

TP3: Estudo de QoS e Mecanismos de Controlo de Tráfego em redes IP

Luís Magalhães¹², Luís Sousa¹³, and Hugo Marques¹⁴

¹ University of Minho, Braga, Portugal

² pg47415@alunos.uminho.pt

³ a89597@alunos.uminho.pt

⁴ pg47848@alunos.uminho.pt

Abstract. O presente ensaio exploratório visa estudo e comparar de mecanismos de controlo de tráfego em redes IP e parametrização correspondente, utilizando o Network Simulator NS-2.

Keywords: NS2 · DropTail · FTP · HTTP · Loss rate · Bandwidth

A Responda às seguintes perguntas:

A.1 Identifique os links em congestão.

Após a execução da simulação pudemos verificar que os links em congestão são todos os links de backbone da rede, mais concretamente os links 3-4 e 4-5.

A.2 Usando os scripts *loss.awk* e *bw.awk*, obtenha os gráficos que ilustram os níveis de perda e utilização da largura de banda ao longo do tempo.

Agora, altere as filas associadas aos links em congestionamento de DropTail para RED e obtenha os resultados e gráficos correspondentes. Comente os resultados.

Analisando os gráficos seguintes, obtidos através da execução dos scripts *loss.awk* e *bw.awk*, podemos observar que, quando utilizamos Droptail nas filas existe uma grande variação entre a largura de banda e perda dos diferentes tráfegos. Já ao utilizar RED, os valores encontram-se mais homogéneos, pelo que podemos concluir que o RED possui uma distribuição mais justa de encaminhamento de pacotes.

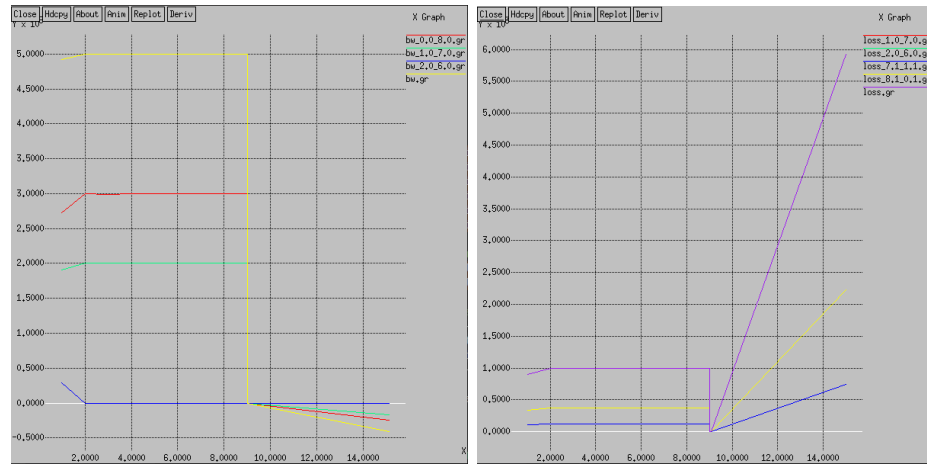


Fig. 1. Largura de banda e perda com DropTail

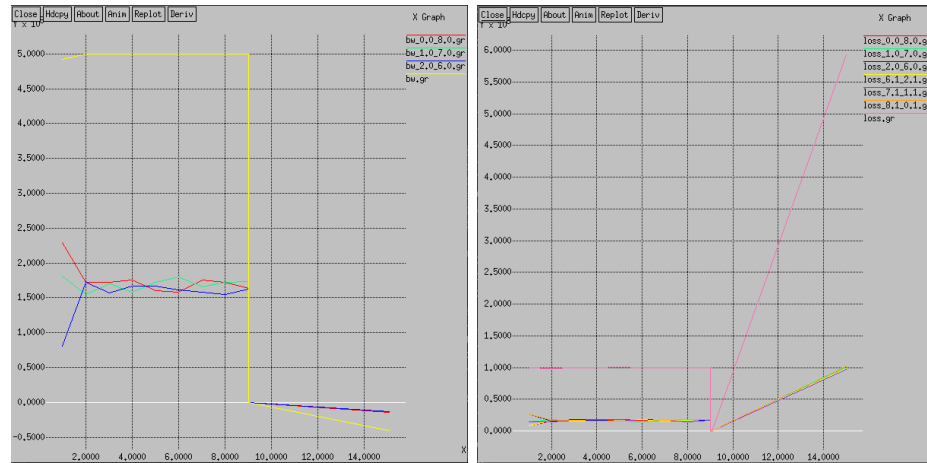


Fig. 2. Largura de banda e perda com RED

B Obtenha os resultados e gráficos que podem ser úteis para análise, a fim de responder às seguintes perguntas:

