

# MC953 - Simulação de Rede Wifi

Luís Felipe Mattos - RA: 107822

24 de Novembro de 2015

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Ferramentas</b>	<b>3</b>
<b>3</b>	<b>Cenários</b>	<b>6</b>
<b>4</b>	<b>Resultados e Análise</b>	<b>8</b>
<b>5</b>	<b>Conclusão</b>	<b>12</b>
<b>6</b>	<b>Referências</b>	<b>13</b>

# 1

## Introdução

A simulação de redes é uma ferramenta muito útil para prever o comportamento de redes sem que haja necessidade de montar toda a estrutura necessária para os testes. Além disso, pode prover informações para o dimensionamento, como por exemplo os protocolos a serem usados, o tamanho dos pacotes, a taxa de transmissão necessária nos links, entre outras.

Neste relatório, iremos montar uma rede simples, com 2 AP's Wifi conectados em um roteador e este conectado em um servidor que representa os serviços de internet que devem ser acessados pelos clientes. Foram feitas análises em cima desta rede utilizando o protocolo UDP e com dois tipos de tráfegos diferentes, em rajadas e com taxa de bits constante (CBR). Além disso, foi variado o número de clientes na rede, a fim de encontrar o ponto em que começa a ocorrer o congestionamento da rede e com isso, o ponto onde existe o aumento do atraso e da perda de pacotes e a diminuição da vazão máxima que a rede suporta.

Esta análise pode ser usada para dimensionar o tipo de tráfego de uma rede, o tamanho dos pacotes, além da quantidade de dados que pode ser gerada pelos clientes e pelo servidor, sem que haja o congestionamento.

O objetivo desta análise é demonstrar que a simulação pode ser bastante útil para fazer este dimensionamento sem que haja necessidade da construção de um protótipo físico.

## 2

# Ferramentas

Para este projeto foi utilizado o simulador de redes NS-3. Este simulador utiliza uma biblioteca (em C++ ou em Python) que é utilizada para especificar a topologia de redes e o comportamento desejado de clientes e servidores.

Para esta implementação, foram utilizados alguns módulos do NS-3:

- ApWifiMac: Cria o endereço MAC para os AP's e faz a comunicação do sinal da rede sem fio (camada física) com a camada de rede
- StaWifiMac: Cria o endereço MAC para os clientes e faz a comunicação do sinal da rede sem fio (camada física) com a camada de rede
- CsmaDevice - Responsável por fazer a comunicação na parte cabeada da rede, ou seja, entre os AP's, o switch e o servidor
- Mobility: Responsável por alocar as posições e a mobilidade dos nós da rede
- BridgeNetDevice: Responsável por fazer a comunicação da parte sem fio com a parte cabeada nos AP's
- OLSR: Responsável pelo roteamento de pacotes pela rede Wifi. Mantém uma tabela de redirecionamento de pacotes que é atualizada dinamicamente
- Stack: Responsável por criar as pilhas e buffers utilizadas pelos protocolos da camada de transporte e da camada de rede
- Ipv4Address: Responsável por atribuir os IP's para os nós da rede
- OnOffApplication: Define o comportamento dos clientes, como por exemplo, o tamanho dos pacotes e o intervalo de envio

Para verificar a corretude da topologia e do comportamento, foi utilizado o plugin nativo do NS-3 chamado Visualizer. Ele é bem simples de utilizar, bastando somente a utilização do seguinte comando:

```
cmd.Parse (argc, argv);
```

E também é necessário incluir a flag `--vis` na linha de execução do programa. Desta forma, obtemos uma janela que mostra graficamente a topologia da rede e se pressionado o botão `F3`, a simulação é iniciada e todos os fluxos são mostrados, bem como as suas respectivas vazões. Um exemplo para 10 clientes pode ser visto na figura 2.1 e na figura 2.2:

