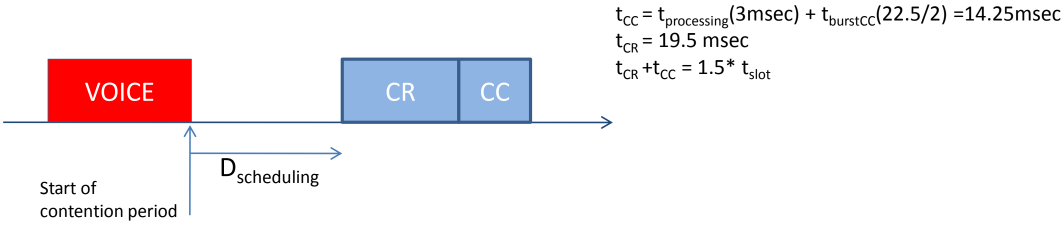

Figure 3.1 The busy nodes draw a random access delay when the contention period starts.

The adaptive MAC scheduling scheme is comprised by the following elements of procedure:

1) *Broadcast of state information*.
   Each node signals its MAC load state (idle/busy[7]) regularly in the dedicated access intervals (TDMA slot 7 or 8).
2) *Data harvesting*.
   All nodes estimate the number of busy nodes in their neighbourhood periodically based on the MAC load level reports received.
3) *Data consumption*.
   When a node shall draw a new random delay, firstly the node selects the t_u-parameter from the set $\{t_{u,low}, t_{u,high}\}$ according to the load level state.

A MAC Load Level (MLL) report uses one bit to signal the MAC load state and 8-bits to signal its address. This report is sent in TDMA slot 7 which carries one and only one report at a time. Hence the periodic MLL-report cycle per node is $0.2025 \cdot n$ seconds, which is 10 seconds in an AHAn50 network. This should be fast enough in a narrowband network but be aware of that background noise may increase the report period due to lost messages.

Only one node can send an MLL-report in a slot, and the picture at the top right corner of Figure 3.2 gives an example of an MLL-report sequence sent over TDMA slot 7. The green/red colour indicates an idle/busy node while the number is the node source address (e.g., green 0 means that node 0 signals idle). Let *N(t)* represent the continuous process of sampling the number of busy nodes. The progress of *N(t)* as the time passes is shown at the bottom of Figure 3.2. For example, in the time period $<t_2,t_3>$, *N(t)*=2 since node 0 and node 1 have signalled busy.

---

[7] Busy means the node has one or more packet ready for transmission. One bit PCI is required to signal this state.

The process of determining the load level in Figure 3.3 is based on the exponential moving average *Ema(N(t)):*

$$n_{i+1} = (1-\gamma)\cdot n_i + \gamma \cdot N(t_{i+1}), \quad n_0 = 0, \quad 0 < \gamma < 1 \tag{3.3}$$

where $N(t_{i+1})$ is $N(t)$ sampled at $t = t_{i+1}$. A random access protocol degrades faster with short t_u-values than long t_u. Therefore we have included an additional condition for switching from low to high load state based on the instantaneous number of busy nodes *Inst(N(t)).* Which random delay parameter t_u to use, is determined by the state transition diagram in Figure 3.3. We have now described the adaptive MAC scheduling algorithm but have not assigned values to the parameters a, b, c, γ and $\{t_{u,low}, t_{u,high}\}$. This is a subject for the next chapter.

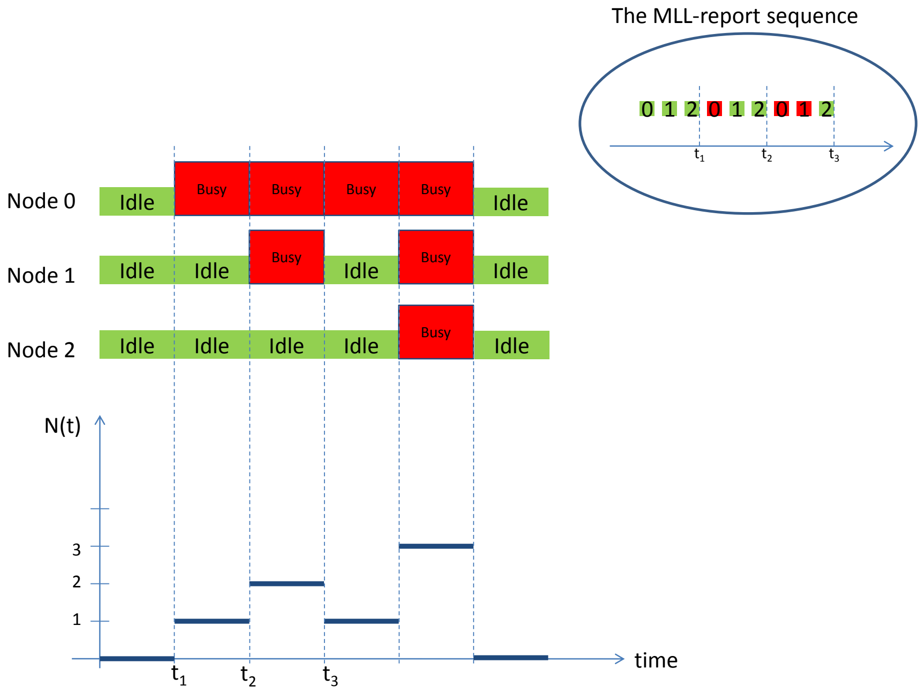

Figure 3.2 All nodes broadcast their state (idle/busy) periodically and each node tracks the load level in their neighbourhood by counting the number of busy nodes. Each node signals their load state periodically (see the rightmost picture) in TDMA slot 7.

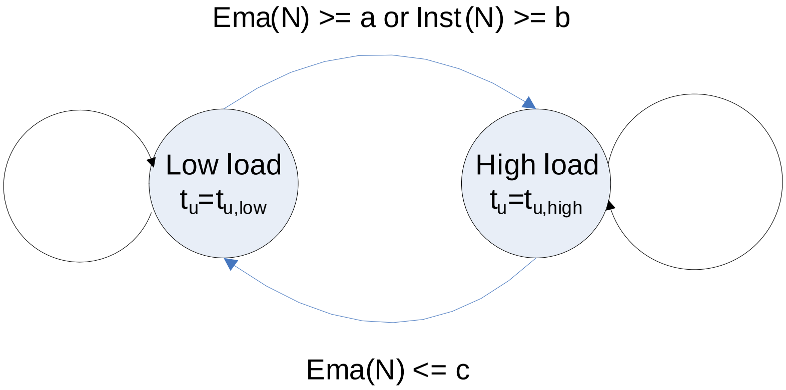

Figure 3.3 State transition diagram for switching between load states.

4 Single Level Priority Networks

A single-level priority network is a network which does not need to differentiate between priority levels. Based on chapter 3, the challenge we shall address now is that of determining the optimum t_u –value for single-priority networks. It is much easier to study the performance of an adaptive algorithm in a network without priority handling. The MAC random access function used in this chapter is specified by (3.1). A short t_u gives a higher collision rate than a long t_u. The drawback with a long t_u is the increased channel idle time which gives loss off transmission capacity and longer link delays. Generally, the optimum t_u-value depends on the packet length; loss of long packets represents more wasted capacity than short packets. An attractive characteristic of NBWF is that it is the short CR PDU that competes for channel access and not a long IP packet.

The most demanding traffic conditions arise in large networks and therefore this chapter considers a large AHAn50 network; a fully meshed network where 50 nodes share one radio channel. We use short packets (10 bytes layer-7 payload) because the optimum t_u depends on the size of the CR PDU only and not the length of the IP packet.

Our first step is to find the optimum t_u-value under high load conditions and Figure 4.1 presents an experiment where the throughput is estimated for a set of t_u-values. Here we observe that 25 msec leads to network collapse, while 1000 msec gives a stable throughput curve, but the network suffers from long channel idle time periods. 300 msec and 400 msec have nearly identical throughput performance. We select 300 msec because a small number gives lower link delays.

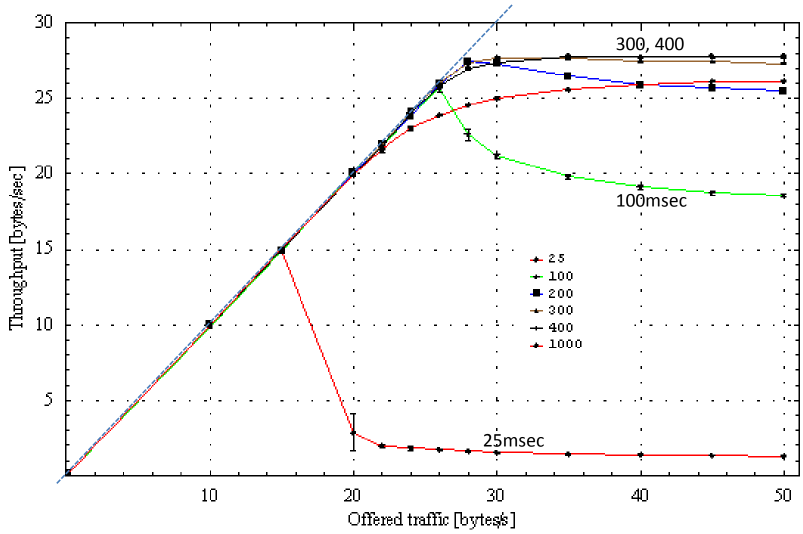

Figure 4.1 Throughput vs. offered traffic for $t_u \in \{25, 100, 200, 300, 400, 1000\}$ *(simJune4b).*

The next question is how short t_u-value we dare to use in a low traffic state. If we use 25 msec and the load estimation process fails, the network collapses. 100 msec seems to be a better choice since the network has reasonable throughput capacity in situations where the load estimation may fail (e.g. network jamming). The purpose of the next experiment is to estimate the gain of an adaptive t_u scheme based on the following parameters[8]:

t_u values

Low load	*100 msec*
High load	*300 msec*

Traffic load threshold set 1 (TS1):

Low $\rightarrow$ *High*	*Ema(N):* >2 *or*	*Inst(N):* ≥ 3
High $\rightarrow$ *Low*	*Ema(N):* ≤ 2 *and*	*Inst(N):* <2

Figure 4.2 and Figure 4.3 show the simulated results where the 100 msec and 300 msec fixed scheduling from Figure 4.1 are included to make it easier to see the change in performance. Adaptive scheduling gives the same throughput performance as 300 msec fixed scheduling. However, from the link delay plot we conclude that adaptive t_u is the winner since a lower link delay is achieved.

