University of Würzburg
Institute of Computer Science
Research Report Series

Performance of TCP/IP with MEDF Scheduling

Ruediger Martin, Michael Menth, Vu Phan-Gia

Report No. 323 February 2004

University of Wuerzburg, Germany
Institute of Computer Science
Department of Distributed Systems
Am Hubland, D-97074 Würzburg, Germany
phone: (+49) 931-8886652, fax: (+49) 931-8886632
{martin|menth|phan}@informatik.uni-wuerzburg.de

Performance of TCP/IP with MEDF Scheduling

Ruediger Martin, Michael Menth, Vu Phan-Gia
University of Wuerzburg, Germany
Institute of Computer Science
Department of Distributed Systems
Am Hubland, D-97074 Würzburg, Germany
phone: (+49) 931-8886652, fax: (+49) 931-8886632
{martin|menth|phan}@informatik.uni-wuerzburg.de

Abstract

To achieve Quality of Service in next generation networks, DiffServ implements appropriate Per Hop Behavior (PHB) for service differentiation. Different flow classes compete for buffer space and forwarding speed in the routers. This is influenced by buffer management and packet scheduling. We examine space priority mechanisms for buffer management like Full Buffer Sharing (FBS), Buffer Sharing with Space Priority (BSSP), and Random Early Detection (RED) gateways and time priority mechanisms for packet scheduling like First In First Out (FIFO), Static Priority (SP) and Modified Earliest Deadline First (MEDF). In particular, we focus on the characteristics of MEDF in TCP/IP networks. Our performance study reveals that MEDF is an attractive mechanism to achieve service differentiation for TCP flows already in presence of low and medium overload. Other mechanisms appear less adequate. RED, for instance, is only effective in extremely high overload situations.

Keywords: Scheduling, TCP-Performance

1 Introduction

Current research for next generation networks focuses amongst others on the provision of Quality of Service (QoS) for different service classes. The differentiated services architecture [1], [2] achieves QoS by implementing appropriate Per Hop Behavior (PHB) for different Transport Service Classes (TSC). This involves the definition of suitable TSC parameters and mechanisms to enforce the respective preferential service in the network. Hence, flows of different TSCs compete for the resources buffer space and forwarding speed in the routers. Mechanisms that assign those resources divide buffer space among different TSCs (buffer management) and control the order in which packets are dequeued and forwarded (scheduling). Therefore, those mechanisms can be characterized along two dimensions: space and time.

In this work we examine space priority mechanisms like Full Buffer Sharing (FBS), Buffer Sharing with Space Priority (BSSP), and Random Early Detection (RED) gateways and time priority mechanisms like First In First Out (FIFO), Static Priority (SP) and Modified Earliest

Deadline First (MEDF). In particular we focus on the characteristics of MEDF in TCP/IP networks. MEDF [3] implements priorities on traffic classes by introducing relative delay factors. We combine it with the space priority mechanisms and contrast it to the other time priority algorithms.

This work is structured as follows. In Section 2 we present the algorithms under study in detail. Section 3 discusses the simulation environment, the respective parameters used for our performance evaluation study, and presents the results obtained from our simulations. Section 4 finally concludes this work with a short summary.

2 Space and Time Priority Mechanisms

Network congestions arises where different flows compete for resources at routers in the network. To avoid this problem at least for a certain subset of high priority flows, network operators classify flows with different priorities. Flows of higher priority should receive preferential service as opposed to low priority flows. Basically, if packet arrivals exceed the router forwarding speed temporarily or permanently, congestion arises and buffers fill up. This leads to longer network delays and high packet loss rates, to degraded Quality of Service. Buffer sizes and forwarding speed are fixed parameters for given networks. To assign these scarce resources to flows of different priority classes, we can limit the space available to the respective flows (buffer management) or we can dequeue the packets depending on their priority (scheduling). Thus, mechanisms to achieve service differentiation can be divided along two dimensions: space and time. Combinations of both are also possible.

2.1 Space Priority Mechanisms

We use three kinds of space priority mechanisms for our performance evaluation: Full Buffer Sharing, Shared Buffers with Space Priority, and Random Early Detection gateways [10]. In the following sections, we denote the router buffer by B and packets by P. The function $S(B)$ refers to the maximum buffer size and $F(B)$ to the current fill level of the buffer. The function $enqueueTail(P, B)$ enqueues the packet P into the buffer B. The function $drop(P)$ drops the packet P if the algorithms cannot accept the packet.

