Using Buffer Management in 3G Radio Bearers to Enhance End-to-End TCP Performance

Juan J. Alcaraz, Fernando Cerdan
Department of Information Technologies and Communications, Polytechnic University of Cartagena, Plaza del Hospital, 1, 30202 Cartagena, Spain
{juan.alcaraz, fernando.cerdan}@upct.es

Abstract
TCP performance over 3G links can be enhanced by means of Active Queue Management (AQM) techniques at the downlink buffer of the mobile network link layer (RLC). In this paper we describe Slope Based Discard (SBD), a novel deterministic buffer management technique whose discarding policy is based on the observation of the filling rate of the buffer. By means of extensive simulation experiments we compare the performance of TCP over three different RLC buffer management schemes: drop-tail (no AQM), SBD and Random Early Detection (RED). The simulation scenarios are configured to evaluate how the end-to-end goodput and delay are influenced by the number of multiplexed flows, the TCP flavor, the buffer size and the wireless channel error rate.

1. Introduction
Third generation cellular networks (3G) provide Internet access at data rates of up to 384 Kbit/s for wide area coverage. The link layer protocol of the 3G radio access network, the Radio Link Control (RLC), can recover from frame losses in the wireless link by means of an Automatic Repeat reQuest (ARQ) algorithm, when configured in Acknowledged Mode (RLC AM) [1].

Given that TCP suffers excessive rate reductions in the presence of propagation errors [2], TCP benefits from the reliable radio bearer provided by RLC. However, several characteristics of 3G radio bearers like high and variable latency and buffer overflow of the downlink buffers [2, 3, 4], have undesired effects on TCP performance.

A new approach to enhance TCP – RLC interaction is the application of Active Queue Management (AQM) techniques at the downlink RLC buffers. Simulation experiments have shown that this solution improves TCP performance over 3G links [5, 6] with a small change at the RLC operation in the Radio Network Controller (RNC) nodes. One of the most extended AQM mechanisms in the Internet routers, Random Early Detection (RED) [7], can be adapted to the particularities of 3G links [6]. However [5] argues that the complex parameter configuration of RED in 3G links is a drawback that deterministic approaches may overcome.

In this paper we describe and evaluate a novel deterministic AQM algorithm called Slope Based Discard (SBD) especially suitable to the characteristics of 3G radio bearers. By means of simulation, SBD and RED are compared with the simple drop-tail scheme in several scenarios. The simulator implements in detail the protocols involved and is configured according to the recommendations given in [6, 8, 9]. Analyzing the results, we disclose the influence of TCP flavor, the buffer size, the number of concurrent TCP flows and the wireless error rate.

The rest of the paper is organized as follows. Section 2 describes the characteristics of 3G radio bearers concerning TCP performance. Section 3 explains the SBD algorithm in detail. Section 4 provides a brief description of the simulation environment. Section 5 compares the performance figures of RED, SBD and drop-tail schemes in several scenarios. The findings of this paper are summarized in section 6.

2. Characteristics of 3G links
Previous experimental [3, 10] and simulation [4, 6, 9] results provide a clear view of the characteristics of 3G wireless links. The behaviour of
the link buffer occupancy has shown a great impact on TCP performance.

3G links employ per-user buffering. A reliable RLC layer stores upper layer packets in a buffer until they are fully acknowledged by the receiver side. Therefore, as described in [3, 4, 5, 6] frame losses in the downlink channel result in higher buffer occupancy at the RLC network side. 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 a high reduction of the source rate.

The buffer should be large enough to avoid frequent overflow. However, excessive queueing cause some additional problems [3, 5] like Round Trip Time (RTT) inflation, unfairness between competing flows and viscous web surfing.

Figure 1 illustrates the effect of the buffer size on the performance of TCP over RLC without AQM and a wireless Frame Error Ratio (FER) set to 10%. The buffer size is given in RLC Service Data Units (SDU) of 1500 bytes. The Figure shows the end-to-end goodput (successfully received packets at the receiver) and delay (packet latency) for a 384 Kbit/s radio bearer multiplexing from 1 to 4 simultaneous TCP flows. TCP flavors tested are TCP Reno, New Reno and SACK. Table 1 shows the parameter values for the simulations performed. Further details about the simulator are given in Section 5.

From Figure 1 we conclude that larger buffer sizes benefit the goodput performance, especially when the radio bearer carries only one TCP flow. On the other hand, an oversized buffer causes higher latency values. These results also show the improvement achieved by the more advanced TCP versions, New Reno and SACK compared to TCP Reno, which is in agreement with the recommendations in [2].

