Traffic Phase Effects with RED and Constant Bit Rate UDP-Based Traffic*

Jörg Diederich[1], Thorsten Lohmar[2],
Martina Zitterbart[1], and Ralf Keller[2]

[1] Institute of Operating Systems and Computer Networks,
Technical University Braunschweig
{dieder | zit}@ibr.cs.tu-bs.de,
[2] Ericsson Research, Ericsson Eurolab Deutschland GmbH, Aachen
{Ralf.Keller | Thorsten.Lohmar}@ericsson.com

Abstract. One building block to provide service quality is queueing disciplines such as Random Early Detection (RED). Besides the topic of service differentiation (e.g., Differentiated Services), data flows *within* a service class are expected to receive the same service quality which is expressed in the service quality metric 'fairness'. This document evaluates traffic phase effects when using RED gateways in conjunction with UDP-based constant bit rate sources. These effects can lead to an unfair division of the available bandwidth among CBR data flows with the same bandwidth share. It is shown that the introduction of randomization may help to improve the fairness.

1 Introduction

The Internet Engineering Task Force has recommended active queue management algorithms such as Random Early Detection (RED) [3] for usage in IP gateways to achieve congestion avoidance while keeping a high utilization of the underlying link [1]. RED gateways control the average queue size by dropping (or marking) single packets "early", i.e., before the maximum queue size is reached. This way, traffic sources are informed about the raising level of congestion. This congestion avoidance method relies on the ability of the transport protocol to interpret a packet drop as a sign of congestion so that it reduces the transmission rate accordingly, such as TCP does. Further objectives of RED are the avoidance of global synchronization to keep a high link utilization and the avoidance of a bias against bursty traffic (cf., [7, 9] for an evaluation of these objectives). Additionally, RED gateways are designed to be able to maintain an upper bound

---

* This work has been supported partially by a grant from the German Ministry for Research and Education (BMB+F) within the project "Communication and Mobility by Cellular Advanced Radio" (COMCAR). Within this project, we are investigating high quality IP traffic to and from vehicles in the wide area. The work described here belongs to our studies on how to improve quality of service in a mobile wireless environment.

of the average queue size even when the transport protocol on the traffic source does not reduce the transmission rate on congestion notification. An example for such a congestion insensitive transport protocol is UDP.

Since the amount of UDP-based traffic in the Internet is assumed to grow in the future, the last mentioned objective of the RED design becomes more important. When an RED gateway is configured, it may not be known in advance if UDP or TCP traffic is dominating.

Furthermore, many users of the Internet are attached to the Internet via circuit-switched lines with a limited maximum bit rate (e.g., based on a modem). To accommodate these users, many coders exists for audio [4, 6] or video [5] which generate constant bit rate traffic.

This document evaluates the usage of RED gateways in conjunction with UDP-based constant bit rate (CBR) traffic flows. It focuses on fairness issues between UDP-only traffic.

2 Random Early Detection Gateways

2.1 Basic Operation

On packet arrival, RED gateways [3] must execute two different algorithms:

1. One algorithm for calculating the average queue size avg which is performed by using a low-pass filter with an exponential weighted moving average method, using w_q as the weight.
2. Another one for calculating the probability of dropping a packet "early", i.e., before the queue overflows. The probability depends on two factors:
   (a) The degree of congestion, represented by the value of avg. Early packet drop is performed only when avg is between min_{th} and max_{th}. When avg is above max_{th}, all incoming packets are dropped permanently.
   (b) A random factor which ranges linearly from 0 to max_p as avg ranges from min_{th} to max_{th}.

   Figure 1 depicts the value of the drop probability as a function of the average queue size.

The objective of the first algorithm is to detect the current degree of congestion while allowing a certain degree of burstiness which is determined by w_q. With the second algorithm, the RED gateway decides whether to drop the incoming packet or not. The random factor is introduced to avoid the global synchronization of congestion notification and to avoid a bias against bursty traffic.