Require: Packet P, Buffer B, max Buffer Size $S(B)$
  if $F(B) < S(B)$ **then**
    $enqueueTail(P, B)$
  else
    $drop(P)$
  end if

Algorithm 1: Full Buffer Sharing ENQUEUE

Full Buffer Sharing (FBS). The FBS strategy (cf. Alg. 1) allows all flows to share the same buffer irrespective of their priority. If not mentioned differently, we use this mechanism as default in our simulations.

Buffer Sharing with Space Priority (BSSP). The BSSP queueing strategy (cf. Alg. 2) allows packets to occupy buffer space available for their TSC and for all TSCs of lower priority. Let TSC_i, $i \in \{0, \ldots, n-1\}$ be TSCs of different priority, 0 being the highest priority. TSC_i can at most demand space BS_i^{max} in the buffer, where $BS_i^{max} \geq BS_{i+1}^{max}$ and BS_0^{max} is set to the actual buffer size. If the function $F(B, TSC_i)$ defines the space in the buffer B that is currently filled by TSC_i, the following condition has to hold: $\sum_{j=i}^{j=n} F(B, TSC_j) \leq BS_i^{max}$, i.e., the classes of lower priority including TSC_i are allowed to claim a share of BS_i^{max} altogether. The concept is illustrated in Figure 1 for three TSCs. There is a guaranteed amount of buffer space for the highest priority class only, lower priority classes possibly find their share taken by classes of higher priority. This concept resembles the Russian dolls bandwidth constraints model (RDM) suggested by the IETF traffic engineering working group (TEWG) in [11].

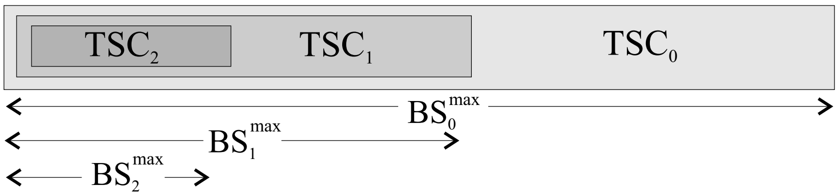

Figure 1: Buffer Sharing with Space Priority for $i = 3$ TSCs.

Require: Packet P, Buffer B, max TSC Buffer Size BS_i^{max} for $i = 0 \ldots n-1$
{ max Buffer Size $S(B) = BS_0^{max}$ }
i = TSC(P)
if $\sum_{j=i}^{j=(n-1)} F(P, TSC_j) \leq BS_i^{max}$ **then**
$enqueTail(P, B)$
else {space limit exceeded for TSC i}
$drop(P)$
end if

Algorithm 2: Buffer Sharing with Space Priority ENQUEUE

Random Early Detection (RED). The RED gateway presented in [10] and recommended for deployment in the Internet in [12] is designed to detect incipient congestion by measuring the average queue length.

Packets are dropped or marked to indicate congestion to senders. RED calculates the average queue length using an exponential weighted moving average (EWMA) to filter sudden increases due to traffic bursts. Several improvements have been suggested for instance in [13] and [14] to achieve fairness in the presence of non-adaptive connections and to introduce TSC priorities. As we are interested in the general potential of various space and time priority mechanisms, we apply RED to the low priority TSC (TSC_{low}) only in the following manner. The packets of the TSC with high priority (TSC_{high}) enter the queue as long as buffer space is available. RED monitors the average queue length for TSC_{low} packets using the EWMA filter. If this queue length reaches a threshold min_{th}, it starts dropping TSC_{low} packets according to a probability function taken from the original RED algorithm in [10]. The probability increases with the queue length. This algorithm complies with Weighted RED where the minimum and maximum allowable buffer size for TSC_{high} are identical with the overall buffer size. Calculating the average queue length as a common value for both classes as in the original RED algorithm leads to starvation of the class with low priority (higher drop probability). With the queue size growing, more and more TSC_{low} packets are dropped, the senders decrease their rates, and the TSC_{high} flows can occupy the buffer space that becomes available now. Finally, all buffer space is taken by TSC_{high} and no space is left for TSC_{low}. Therefore we compute the average queue length for TSC_{low} only and adjust the parameters accordingly.

