Performance Evaluation of AQM Schemes in Rate-Varying 3G Links

Juan J. Alcaraz, Fernando Cerdan

[1] Department of Information Technologies and Communications, Polytechnic University of Cartagena, Plaza del Hospital, 1, 30202 Cartagena, Spain
{juan.alcaraz, fernando.cerdan}@upct.es

Abstract. When TCP is carried over 3G links, overbuffering and buffer overflow at the RLC layer degrades its performance. AQM techniques at the RLC buffer can bring noticeable enhancements to TCP performance without introducing changes in 3G specifications. We show that the optimum parameter setting of AQM algorithms in RLC buffers is strongly related to the radio bearer rate, which can be changed dynamically by control layer protocols. By means of extensive simulation experiments we propose, for each specified nominal rate, optimum configurations that keep the goodput near the maximum while the delay is reduced up to 50%. We consider two AQM schemes, an adapted RED algorithm and a novel deterministic one, SBD described in this paper. We illustrate how an automatic reconfiguration of AQM parameters avoids the degradation caused by sudden changes in the radio bearer rate.

1 Introduction

Third generation cellular networks (3G) are expected to be an important part of the Internet. Many Internet applications like e-mail, web surfing and file transfer rely on TCP for the end-to-end transport. In 3G radio access networks, the link layer is managed by the Radio Link Control (RLC) protocol [1] which can be configured to provide a reliable service, recovering from propagation errors. A reliable RLC layer reduces packet losses perceived at TCP layer, avoiding the triggering of unnecessary congestion control measures [2, 3]. However, several characteristics of 3G links like high and variable latency and buffer overflow of the downlink buffers [4, 5, 6], have undesired effects on TCP performance.

In order to overcome these effects, recent works [5, 6] propose the application of Active Queue Management (AQM) techniques at the downlink RLC buffers. AQM can improve TCP performance over 3G links with a small change at the Radio Network Controller (RNC) nodes. In contrast to other proposals, this approach does not require changes in TCP itself and does not break the end-to-end semantics of TCP.

This paper addresses the configuration of AQM parameters in an RLC buffer considering the variations on the RB nominal rate. These variations have a significant

---

This work was supported by the Spanish Inter-Ministerial Sciece and Technology Commission under project TEC2005-08068-C04-01/TCM.

effect on TCP performance, as we illustrate, and may be caused, e.g. by the 3G scheduling mechanisms or handovers among cells. We propose two alternative AQM schemes, Random Early Detection [7] (RED) and Slope Based Discard (SBD).

RED is one of the most extended AQM mechanisms for Internet routers, and its adaptation to the particularities of 3G links is described in [6]. We contribute providing further insight into the parameter setting of RED in radio bearers. Based of extensive simulation experiments we evaluate multiple RED configurations in different RBs.

SBD is a novel deterministic AQM algorithm especially suitable to the characteristics of 3G links. In this paper we describe SBD and disclose how its parameters should be configured regarding the RB bandwidth.

In the 3G protocol stack, the Radio Resource Control (RRC) protocol handles the resource management algorithms, setting up and modifying layer 1 and layer 2 protocol entities [8]. The operating scheme that we propose is completely compatible with this architecture: the RRC entity, responsible of changing the RB rate, should reconfigure AQM parameters according to the rate value. In this paper we provide examples of AQM reconfiguration under abrupt RB bandwidth changes. It should be noticed that these schemes do not require additional signalling because they operate only on the RNC side, and therefore could be implemented without introducing changes in 3GPP specifications.

The rest of the paper is organized as follows. Section 2 describes how a rate-varying RB degrades TCP performance. Section 3 explains the SBD algorithm in detail. Section 4 provides a brief description of the simulation environment and the simulation scenarios. Section 5 summarizes the parameter configuration guidelines for SBD and RED derived from the simulation results. Section 6 shows the operation of each algorithm under rate variations. The paper concludes in section 7.