2.2 RED and UDP-Based Traffic

It is assumed that in general UDP-based sources do not implement a congestion avoidance mechanism to reduce their transmission rate due to packet drops. Depending on the amount of UDP-based traffic arriving at the RED queue and the value of max_p, the average queue size cannot be kept within the min_{th},

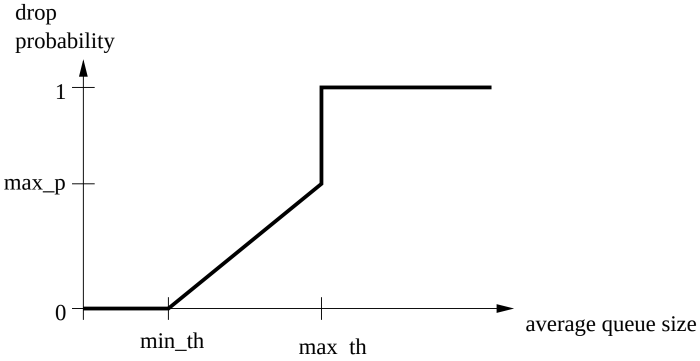


Fig. 1. The probability of early drop

max_{th} interval. For example, if a value of $max_p = 0.02$ is used (as recommended for congestion notification of TCP sources [3]), a constantly arriving amount of traffic being 102% of the bottleneck link capacity or above will keep the average queue size at a level of around max_{th}. This is due to the fact that only a maximum of 2% of the arriving packets is dropped early. If max_p is set to higher values, for example, 0.25, 0.5, or 0.75, an arriving amount of traffic being 133%, 200%, or 400% of the bottleneck link capacity or above will compensate the early drops. That means that in spite of the early drops, the amount of traffic trying to cross the bottleneck link is still larger than the link capacity. Only a value of $max_p = 1$ cannot be compensated by a higher amount of arriving traffic.

For all these cases, the influence of the early drop algorithm is limited since the RED queueing discipline changes periodically between the early-drop phase, where packet drops are based on a random factor, and the permanent-drop phase, in which randomization does not play any role. Therefore, effects of global synchronization may be reintroduced again which lead to an unfair division of bandwidth. The remainder of this paper addresses the issue of the fairness of RED in a scenario with UDP-based CBR sources.

3 Simulation Scenario

Simulations are performed with the network simulator (ns) [8]. Figure 2 depicts the simulation topology used for the following investigations.

50 nodes send CBR traffic via UDP over a single bottleneck link with an RED queuing discipline to 50 receiving nodes. The delay on all links does not have an influence on the simulation results since UDP has no congestion control mechanism which depends on the round-trip time and, thus, on the delay.

In [3] it is proposed to set the RED parameters max_{th} to at least twice the value of min_{th} and w_q to larger than 0.001. According to these rules, RED is configured with $min_{th} = 30$ packets, $max_{th} = 60$ packets, and $w_q = 0.002$.

