Study of QoS and Traffic Control Mechanisms in IP Networks

Filipe Oliveira, João Rua, Miguel Zenha
Computer Science Department
Universidade do Minho
Email: a57816@alunos.uminho.pt, a41841@alunos.uminho.pt, a66551@alunos.uminho.pt,

***Abstract*—In traditional networks, all connections and services get the same treatment. However, since network resources are limited, and the overall Internet only offers a "Best-Effort" approach, it is important to differentiate between connection classes, and to be able to treat them accordingly to standardised and well documented parameters.**

This exploratory essay focus on developing a comparative study of traffic control mechanisms in IP networks and corresponding parametrisation, using the Network Simulator NS-2. In order to do so, a test platform will be presented and several Diffserv parameters will be discussed.

1. Network topology to be used

The network topology to be used as test platform is illustrated in figure 1. The network topology includes six clients (from Cli1 to Cli6), two edge routers (E1 and E2), and a core router (C0). The clients' access links have a capacity of 5Mbps and a delay of 5ms, and the core network links have a capacity of 5Mbps and a delay of 10ms.

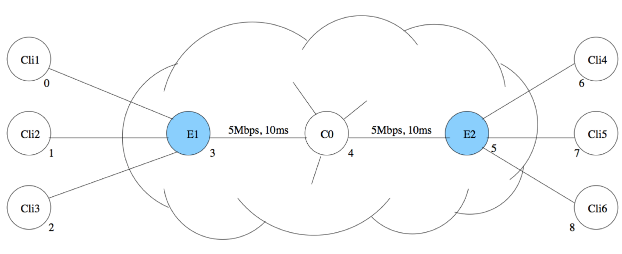

Figure 1. ISP network topology

The topology is deliberately symmetric to simplify traffic analysis. During this exploratory essay several changes will be made regarding the services/applications that every Client holds, however, the topology remains unchanged.

In most of the cases, it will be enough to analyse flow in one way, however, in the last analyse on chapter **??**, flows in both ways will be analysed due to the bigger complexity of the simulation.

As the topology evidences, if all clients use the link capacity simultaneously then congestion will occur in the network backbone, and the service provider will not be able to guarantee proper traffic delivery. To minimise or solve this effect, several traffic control mechanisms will be used in order to promote quality of service (QoS) in the domain.

Simulations for all scenarios were 15 seconds long. This was a very short simulation time, but enabled us to achieve a confidence interval, producing a stable final state.

2. Applications/Services to be used

- **CBR over UDP** - generates Constant Bit Rate (CBR) traffic over UDP. This may correspond to the transmission of audio or video traffic at a regular/periodic rate.
  - Parameters: rate (bits/sec) e packet size (Bytes);
- **FTP** - transfer of large files over TCP;
- **Voice over UDP** - simulates a voice call over UDP; This traffic is characterised by having a constant rate, alternating between talk and silence time periods.
  - Parameters: rate (bits/sec) and burst size (in seconds).

3. Tools and evaluation metrics

In order to infer the network quality of service we will take in consideration the following parameters Metrics to use in the simulations:

- **Loss rate** (total and per flow), in packets/sec.
- **Bandwidth** in use (total and per flow), in bits/sec.

4. A - Simulating the "Best-Effort" scenario

By default, routers handle packets based on a simple FIFO queueing system, trying to forward them in the best possible way according to the available resources (memory and CPU).

This well-known model is called Best-Effort as there are QoS guarantees on packet delivery (in terms of bounded delays, loss and/or bandwidth utilisation).

For a first approach we considered similar clients with CBR applications, generating each one a rate of 3Mbps, for a total of six flows ($0 \rightarrow 8$, $1 \rightarrow 7$, $2 \rightarrow 6$, $8 \rightarrow 0$, $7 \rightarrow 1$, $6 \rightarrow 2$).

4.1. Identification of the links under congestion

Producing the graphs illustrating the levels of loss and bandwidth utilisation along the time, we can infer that the core network links ($3 \leftrightarrow 4$, $4 \leftrightarrow 5$), since the full bandwidth is being used, as stated in figures 2 and 3.