### 4. Slope Based Discard Algorithm

Slope Based Discard (SBD) is a novel deterministic AQM algorithm that can be implemented at the downlink RLC buffer. SBD is based on the following ideas:

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 Figure 2). The rate reduction caused by this signal is perceived at the buffer after an additional propagation time, $T_f$, corresponding to the fixed (wired) network.

The discarding policy is driven by the buffer filling rate, $r$. When $r$ exceeds a critical value, $r_c$, a packet will be dropped. The buffer occupancy level determines the value of $r_c$.

The value of $r_c$ represents the filling rate that, if sustained, would fill the buffer entirely before the rate reduction (due to a packet discard) can be detected at the RLC buffer. In Figure 2 this reaction time equals $T_s + T_f$. Additional discards should be avoided until the end of this time.

![Figure 2. Diagram of the end-to-end connection](image-url)
The packet chosen for discard will be as close as possible to the front of the queue, thus the signalling time \( T_S \) does not include the queueing time, \( T_Q \). Additionally, the algorithm should not discard a packet whose transmission over the RLC link has already started, because it would trigger the signalling procedure required to synchronize RLC sender and receiver sides [1].

The following parameters control the SBD algorithm:

- \( \text{minth} \) is the occupancy level above which packets can be dropped.
- \( \text{maxth} \) is the maximum occupancy allowed in the buffer.
- \( T_R \) is the estimated reaction time.
- \( \alpha \) is a constant number of bits in the buffer. In order to measure the slope of the buffer occupancy process curve, \( r \), the algorithm controls the time used to reach the threshold (\( \text{ thn } \)) of an occupancy interval of \( \alpha \) bits. In contrast with a measure based on estimating the slope in fixed periods of time, this method provides an immediate reaction when \( r \) exceeds \( r_c \).

Figure 3 shows the RLC buffer occupancy (\( BO \)) curve to provide a graphical example of the algorithm.

When \( BO \) is above \( \text{minth} - \alpha \) the algorithm sets \( \text{ thn } = BO + \alpha \) and calculates with (1) the value of \( r_c \) corresponding to this \( \text{ thn } \).

\[
r_c = \frac{\text{maxth} - \text{ thn }}{T_R}
\]

The time \( C_n \) that it takes the buffer to store \( \alpha \) additional bits at \( r_c \) is \( C_n = \alpha / r_c \). A timer for \( C_n \) is started. If the timer expires and the buffer occupancy is below \( \text{ thn } \), then the actual filling rate is lower than \( r_c \), and no packet is dropped. On the other hand, if \( \text{ thn } \) is reached before the expiration of the timer, the current filling rate surpasses \( r_c \) and a packet is discarded.

In Figure 3, the dotted segments starting at each measuring interval represent \( BO \) when the buffer is filled at the critical rate (critical curve). In the intervals \( C_0, C_1 \) and \( C_2 \), no packet is discarded because \( BO \) is below the critical curve. In contrast, in the \( C_3 \) period, the filling rate is above \( r_c \). Hence, the threshold \( \text{ th3 } \) is reached before the timer \( C_3 \) expires. When \( \text{ th3 } \) is reached, a packet is dropped. The monitoring and discarding algorithm is deactivated for a period \( T_R \), avoiding consecutive packet discards. A description of the algorithm in pseudocode can be seen in Figure 4.

5. Simulation Environment

The simulation environment employed in this research was developed in OMNeT++ [11] and comprises a complete implementation of TCP and RLC protocols. Similar simulators were described in [4, 9]. The simulation scenario consists of one or several TCP sources connected to their respective receivers in a mobile equipment. Table 1 shows the parameters setting. The radio bearer has a round trip time of 100 ms and a bidirectional nominal rate of 384 kbit/s. The fixed network is modeled with a fixed rate of 1 Mb/s and a round trip delay of 200 ms [12].

The wireless channel generates error bursts according to the model in [13]. The normalized Doppler frequency (0.01) is the product of the Doppler frequency and the radio frame duration.

Uplink and downlink FERs are equal and fixed to 10% (a typical UMTS design value [12]), 15% or 20%.

The simulation results are obtained averaging 20 runs per sample. Each run is a 60 seconds download session. The radius of the confidence interval is less than 3% of the magnitude of the averaged sample, with a confidence degree of 90% (t-student distribution).

For each packet arrival
- update \( BO \)
  - if timer \( C_n \) on
    - drop packet
    - deactivate timer \( C_n \)
    - start timer \( T_R \)
  - if timer \( C_n \) off
    - if \( BO \geq \text{minth} - \alpha \)
      - calculate \( C_n \)
      - timer \( C_n \leftarrow C_n \)
      - \( \text{ thn } \leftarrow BO + \alpha \)
      - start timer \( C_n \)
- upon timer (\( C_n \) or \( T_R \) expiration)
  - if \( BO \geq \text{minth} - \alpha \)
    - calculate \( C_n \)
    - timer \( C_n \leftarrow C_n \)
    - \( \text{ thn } \leftarrow BO + \alpha \)
    - start timer \( C_n \)

Figure 3. \( BO \) of a buffer implementing SBD.