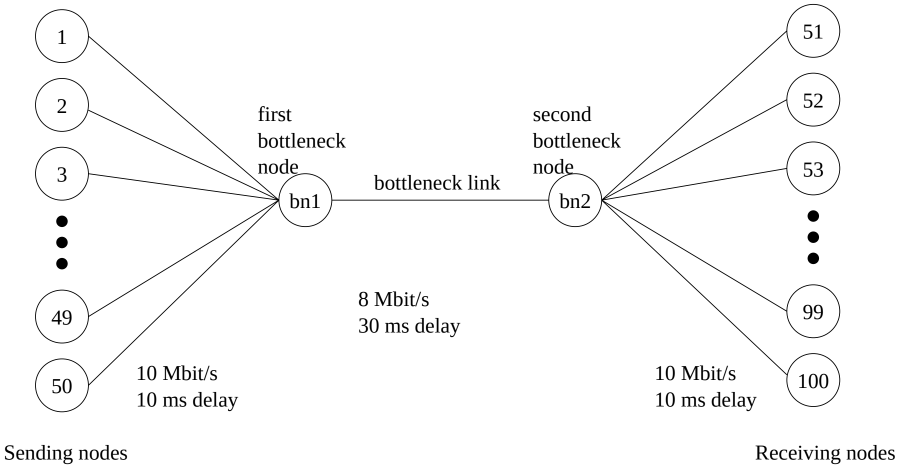


Fig. 2. The simulation topology

However, since there is no bursty traffic within the simulations, the setting of these three parameters is less important. max_p is set to 0.02 (a recommended value for TCP-based traffic sources [3]) so that the influence of the early drop algorithm is low at first. There is no recommended value for max_p in case of a congestion insensitive transport protocol such as UDP. The buffer size of the bottleneck link is 180 packets so that packet drops due to overflow of the packet queue never occur.

For all simulations, there are five different CBR traffic generators on each of the sending nodes which send UDP packets with 16, 20, 30, 50, and 100 kbps. There is a total amount of 10,800 kbps arriving at the bottleneck link corresponding to an overload of 135 %. The packet size is constant 125 bytes for each data flow.

Each simulation lasts 100 seconds. The traffic generators start sending in an arbitrary, but deterministic order (using a random number generator with the same seed for all simulations) within the first second of all simulations. The five generators on each node start simultaneously. The measurements start after 4 s of simulation time so that both, the actual queue size and the weighted average queue size of RED have already reached a stable state.

4 Fairness of RED with UDP-Based CBR Traffic

The traffic generators create data flows with a total bit rate of 10,800 kbps which have to compete for the available 8 Mbps on the bottleneck link. Intuitively, it is expected that each data flow gets a fair share of roughly about $\frac{8}{10,8} = 74.1\%$ of its source data rate. Table 1 shows the result of the first simulation run.

Table 1. Statistical simulation output for the different flows.

Data rate sent [kbps]	Mean of the received throughput [kbps]	Throughput share [%]	Standard deviation [kbps]	Minimum data rate [kbps]	Maximum data rate [kbps]
16	11.7	73.3	0.9	9.6	13.2
20	14.5	72.6	2.0	10.8	18.0
30	21.9	73.0	1.7	18.9	25.3
50	37.6	75.1	3.1	31.9	42.8
100	74.8	74.8	2.6	68.0	78.3

The arithmetic mean of the received throughput, measured at the traffic sinks, is close to the expected share of 74.1%. However, the throughput varies significantly between single data flows.

For example, for the data flows from the 16 kbps traffic generators, one particular data flow achieves a throughput of only 9.6 kbps while another data flow achieves 13.2 kbps, 37 % more than the former. This situation even does not change if the simulation time is extended to 1,000 seconds. The variation coefficient, which is the standard deviation σ divided into the arithmetic mean, is about 0.03–0.14 (3–14% of the mean) for this particular simulation, too high to be negligible. Further simulations with different randomly chosen start times for the traffic generators show an arithmetic mean of about 10% for the variation coefficient.

As a result, RED seems to be no longer fair when the average queue size oscillates around max_{th}.