2 Characteristics of 3G Links

Previous works [5, 9, 10] provide a clear view of the characteristics of 3G wireless links. Radio bearers are expected to multiplex a number of simultaneous connections ranging from 1 to 4 TCP flows [5, 11], because 3G links employs per-user buffering. At the RLC layer, the upper layer packets will be stored in the downlink buffer until they are fully acknowledged by the receiver side. In consequence, as described in [4, 10] frame losses in the downlink channel result in higher RLC buffer occupancy at the network side. Considering that the current RLC specification [1] proposes a drop-tail buffer, the buffer may overflow causing consecutive packet losses. This situation is especially harmful in the first stages of a TCP connection (slow start) and has a higher impact in TCP Reno, which can only recover from consecutive losses with a Retransmission TimeOut (RTO), causing the highest reduction of the rate.

The buffer should be large enough to avoid frequent overflow. However, excessive queuing cause some additional problems [4, 5] like Round Trip Time (RTT) inflation, unfairness between competing flows and viscous web surfing.

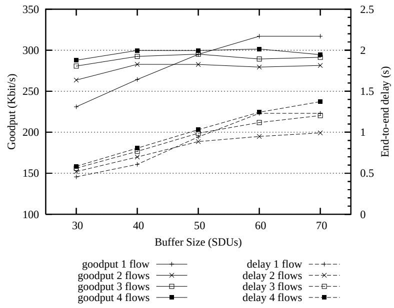

Fig. 1. TCP performance over a 384 kbit/s RB, with drop tail operation at RLC

Table 1. Simulation parameters

3G link parameters	Setting
PDU payload size	320 bits
TTI (Transm. Time Interval)	10 ms
Transmission window	1024 PDUs
maxDAT	10
In-order-delivery	true
Status Prohibit Timer	60 ms
Missing PDU detection	true
Poll Timer	60 ms
Wireless Round Trip Delay	50 ms
Normalized doppler frequency	0,01
Poll window	50 %
Last PDU in buffer Poll	yes
Last retransmitted PDU Poll	yes
Frame Error Ratio (FER)	10%

TCP parameters	Setting
Maximum TCP/IP packet size	1500 bytes
Maximum allowed window	64 kbytes
Initial window	1
Wired Network Round Trip Delay	200 ms

Fig. 1 illustrates the end-to-end goodput and delay of TCP over an RB for different RLC buffer sizes and a number of flows ranging from 1 to 4. The goodput represents the successfully received packets at the receiver and the delay is the transfer time of a packet in the downlink direction, at the TCP layer. The buffer size is given in RLC Service Data Units (SDU) of 1500 bytes. Table 1 shows the parameter configuration for the RLC and TCP protocols. The RLC parameters were set according to the optimizing considerations described in [6, 9]. Further details on the simulator are provided in Section 4. As expected, Fig. 1 reveals that a larger buffer benefits the goodput performance but the overbuffering increases the latency.

The rate variation of the RB may have additional effects on TCP performance. Fig. 2 shows the trace of two TCP connections over an RB which rate starts at 384 kbit/s and switches to 128 kbit/s 50 seconds after the start time. The curve at the top shows the RLC buffer occupancy (*BO*). The buffer capacity is set to 40 SDUs. The TCP congestion windows of each flow (*cwnd 1* and *cwnd 2*) are depicted below the *BO* curve, and the two curves at the bottom show, for each TCP flow, the sequence number of the packets when they are sent, received and dropped.

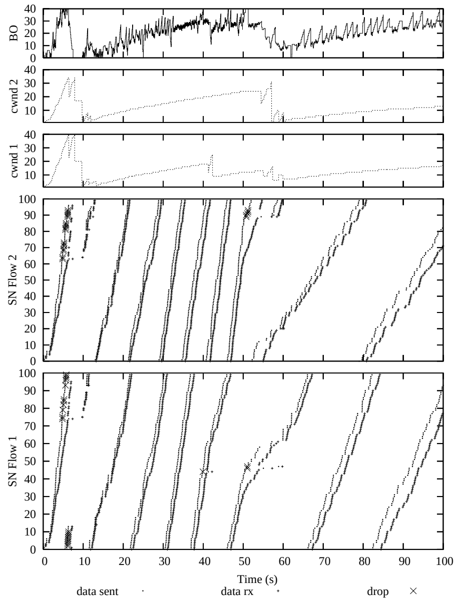

Fig. 2. Trace of two TCP connections carried over RLC

At the first stages of the connection, multiple packets are dropped due to buffer overflow, causing an RTO in both sources. The congestion windows shrink and the sources begin to recover their rate slowly. The overbuffering appears again, causing high delay and higher probability of additional overflows. At t = 50 the buffer overflows again because of the RB rate reduction. With a lower bandwidth in the 3G link, the overbuffering increases the latency even more and causes RTO inflation.