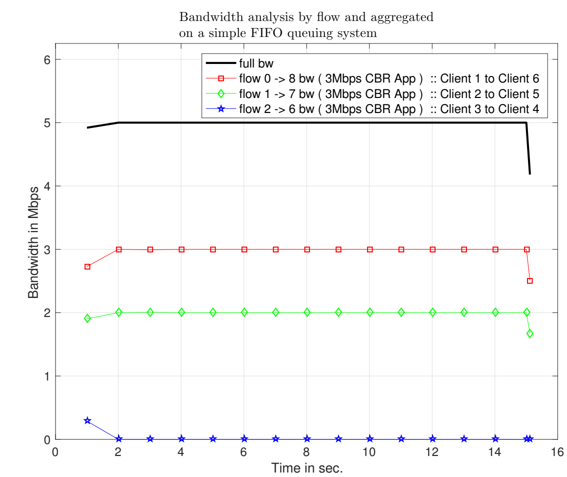

Figure 2. Bandwidth analysis by flow and aggregated on a simple FIFO queueing system, simulation a "best effort" scenario

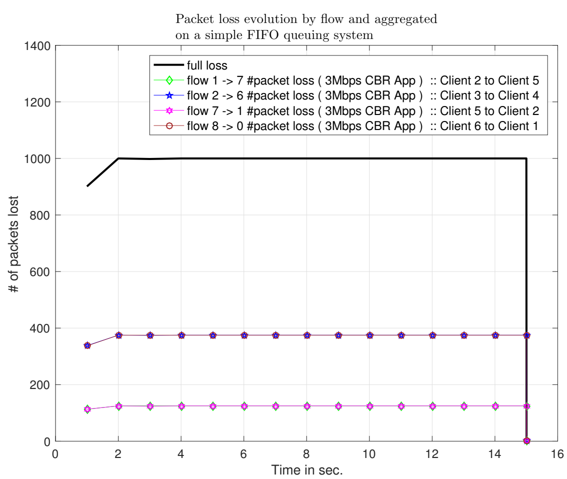

Figure 3. Packet loss evolution by flow and aggregated on a simple FIFO queueing system, simulation a "best effort" scenario

As stated before, the Internet's "best-effort" scenario produces an undesired non-equitable bandwidth distribution. Denote that this simple simulation only deals with one type of service simulation. The inclusion of other, "more sensible" to network congestion, services like for example VOIP, would result in an unacceptable QoS.

Changing the queues associated with the links under congestion from DropTail to RED would theoretically result into in a better service. The corresponding results are shown in figures 4 and 5.

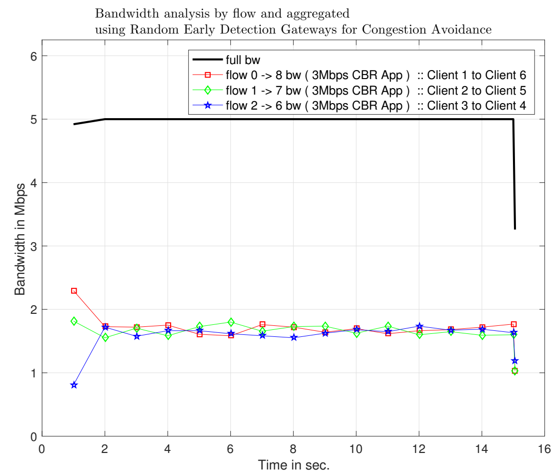

Figure 4. Bandwidth analysis by flow and aggregated on a simple FIFO queueing system, simulation a "best effort" scenario, using Random Early Detection Gateways for Congestion Avoidance