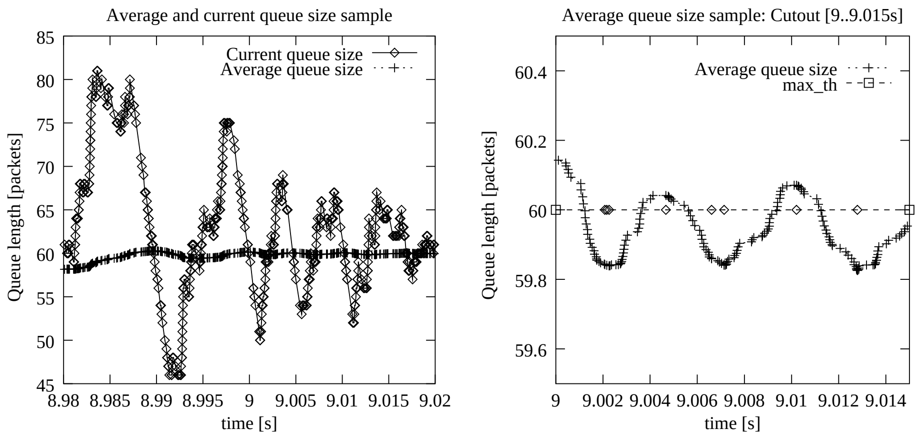
5 Traffic Phase Effects

Effects of traffic phases [2] have been one of the reasons to develop queueing disciplines based on random drop. Since the random factor of RED has a small influence only in case of $max_p = 0.02$ and UDP-only traffic (cf. Sect. 2.2), the temporal interaction of the CBR traffic generators with the average queue size is considered a candidate to be responsible for this effect. This section performs an analysis of this interaction.

5.1 Periodicity of the Average Queue Size

Figure 3a depicts a sample of the average queue size and the current queue size. The y-axis shows the current number of packets in the queue and the calculated average queue size. Figure 3b depicts a cutout from figure 3a showing the average queue size only.

The average queue size oscillates around max_{th} (being 60 packets in the simulations), as it can be seen from figure 3b. This is because packets have a certain probability to be queued if the average queue size is below max_{th} (the so-called ‘early-drop phase’). In this phase, the current queue size raises. As soon



3a 3b

Fig. 3. Sample of the average and current queue size

as the average queue size is above max_{th}, all incoming packets are dropped (the so-called ‘permanent-drop phase’) and the current queue size becomes smaller. Since the average queue size is calculated by a low-pass filter, it raises and falls much slower than the current queue size.

The analysis of the simulation output has shown that the early-drop phase lasts about 9.1 ms on average (with a standard deviation of 5.3 ms) and the permanent-drop phase about 3.2 ms (with a standard deviation of 1.7 ms). Thus, the total period of the average queue size oscillations is $9.1 + 3.2 = 12.3$ ms.

5.2 Synchronization of Periods

This section evaluates the interaction of the average queue size period with the periodicity of the CBR traffic generators. Table 2 shows the inter-packet time for the different CBR traffic generators.

Table 2. Inter-packet times for the different CBR sources

Data rate [kbps]	16	20	30	50	100
Inter-packet time [ms]	62.5	50.0	33.3	20.0	10.0

There is a good chance of a synchronization between the oscillation period of the average queue size and the periodicity of the CBR traffic generators. Therefore, a high variation coefficient is expected from the 16 kbps CBR sources, since their period is nearly five times as large as the average queue size oscillation period (5*12.3 ms=61.4 ms vs 62.5 ms). The same holds for the 20 kbps sources (4*12.3 ms = 49.2 ms vs. 50 ms) and for the 100 kbps sources (4*12.3 ms = 49.2 ms vs. 5*10 ms = 50 ms).

Table 3 presents the mean values of the variation coefficient for each class of CBR traffic sources averaged over 15 simulation runs. For the 20 kbps CBR sources, the mean of the variation coefficient is as high as expected, but for the other CBR sources it is either unexpected high or unexpected low. This might be because a perfect synchronization will not happen since the average queue size oscillations have a high variation coefficient of more than 50%.

Table 3. Variation coefficients for the different CBR sources

Data rate [kbps]	16	20	30	50	100
Variation coefficient [%]	7.3	12.2	9.1	15.3	11.8

Therefore, theoretical predictions about what data flows becomes synchronized seems to be complex. Instead, simulations should confirm that traffic phase effects are responsible for the unfairness between the data flows.

The Role of Start Times Synchronization effects should depend on the particular point of time when a CBR traffic generator starts sending packets. The following section analyzes this dependency.

In the following set of simulations, the start of a particular data flow of the 20 kbps data source is delayed from 1 to 50 ms in 1 ms steps. In order to have a high possibility to get a significant improvement in the achieved throughput, the data flow with the least throughput (10.8 kbps) of the first performed simulation is chosen to be delayed. Since the inter-packet time is four times the average queue size oscillation period, the achieved throughput of the particular data flow is expected to oscillated with a resulting period of roughly 12 ms, leading to an expected amount of four minima resp. maxima.