Given the behaviour of the buffer occupancy process in RLC, AQM techniques can be considered as a feasible strategy to enhance TCP performance over 3G links. AQM is aimed to maintain the buffer occupancy around a certain level, thus avoiding consecutive packet losses and reducing the delay jitter.

It should be mentioned that although RED is extensively used for Internet routers, its implementation in wireless link layer buffers is relatively novel. Thus, the effect of its parameter configuration on TCP performance is still not fully known. According to [5, 6], an RED algorithm at RLC should use the instantaneous queue size to react faster to sudden *BO* increments and to reduce operational complexity.

3 Slope Based Discard

We propose a novel AQM algorithm for the downlink RLC buffer in 3G networks, named Slope Based Discard (SBD).

The SBD algorithm is based on the following ideas:

1. A packet discard is a congestion signal directed to the TCP sender side that takes a certain amount of time, T_S, to arrive at the TCP source (see Fig. 3). The rate re-

duction is perceived at the buffer after the propagation time, T_f, of the fixed (wired) network.

2. The discarding policy is driven by the buffer filling rate, r. In normal operation, whenever r exceeds a critical value, r_c a packet is dropped. The buffer occupancy level determines the value of r_c.
3. r_c represents the filling rate that, if sustained, would fill the buffer entirely before the rate reduction can be perceived at the buffer after a packet discard.
4. After a packet drop, additional packet discarding should be avoided until the rate reduction at the TCP source can be noticed at the RLC buffer. In Fig. 3 this reaction time equals $T_S + T_f$.
5. The packet chosen for discard will be as close as possible to the front of the queue, in order to reduce the reaction time. Additionally, the algorithm should not discard a packet if its transmission over the RLC link has already started. Otherwise, upon a packet discard, the RLC would start the signalling procedure required to synchronize RLC sender and receiver sides [1]. Consequently, our protocol discards the first packet whose transmission has not started, thus reducing complexity and avoiding changes in the 3GPP specification itself

The following parameters control the SBD algorithm:

– *minth*: buffer occupancy level above which packets can be dropped.
– *maxth*: maximum occupancy allowed in the buffer.
– T_r: estimated reaction time.
– α: occupancy interval used for the estimation of the slope of *BO* process curve.

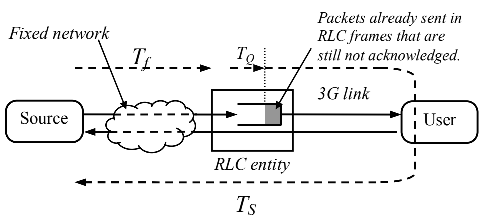

Fig. 3. Schematic diagram of the end-to-end connection with the delay times of each section

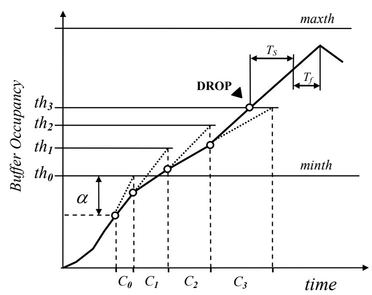

Fig. 4. Buffer ccupancy curve of a buffer implementing SBD

Fig. 4 depicts the buffer occupancy (*BO*) process in an RLC buffer as a graphical example of the algorithm.

When *BO* is higher or equal to $minth - \alpha$, the algorithm calculates the time C_n that it will take the buffer to store α additional bits at a filling rate equal to r_c ($C_n = \alpha / r_c$). The threshold level for the measuring interval is also determined ($th_n = BO + \alpha$). The value of r_c depends linearly on th_n according to the definition and expressed in (1).

$$r_c = \frac{maxth - th_n}{T_r} \quad (1)$$

A timer for C_n is started. If the timer expires and BO is below the threshold th_n, then the actual filling rate is lower than r_c, and no packet is dropped. If th_n is reached before the expiration of the timer, the current filling rate surpasses r_c and therefore a packet will be discarded.