---

[8] $\gamma = 0.5$

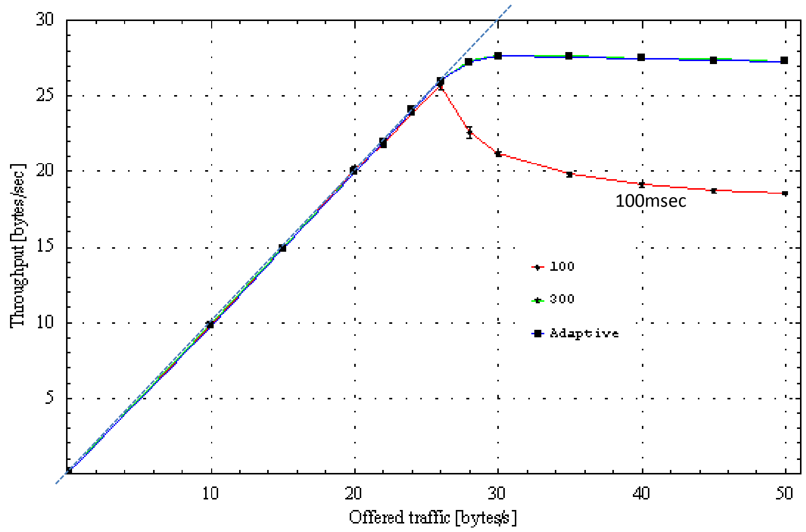

Figure 4.2 Throughput comparison for fixed and adaptive MAC scheduling (simJune4b).

From the link delay plot, we observe that a shorter delay might be reached if the adaptive t_u algorithm switches to high load state later. This is achieved by modifying the threshold parameters and we want to test the following three sets:

Threshold set 2 (TS2):

$Low \rightarrow High$	$Ema(N): \geq 5$	$Inst(N): \geq 20$
$High \rightarrow Low$	$Ema(N): \leq 3$	

Threshold set 3 (TS3):

$Low \rightarrow High$	$Ema(N): \geq 10$ or	$Inst(N): \geq 20$
$High \rightarrow Low$	$Ema(N): \leq 4$	

Threshold set 4 (TS4):

$Low \rightarrow High$	$Ema(N): \geq 20$ or	$Inst(N): \geq 40$
$High \rightarrow Low$	$Ema(N): \leq 6$	

These sets use a more aggressive switching strategy by allowing an increasing number of nodes to compete for access before shifting to a longer t_u. We have also omitted the test against the instant number of nodes (*Inst(N)*) when switching from high to low load.

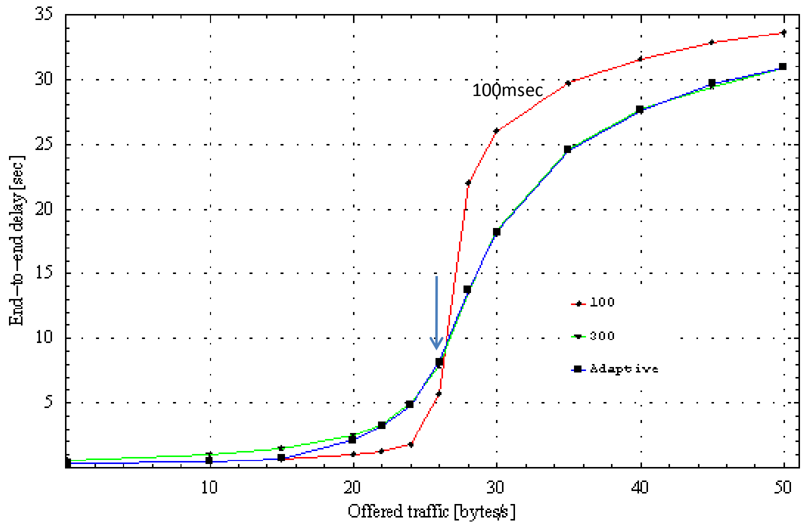

Figure 4.3 Link delay comparison for fixed and adaptive MAC scheduling (simJune4b). The vertical arrow points out the network saturation point.

All the threshold sets produced identical throughput plots but the link delays were different, see Figure 4.4. Since the threshold set TS4 gives the minimum average link delay, TS4 is the preferred choice. Let’s consider the probability to receive a CC PDU, given that a CR PDU is sent, see Figure 4.5. If a collision occurs, none of the CR PDUs will be received successfully and the sending nodes register a “no CC PDU received”-event. The plot shows that the collision rate is insignificant up to 1 packet/s (10bytes/s) and becomes high at maximum load. We have included the “Fixed 100 msec”-scheduling results in the figure to illustrate the improvement of the adaptive scheduling. It is also interesting to note how the displacement of the high/low switching point in Figure 4.6 moves as we alter the threshold set.

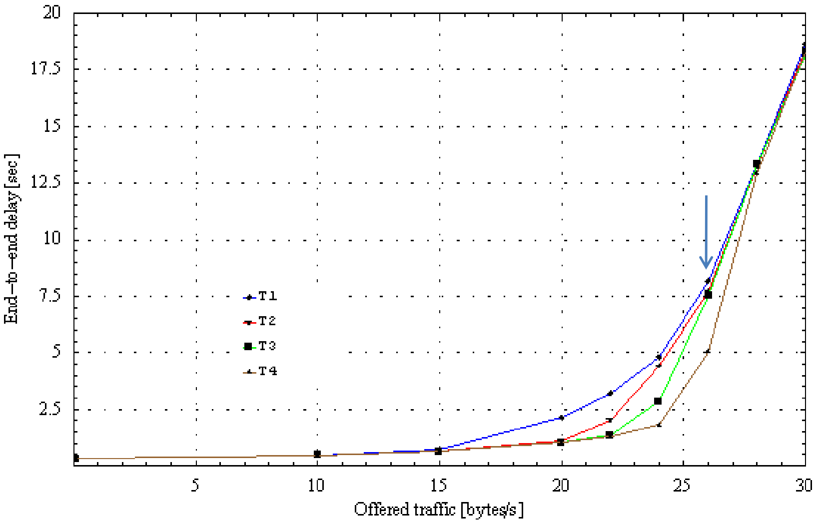

Figure 4.4 Link delay comparison for adaptive MAC scheduling using different threshold sets (simJune22a). The vertical arrow points to the network saturation point.

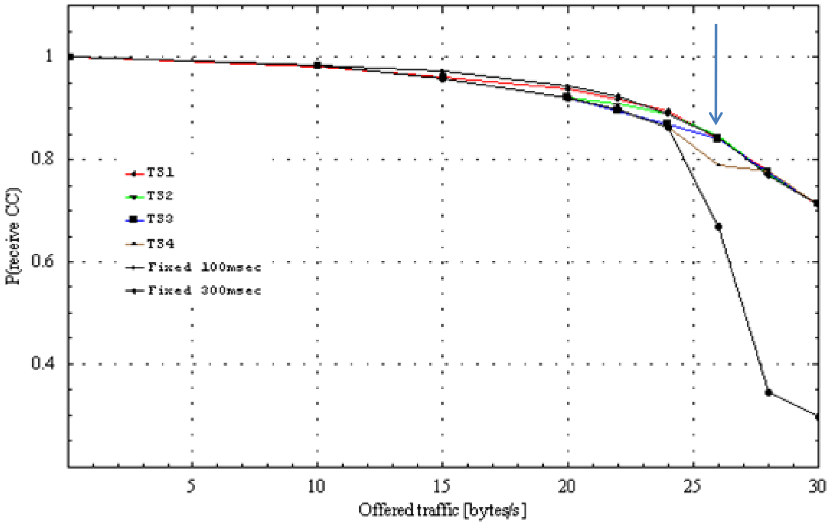

Figure 4.5 Probability to receive a CC PDU given that a CR PDU is sent. The vertical arrow points to the network saturation point.

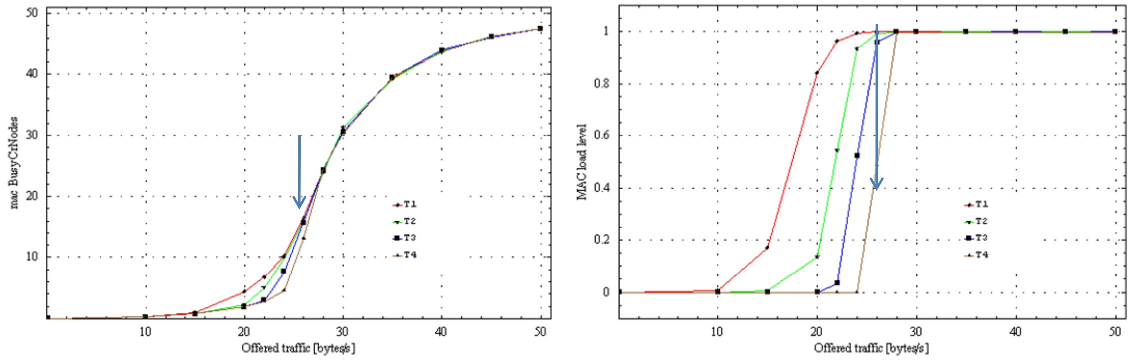

Figure 4.6 Number of busy nodes measured from periodic MLL-reports (left) and the MAC load level. (simJune22a). The vertical arrow points to the network saturation point.

4.1 Conclusions and remarks

This chapter conclude that we shall apply the threshold set TS4 which gives $a = 20$, $b = 40$ and $c = 6$, see Figure 3.3. All the experiments have been based on $\gamma = 0.5$ but we have not tested other values since this is not important at the current stage of the NBWF project. For an AHAn50, we recommend the set $\{t_{u,low} = 100, t_{u,high} = 300\}$ when using adaptive MAC scheduling. In smaller networks, we will find that these values should be lower.

A network without adaptive MAC scheduling must select a t_u-value which gives a stable throughput at all load levels since the sudden drop in throughput seen in Figure 4.1 is unacceptable. The consequence is higher link delay as can be seen from the green line (“Fixed 300”) in Figure 4.7.

Adaptive MAC scheduling and fixed ($t_u = 300$)-scheduling have the same throughput performance (Figure 4.2) but the former consumes some transmission capacity in TDMA slot 7. However, we recommend use of adaptive MAC scheduling to achieve a lower link delay compared to fixed ($t_u = 300$)-scheduling. The profit of an adaptive scheme decreases with decreasing network size.