Figure 4 depicts the result of the simulations. The x-axis shows the delay of the starting time in milliseconds. The y-axis shows the achieved throughput for this particular data flow in each simulation.

As expected, the throughput improves significantly if the starting time of the data flow is delayed. However, an oscillation of the achieved throughput could not be seen as expected. Further simulations with different data flows to be delayed have shown not even a significant change in the achieved throughput.

Figure 5 depicts a sample of the average queue size of the simulation with no delayed start time and the simulation with a 1 ms delay. The x-axis shows the simulation time whereas the y-axis shows the average queue size in packets.

Although the achieved throughput of the particular data flow does not differ significantly (10,750 bps vs. 10,710 bps) for both simulations, the average queue sizes are completely different already after a simulation time of around 4 s.

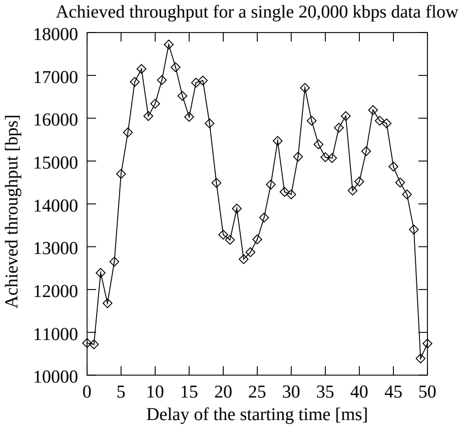


Fig. 4. Achieved throughput of a a single flow with delayed starting times

Thus, these simulation results cannot be used to show a dependency between the periods of the average queue size and the CBR traffic generators since already a delay of 1 ms changes the phase of the average queue size oscillations significantly.

However, since randomization has been shown to avoid phase effects, the next section performs further simulations with randomization included at different places.

6 Introducing Randomization

Two possible sources of the phase effects can be identified: synchronization of the five traffic sources on a single node (intra-node synchronization) or synchronization of the traffic sources on different nodes (inter-node synchronization). To examine the effect of the former, the first simulation was repeated with random starting times for each CBR traffic generator. As a result (cf. table 4), the variation coefficient is slightly lower (3–11%). Hence, the intra-node synchronization

Table 4. Statistical simulation output: Fully randomized start times

Data rate sent [kbps]	Mean of the received throughput [kbps]	Throughput share [%]	Standard deviation [kbps]	Minimum data rate [kbps]	Maximum data rate [kbps]
16	11.5	71.8	1.2	8.3	13.7
20	14.4	72.1	1.6	11.9	18.0
30	22.1	73.8	1.8	19.2	25.4
50	37.3	74.6	2.5	30.7	40.8
100	75.0	75.0	2.1	69.5	78.2

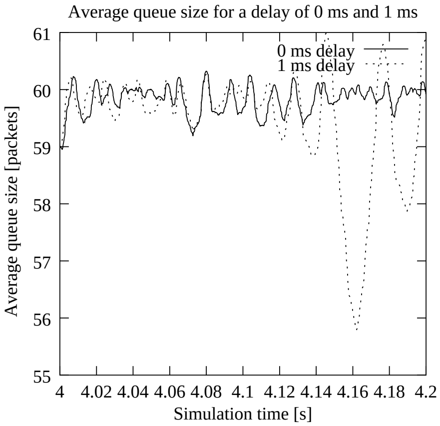


Fig. 5. Sample of the average queue size for the no delay and 1 ms delay simulation

has an influence on the traffic phase effects. However, since the variation coefficient remains to be high, the intra-node synchronization cannot be the only source of the traffic phase effects.