In Fig. 4, the dotted segments starting at each measuring interval represent the buffer filling at the critical rate (critical curve). In the intervals C_0, C_1 and C_2, no packet is discarded because the BO curve is below the critical curve. In contrast, in the C_3 period, BO is above r_c. Hence, the threshold th_3 is reached before the timer C_3 expires. When th_3 is reached, a packet is dropped. The monitoring and discarding algorithm is deactivated for a period T_r, avoiding consecutive packet discards. The following pseudocode summarizes the algorithm operation.

Pseudocode of the SBD operation.

```
for each packet arrival
  update BO
  if timer Cn on
    if BO ≥ thn
      drop packet
      deactivate timer Cn
      start timer Tr
  if timer Cn off
    if (BO ≥ minth - α)
      calculate Cn
      thn = BO + α
      start timer Cn

upon timer (Cn or Tr) expiration
  if (BO ≥ minth - α)
    calculate Cn
    thn = BO + α
    start timer Cn
```

One of the main advantages of SBD compared to random or RED-like mechanisms is that it does not need to generate random numbers to compute the discarding probability because of its deterministic operation. This reduces the computational cost of the algorithm, and makes it more feasible for its implementation at the RLC level where the buffering is done in a per-user basis.

The maintenance of a new timer, C_n, does not add too much complexity to the RLC operation, which already handles several timers, e.g. *Poll Timer* and *Status Prohibit Timer* [1]. The RLC can synchronize C_n to the Transmission Time Interval (TTI) which is equivalent to a clock signal with a granularity of 10 ms (similar to that of other RLC timers).

4 Simulation Environment

The simulation environment employed in this research has been developed in OMNeT++ [12] and comprises a complete implementation of TCP and RLC protocols. Similar simulators were described in [9, 10]. The simulation topology, shown in Fig. 5 consists of one or several TCP sources connected to their respective receivers in the user's equipment (UE). The end-to-end connection consists of two sections, the wired network and the radio bearer. The wired network comprises the Internet and the 3G core network. The radio bearer has a round trip time (RTT_w) of 50 ms [2, 8] and a bidirectional nominal rate ranging from 384 kbit/s to 64 kbit/s, representing the bottleneck link, which is the situation expected in most cases [11]. The wired (fixed) network is modeled with a 1 Mb/s link with a round trip delay (RTT_f) of 200 ms [2].

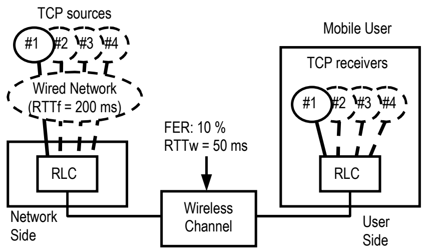

Fig. 5. Schematic representation of the simulation environment

The wireless channel generates error bursts according to the model described in [13] where the Doppler frequency, f_d, of the UE determines the average burst length. Lower f_d causes longer bursts of errors. It is usual to employ the normalized Doppler frequency, equal to the product of f_d and the radio frame duration (10 ms).

In order to obtain more realistic results, the error probability in our model is the same in the uplink and in the downlink direction. The frame loss ratio is 10%, a typical UMTS design value [2, 8].

The simulation results exposed in this paper are obtained averaging 20 runs per sample and the radius of each confidence interval is estimeated with a confidence degree of 90% according to a t-student distribution.

The TCP flavour employed is TCP Reno, one of the most extended in the Internet [11]. RLC and TCP parameter setting is shown in Table 1.

5 Parameter Configuration

In our simulations, multiple parameter combinations of RED and SBD were tested, in order to disclose the effect of each one on the end-to-end performance. Four standard RB rates are considered: 384, 256, 128 and 64 kbit/s.

In both SBD and RED the following values of *minth* were evaluated: 5, 10, 15, 20, 25 and 30 SDUs. The value of *maxth* equals the size of the RLC buffer, 40 SDUs, enough to prevent buffer overflow, keeping a low delay. In SBD the values of T_r ranged from 50 ms to 5000 ms, and α is equal to 5 SDUs, which was found to be an optimum value and a compromise value between fast detection of congestion and excessive sensitivity to occupancy oscillations.