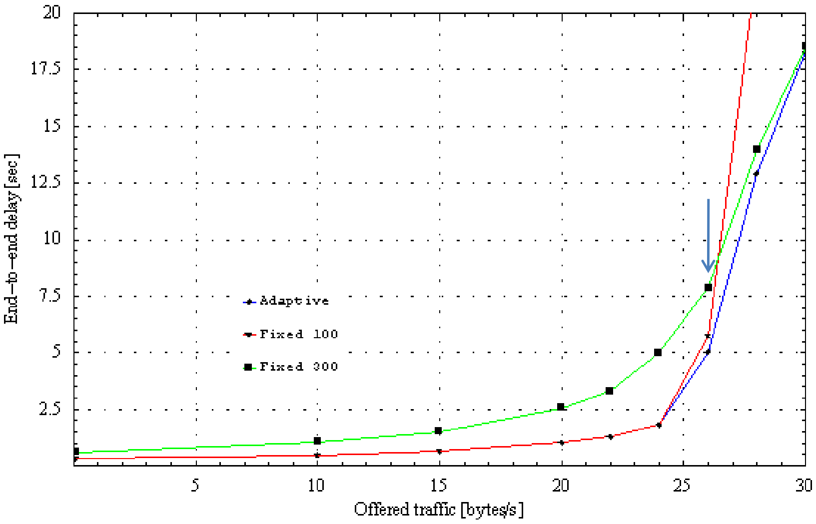

Figure 4.7 Link delay comparison for adaptive and fixed MAC scheduling (simJune22d). All the simulations use the threshold set TS4.

5 Multilevel Priority Networks

The basic principle of implementing priority handling within the MAC layer is to delay low priority packets such that high priority packets get a higher probability to grab the radio channel. The MAC random delay function for a four-level priority network is given by

$$D_{scheduling,Pi} = t_{pri,Pi} + RandUniform[0, t_{u,Pi}], \qquad i = 0,1,2,3 \tag{5.1}$$

where $t_{pri,Pi}$ is named the *priority delay* for priority level P_i. The priority delay is not a random number and does not affect the collision rate directly but determines the differentiation between the priority levels. For example, by assigning a short priority delay to P3-packets and a long value to P0-packets, we achieve that P3-packets grabs the channel before the P0-packets. Given that a network operates with one priority level at a time, our earlier expressions are still valid

$$p_{coll,Pi} = 1 - (1 - t_v / t_{u,Pi})^n \text{ and } E[C_{I,Pi}] = t_{pri,Pi} + t_{u,Pi} / (n+1) \tag{5.2}$$

Here we see an enlarged channel idle period but no change of the collision rate. Based on the simulation experiments from the earlier chapters, the task is now to find parameter values suitable for the design goals presented in chapter 2.

This chapter follows the recommendation given in section 4.1 and implements a two-level MAC random delay function per priority, one for each load level; low or high. Hence, we implement eight different random delay functions.

Firstly, consider a network operating in a low traffic state; one or two nodes are active at each priority level and collision events are a minor problem. The optimization challenge is now: How much priority delay must be introduced at each priority level to have good differentiation between the priorities (cf., Figure 2.2) without sacrificing too much throughput capacity? Figure 5.1 proposes a set of start values. P3 has been assigned zero priority delay since P3-packets shall win. Even though we have assumed a few active nodes, P3-traffic must be assigned random access delay but we can use a low value ($t_{u,P3} = 25m\sec$). P2 and P3 have overlapping random delays. The drawback is degraded sorting performance but a shorter link delay on the P2-traffic is achieved. Consider P0 (lowest priority) for which the priority delay area has been enlarged to fall outside the random delay for P2 and P3. Hence P3- and P2-packets always win over P0-packets. We have decided not to let P1-packets always win over P0 since this would have increased the priority delay on P0 even more.

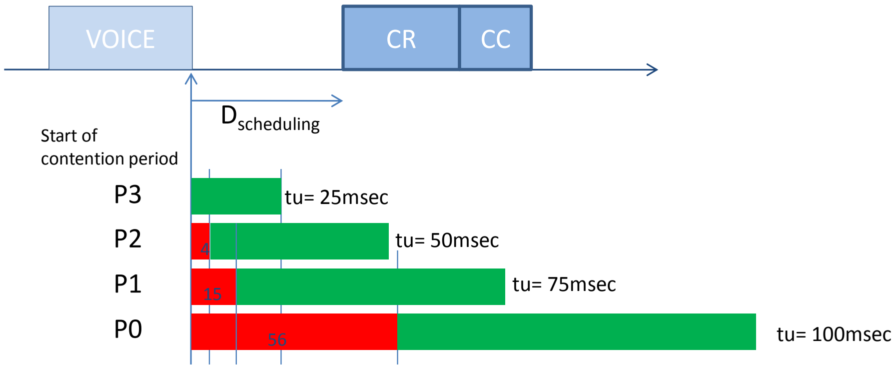

*Figure 5.1 MAC scheduling parameter values during **low traffic** periods. The red areas are priority delays while the green areas are the random delays.*

Let's look at high traffic periods where all the network nodes compete for channel access but the network is still in the steady-state, which means that the network has capacity to serve the lowest priority packets. Under this condition, the priority distribution has a great impact on the traffic volume per priority level handled by the MAC layer[9]. Figure 4.1 shows that t_u=100msec gives degraded throughput performance in a single-priority AHAn50 network. Despite this fact, we use 100msec for the P3-traffic since only 10% of the total traffic volume uses the P3-level. t_u-values for the other priorities are given in Figure 5.2. Also the P0-traffic amounts to only 10% of the traffic volume but we apply a much longer random delay to make it easier for P1-packets to grab

---

[9] When the network reaches the saturation point, the P0 fresh traffic queue grows very fast with increasing traffic. From this point the network will always have P0-packets to serve despite that P0 amounts to only 10% of the traffic volume. As the network reaches a high congestion state, the impact of the priority distribution vanishes. The same can be said about the packet arrival distribution.

the channel before P0-packets. The same could have been achieved by increasing the priority delay but that would not have decreased the collision rate in a saturated network.

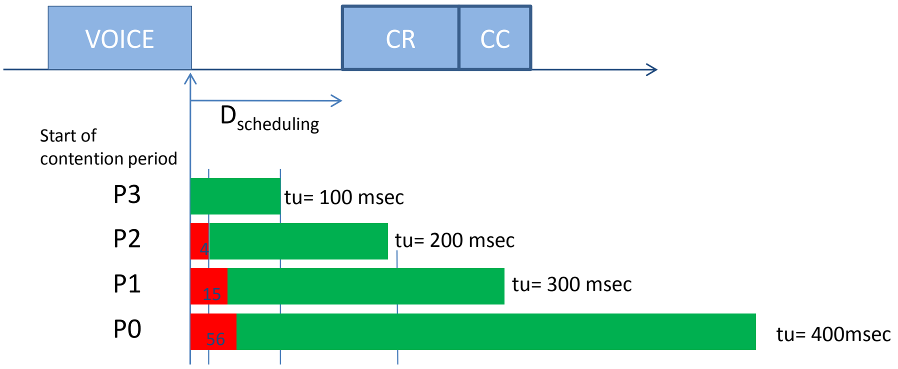

Figure 5.2 MAC scheduling parameter values during ***high traffic*** *periods.*

We have now described eight different random delay functions, four for each traffic load state; high or low. If we decide later not to use the adaptive MAC scheduling in conjunction with the MLPP service then we must select the four functions in Figure 5.2 to get an acceptable collision rate. In a multilevel priority network, each node must at least signal its busy state for the highest priority under service, and each node must maintain the Finite State Machine (FSM) in Figure 5.3 for each of its neighbour[10].

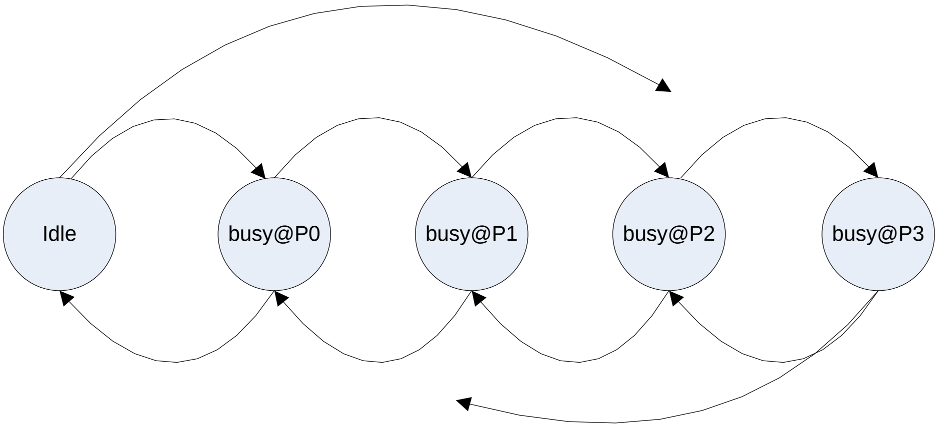

Figure 5.3 The periodic MLL-report signals one of five states and each node maintains one FSM for each node in its neighbour set. This FSM is updated when receiving the periodic MLL-report from the neighbours. If busy, a neighbour signals the highest priority level only and not the busy/idle state for each priority level. Thus we have random transitions from any state to any state.

---

[10] In the single-priority case we had two states and have now extended the machine to have five states.

This means that the *N(t)*-function in Figure 3.2 becomes $N_p(t)$, where $p \in \{P0, ..., P3\}$. Based on $N_p(t)$, the network nodes use the same state diagram as earlier (see Figure 3.3) to switch between the load states and they calculate periodically[11] the load state as follows:

```
updateStateDiagram()
{
    // Note: The state diagrams must be updated in the order P3...P0
    updateState(P3); // Executes Figure 3.3 using Np(t) with p=P3.
    if( isHighLoad(P3) )
    { // Set high load on P2...P0 to maintain the sorting characteristics
       set {P0,P1,P2} to high load state;
       return;
    }
    updateState(P2); // Executes Figure 3.3 using Np(t) with p=P2.
    if( isHighLoad(P2) )
    { // Set high load on P1...P0 to maintain the sorting characteristics
       set {P0,P1} to high load state;
       return;
    }
    updateState(P1); // Executes Figure 3.3 using Np(t) with p=P1.
    if( isHighLoad(P1) )
    { // Set high load on P0 to maintain the sorting characteristics
       set {P0} to high load state;
       return;
    }
    updateState(P0); // Executes Figure 3.3 using Np(t) with p=P0.
} // end updateStateDiagram
```

Note the very important rule to maintain the sorting property is: If the (i+1)-priority FSM (Figure 3.3) is in high load state then all lower priority FSMs shall also make a transition to high load state. If this rule is broken, the network may end up in a situation where e.g. P2-traffic uses $t_{u,low,P2} = 50$ and P3-traffic uses $t_{u,high,P3} = 100$ for long time periods. However, this rule does not prevent this situation to occur for short time periods since we cannot ensure that the nodes switch load state at the same time instances (e.g. background noise may lead to loss of reports).