Thus, randomization was introduced at two further places (while intra-node synchronization was reintroduced again):

1. At the traffic sources
2. At the RED gateway

6.1 Randomization at the Traffic Sources

At first, the CBR sources were modified to send with a uniformly distributed inter-packet time (±10 ms). As a result, the variation coefficient falls below 5% of the mean value (cf., table 5) although intra-node synchronization was introduced again. Thus, the fairness of dividing the available bottleneck link capacity among the CBR flows is improved significantly. In case of a larger interval of the inter-

Table 5. Statistical output: Randomized inter-packet times at the CBR sources

Data rate sent [kbps]	Mean of the received throughput [kbps]	Throughput share [%]	Standard deviation [kbps]	Minimum data rate [kbps]	Maximum data rate [kbps]
16	11.8	73.5	0.3	11.2	12.5
20	14.6	73.2	0.4	13.7	15.6
30	21.9	73.1	1.1	20.2	24.1
50	37.3	74.5	1.7	33.9	40.7
100	74.6	74.6	1.6	70.9	77.6

packet time (±100 ms), the variation coefficient falls below 1.6%. However, it will not always be possible to change all CBR traffic sources in practice.

In a second step, the traffic generators were changed to be switched on/off dynamically during the simulations (so called "Exponential On/Off" traffic generators). The on-time for each traffic generator was chosen randomly from an exponential distributed random variable with a mean of 5, 25, 33, 50, 75, 100, and 125 seconds. During the on-time, the traffic generators send traffic with the same constant bit rate as in the previous simulations. The off-time was chosen randomly from an exponential distributed random variable as well with a mean of 650 milliseconds. Figure 6 shows five graphs of the variation coefficient for the five different classes of traffic.

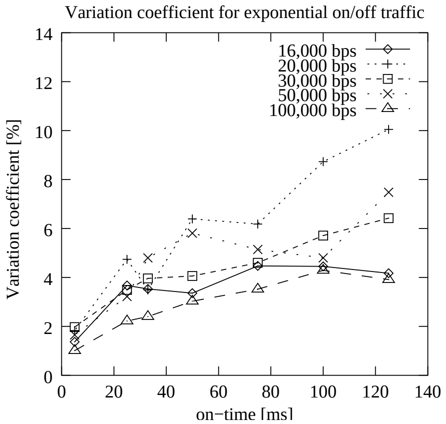


Fig. 6. Variation coefficient for the exponential on/off traffic generators

On the x-axis, the mean of the on-time is depicted. The y-axis shows the resulting variation coefficient for each of the five different classes of traffic. As a result, the variation coefficient raises with the length of the on-time. In case of the highest on-time, the variation coefficient is as high as in the CBR traffic generator scenario. Thus, the introduction of Exponential On/Off traffic generators led to the necessary introduction of randomization to avoid the traffic phase effects.

6.2 Randomization at the RED Gateway

One possibility to introduce randomization at the RED gateway is not to drop the arriving packet, but to drop a randomly chosen packet from the queue. This way, the standard deviation can be reduced, but only slightly below 10% of the mean value (cf. table 6). This randomization leads to a small improvement of the fairness only. The introduction of randomization at the traffic sources performs much better. Dropping an arbitrary packet from the packet queue is

Table 6. Statistical simulation output for randomized drop.

Data rate sent [kbps]	Mean of the received throughput [kbps]	Throughput share [%]	Standard deviation [kbps]	Minimum data rate [kbps]	Maximum data rate [kbps]
16	11.8	73.8	0.5	10.8	12.7
20	14.4	72.0	1.4	12.0	16.9
30	22.0	73.3	1.1	20.0	24.6
50	37.1	74.2	1.7	34.3	40.1
100	74.7	74.7	0.9	72.5	76.2

easier compared to changing all CBR traffic sources as only changes at a single place in the network are necessary. However, dropping a packet from the middle of the queue is difficult to implement efficiently. Therefore, an efficient solution to introduce randomization, which is also easy to implement, is still missing.

Another possibility to introduce randomization at the RED gateway is to increase max_p so that there is a higher influence of the early-drop phase (cf. sect. 2.2). Figure 7 depicts how the variation coefficient changes for each class of data flows when max_p is varied.