2.2 Time Priority Mechanisms

Once packets arrive at the queue and the space priority mechanism assigns available buffer space, i.e., it decides whether the packet is accepted or dropped, the time priority mechanism decides which packet to dequeue next. We evaluate three time priority mechanisms: FIFO, Static Priority (SP), and Modified Earliest Deadline First (MEDF).

First in First Out (FIFO). FIFO leaves the prioritization to the enqueueing option and is used as the performance baseline to compare with. Packets proceed in the order they arrive and are accepted by the space priority mechanism.

Static Priority (SP). The Static Priority concept chooses TSC_{high} packets in FIFO order as long as packets of that class are in the buffer. TSC_{low} packets wait in the router queue until low priority packets only are available. Then they are also dequeued in a FIFO manner until new TSC_{high} packets arrive.

Modified Earliest Deadline First (MEDF). In the context of the UMTS Terrestrial Radio Access Network, the authors of [3] introduced a modified version of the Earliest Deadline First (EDF) algorithm called Modified Earliest Deadline First (MEDF). It supports n only different TSCs, but in contrast to EDF it is easier to implement. Packets are stored in n TSC specific queues in FIFO manner. They are stamped with a modified deadline that is their arrival time plus an offset $M_i, 0 \leq i < n$, which is characteristic for each TSC. The MEDF scheduler selects the packet for transmission that has the earliest due date among the packets in the front positions of all queues. For only two TSCs, this is the choice between two packets and sorting according to ascending deadlines is not required. The difference $|M_i - M_j|$ between two TSCs i and j is a relative delay advantage that influences the behavior of the scheduler. We are interested in the performance of this scheduling algorithm in the presence of adaptive traffic, here TCP.

For our simulations we use two TSCs whose queues are implemented as shared buffers such that all space priority mechanisms are applicable. With two TSCs we set the MEDF parameters to $M_{high} = 0$ and $M_{low} = x, x \in \{0.01, 0.1, 0,5, 1.0, 1.5\}$. Thus, TSC_{high} obtains no additional delay. The deadline for TSC_{low} packets is increased by the M_{low} parameter.

3 Priority Mechanisms Performance Evaluation

In this section we describe the general goals and approach of our performance evaluation study and present the results. We used the network simulator (NS) version 2 [15] to run the experiments deploying the RENO TCP implementation [16]. Standard simulation methods as replicate-delete were applied to obtain statistically reliable results of the non-ergodic random processes. In the following sections we only give average values as the simulated time was chosen to yield very narrow confidence intervals. Our goal is the measurement of the prioritization of TSC_{high} traffic. For that purpose, we define the bandwidth ratio, the amount of bandwidth used by TSC_{hihg} divided by the amount of bandwidth used by TSC_{low}. We use the same number of saturated TCP sources for both TSCs, i.e., the traffic offer for TSC_{high} and TSC_{low} are the same.

3.1 MEDF characteristics

First we evaluate the MEDF characteristics as this work focuses especially on MEDF. To isolate the general behavior of the algorithms more easily and to eliminate unpredictable side effects, we started with single link simulations and extended it to multiple links.

3.1.1 MEDF Single Link Scenario

Figure 2: Single link simulation topology.

Simulation environment. Figure 2 shows the simulation topology for the single link experiment, the so-called dumbell topology. A number of TSC_{high} TCP traffic sources and a number of TSC_{low} TCP traffic sources connect to Router A. Router A uses a space and a time priority mechanism described above and sends the packets over the link to router B. Router B has sufficient capacity to serve the link and its single task is to distribute the arriving packets to the corresponding destination. Thus, the effects of the various priority management mechanisms can be monitored and analyzed easily.

We choose the number of simultaneous TCP connections n as $n_{min} \cdot 2^i, i \in \{0, \ldots, 8\}$, n_{min} being the minimum number of TCP connections to get a theoretical load of 100% on the link. Otherwise there is no overload, space and time priorities do not have effect, and the flow control is not active. Here $n_{min} = 2$. In this work we divide the number of concurrently active saturated TCP sources equally among both TSCs. The packet size $S(P)$ is a common standard value of 500 Bytes including headers. Regarding the link parameters, with the link bandwidth being $C_l = 1.28Mbit/s$, we set the link propagation delay D_{prop} to 46.875 ms so that the theoretical round trip time RTT sums up to $RTT = 2 \cdot (n_{links} \cdot D_{prop} + (n_{links} + 1) \cdot D_{TX}) = 2 \cdot (1 \cdot 46.875ms + 2 \cdot 3.125ms) = 100ms$, where $D_{TX} = \frac{S(P)}{C_l}$ is the transmission delay to send a packet and n_{links} the number of links between routers A and B.