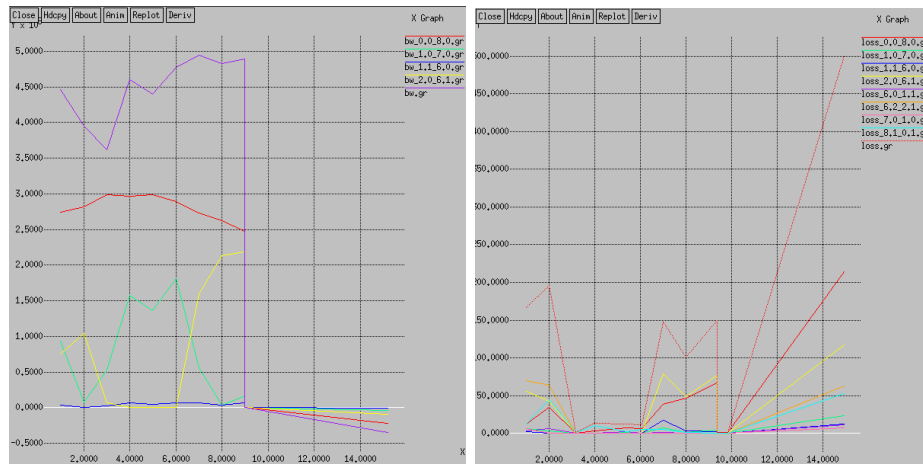


Fig. 3. Largura de banda e perda com tráfego real

B.1 Identifique quais são as aplicações são mais afetadas pelo congestionamento? e porquê?

A mais afetada foi a conexão FTP, visto se tratar de uma aplicação que lida com transferências de arquivos grandes por TCP.

C Responda às seguintes questões:

- C.1** Com base no código do *script* e no manual, e para o link E1 - C0 (ou C0 - E2), identifique:
- o número de filas existentes e o agendador de tráfego em uso;
 - a disciplina de filas em uso e a configuração de cada fila;
 - a quantidade de memória alocada para as filas;
 - as filas que manipulam fluxos de dados.

Existe uma fila física e duas filas virtuais. O agendador de tráfego utilizado é o "default", mais concretamente o Round Robin.

É usada como disciplina o FIFO e cada fila virtual é configurada através do comando *configQ*, usando 5 parâmetros: os 2 primeiros parâmetros indicam que a fila física é 0/1 e a fila virtual é 0/1. Os 2 parâmetros de seguida referem que essas filas possuem um valor mínimo de *threshold*, i.e, um valor mínimo do tamanho médio das filas de espera em pacotes, assim como um valor máximo, sendo que o

valor médio tem de ser inferior ao valor máximo, para não se perder a totalidade dos pacotes. O último parâmetro indica um valor máximo de *pb* (*parâmetro de configuração do algoritmo de implementação da fila*) e que representa uma determinada probabilidade.

C.2 Considerando os resultados/estatísticas da simulação, identifique a fila que sofre maior perda de pacotes. Tente justificar esse comportamento.

A fila que sofre maior perda de pacotes, é a fila de espera de E1 para C0. Como esta fila recebe pacotes de 3 fontes diferentes com taxas elevadas de uso de largura de banda, o link entre E1 e C0 não é capaz de encaminhar os pacotes com velocidade suficiente, pelo que a fila de espera tem tendência a encher, sendo obrigada a perder pacotes.

<p>Estadísticas da Fila de Espera de E1 para C0</p> <pre> Packets Statistics ===== CP TotPkts TxPkts ldrops edrops -- - All 5619 3136 2897 386 10 1252 1252 0 0 11 4367 1884 2097 386 </pre> <p>Estadísticas da Fila de Espera de C0 para E2</p> <pre> Packets Statistics ===== CP TotPkts TxPkts ldrops edrops -- - All 3111 3111 0 0 10 1242 1242 0 0 11 1869 1869 0 0 </pre>	<p>Estadísticas da Fila de Espera de E1 para C0</p> <pre> Packets Statistics ===== CP TotPkts TxPkts ldrops edrops -- - All 11244 6261 4197 786 10 2582 2582 0 0 11 8742 3759 4197 786 </pre> <p>Estadísticas da Fila de Espera de C0 para E2</p> <pre> Packets Statistics ===== CP TotPkts TxPkts ldrops edrops -- - All 6236 6236 0 0 10 2492 2492 0 0 11 3744 3744 0 0 </pre>	<p>Estadísticas da Fila de Espera de E1 para C0</p> <pre> Packets Statistics ===== CP TotPkts TxPkts ldrops edrops -- - All 16869 9386 6298 1185 10 3752 3752 0 0 11 13117 5634 6298 1185 </pre> <p>Estadísticas da Fila de Espera de C0 para E2</p> <pre> Packets Statistics ===== CP TotPkts TxPkts ldrops edrops -- - All 9361 9361 0 0 10 3742 3742 0 0 11 5619 5619 0 0 </pre>
---	--	--

Fig. 4. Estatísticas das filas de espera

C.3 Gere os gráficos que reflectam a perda de pacotes e a utilização da largura de banda ao longo do tempo. Comente os resultados quando comparado ao cenário de melhor esforço e explique as diferenças.