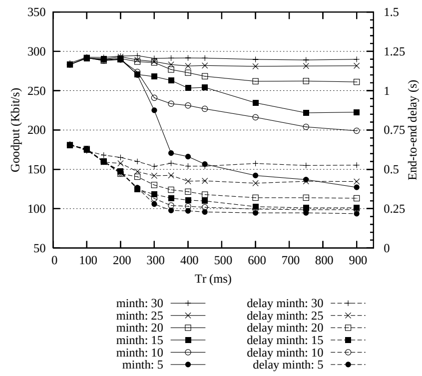

Fig. 6. Performance of one TCP flow over a 384 kbit/s radio bearer with SBD

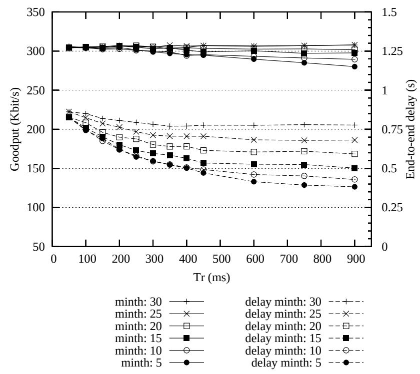

Fig. 7. Performance of 4 TCP flows over a 384 kbit/s radio bearer with SBD

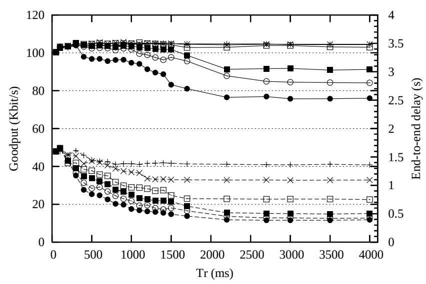

Fig. 8. Performance of one TCP flow over a 128 kbit/s radio bearer with SBD (see callouts in Fig. 6)

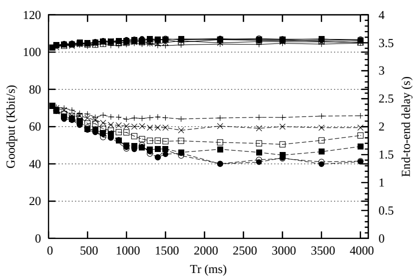

Fig. 9. Performance of 4 TCP flows over a 128 kbit/s radio bearer with SBD (see callouts in Fig. 7)

Figures 6 and 7 show the average goodput and delay of 1 and 4 TCP connections over a 384 kbit/s RB for different combinations of SBD parameter values. Figures 8 and 9 show the performance figures for a 128 kbit/s RB.

The following conclusions are derived regarding each parameter:

1) T_r has a direct impact on the aggressiveness of the discarding policy. According to (1), a higher T_r reduces r_c, and thus, the delay decreases.

2) minth determines a lower bound for the goodput and the delay because it sets the minimum TCP *cwnd* to allow a packet discard. To avoid link underutilization, *minth* should be above the Bandwidth Delay Product (DBP) of the connection, defined as the product of the Round Trip Propagation Delay (RTPD = RTT_w + RTT_f) with the bottleneck link rate. Our results show that *minth* should be set, at least, to 2×BDP in order to maintain the goodput near the optimum for a wide range of T_r values.

Comparing the performance for one and several TCP flows, we conclude that the optimum parameter setting for each nominal rate should be determined for the single TCP flow scenario. In this case, the goodput decays when the discarding policy is too aggressive (high T_r values and/or low *minth*). This fact is easily explained considering TCP *cwnd* dynamics. Using AQM, an early packet discard halves the window of the TCP connection. Obviously, in a single flow scenario this measure halves the overall user rate. This avoids buffer overflow but limits the goodput improvement. In a multiple flow scenario the overall user rate reduction is less severe because it only affects one connection upon each packet discard.