The default value for the buffer size S_{Buffer} is 160 packets so that a router is able store packets for 0.5 seconds transmission. We use the parameters mentioned here as default parameters and write down the respective values in the following text only if they are set differently. Other parameters like algorithm specific settings are subject to the analysis and we indicate their values appropriately.

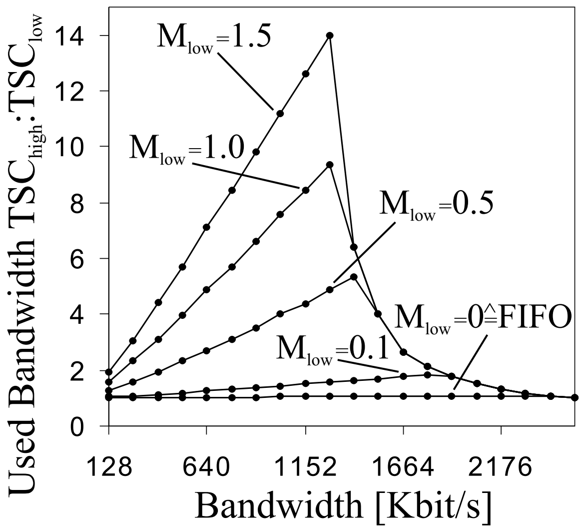

Figure 3: MEDF prioritization for two TSCs.

Simulation. Basic experiments show that there is virtually no priority for the minimum number of users ($n_{min} = 2$ here). The link capacity is fully shared between the single user of each class, thus, they reach the maximum rate. This behavior – as expected – is sound for lack of competition on the link. Prioritization of TSC_{high} traffic reaches its maximum at $n = 4$ users (2 users per TSC) and degrades with a rising number of users. As we cannot simulate any value between two and four users – one and two users per TSC – we vary the bandwidth while keeping the number of users fixed at a value of 4 to derive the basal characteristics of the algorithm by having a more continuous range.

Figure 3 shows the bandwidth used by TSC_{high} in multiples of the bandwidth used by TSC_{low}. The link bandwidth is the x-axis parameter. At a bandwidth of 1.280 Mbit/s this experiment corresponds to a simulation with default values and 4 users, at a bandwidth of 2.560 Mbit/s it is equivalent to 2 users. Higher offset values M_{low} lead directly to a higher prioritization of TSC_{high} packets. If the bandwidth is low, which corresponds to many users per TSC and heavy competition for the link resources, the throughput ratio is low as well and increases with the bandwidth. Low bandwidth (same holds for many users) limits the rate connections for TSC_{high} can achieve dramatically. Besides, the actually measured round trip time increases and shortens the maximum obtainable rate. Thus, TSC_{low} connections are able to grasp a higher relative share of the bandwidth. The bandwidth ratio rises until it reaches a maximum. Here, slowly sufficient capacity becomes available for both TSCs and low priority packets can use more of the additional bandwidth. At 2.560 Mbit/s there is virtually no competition for bandwidth anymore.

The MEDF parameter M_{low} can be used to adjust the priority ratio for the anticipated level of competition for network resources. If sufficient resources are available, the MEDF algorithm does not influence normal network operation. For very scarce resources – here large numbers of users and low bandwidth, respectively – the network is under heavy overload and anticipatory action like admission control to block some of the connections must be taken to prevent such situations. Otherwise, only a very small portion of the overall bandwidth remains for each TSC_{high} flow anyway — no matter whether they receive preferential service or not. For low and medium overload, MEDF shows a very clear behavior.

3.1.2 MEDF Multi Link

We now extend our single link experiment to multiple links to assess the influence of MEDF on TSC priority if applied multiple times.