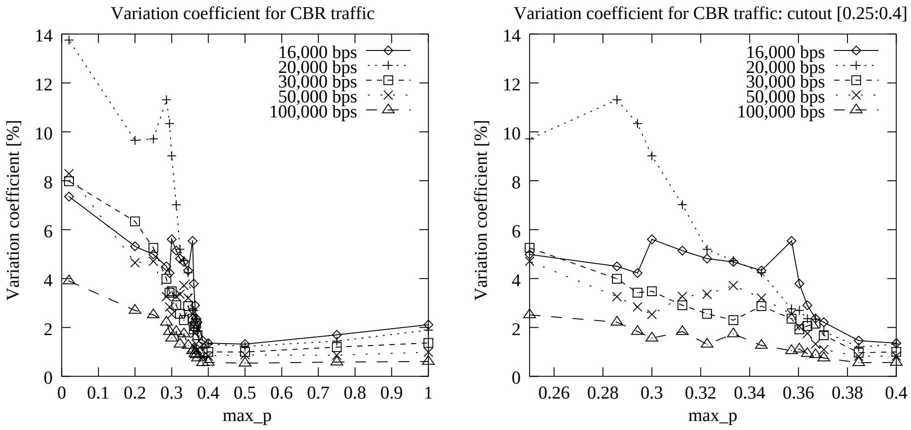


Fig. 7. Variation coefficient for simulations with different max_p values

As long as max_p is small enough (below 0.25) so that the early drops cannot prevent the link from being overloaded, the variation coefficient remains high. If max_p becomes sufficiently large (above 0.36) so that the load of the link falls below 100%, the traffic phase effects disappear. In between both values, the variation coefficient quickly becomes smaller. Thus, a sufficient amount of randomization to improve the fairness situation may be introduced by setting max_p to a sufficiently high value, depending on the load on the bottleneck link.

In summary, this section has led to the strong presumption that the unfairness of RED with regard to CBR data flows is caused by traffic phase effects. Randomization at the traffic sources (either for CBR or for Exponential On/Off traffic generators) does improve the fairness as well as tuning the max_p parameter at the RED gateways.

7 Summary

In this paper, the problems of UDP-based CBR traffic sources in conjunction with RED gateways have been analyzed briefly. Traffic phase effects can lead to an unfair division of bandwidth among CBR data flows. The introduction of randomization at either the traffic sources or at the RED gateway improves the fairness either when the traffic sources do not send strictly CBR traffic anymore or when it is ensured that randomization at the RED gateway is in place.

Further work will be done evaluating the Generalized RED queueing discipline in our simulation to enable the usage of service differentiation by drop priorities. Additionally, the traffic model will be extended to variable bit rate traffic, e.g., as produced by video coders.

References

[1] B. Braden, D. Clark, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclawski, and L. Zhang. Recommendations on queue management and congestion avoidance in the Internet. Request for Comments (Informational) 2309, Internet Engineering Task Force, Apr. 1998.

[2] S. Floyd and V. Jacobson. On traffic phase effects in packet-switched gateways. *Internetworking: Research and Experience*, 3(3):115–156, Sept. 1992.

[3] S. Floyd and V. Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM trans. on networking*, 1(4):397–413, Aug. 1993.

[4] International Telecommunication Union. Pulse code modulation (PCM) of voice frequencies. Recommendation G.711, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, 1988.

[5] International Telecommunication Union. Video codec for audiovisual services at p x 64 kbit/s. Recommendation H.261, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, Mar. 1993.

[6] International Telecommunication Union. Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s. Recommendation G.723.1, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, Mar. 1996.

[7] M. May, T. Bonald, and J.-C. Bolot. Analytic evaluation of RED performance. In *Proc. of INFOCOM 2000*, Tel-Aviv, Israel, Mar. 2000.

[8] The UCB/LBNL/VINT network simulator - ns (version 2). UCB Multicast Network Research Group, University of California at Berkeley. URL: http://www-mash.cs.berkeley.edu/ns/.

[9] H. Wang and K. G. Shin. Refined design of random early detection gateways. In *Proc. of GLOBECOM 1999*, Rio de Janeiro, Brasil, Dec. 1999.