Figure 4. Pseudocode of SBD.
6. Performance Evaluation

Our performance simulation analysis compares two metrics, goodput and delay, for the simple drop-tail scheme (DT) and two AQM mechanisms: RED and SBD. The values considered for RED parameters are:

- Distance between maxth and minth (Δ): 10, 15, 20 and 25 SDUs. (maxth equals the buffer size)
- maxp: 0.01, 0.02, 0.05 and 0.08.

It should be mentioned that, as stated in [5, 6], given the relatively low rate of a radio bearer, an RED implementation adapted to the RLC should not average the buffer occupancy.

The values considered for SBD parameters are:

- maxth = 10 and minth = 5. (maxth = 10 means that the RED scheme is configured to (1).
- T_R: 100, 200 and 300 ms.
- R: 10, 15, 20 and 25 SDUs.

The value of R is set to the size of 5 SDUs (1500 bytes), which was found to be an optimum value and a compromise value between fast detection of congestion and excessive sensibility to BO oscillations.

The different scenarios combine the following values of the simulator parameters:

- RLC Buffer Size: 30, 40, 50, 60 and 70 SDUs.
- TCP flows multiplexed: 1, 2, 3 and 4 TCP flows.
- TCP flavors: Reno, New Reno and SACK.
- Frame Error Rate (FER) in the wireless channel: 10%, 15% and 20%

Figure 6 shows the average goodput and delay values of the three buffer management schemes for each TCP flavor with a buffer of 30 SDUs. The configuration of RED: maxp = 0.05 and Δ = 15 for TCP Reno. For New Reno and SACK, maxp = 0.01 and Δ = 10. Figure 7 correspond to a buffer size of 60 SDUs. The configuration of RED: maxp = 0.05, Δ = 15 for TCP Reno and New Reno. For SACK, maxp = 0.01, Δ = 10. The configuration of SBD is the same in both figures: T_R = 100 ms and Δ = 12 for TCP Reno and Δ = 6 for the other flavors.

Figure 8 shows the performance results for TCP Reno sources over radio bearers with different FER and an RLC buffer size of 40 SDUs. For FER = 10%, RED parameters are: maxp = 0.05, Δ = 10; SBD parameters: T_R = 100 ms, Δ = 12. For FER = 15% and 20%, RED parameters are: maxp = 0.05, Δ = 20; SBD parameters: T_R = 200 ms, Δ = 10.

In contrast with RED, SDB can be easily configured because the effect of its parameters on TCP performance is very predictable. In SBD, a lower minth tends to reduce the goodput and the delay. A higher T_R implies a more aggressive policy, because, according to (1), r_c is reduced, and therefore the delay decreases, but also does the goodput.

![Figure 6. TCP performance (30 SDUs buffer).](image1)

![Figure 7. TCP performance (60 SDUs buffer).](image2)
goodput improvement of AQM over drop-tail is higher (Figure 8(a) and Figure 7(a)). An early packet discard, in an AQM buffer, halves the window of the TCP connection. Obviously, in a single flow scenario this measure halves the overall user rate. In contrast, in a multiple flow scenario, this only affects to one connection. The goodput gain is higher because buffer overflow is avoided with less rate reduction.

3) Effect of the buffer size: With larger buffers, AQM is able to prevent overflow and to reduce the delay while keeping BO above TCP’s bandwidth delay product, thus providing better goodput to TCP.

4) Effect of the wireless error rate: AQM algorithms counteract the buffer occupancy increment caused by a higher frame loss ratio in the channel. Figure 8 shows better performance of AQM schemes compared to DT. In scenarios with higher FER, $T_S$ should be higher (as well as $maxp$ in RED) to react faster to changes in the buffer occupancy.

7. Conclusions

By means of simulation experiments we show that AQM techniques applied to RLC downlink buffers improve goodput and delay performance of TCP over 3G radio bearers. This improvement depends on several features. The buffer overflow reduction benefits especially TCP Reno sources. The delay is reduced in every scenario because AQM maintains the buffer occupancy around a certain level. AQM is more effective in large buffers and when the radio bearer multiplexes more than one TCP flow.

The main advantage of SBD compared to RED is its easier implementation and configuration. SBD is deterministic; hence it does not need the generation of random numbers required for the computation of discarding probabilities. This fact makes SBD more feasible for buffer management at RLC level, where the buffering is done in a per-user basis.