Table 2. Maximum average goodput values in SBD simulations for the single flow case

RB	T_r (ms)	*minth* (SDUs)	Goodput (kbit/s)	Delay (s)
384	250	30	294.5 ±5.6	0.55 ± 0.02
256	300	25	203.6 ± 3.6	0.71 ± 0.02
128	900	20	105.0 ± 1.6	0.99 ± 0.04
64	1500	10	52.5 ± 0.8	1.97 ± 0.14

Table 3. Proposed SBD configuration and its performance values in the single flow scenario

RB	T_r (ms)	*minth* (SDUs)	Goodput (kbit/s)	Delay (s)
384	200	20	290.5 ± 4.2	0.47 ± 0.03
256	500	20	200.1 ± 3.7	0.53 ± 0.02
128	1300	15	102.1 ± 1.5	0.72 ± 0.03
64	2500	10	50.4 ± 1.5	1.12 ± 0.12

The maximum achievable goodput values with SBD, shown in Table 2, are tied to high delays. Considering these performance values as a reference, a more aggressive setting of *minth* and T_r can lead to a delay reduction of up to 30% with a negligible reduction of the goodput (between 1% and 4%), as shown in Table 3.

In the simulations of the RED schemes, the values of *maxp* ranged from 0.005 to 1. Figures 10 and 11 show the average goodput and delay of 1 and 4 TCP connections over a 384 kbit/s RB, and Figures 12 and 13 show the performance figures for a 128 kbit/s RB. As expected, the single flow scenario is more responsive to RED parameter changes, and therefore it is the worst case for configuration.

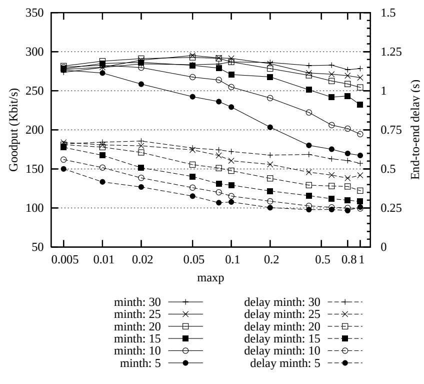

Fig. 10. Performance of one TCP flow over a 384 kbit/s radio bearer with RED

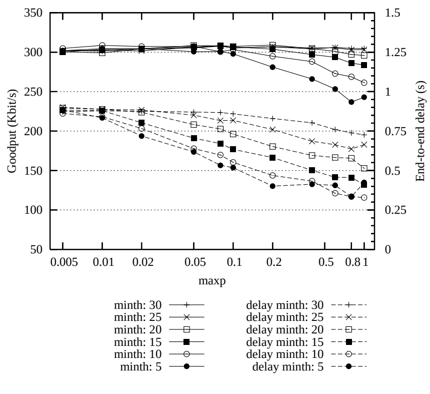

Fig. 11. Performance of 4 TCP flows over a 384 kbit/s radio bearer with RED

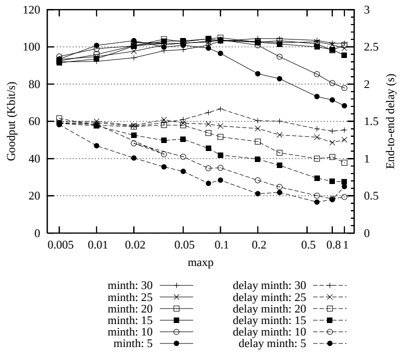

Fig. 12. Performance of one TCP flow over a 128 kbit/s radio bearer with RED

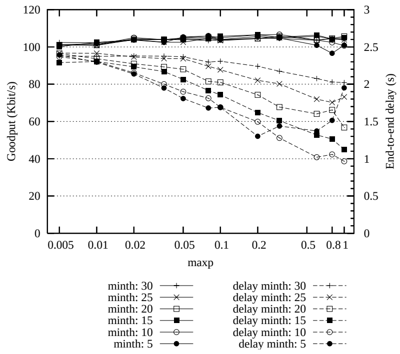

Fig. 13. Performance of 4 TCP flows over a 128 kbit/s radio bearer with RED

Following the same criteria used in SBD, the maximum goodput values, shown in Table 4, are taken as a reference to select parameters with a better balance between goodput and delay performance. The proposal is shown in Table 5.