Figure 5.4 gives simulated throughput for short packets (10bytes IP payload) in case of two different network sizes. Both networks conform well to the idealised throughput plot in chapter 2, but the AHAn50-network should have blocked slightly more of the P1-traffic. Figure 5.5 presents the results for long packets (1500 bytes IP payload) which show that the network throughput curves are very close to the course wanted. However, the P1-throughput for the AHAn50-network shows a significant deviation from the idealised throughput plot. Even worse, the P1-traffic should have been blocked at $\Lambda = 1500$ but increases from $\Lambda = 1650$ to $\Lambda = 1950$. Also the P3-traffic experiences lost packets too early. We conclude that we need to find better MAC scheduling parameters for the high-traffic state (Figure 5.2) such that the P1-traffic is choked down earlier. This will improve the situation for the P2- and the P3-traffic.

---

[11] In the simulator, each node use a random periodicity given by randomUniform[5,10] seconds.

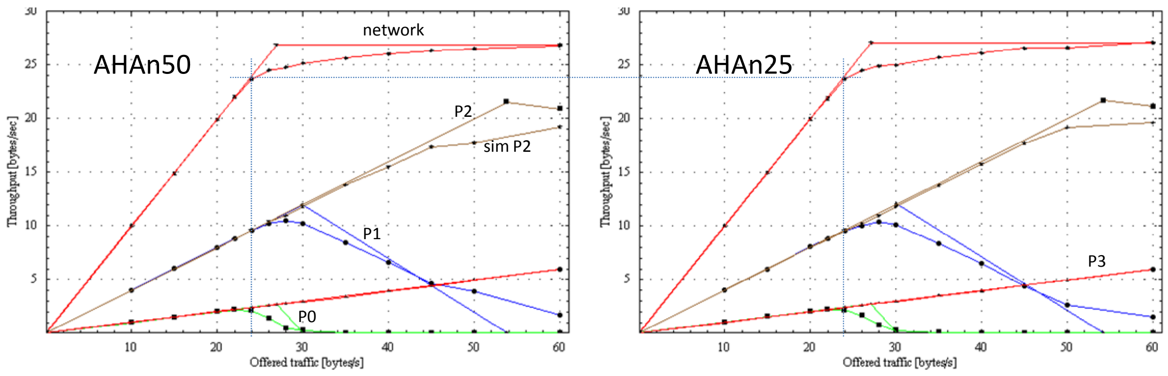

Figure 5.4 Simulated throughput vs. offered traffic (Λ) for short packets (10bytes) (simJuly12a/13a).

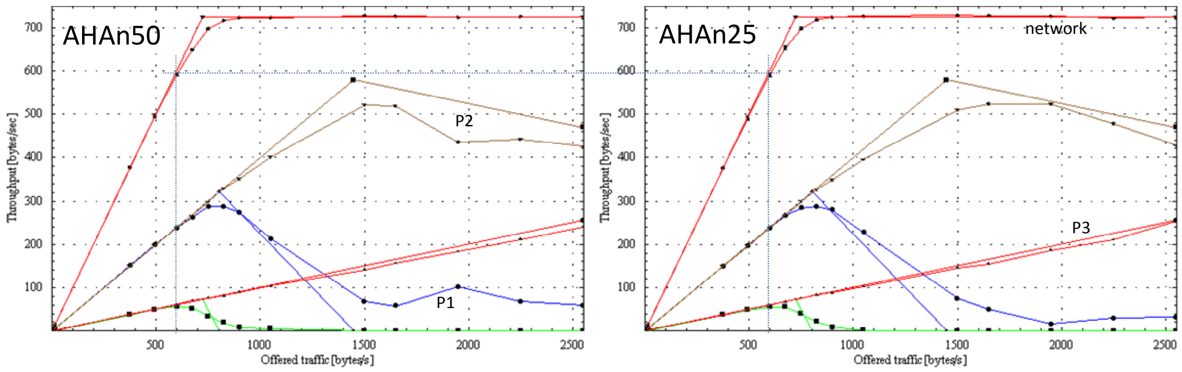

Figure 5.5 Simulated throughput for long packets (1500 bytes).

Pictures showing the link delays and the MAC connection setup delays (Figure 1.5) are given in Figure 5.6. We observe a significant deviation from the idealised delay plot in chapter 2. Consider the P3-traffic which has a low connection setup delay and shows a small variation over the network load. Despite the low and nearly constant connection setup delay, we have a high increasing P3 link delay. As seen from Figure 5.7, the layer 3a queue delay plot has a similar shape as the link delay plot and its delay values are lower but very close to the corresponding points on the link delay curve. Hence, it is the input buffer delay that causes the deviation from idealised link delay.

As seen from Figure 5.7, an arriving P3-packet (10% of the traffic volume) enters a small queue in contrast to the P1-packets (40% of the traffic volume). The reason for the long P3 queue delay must then be caused by a long LLC service time despite the fact the P3-connection setup time is short. This indicates that we should increase the random delay and the priority delay for the P2-traffic such that the P3-packets win more frequently. When adding a delay to P2, we must also add delays to the other lower levels to maintain the sorting property. The consequence is less network throughput.

Also note the break in the P3 link delay curve; the delay drops suddenly at $\Lambda = 50$. At this point, the MAC scheduling process switches to high load (see Figure 5.9) for the P2-traffic and adds longer random delays. This is beneficial for the P3-packets which are served by the low load random delay function and then grabs the channel more frequently.

The adaptive MAC scheduling function demands exchange of periodic status packets to track the network load state, represented by the number of active nodes N_{busy}, see Figure 3.2[12]. An MLPP service demands a value per priority level and the leftmost plot in Figure 5.8 shows the simulation results for the N_{busy}.

Firstly, note that the sum of the N_{busy} over the priority levels may be much larger than the network size because the periodic MLL-reports signal the upper layer queue status in addition to packets under service at the MAC level. Then we may have a situation where N_{busy} =25[13] for P0, P1 and P2 and N_{busy} =2 (say) for P3, giving the sum 77 in a network with 25 nodes. In section 1.1, we specified a probe ($N_{\mathbf{G}busy}$) which measures perfectly how many nodes that compete for access. When the network operates at its throughput capacity, 6.7 nodes compete. From this point the number of competitors accelerates rapidly with the offered traffic. $N_{\mathbf{G}busy}$ can never be larger than the network size.

Note the drop in $N_{\mathbf{G}busy}$ at $\Lambda = 28$ for priority P0. From this point, the 3a layer entity has so many P1-packets that it must reduce the outgoing P0-traffic. The same happens at $\Lambda = 45$ for the P1-traffic and the P2-traffic takes over.

Without background noise, lost CC PDUs are due to CR PDU collisions. The MAC entity adjusts the random delay to keep the collision rate at an acceptable level. It is therefore interesting to measure the probability-of-receiving-a-CC (p_{CC}, see section 1.1). Figure 5.9 illustrates the impact of a changing random delay functions on p_{CC}.

Generally p_{CC} decreases with increasing offered traffic due to CR PDU collision events. However, for P0 p_{CC} starts to ascend at $\Lambda = 28$ when the P0-packets are served by the high-load random delay function. The load level for the P3-traffic is always low, MAC uses much shorter random delays and then the P3-packets get an additional advantage as the lower priority packets are served by high-load random delay function.

---

[12] N_{busy} is a probe defined in section 1.1.

[13] This number cannot be larger than the network size.

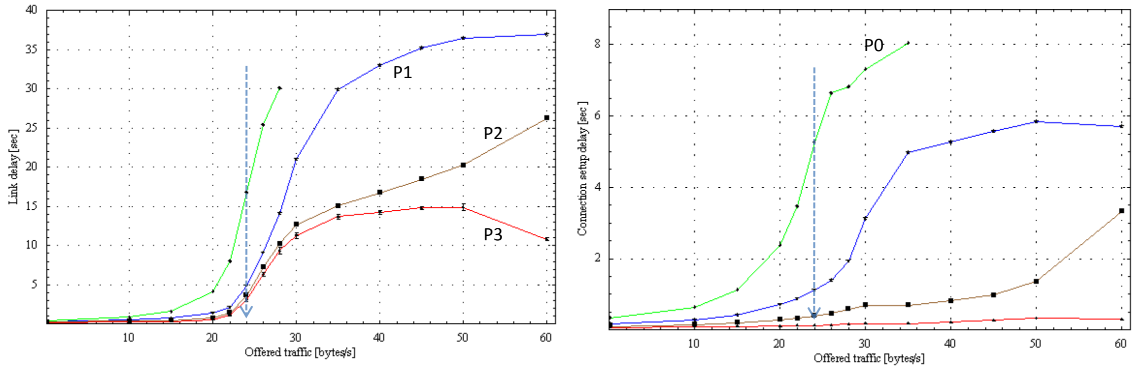

Figure 5.6 Link delay (left) and MAC connection setup delay for the AHAn25 network using short packets (10 bytes) (simJuly13a).

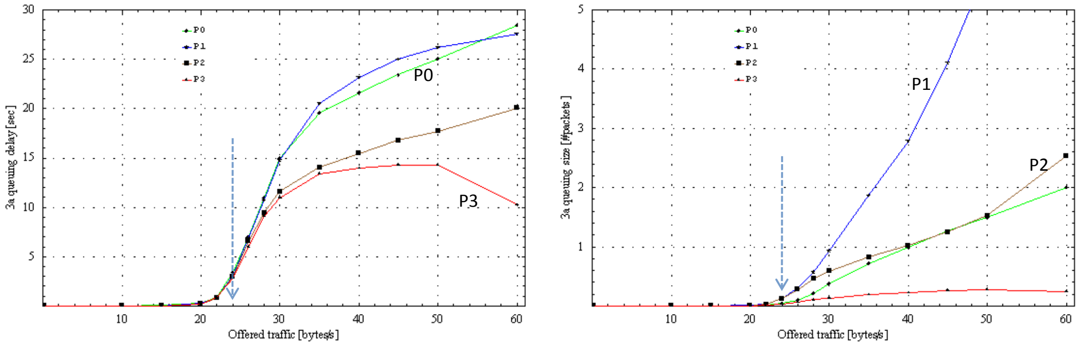

Figure 5.7 Layer 3a queuing delay (left) and 3a fresh traffic queue size for the AHAn25 network using short packets (10 bytes). The queue size is the number of packets seen by an arriving packet (simJuly13a).