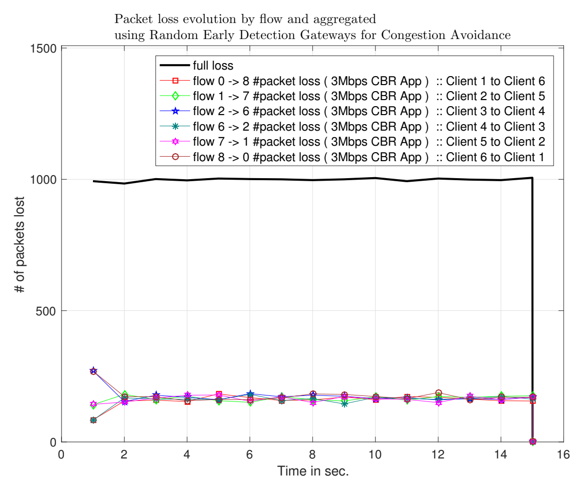

Figure 5. Packet loss evolution by flow and aggregated on a simple FIFO queueing system, simulation a "best effort" scenario, using Random Early Detection Gateways for Congestion Avoidance

Notice that this "solution" only improves the equitable bandwidth distribution across flow because they all are produced with the same service/traffic type. If we included for exemplo some TCP over IP service, since it behaves in order to prevent/diminish congestion, it would suffer more from bandwidth "starvation" than any service using UDP over IP.

In figures 6 and 7, simulated results are show if one client would generate more CBR traffic than the others, on a simple FIFO queuing system, simulating a "best effort" scenario.

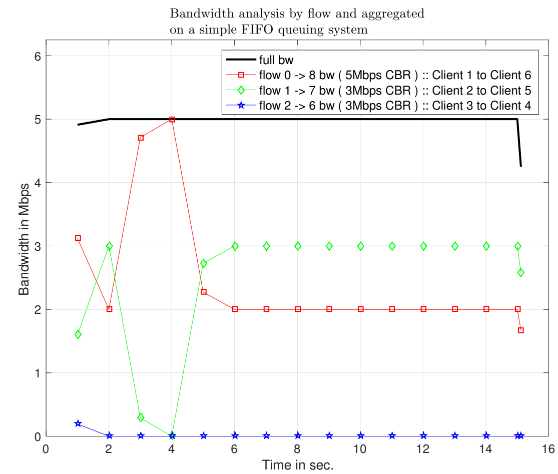

Figure 6. Bandwidth analysis by flow and aggregated on a simple FIFO queueing system, simulation a "best effort" scenario, in which one client would generate more CBR traffic than the others.

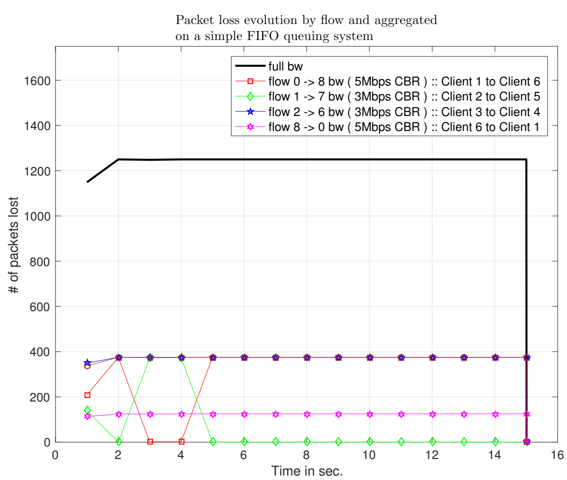

Figure 7. Packet loss evolution by flow and aggregated on a simple FIFO queueing system, simulation a "best effort" scenario, in which one client would generate more CBR traffic than the others.

5. B - Simulating a multi-service network in the "Best-Effort" scenario

In a more realistic scenario, it would be expectable to have both UDP and TCP traffic with other characteristics (FTP, HTTP, etc.). Using the procedures already included in the simulation script, several changes were made in order to obtain the following scenario:

- **a CBR application** sending 4Mbps from client 1 to client 6, and other from client 6 to client 1;
- **a FTP connection** from client 2 to client 5, and other from client 4 to client 2;
- **a voice connection over UDP** from client 3 to client 4, and vice-versa. Since VOIP Bandwidth consumption naturally depends on the codec used, we selected G.711 - 64 Kbps Bitrate and 87.2 Kbps Nominal Ethernet Bandwidth, and simulated a maximum of 30 calls at any given simulation time. The presented graphic results for VOIP are a aggregation of all the 30calls.