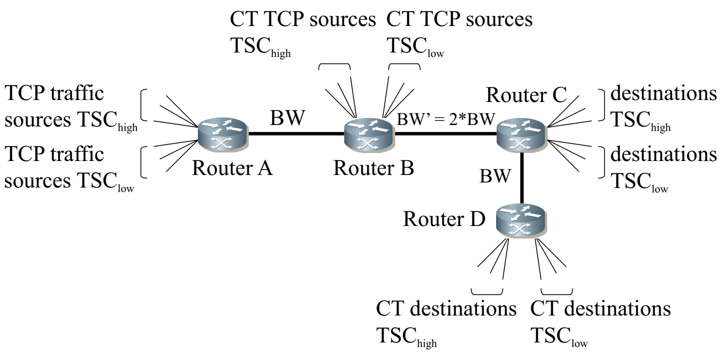

Figure 4: Multi link simulation topology.

Simulation environment. Figure 4 shows the simulation topology for the multi link experiment in the case of two links. If we simply add additional links and routers, the first router receives the packets from the TCP sources in an unordered way and applies the priority algorithm. Thus, the packets arrive at the router serving the next link one by one and the priority algorithm has no additional effect. To overcome this problem, we introduce cross traffic. Additional TCP sources connect to the interior routers and generate traffic that crosses the way of the measured traffic.

It is important to send the cross traffic over the same number of links to account for comparable round trip times for the measured traffic and the cross traffic. Furthermore, the round trip time for both the single link and the multi link experiment should be the same. Otherwise, significant parameters that depend on the round trip time such as the maximum rate that can be achieved by a TCP connection are different and the experiments are not comparable. Therefore, we calculate the new link propagation delay $D_{prop} = \frac{46.875-(n_{link}-1)\cdot D_{TX}}{n_{link}} ms$.

The TCP connections need the same bandwidth per flow on all links. If the bandwidth differs from link to link, the link with the lowest capacity becomes the bottleneck and dominates the observable effects. However, doubling the bandwidth of the links with cross traffic solves this problem.

Simulation. Figure 5 shows the effect of MEDF over multiple links. We used the standard parameters with $M_{low} = 1$ and the default Full Buffer Sharing mechanism as buffer management.

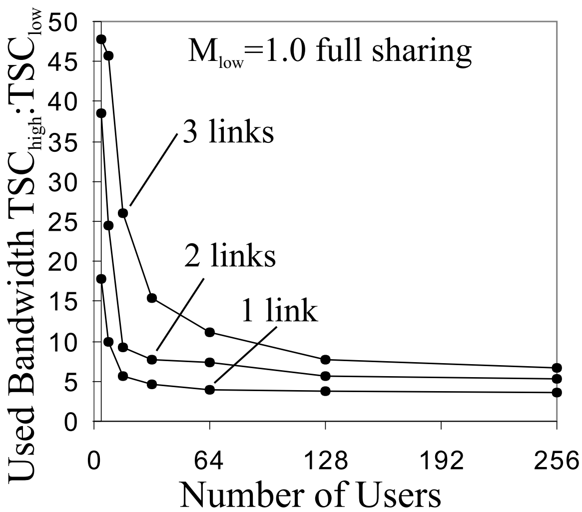

Figure 5: MEDF prioritization in a multi link topology.

In general, the degree of prioritization of TSC_{high} increases with the number of links on the path, hence, with the number of applications of MEDF scheduling instances. However, when the competition for network resources is low, the increase in priority is much more obvious. The reason behind this is similar to the situation for the single link experiment. The bandwidth theoretically available to a single connection is higher, hence, the actually measured round trip time is lower. Therefore, few TSC_{high} connections achieve higher rates in contrast to the situation when the network is highly overloaded. Rising competition for network resources makes the conditions for TSC_{high} more disadvantageous. TSC_{low} now obtains a larger share of the bandwidth. The priority does not increase linearly if additional links are added. The overall bandwidth ratio can be controlled by setting the MEDF parameter appropriately.

3.1.3 MEDF and Space Priority

We now consider the MEDF characteristics with the usage of space priority mechanisms. Figure 6 shows the influence of the buffer sharing option. FIFO with FBS leads to an even division of available bandwidth between both TSCs as no packet preferences exist. FIFO with BSSP spreads the bandwidth equally as long as there is enough buffer space available (n ≤ 2). Then it reaches its maximum when router buffers fill completely and slightly flattens under heavy traffic load.