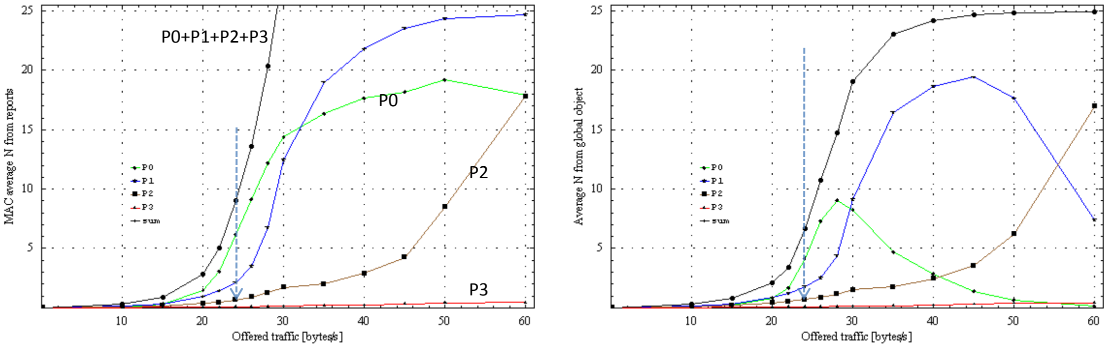

Figure 5.8 Number of busy nodes vs. offered traffic for the AHAn25 network using short packets (10 bytes).

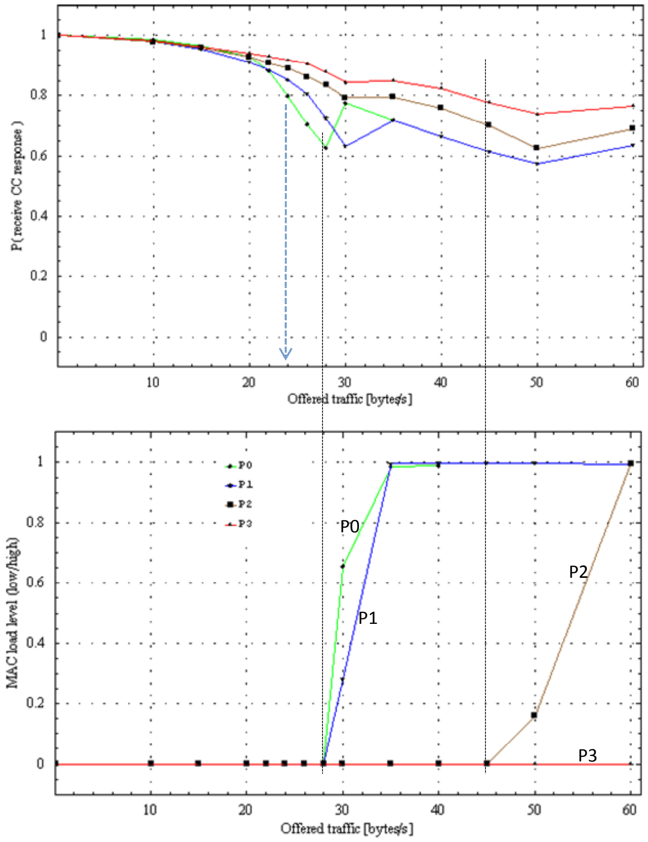

Figure 5.9 Probability to receive a CC response and MAC load level vs. offered traffic for the AHAn25 network using short packets (10 bytes).

5.1 Conclusions and remarks

This chapter has illustrated how the MLPP service improves the quality of service for the high priority traffic as the network saturation level increases. Best results were achieved for short IP packets but we expect that by adjusting some of the parameters, long packets will achieve better results. We argue that the present analysis is sufficient for the current stage of the NBWF project. It is more important to find answers to the following questions: 1) what is the cost of implementing MLPP in the MAC protocol? and 2) Do we actually need to implement MLPP in the MAC protocol? These questions are topics for the next chapter.

Chapter 4 quantified the performance enhancements by using adaptive MAC scheduling instead of fixed scheduling. Now we conduct the same exercise for an MLPP network. A network using fixed scheduling must use the high-traffic random delays in Figure 5.2. Otherwise the network suffers from a high collision rate under high traffic periods. Figure 5.10 gives the simulated results and shows that the adaptive scheme improves the network throughput with approximately 9%. The link delay simulations are presented in Figure 5.11. It is most important to have performance improvements in the steady-state region (the area to the left of the vertical arrows in the figures) since this is the normal operating area. We see that the adaptive scheme outperforms the fixed scheme. From the following table we clearly see the benefit of adaptive scheduling when the network operates near its throughput capacity:

End-to-end delays [sec] at 22 bytes/s

	priority P0	priority P1	priority P2	priority P3
Fixed	17.8	7.7	4.9	3.4
Adaptive	8.0	2.2	1.4	1.2

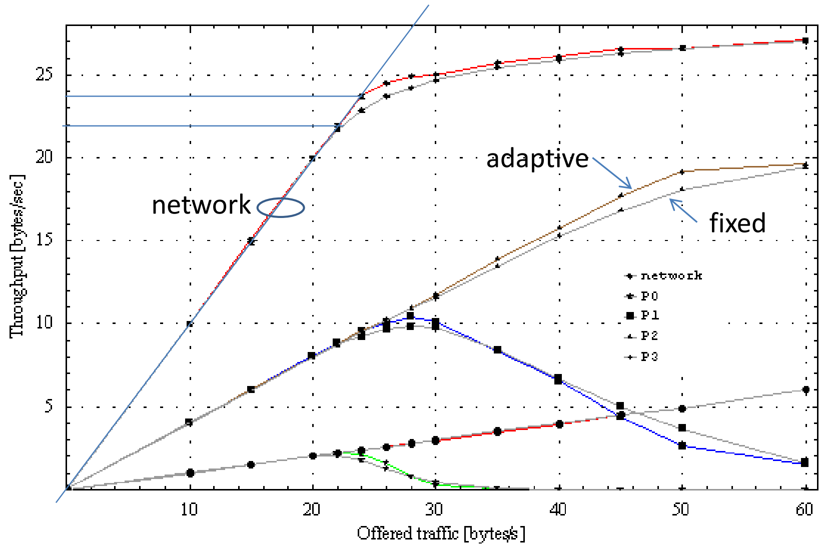

Figure 5.10 Throughput comparison for networks using fixed or adaptive MAC scheduling. The payload size is 10 bytes. (simSept13a).

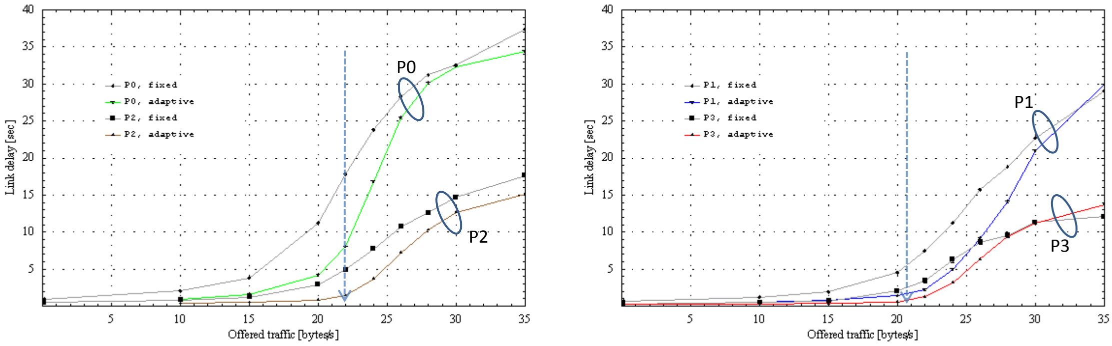

Figure 5.11 Link delays (without confidence control) for networks using fixed or adaptive MAC scheduling. The vertical line points to the network saturation point.

A drawback by not signalling the load level per priority[14], but only the highest level, is that load state changes at lower levels are not taken into account in the load estimation process. Consider a network serving P2- and P3-traffic only. As long as the nodes have P3-packets in their queues, they shall signal the load level for P3 and any status changes for P2 are not brought to the other nodes. In the presence of at least one P3-packet in each node, assume that one of the following two events happen:

---

[14] The benefit is fewer PCI bits (i.e., less overhead).

1) A new P2-packet arrives and the P2-queue size increases
2) The lifetime control function deletes a P2-packet and the P2-queue size decreases

The priority distribution determines the frequency of event 1, and the consequence of neglecting this event might be an increased collision rate due to use of a shorter t_u than optimum. The P2-packet arrival rate is higher than for P3 and Figure 5.7 indicates short P3-packet queue sizes. Hence the P3-queues are emptied fast and the load level for the P2-traffic is signalled again in TDMA slot 7. Event 2 occurs seldom compared to the other event and can be neglected.

Even though we have identified a weakness by not signalling the load state for all the priority levels in each periodic MLL-report, we do not recommend use of resources to analyse this alternative solution. If any, only marginal performance improvement is expected.

6 Why Priority Handling in the MAC Layer?

The purpose of MLPP is to provide expedited transport of high priority traffic. By comparing simulation experiments from two AHAn25-networks, with and without MAC level priority, we want to quantify the benefits of implementing MAC level priority. A negative consequence of priority handling at the MAC layer might be less overall/network throughput capacity.

Both AHAn25-networks operate under identical conditions but the network that does not support priority handling at the MAC level uses a MAC access delay function without a priority delay component:

$$D_{scheduling} = RandUniform[0, t_u], \quad t_{u,low} = 100, \quad t_{u,high} = 300 \text{ [msec]} \tag{6.1}$$

Also remember that the MAC protocol does not implement pre-emption; an established MAC connection is never disconnected if a higher priority packet arrives[15]. This chapter uses a short layer 7 payload size (10 bytes) and the lack of a MAC pre-emption function will be less distinctive. A large payload size (e.g. 1500 byes) gives longer blocking delays.

The scenarios considered up to now have assumed that the priority distribution $\alpha_2 = \{0.1, 0.4, 0.4, 0.1\}$ is uniformly distributed over the network nodes; the average queue size per priority is identical for all the nodes. Section 6.1 below presents simulation experiments under this condition. Assume a traffic case where the priority distribution is *not* uniformly distributed over the nodes, for example, one group of nodes receives only P3-traffic from their IP clients and then the queue sizes for the other priority levels are zero. This situation is referred to as an asymmetric traffic condition. Section 6.2 conducts experiments under the new traffic condition.

---

[15] To implement pre-emption between two nodes is not straightforward since it demands signalling over the radio channel.

6.1 Symmetric traffic conditions

Figure 6.1 gives the throughput plots. MAC level priority costs 2.2 bytes/s (8.4% reduction in throughput capacity) but the P2-traffic gets 3.8 bytes/s (27%) higher throughput. This improvement is a result of the bad priority sorting in the network without MAC priority; too many of the P0- and P1-packets win over the P2-packets after the network has reached the saturation point.