Table 4. Maximum average goodput values in RED simulations for the single flow case

RB	*maxp*	*minth* (SDUs)	Goodput (kbit/s)	Delay (s)
384	0.05	25	295.1 ± 5.9	0.62 ± 0.02
256	0.08	20	204.5 ± 3.6	0.75 ± 0.03
128	0.1	10	103.8 ± 1.3	0.88 ± 0.03
64	0.1	10	52.3 ± 0.9	1.57 ± 0.08

Table 5. Proposed RED configuration and its performance values in the single flow scenario

RB	*maxp*	*minth* (SDUs)	Goodput (kbit/s)	Delay (s)
384	0.05	20	292.7 ± 4.6	0.52 ± 0.02
256	0.06	15	203.3 ± 2.4	0.63 ± 0.02
128	0.2	10	100.9 ± 1.8	0.70 ± 0.03
64	0.3	10	50.0 ± 1.9	1.32 ± 0.14

Table 6. Performance of a drop-tail buffer and the relative improvement of SBD and RED

RB	Flows	Goodput (kbit/s)	SBD	RED	Delay (s)	SBD	RED
384	1	277.1 ± 6.1	+5%	+6%	0.72 ± 0.04	–34%	–27%
	4	298.6 ± 4.8	+3%	+3%	0.89 ± 0.01	–21%	–21%
256	1	185.1 ± 4.5	+8%	+10%	0.94 ± 0.07	–43%	–32%
	4	201.9 ± 3.1	+4%	+2%	1.28 ± 0.02	–25%	–19%
128	1	89.9 ± 5.8	+14%	+12%	1.58 ± 0.14	–54%	–55%
	4	93.4 ± 2.6	+8%	+7%	2.44 ± 0.06	–35%	–39%
64	1	42.1 ± 1.4	+20%	+19%	3.12 ± 0.21	–64%	–58%
	4	48.5 ± 1.6	+8%	+8%	4.33 ± 0.09	–23%	–40%

Finally, Table 6 shows the performance of a drop-tail buffer and the improvement achieved by the proposed SBD and RED configurations. While both schemes are similar in terms of goodput improvement, SBD tends to achieve greater delay reductions than RED, especially for the single flow case. Besides, SBD configuration is somewhat easier than RED because the delay is directly reduced increasing T_r (could be tied to application requirements), while *minth* acts as a "security" limit, assuring that the goodput does not fall below certain value.

6 Operation in Rate-Varying 3G Links

In this section we show how an RLC buffer with automatic AQM reconfiguration reacts to a sudden change in the RB rate. The chosen example consists of two TCP flows served by a 3G link that starts at 384 kbit/s and switches to 128 kbit/s after 50 seconds. The AQM parameter configuration is automatically changed when the RRC modifies the RB rate. Fig. 14 shows a trace for the SBD algorithm, using the parameter setting proposed in Table 3. When the bandwidth reduction takes place, the over-buffering is successfully avoided by means of reconfiguration.

Fig. 15 shows the same situation for an RED buffer. For each RB rate, RED is configured according to Table 5. The adaptive strategy is also effective for RED, although RED is less capable to avoid the overflow at the first stages of the connection, when the sources are in the *Slow Start* state. The increment of the buffer occupancy is too steep for RED to react. SBD is specially designed to react against fast and sustained *BO* increments; therefore the overflow is avoided even in *Slow Start*.