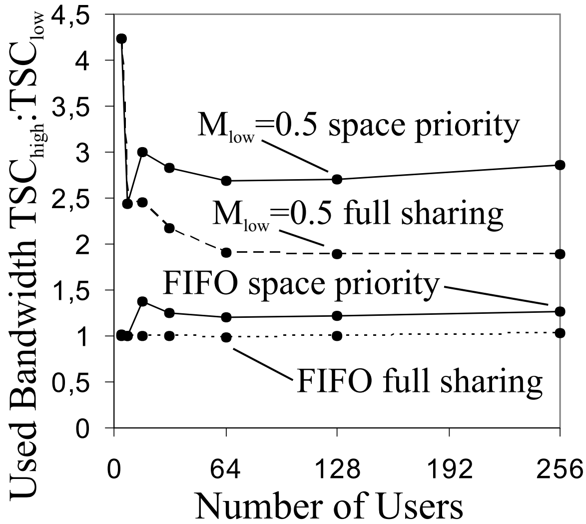

Figure 6: MEDF and the impact of space priority.

MEDF with parameter $M_{low} = 0.5$ and FBS clearly outperforms both FIFO experiments and exhibits the behavior characterized in the preceding sections. If we add BSSP, we observe a superposition of the MEDF curve and the curve for FIFO with BSSP. For few users we clearly identify the typical MEDF characteristics, for more users the router buffers fill completely and the space priority comes into play. Thus, space priority prohibits the typical decrease of the bandwidth ratio.

3.2 MEDF in comparison to other priority mechanisms

Finally we examine other priority mechanisms like Static Priority and RED with priority and contrast it to MEDF.

3.2.1 Static Priority

Under network congestion, the time priority mechanism Static Priority leads to starvation of TSC_{low} regardless of the buffer management in use. There are always TSC_{high} packets waiting in the router queues. SP dequeues those packets and even though the TSC_{low} packets occupy most of the buffer space, their chance to leave the buffer is very low and, thus, the

TCP timers for those connections expire. Accordingly, the TCP source tries to re-establish the connection but will suffer from starvation again. As a consequence, SP is completely inadequate for severely congested networks. In contrast to other time priority mechanisms — specifically MEDF — it does not consider a maximum delay for low priority traffic to prevent this effect.

3.2.2 Random Early Detection (RED)

RED gateways can be used to introduce space priority into router queues as described in Section 2. The average queue length is calculated for TSC_{low} traffic only to prevent starvation of those flows.

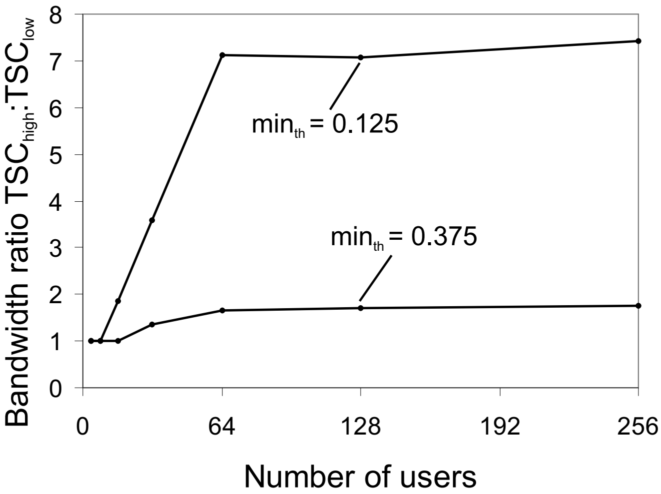

Figure 7: RED prioritization for two TSCs.

Figure 7 shows the simulation results for RED. In contrast to MEDF, the bandwidth ratio starts at a value of 1, i.e., TSC_{high} receives no preferential service. The network is slightly congested and the TSCs compete for network resources but RED as a pure space priority mechanism cannot take effect. Router buffers are not constantly filled at a high level and, therefore, the drop probability is too low. This effect becomes even more apparent for higher thresholds min_{th}. For medium network congestion the bandwidth ratio rises and finally reaches a nearly constant level for high network congestion. The threshold value influences the bandwidth ratio directly. Lower values make RED drop packets earlier and prefer TSC_{high} packets stronger.

In contrast to RED, MEDF strongly prioritizes TSC_{high} already in low and medium congested networks due to its scheduling capabilities. Indeed, MEDF reaches its maximum when the network load slightly exceeds 100% (cf. Figure 3 at 1.280 Mbit/s).