The corresponding results are shown in figures 8 and 9.

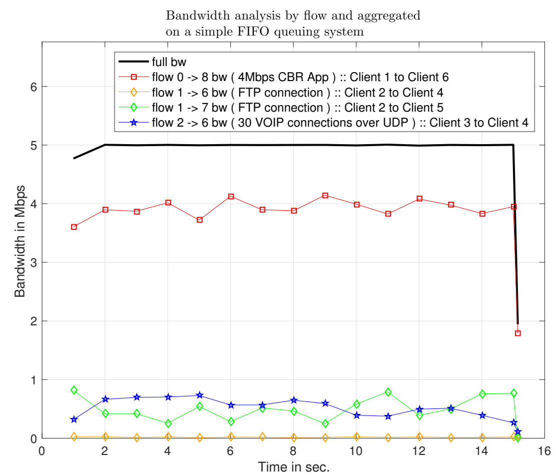

Figure 8. Bandwidth analysis by flow and aggregated on a simple FIFO queueing system, simulating a multi-service network in a "best effort" scenario.

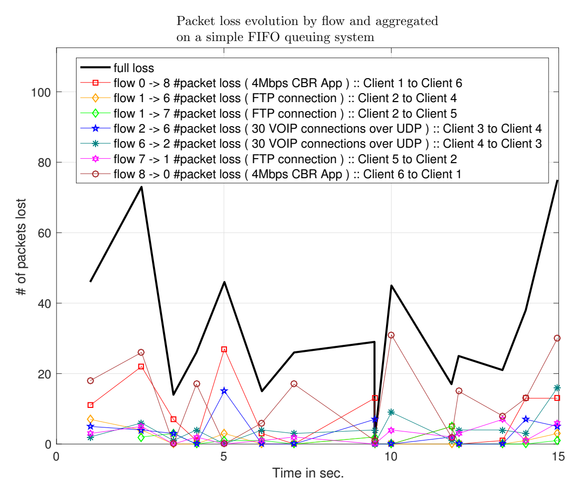

Figure 9. Packet loss evolution by flow and aggregated on a simple FIFO queueing system, simulating a multi-service network in a "best effort" scenario.

As you can state in figure 9, services like VOIP connections over UDP and FTP connections suffer the most when the network is fully congested, being the flows 1 $\rightarrow$ 7 (FTP connection), 2 $\rightarrow$ 6 (30 VOIP connections over UDP), and 6 $\rightarrow$ 2 (30 VOIP connections over UDP), the ones that are most affected.

The relation between flow, total number of packets lost, and percentage of loss/sent packages, is presented in table

1, and lets us fully understand the harm of treating all traffic with the same priority.

TABLE 1. RELATION BETWEEN FLOW, TOTAL NUMBER OF PACKETS LOST, TOTAL NUMBER OF PACKETS SENT, AND PERCENTAGE OF LOSS/SENT PACKAGES, ON A SIMPLE FIFO QUEUEING SYSTEM, SIMULATING A MULTI-SERVICE NETWORK IN A "BEST EFFORT" SCENARIO.

Flow	#packets loss	#packets received	% loss/received
$0 \rightarrow 8$ (4 Mbps CBR App)	116	29652	0.3912 %
$1 \rightarrow 6$ (FTP connection)	20	3720	0.5376 %
$1 \rightarrow 7$ (FTP connection)	15	3743	0.4007 %
$2 \rightarrow 6$ (30 VOIP connections over UDP)	48	4016	1.1952 %
$6 \rightarrow 2$ (30 VOIP connections over UDP)	61	4349	1.4026 %
$7 \rightarrow 1$ (FTP connection)	36	3620	0.9945 %
$8 \rightarrow 0$ (4 Mbps CBR App)	185	29445	0.6283 %

Please denote that despite the loss percentage doesn't seem to high for all flows, those values are presented as a mean value, giving the possibility of loss increase in certain time intervals, and decrease in others. We should therefore analyse the percentage of loss per flow by connection time. The corresponding results are shown in figure 10.