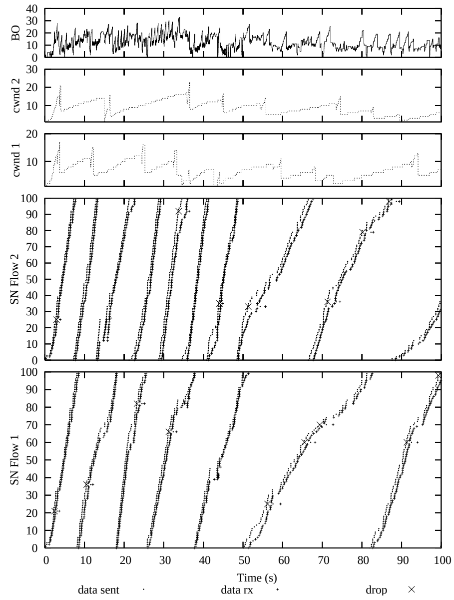

Fig. 14. Two TCP connections over RLC with automatic SBD reconfiguration

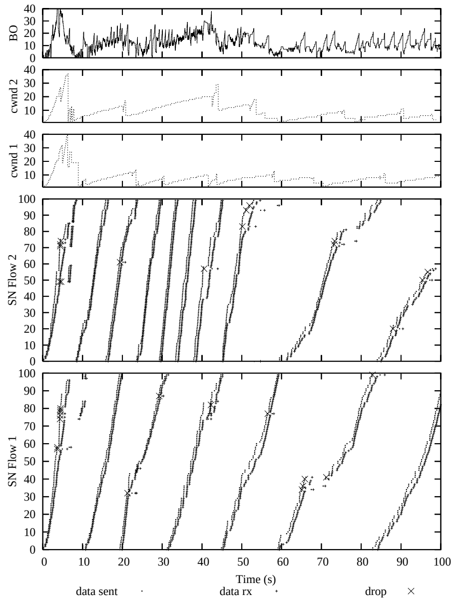

Fig. 15. Two TCP connections over RLC with automatic RED reconfiguration

7 Conclusions

This paper provides further insight into the effectiveness and configuration guidelines of AQM techniques on an RLC buffer. We focus on two AQM algorithms, RED and a novel deterministic scheme, SBD. By means of extensive simulation experiments we disclosed the effect of each parameter on TCP goodput and delay at different RB rates. Based on these results, we propose an optimum configuration for each algorithm aiming to reduce the delay while maintaining the goodput near the maximum. Compared to the current drop-tail specification of RLC, the delay performance is reduced about 45% and the goodput increases 12% for the single flow case.

SBD performs slightly better than RED in terms of delay reduction and its configuration is more straightforward. In addition, SBD is more effective than RED in avoiding buffer overflow in the first stages of the connection. This makes SBD a better choice than RED for applications requiring multiple downloads of short files, like Web surfing. Finally, SBD's deterministic operation makes it more feasible to implement on RLC, which operates in a per-user basis.

Finally, for each AQM scheme, we illustrate how a dynamic parameter reconfiguration is capable of maintaining an optimum performance, avoiding buffer overflow and excessive latency in situations of sudden changes in the RB bandwidth.

References

1. 3GPP TS 25.322, "Radio Link Control (RLC) protocol specification", v. 6.4.0., Jun. 2005.
2. M. Meyer, J. Sachs and M. Holzke, "Performance Evaluation of a TCP Proxy in WCDMA Networks", *IEEE Wireless Communication*, Oct. 2003, pp.70-79.
3. H. Inamura et al., "TCP over Second (2.5G) and Third (3G) Generation Wireless Networks" IETF RFC 3481, Feb. 2003.
4. R. Chakravorty, A. Clark and I. Pratt, "Optimizing Web Delivery over Wireless Links: Design, Implementation and Experiencies", *IEEE J. Select. Areas Commun.*, vol. 23, no. 2, Feb. 2005, pp. 402- 416.
5. M. Agfors, R. Ludwig, M. Meyer and J. Peisa, "Queue Management for TCP Traffic over 3G Links", in *Proc. IEEE WCNC 2003*, pp. 1663 -68.
6. J. J. Alcaraz, F. Cerdan and J. García-Haro, "Optimizing TCP and RLC Interaction in the UMTS Radio Access Network", *IEEE Network*, vol. 20, no. 2, Mar. 2006, pp. 56 – 64.
7. S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Networking*, vol. 1, pp. 397-413, Aug. 1993.
8. H. Holma, A. Toskala, *WCDMA for UMTS: Radio Access for Third Generation Mobile Communications, Third Edition*, Wiley, July 2004.
9. M. Rossi, L. Scaranari and M. Zorzi, "On the UMTS RLC Parameters Setting and their Impact on Higher Layers Performance", in *Proc. IEEE 57th VTC*, vol. 3, Oct. 2003, pp. 1827- 32.
10. R. Bestak, P. Godlewski and P. Martins, "RLC Buffer Occupancy when Using a TCP Connection over UMTS", in *Proc. IEEE PIMRC*, vol. 3. Sep. 2002, pp. 1161-65.
11. A. Gurtov, S. Floyd, "Modeling Wireless Links for Transport Protocols", *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, Apr. 2004, pp. 85-96.
12. A. Varga, "The OMNeT++ Discrete Event Simulation System", in *Proc. European Simulation Multiconference*. June 2001.
13. A. Chockalingam and M. Zorzi, "Wireless TCP Performance with Link Layer FEC/ARQ", in *Proc. IEEE ICC*, June1999, pp. 1212-16.