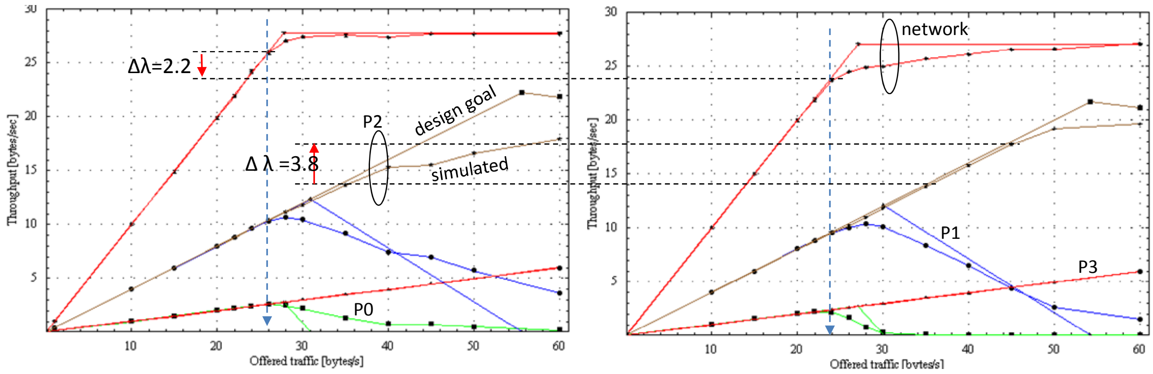

Figure 6.1 A network without MAC level priority handling at the left (simJuly19a). The right picture shows a network where the MAC protocol executes priority handling (simJuly13a). The vertical arrows point out the network saturation point.

Figure 6.2 shows the link delays (without confidence control) and the following two tables summarise the results at the saturation points:

w/o MAC priority

P0	4.3 sec
P1	3.9
P2	3.6
P3	3.4

with MAC priority

P0	16.8 sec
P1	4.9
P2	3.7
P3	3.2

When the networks operate at their throughput limit, the network without MAC priority has scarcely any differences between the links delays. This is expected since two competing nodes have the same probability to grab the radio channel regardless of the priority. First at high load levels, where the queues are long, we see a clear difference in the link delays among the priority classes. Both link delay plots deviate from the ideal shape seen in chapter 2 since the high priority link delays do not have a horizontal course. It is interesting to look at the MAC connection setup delays in Figure 6.3. For the MAC-priority case, we have large variations between the priority levels, and note the short and nearly flat connection setup latency for the P3-packets.

Also note the peculiar shape of the plot for the network without MAC priority, which is caused by the adaptive MAC scheduling. This can be seen more clearly in Figure 6.4 where the probability-of-receiving-a-CC (p_{CC}) estimate and the MAC-load-level estimate are plotted in the same figure. During low traffic periods where typically one node competes, p_{CC} is one. As the offered traffic increases, an increasing number of nodes compete, the collision probability increases and p_{CC} drops. If the MAC load level goes from low to high, the adaptive MAC scheduling function switches to longer random delays, the collision probability sinks and p_{CC} increases.

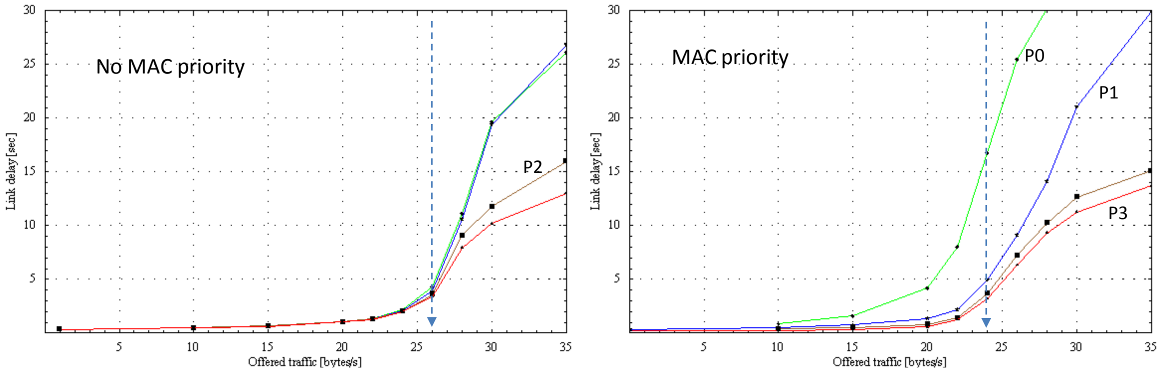

Figure 6.2 Link delays vs. offered traffic.

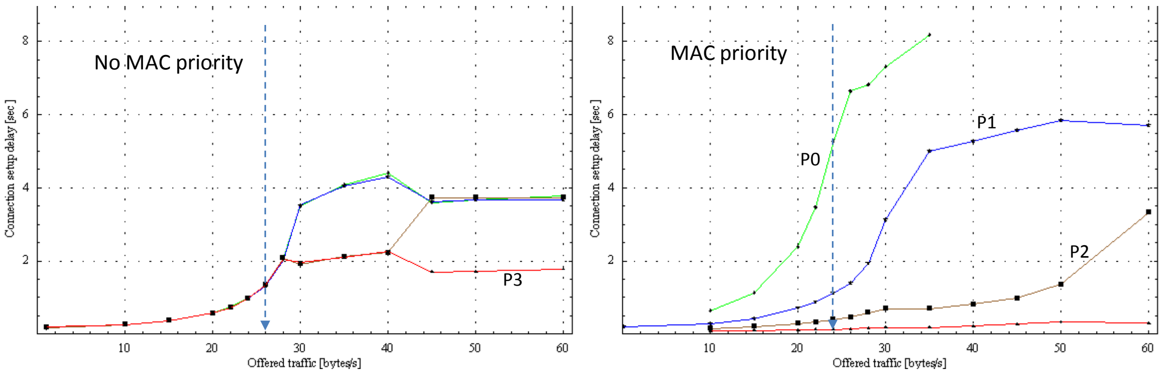

Figure 6.3 MAC connection setup delay vs. offered traffic.

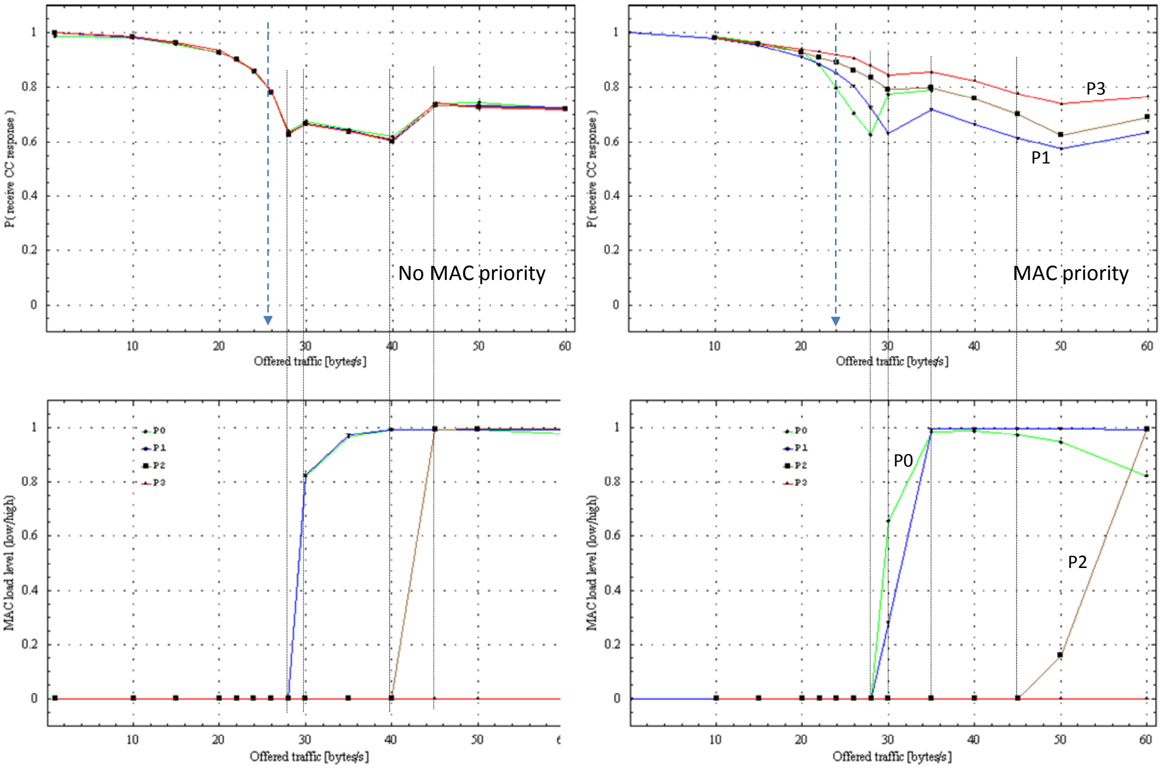

Figure 6.4 Probability to receive a CC response and MAC load level vs. offered traffic.

6.2 Asymmetric traffic conditions

The scenario simulated above had acceptable priority handling characteristics observed from the throughput plot (Figure 6.1) even without MAC level priority. With **MLPP aware LLC/3a-protocols** and symmetric traffic condition, the queue situation per priority is nearly identical for all the nodes and they switch to the next higher priority level at the same load level. In a scenario where the priority distribution is not uniformly distributed over the nodes (i.e., some nodes have P3-traffic only), the benefit of MAC level priority can be seen more clearly. We specify a new scenario using the same relative offered traffic per priority but with an inhomogeneous priority distribution over the nodes:

Node address set[16]	{0,...,6}	Priority P0	P0 traffic	$\Lambda_{P0} = 0.1 \cdot \Lambda$
	{7,...,12}	Priority P1	P1 traffic	$\Lambda_{P1} = 0.4 \cdot \Lambda$
	{13,...,18}	Priority P2	P2 traffic	$\Lambda_{P2} = 0.4 \cdot \Lambda$
	{19,...,24}	Priority P3	P3 traffic	$\Lambda_{P3} = 0.1 \cdot \Lambda$

Here the node set {0,...,6} never serves P3-traffic only regardless of the traffic level, and without MAC priority, the traffic streams from the two groups {0,...,6} and {19,...,24} have the same probability to grab the radio channel.

---

[16] We have 25 nodes and 4 priority levels. The P0-group contains 7 elements, the other 6.

Figure 6.5 gives the simulated results and shows a dramatic change from Figure 6.1 when MAC priority is disabled; the throughput plot deviates largely from the ideal throughput shape in chapter 2. The network without MAC priority has the highest throughput capacity and becomes saturated at 28 bytes/s. From this point, all the (P1+P2)-nodes are active while the (P0+P3)-nodes still have idle periods since the (P0+P3)-offered traffic amounts to only 20% of the offered traffic. As the number of active (P0+P3)-nodes increases, the (P1+P2)-nodes gets more competition and the (P1+P2)-throughput drops. Since all nodes use identical MAC scheduling functions, the throughput capacity per priority reaches the horizontal line *networkThroughput/4* as the offered traffic increases. These experiments show that a network without MAC priority serves P0-traffic while rejecting P1- and P2-traffic.