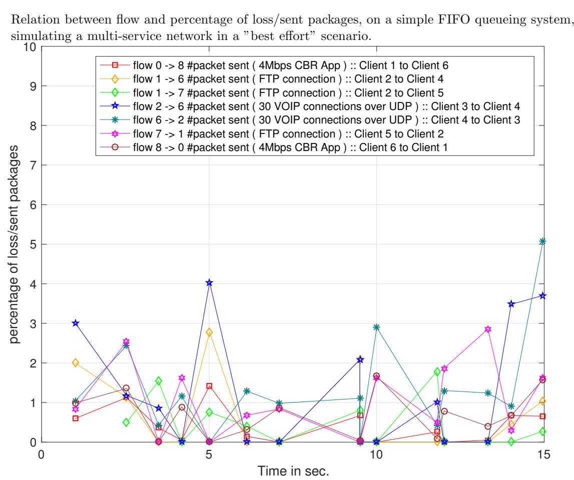

Figure 10. Packet loss/sent percentage by flow evolution, on a simple FIFO queueing system, simulating a multi-service network in a "best effort" scenario.

As stated before, it is in flows $2 \rightarrow 6$ (30 VOIP connections over UDP) and $6 \rightarrow 2$ (30 VOIP connections over UDP) that we observe a bigger loss percentage over time (5%). The service that should be prioritized and treated as the most volatile to delays is the one suffering the most from congestion.

6. C.1 Identify the links under congestion

6.1. E1 - C0 (Edge to Core Configuration)

- **The number of existing queues and the traffic scheduler in use**
  1 physical queue, implementing 2 virtual queues;

- **Policy Entry**
  - **Client1 $\rightarrow$ Client6** – TokenBucket:
    - **Committed Information Rate:** 2 Mbit-s/sec;
    - **Committed Burst Size:** 5 KBytes;
    - **Policer Table** has initial (green) code point 10, and downgraded (yellow) code point 11;
  - **Every remaining initial and end station** – Dumb:
    - **Policer Table** has always downgraded (yellow) code point 11;

- **The queueing discipline in use and the configuration of each queue:**
  Round Robin scheduling and RIO-C Active Queue Management:
  - queue 0:
    - **minimum threshold**: 20 Packets;
    - **maximum threshold**: 40 Packets;
    - **maximum dropping probability**: $2 * 10^{-2}$;
  - queue 1:
    - **minimum threshold**: 10 Packets;
    - **maximum threshold**: 20 Packets;
    - **maximum dropping probability**: $1 * 10^{-1}$;

- **the amount of memory allocated to the queues:**
  Default queue buffer size is 20 packets (Packet size 1 KB) : 20KB per queue;

- **the queues which handle data flows:**
  Code point 10 mapped to physical queue 0 and virtual queue 0, Code point 11 mapped to physical queue 0 and virtual queue 1;

6.2. C0 - E2 (Core to Edge Configuration)

- **The number of existing queues and the traffic scheduler in use**
  1 physical queue, implementing 2 virtual queues;

- **The queueing discipline in use and the configuration of each queue:**
  Round Robin scheduling and RIO-C Active Queue Management:
  - queue 0:

    * **minimum threshold**: 20 Packets;
    * **maximum threshold**: 40 Packets;
    * **maximum dropping probability**: $2 * 10^{-2}$;
  - queue 1:
    * **minimum threshold**: 10 Packets;
    * **maximum threshold**: 20 Packets;
    * **maximum dropping probability**: $1 * 10^{-1}$;

- **the amount of memory allocated to the queues:**
  Default queue buffer size is 20 packets (Packet size 1 KB) : 20KB per queue;

- **the queues which handle data flows:**
  Code point 10 mapped to physical queue 0 and virtual queue 0, Code point 11 mapped to physical queue 0 and virtual queue 1;