No cenário Best Effort, os pacotes são descartados indiferentemente, não existindo garantia quanto à taxa de perda de pacotes e largura de banda. No DiffServ a largura de banda é garantida e se os pacotes forem marcados como conformes, estes são entregues com elevada probabilidade. Comparando os dois cenários verifica-se que a perda de pacotes é bastante constante para os dois e no cenário DiffServ há mais estabilização na Bandwith.

A principal diferença a notar é uma redução na largura de banda da ligação entre o cliente 0 e o cliente 8, a qual deriva da utilização de uma política de TokenBucket.

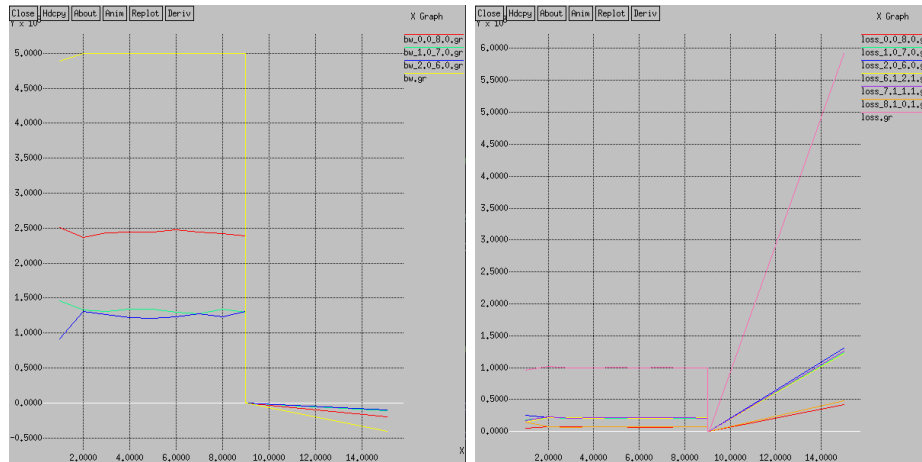


Fig. 5. Largura de banda e perda

C.4 Suponha que o fornecedor de serviços pretenda implementar a seguinte política:

- Para garantir 30% de capacidade para clientes com características idênticas (a capacidade total pode ser usada se disponível).
- O tráfego que excede a taxa negociada deve ser reclassificado, ou seja, encaminhado com prioridade mais baixa.

Espera-se que os alunos proponham e justifiquem uma solução concreta para implementar essa política, usando mecanismos de controle de tráfego adequados disponíveis no NS-2. Inclua resultados de simulação ilustrativos e discuta os prós e contras da solução proposta da perspectiva do provedor de serviços.

Para implementar esta política é necessário adicionar uma Policy Entry para clientes com características idênticas, a qual assegura um CIR de 30% do canal e apenas descarta pacotes quando toda a largura de banda do link de encontra em uso.

Como queremos que todo o tráfego que ultrapassa os 30% da capacidade do link receba uma prioridade menor, utilizaremos a política de TokenBucket.

Aplicando os seguintes filtros, podemos verificar os resultados apresentados no gráfico abaixo:

```
$q(E1C0) addPolicyEntry [$Cli1 id] [$Cli6 id] TokenBucket 10 $cir0 $cbs0
$q(E1C0) addPolicyEntry [$Cli2 id] [$Cli5 id] TokenBucket 10 $cir0 $cbs0
$q(E1C0) addPolicyEntry [$Cli3 id] [$Cli4 id] TokenBucket 10 $cir0 $cbs0
```

Fig. 6. Filtros aplicados

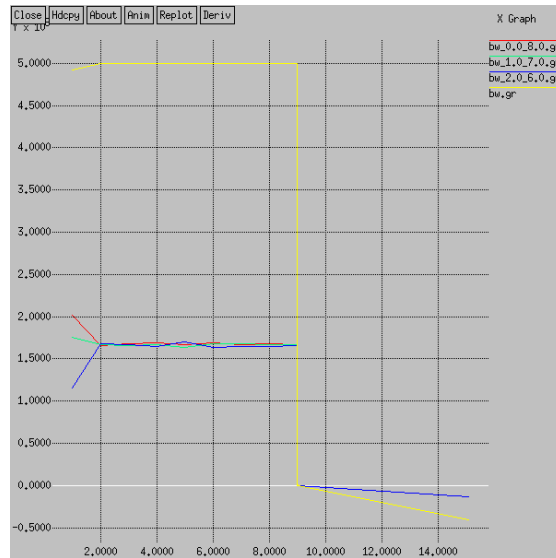


Fig. 7. Largura de banda com as políticas aplicadas

A grande vantagem desta proposta é que, num caso em que todos os clientes necessitem de utilizar a rede, haverá uma distribuição justa da largura de banda do canal, o que se irá refletir numa qualidade mínima de rede para todos os clientes.

Esta solução peca por não permitir a distinção de prioridade entre tipos de tráfego, apenas tendo em conta a fonte do mesmo. Um utilizador pode usar os seus 30% com uma transferência de um ficheiro (a qual não necessita ser realizada rapidamente) impedindo qualquer aplicação não tolerante a atrasos de ter boa qualidade.