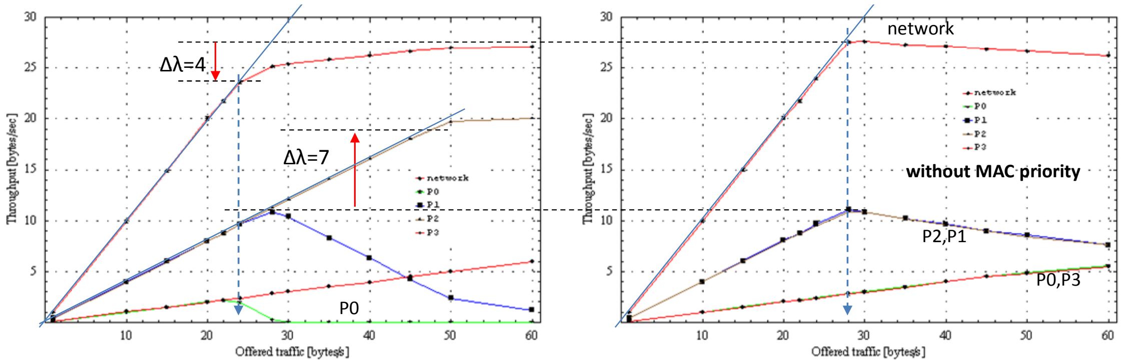

Figure 6.5 A network without MAC level priority handling at the right. The left picture shows a network where the MAC protocol implements priority handling. The vertical arrows point out the network saturation point (simJuly19b).

6.3 Conclusions and remarks

This chapter has shown how the MAC priority based scheduling improves the passability of the high priority packet at the cost of network throughput. Based on the simulation experiments, we have two conclusions depending on the priority distribution:

C1: When the priority distribution is uniformly distributed (symmetric) over the network nodes, the network has acceptable priority sorting performance without MAC priority.

C2: When the priority distribution is not uniformly distributed (asymmetric) over the network nodes, the network must implement MAC priority to have acceptable priority sorting performance.

In a real scenario, we do not expect fully symmetric traffic conditions from the IP clients. Mobility, multihop traffic, routing and ARQ are example of network functions which contribute to asymmetric traffic conditions at the MAC level. A real network must therefore implement priority handling in the MAC protocol.

The MAC load estimation process selects the load state (low or high) according to the load state per priority. In a network with asymmetric traffic conditions and identical MAC scheduling function for all priorities, the load estimation process must not differentiate between the priority levels. However, we have not modified the software which means that the MAC entity will claim low-load for all the traffic levels simulated. This short cut gives a higher collision rate but the magnitude is not so high that it affects the main results. A performance drop due to a high collision rate can be seen in Figure 6.5 by the negative slope of the network throughput plot for the “no-mac-priority”-case.

A design flaw in the load state switching strategy in chapter 5 is that all the priority levels are handled separately. Then the network with asymmetric priority distribution did never enter the high-load state and may suffer from a high collision rate. Both the symmetric and asymmetric cases have identical network throughput plots, so this is not an important issue here. However, enhancements to asymmetric traffic conditions should be a subject for further study.

The simulation experiments presented in this chapter use short 10 bytes packets only. What about long packets? We have conducted the same experiments for 1500 bytes/s and did not discover new results.

7 Operating in an Imperfect World

It is important to identify network environments that are challenging for the MLPP MAC service. If the MLPP MAC service introduces more vulnerability than other protocol functions then we have to find a solution, or remove the MLPP MAC service.

The most obvious challenge in a real network is network jamming and background noise. Both introduce packet loss and the quality of the network load estimates become degraded. Section 7.1 analyses the sensitivity to background noise. We do not consider network jamming since the current version of the NBWF radio does not provide Electronic Protective Measures (EPM).

All the simulation experiments up to now have assumed a Poisson distributed packet arrival process. In a real network, sudden external events may introduce momentary increase of the packet arrival rate which leads to degraded throughput performance for a period of time. However, the MLL-reports sent in TDMA slot 7 is unaffected by collisions and the adaptive MAC scheduling function will increase the average random delay after some time. In an AHAn25 network without background noise, any node has received the MLL-reports from all the neighbours in about 5 seconds. This is fast enough for a narrow band network. The current version of the NBWF simulator does only support steady-state analysis and therefore is not an appropriate tool to analyse transient conditions. But section 7.2 below discusses how the network reacts when going from Poisson packet arrival to a deterministic packet arrival process.

7.1 Sensitivity to background noise

The only part of the MLPP MAC service that introduces vulnerability to background noise is the load estimation process that signals in TDMA slot 7. Loss of MLL-reports leads either to underestimation or overestimation of the number of active nodes. Underestimation is most harmful for the network capacity since the network erroneously may apply the short random delay function and then operates with a higher collision rate than optimum. By activating the NBWF simulator’s zero capture model, we are able to set the packet loss probability over the radio channel to an exact value. A set of experiments are presented in Figure 7.1. When the short 10 bytes payload is employed, each payload is sent as a single transmission burst and then it is the signalling system that is most vulnerable to the background noise. At least two additional transmissions are required; one for connection establishment and one for disconnection. This changes when switching to the long 1500 bytes payload that must be sent as 8.2 bursts[17]. To have a better picture of the situation, we calculate the normalised throughput relative to the noiseless environment case ($p_{loss} = 0$):

p_{loss}	10bytes	1500bytes
0%	1	1
1%	0.92	0.89
5%	0.63	0.73
10%	0.42	0.55

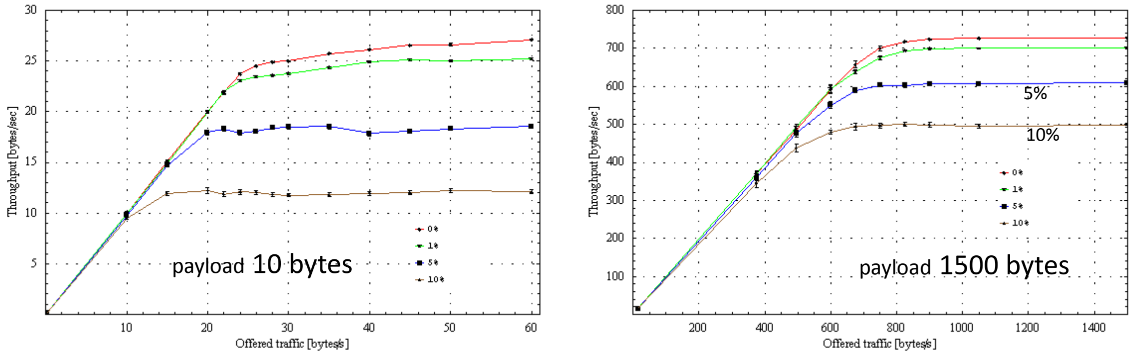

Figure 7.1 Throughput vs. offered traffic in a network with background noise for short 10 bytes and long payload.
The packet loss probability set used is {0,0.01,0.05,0.1} (simSept4a).

These results indicate that the protocol efficiency decreases fast as the packet loss rate becomes higher than 1-2%.

Background noise and collisions decrease the connection establishment success rate. A connection setup phase is successful when a CC PDU is received. Therefore it is interesting to

---

[17] The average LLC segment size was measured to 183 bytes.

compare p_{cc} for noisy and noiseless environments. Figure 7.2 shows how p_{cc} becomes degraded in a noisy environment. At low traffic levels, packets are mostly lost due to background noise and $p_{cc}=1$ for the noiseless case while $p_{cc}=0.9^2=0.81$ for the other case.

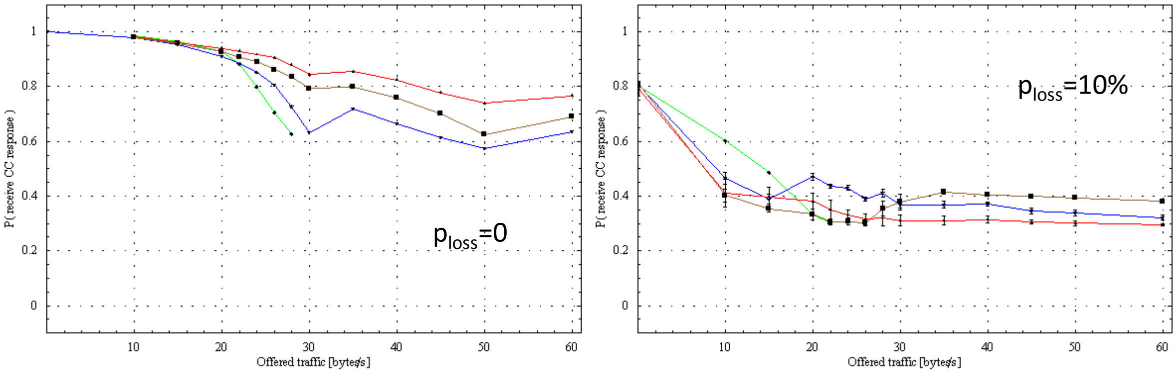

Figure 7.2 p_{cc} per priority vs. offered traffic in a network without background noise (left) and with background noise (right). The payload size is 10 bytes.

7.2 Sensitivity to the packet arrival distribution

All the simulation experiments up to now have used the Poisson packet arrival distribution. A more demanding packet arrival process is a batch arrival process. The simulator has not implemented this distribution, nor does the simulator support transient analysis. A further study for the NBWF-project should be transient analysis of the core protocol under batch arrival. To illustrate the behaviour under another packet arrival distribution, we select a deterministic packet arrival distribution. From the queuing theory, we know that this is a less demanding distribution than the Poisson distribution due to its zero variance.

Figure 7.3 gives the simulation results for the link delay and the MAC connection setup delay (both cases had the same throughput capacity). We clearly see better performance under deterministic packet arrivals.

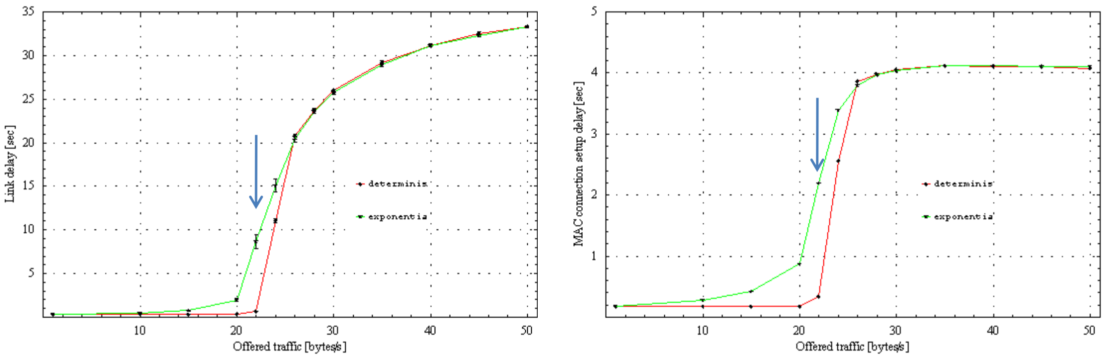

Figure 7.3 Link delay (left) and MAC connection setup delay vs. offered traffic for an AHAn25 using short packets (10 bytes payload). The vertical arrow points out the network saturation point (simJune22c).