4 Conclusion

In this work we examined the influence of three time priority mechanisms (packet scheduling) on the throughput of two different Transport Service Classes (TSCs) in presence of network congestion: First In First Out (FIFO), Modified Earliest Deadline First (MEDF), and Static Priority (SP). SP leads to starvation of low priority traffic while FIFO effects no prioritization at all. MEDF achieves the desired priority ratio of the high priority TSC over the low priority TSC in realistic overload situations by its adjustable parameter M_{low} which reflects a relative delay advantage.

We combined these scheduling algorithms with different buffer management schemes, namely Full Buffer Sharing (FBS), Buffer Sharing with Space Priority (BSSP), and Random Early Detection (RED) Gateways. BSSP enforces the prioritization of MEDF but with FIFO scheduling it has only effect in severly congested networks.

RED realizes preferential treatment of high priority traffic in heavy overload situations. However, it fails in case of low or medium overload, which is more likely than severe overload and for which MEDF has more powerful service differentiation capabilities. Hence, our performance study revealed that MEDF is an attractive mechanism to achieve service differentiation for TCP flows in congested networks.

Acknowledgment

The authors would like to thank Prof. Tran-Gia for the stimulating environment which was a prerequisite for that work.

References

[1] S. Blake, D. L. Black, M. A. Carlson, E. Davies, Z. Wang, and W. Weiss, "RFC2475: An Architecture for Differentiated Services." ftp://ftp.rfc-editor.org/in-notes/rfc2475.txt, Dec. 1998.

[2] D. Grossman, "RFC3260: New Terminology and Clarifications for Diffser." ftp://ftp.rfc-editor.org/in-notes/rfc3260.txt, April 2002.

[3] M. Menth, M. Schmid, H. Heiß, and T. Reim, "MEDF - A Simple Scheduling Algorithm for Two Real-Time Transport Service Classes with Application in the UTRAN," *IEEE INFOCOM 2003*, April 2003.

[4] S. Ramabhadran and J. Pasquale, "Stratified round Robin: a low complexity packet scheduler with bandwidth fairness and bounded delay," *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications Karlsruhe, Germany*, pp. 239–250, 2003.

[5] D. Stiliadis and A. Varma, "Efficient fair queueing algorithms for packet-switched networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 6, pp. 175–185, April 1998.

[6] B. Davie *et al.*, “An Expedited Forwarding PHB (Per-Hop Behavior).” ftp://ftp.rfc-editor.org/in-notes/rfc3246.txt, March 2002.

[7] A. Charny *et al.*, “Supplemental Information for the New Definition of the EF PHB (Expedited Forwarding Per-Hop Behavior).” ftp://ftp.rfc-editor.org/in-notes/rfc3247.txt, March 2002.

[8] M. Shreedhar and G. Varghese, “Efficient Fair Queueing Using Deficit Round-Robin,” *IEEE/ACM Transactions on Networking*, vol. 4, June 1996.

[9] S. Floyd and V. Jacobson, “Link-sharing and Resource Management Models for Packet Networks,” *IEEE/ACM Transactions on Networking*, vol. 3, August 1995.

[10] S. Floyd and V. Jacobson, “Random Early Detection Gateways for Congestion Avoidance,” *IEEE/ACM Transactions on Networking*, vol. 1, pp. 397–413, August 1993.

[11] F. Le Faucheur, “Russian Dolls Bandwidth Constraints Model for Diff-Serv-aware MPLS Traffic Engineering.” Internet Draft TEWG, June 2003.

[12] B. Braden *et al.*, “Recommendations on Queue Management and Congestion Avoidance in the Internet,” *IETF RFC2309*, April 1998.

[13] F. Anjum and L. Tassiulas, “Balanced-RED: An Algorithm to Achieve Fairness in the Internet,” *IEEE INFOCOM 1999*, March 1999.

[14] U. Bodin, O. Schelén, and S. Pink, “Load-tolerant Differentiation with Active Queue Management,” *SIGCOMM Computer Communication Review*, July 2000.

[15] K. Fall and K. Varadhan, “The ns Manual.” http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf, December 2003.

[16] W. R. Stevens, *TCP/IP Illustrated, Volume 1*. Addison-Wesley Longman, 1994.