Figure 7.4 plots the number of busy nodes as a function of time. This is a single sample path from time instance zero where the network queues are empty. Note that the scales of the y-axis are very different. The exponential distribution gives a much higher fluctuation of the number of busy nodes.

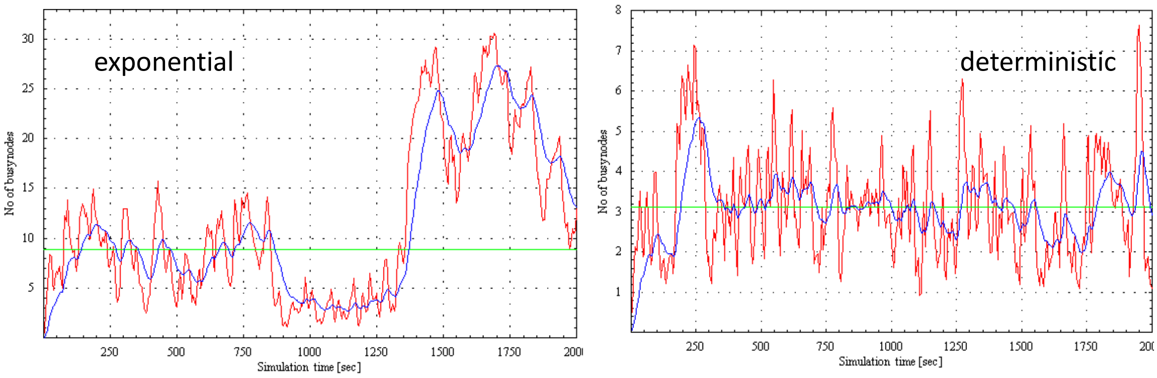

Figure 7.4 Number of busy nodes as time series for two different packet arrival distributions. The measurements are taken at the load level where the network reaches saturation (26 bytes/s, 2.6 packets/s). The blue graph represents the exponential moving average using weighting factor 0.1.

7.3 Sensitivity to nonconforming priority distributions

We have defined a priority distribution for the offered traffic from the IP clients. The impact of this distribution is reduced when the network becomes congested since the lower priority queue size becomes larger than one and then the MAC layer always has low priority traffic to serve. However, the most important part of the throughput plot in Figure 1.1 is the part that covers the steady-state, and the MAC parameter should be optimised for this part of the plot. We have no guarantee that the IP clients in a real network apply the same distribution. Therefore it is necessary to analyse the sensitivity to different priority distributions after the MAC parameter set has been specified. This is a subject for further study.

8 Conclusions and Remarks

Multi-Level Precedence and Preemption (MLPP) is a network service which facilitates priority labelling of the IP packets according to their importance as determined by the IP clients. During network congestion, the NBWF core protocols shall first serve the packets with the highest priority.

IP QoS classes [10,11] and MLPP priority are two different subjects. The former specifies a method to differentiate between **application types** which demand different communication channel characteristics (e.g., low jitter, short delay,...). This in contrast to the MLPP priority level that shall reflect the importance of the **information content**.

A narrow band network has low throughput capacity and to experience network congestion during normal usage must be regarded as a normal event. Therefore it is important to have a robust MLPP service which increases the passability of the highest priority traffic during periods of network congestion. Many tactical communications scenarios may take benefit of the MLPP service: 1) Unforeseen external events imply need for more network capacity than predicted, see Figure 1.1; 2) The radio environment became more difficult than expected (e.g. higher pathloss, network jamming or network interference from friendly forces); or 3) Higher user mobility than planned introduce more routing traffic and more radio hops.

The MLPP service is based on a cross layer design where each layer implements priority handling and the local flow control function between adjacent layers provides for service of the highest priority traffic. Each node implements a priority aware buffer system for storing the input traffic from the IP clients. These buffers are located at layer 3 (network layer) and may store a significant number of packets compared to the lower layers. We do not see a need for a pre-emption function above the MAC layer because the packet lifetime control function will delete packets before lack of buffer space becomes a practical problem.

MAC pre-emption means to disconnect the existing connection for a lower priority data packet when a higher priority data packet arrives. The cost of releasing a MAC connection and allocating a new connection for a pre-empted data packet is high. Moreover, implementation of a pre-emption service between two remote nodes demands signalling over the radio channel and is regarded as a complex lengthy process. Pre-emption is therefore not recommended on data traffic. The maximum LLC Service Data Unit (SDU) size shall be short[18] and we assume the blocking delay of high priority data becomes acceptable.

This document also addresses adaptive MAC scheduling. The purpose of an adaptive MAC scheduling process is to maintain an optimum throughput/delay-performance under changing traffic load in a fully meshed network[19]. NBWF uses a radio with a large coverage area and a typical network has good connectivity and most of the traffic is single-hop.

---

[18] less than 1500 bytes

[19] This function is not a countermeasure to the hidden-node problem that may occur in multihop network.

The cost of implementing adaptive MAC scheduling is the signalling capacity used in the fixed slots (TDMA slot 7 and 8) for the periodic MAC Load Level (MLL) reports and the increased software complexity. A network without adaptive MAC scheduling must select a larger random delay parameter (t_u) to prevent network collapse (Figure 4.1). The main benefit of adaptive MAC scheduling is a shorter link delay at low to medium load since an adaptive scheme can use a lower t_u here (Figure 4.7). We also observed improved connection setup success rates (Figure 4.5). The benefit of adaptive MAC scheduling increases as the network size increases. Some adjustments to the MAC load estimation process is needed (see section 6.3) and we need to tune some of the parameters. However, we find the present study sufficiently complete at the current stage of the NBWF-project.

The adaptive MAC scheduling works in consort with the MLPP but is not a prerequisite for the MLPP service. The MLPP service shall be a mandatory service for NBWF while the adaptive MAC scheduling may be specified as optional since this function has less importance for the network users than the MLPP service. Nodes supporting adaptive MAC scheduling must disable this function before deployment in a network where this function is not supported.

References

[1] “Requirements for a Narrowband Waveform”, AC/322(SC/6-AHWG/2)M(2008)0003, August 2008.

[2] Svein Haavik, ”Initial link layer protocol design for NBWF – input to NATO SC/6 – AHWG/2”, FFI-rapport 2009/01895.

[3] Vivianne Jodalen, ”Modelling the NBWF radio”, TIPPER/FFI project document, FFI June 2008.

[4] Tore J Berg, “The design of an initial NBWF network simulator”, FFI-report 2008/01921, FFI November 24th 2008.

[5] Bjørnar Libæk, et.al, “Enhancements to the Narrowband Waveform (NBWF) network simulator”, FFI-report 2009/01765, FFI June 10th 2008.

[6] Bjørnar Libæk and Bjørn Solberg, “A simulator model of the NATO Narrowband Waveform physical layer”, FFI-notat 2011/00533, FFI October 19th, 2011.

[7] Vivianne Jodalen, et.al, “NATO Narrowband Waveform (NBWF) – overview of link layer design”, FFI-report 2009/01894, FFI March 28th, 2011.

[8] Tore J Berg, “Design of an initial LLC Data Protocol for the NBWF Simulator”, FFI-report 2011/00537, FFI March 21st, 2012.

[9] R. G. Cole and B. S. Farroha, “Implications of Precedence and Preemption Requirements on Packet Based Transport Architectures”, MILCOM 2007.

[10] S. Blake et al., “An Architecture for Differentiated Services”, RFC 2475, 1998, http://www.ietf.org.

[11] D. Grossman, “New Terminology and Clarifications for DiffServ”, RFC 3260, 2002, http://www.ietf.org.

Terms and Acronyms

AHA	All hearing all
ARQ	Automatic Repeat Request
CAS	Carrier sense
CaP	Capability Panel
CaT	Capability Team
CC	Connect Confirm
CC PDU	Connect Confirm PDU
CCCH	Common Control Channel
CEID	Connection Endpoint Identifier
CL	ConnectionLess
CNR	Combat Net Radio
CO	Connection Oriented
CODTC	Connection oriented data traffic channel
CR	Connect Request
CR PDU	Connect Request PDU
CTS	Clear To Send
DOM	Document Object Model
DR-PDU	Disconnect Request PDU
DSSS	Direct Sequence Spread Spectrum
DT PDU	Data PDU
EPM	Electronic Protective Measures
FBN	First Bit Number
FSM	Finite State machine
GiD	Global identifier
ICI	Interface Control Information
IP	Internet Protocol
IP-SAP	Internet Protocol SAP
LBN	Last Bit Number
LLC	Logical Link Control
LLC-AM	LLC Acknowledged Mode
LLCE	LLC Entity
LLCP	LLC Protocol
LLC-TM	LLC Transparent Mode
LLC-UM	LLC Unacknowledged Mode
MAC	Medium Access Control
MACE	MAC Entity
MAC-SP	MAC Service Provider
MANET	Mobile Ad-hoc NETwork
MIP-SAP	Multicast IP SAP
MLL-report	MAC Load Level Report
MRATCH	Multicast Random Access CHannel
MTCH	one-to-Multipoint Traffic CHannel

MV	Multicast Voice
MV-SAP	Multicast Voice SAP
NBWF	Narrow Band Wave Form
NC3B	NATO C3 Board
NM-SAP	Network Management SAP
OSI	Open System Interconnection
OSI	Open system
OTCH	one-to-One Traffic CHannel
PCAS	Premature CAS
PCI	Protocol Control Information
PDP	Packet Data Protocol
PDU	Protocol Data Unit
PHY	Physical
PTT	Push To Talk
RATCH	Random Access Traffic CHannel
RF	Radio Frequency
RLC	Radio Link Control
RM	Reference Model
RRC	Radio Resource Control
RTS	Request To Send
SAP	Service Access Point
SDU	Service Data Unit
SNR	Signal to Noise Ratio
SP	Service Provider
SQL	Structured Query Language
TDM	Time Division Multiplexing
TDMA	Time Division Multiple Access
UE	User Environment or User Equipment
UTL	Utility
UV	Unicast Voice
UV-SAP	Unicast Voice SAP
XML	Extensible Mark-up Language
xxx-E	xxx Entity (e.g., LLC-E)
xxx-SAP	xxx Service Access Point (e.g., LLC-SAP)
xxx-SP	xxx Service Provider (e.g. MAC-SP)