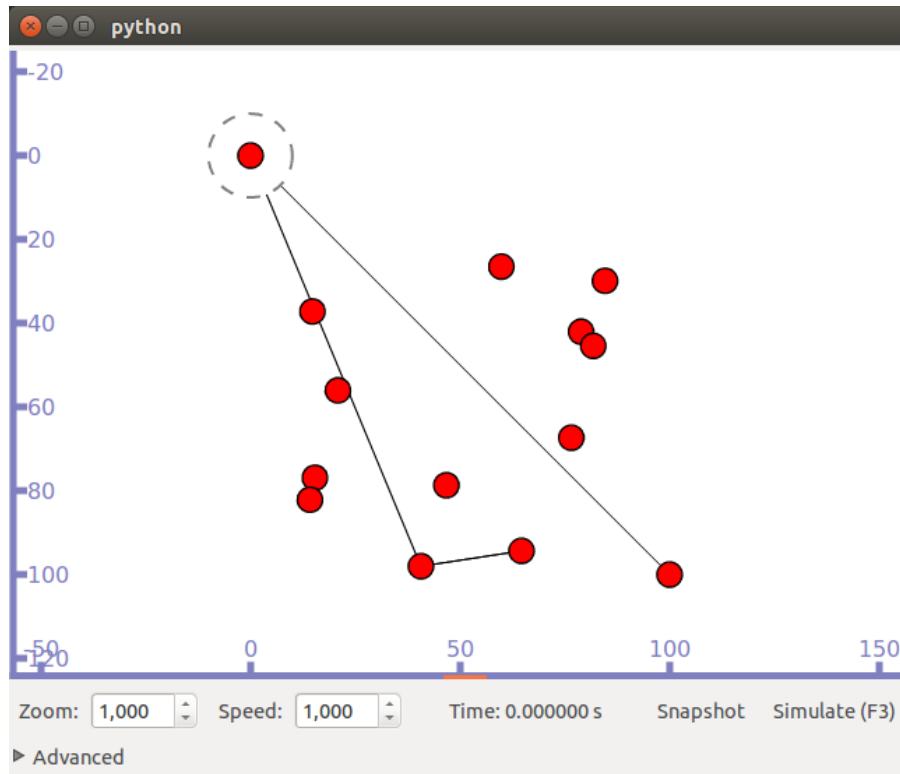


Figura 2.1: Topologia da rede no Visualizer

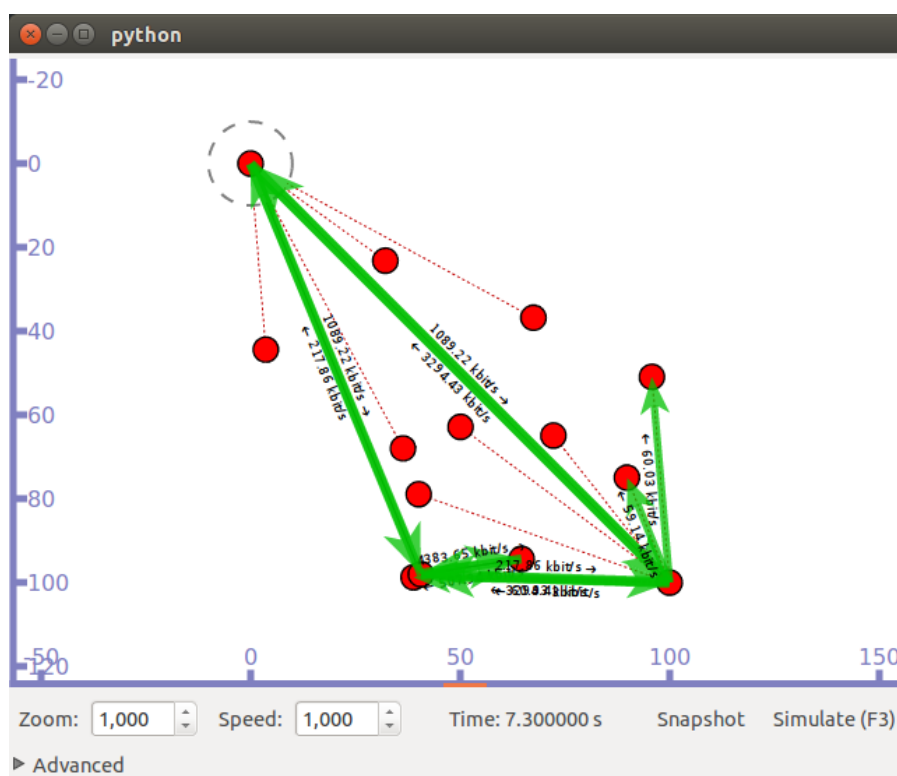


Figura 2.2: Fluxos da rede no Visualizer

Foram utilizados também scripts escritos em Python, que utilizam o pacote Parse. Estes scripts fazem o parsing dos resultados obtidos, calculam as médias da vazão, atraso e perda para uma única execução, a fim de calcular os valores médios para a rede como um todo.

Este procedimento é feito para cada execução separadamente e então para todas as execuções é calculada a média dos valores, bem como o intervalo de confiança de 95%, como foi pedido na especificação do projeto.

### 3

## Cenários

O cenário utilizado para as execuções é bem simples, mas foi eficiente o bastante para se conseguir obter resultados expressivos. Foi feito um quadrado de 100m de lado, onde os dois AP's ficam localizados nos  $(0, 0)$  e  $(100, 100)$ . Os cliente foram gerados em posições aleatórias dentro deste quadrado e se movem também aleatoriamente dentro do quadrado. Como pode ser visto na figura 3.1:

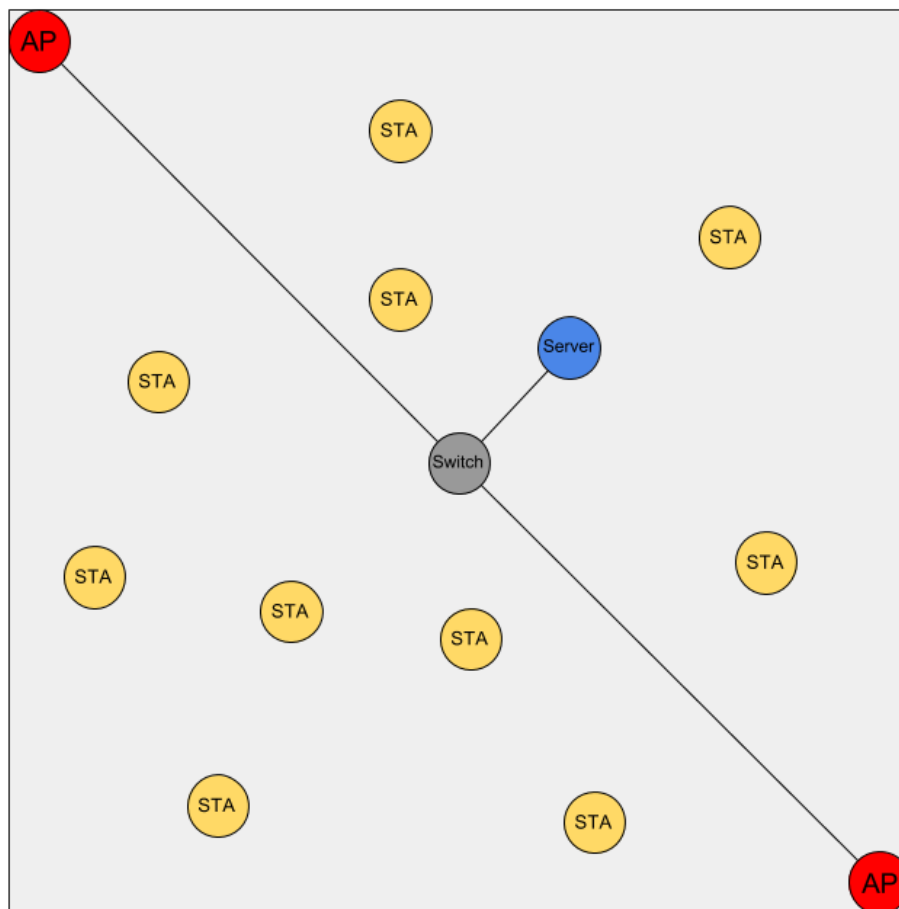


Figura 3.1: Topologia da rede

Apesar de não existir a necessidade de calcular o hand-off (para alunos de graduação), a movimentação dos clientes foi utilizada para garantir um resultado mais próximo daquilo que encontramos na realidade. Além disso, os resultados foram melhores, uma vez que clientes poderiam ser gerados em posições fora do alcance da área de cobertura da Wifi mas que em algum momento se movem para perto de um AP e conseguem se conectar.

Além disso, os testes foram realizados se alterando diferentes parâmetros a cada execução. Alguns desses parâmetros incluem a semente que gera aleatoriamente as posições e o movimento dos clientes, o número de clientes e o tipo de fluxo que será gerado por cada cliente.

Para o número de clientes, foram utilizados valores entre 5 e 40, variando de 5 em 5, ou seja, 5, 10, 15, 20, 25, 30, 35 e 40 clientes. Além disso, para o tipo do fluxo, duas opções foram utilizadas, o fluxo em rajada e o fluxo com taxa de bits constante.

Para o fluxo em rajada, foi escolhido que os clientes deveriam enviar dados durante 0.1s e em seguida entrariam em pausa por 1s, sendo assim, o fluxo seria enviado por um período curto e ficaria em pausa por um período maior, 10 vezes mais longo do que o pico de envio. Além disso, foi determinado que o pacote teria o tamanho de 1500 bytes.

Para o fluxo em CBR, o fluxo tem uma taxa constante (500kbps, por padrão do NS-3) e um pacote de tamanho 512 bytes.

Para a execução, tivemos que para cada tipo de fluxo, foram executados 5 iterações para cada número de clientes na rede, dando um total de 40 execuções para cada tipo de fluxo.



## 4

# Resultados e Análise

Após as execuções, os seguintes gráficos foram gerados a partir dos dados obtidos:

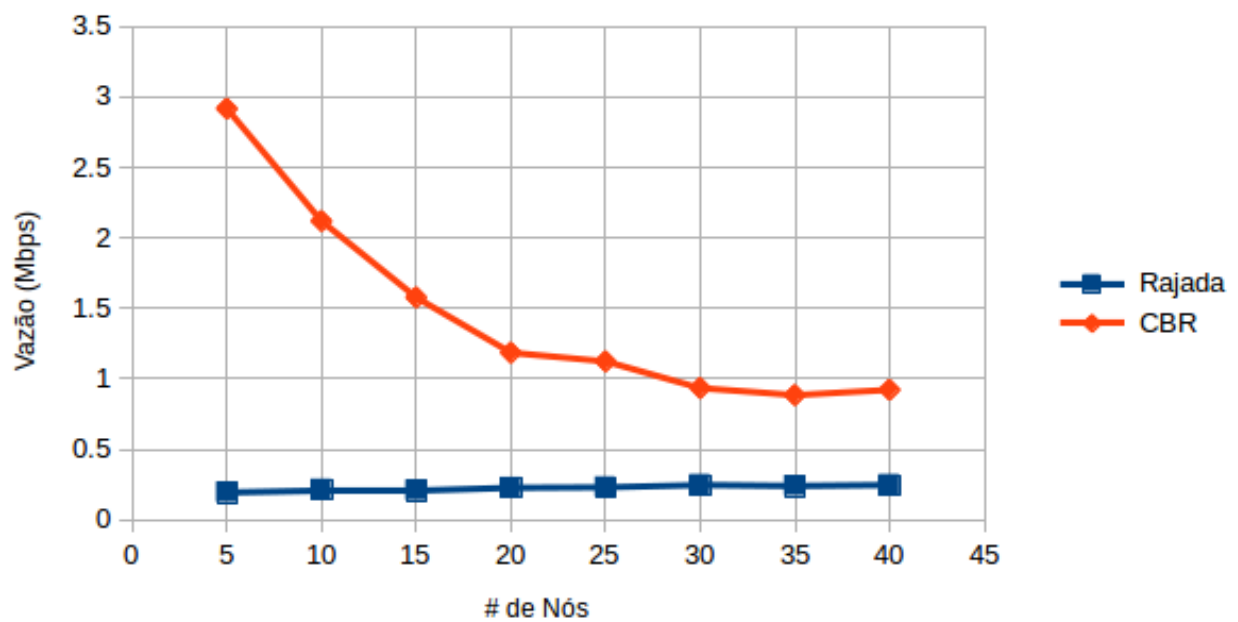


Figura 4.1: Vazão média da rede

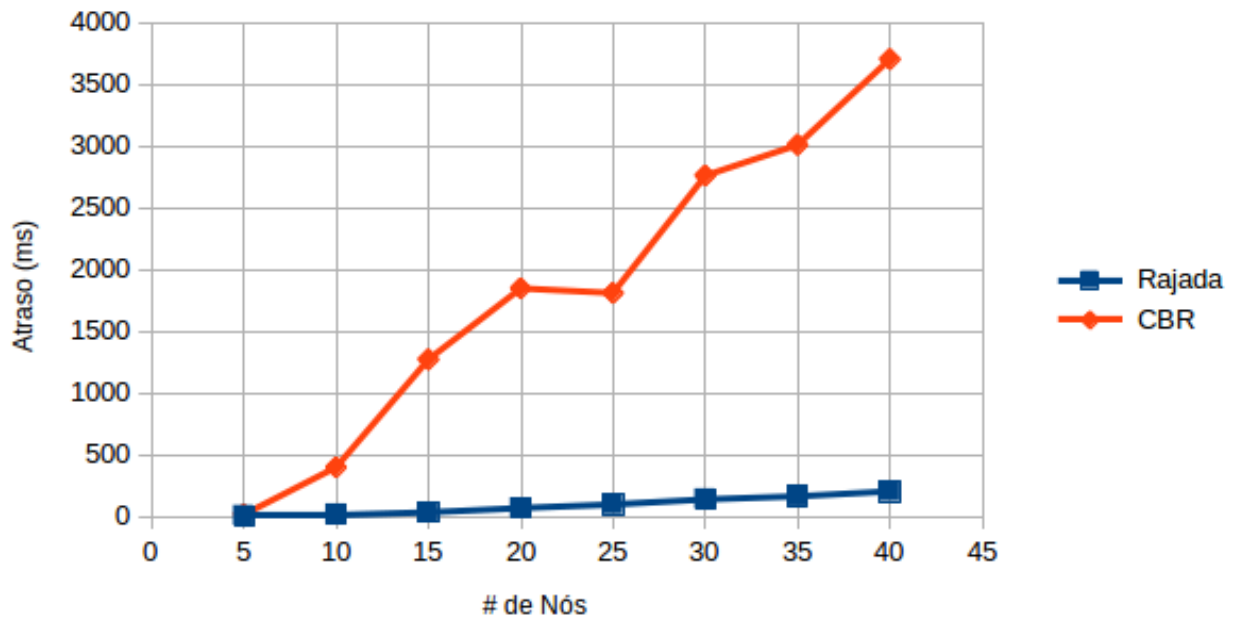


Figura 4.2: Atraso médio da rede

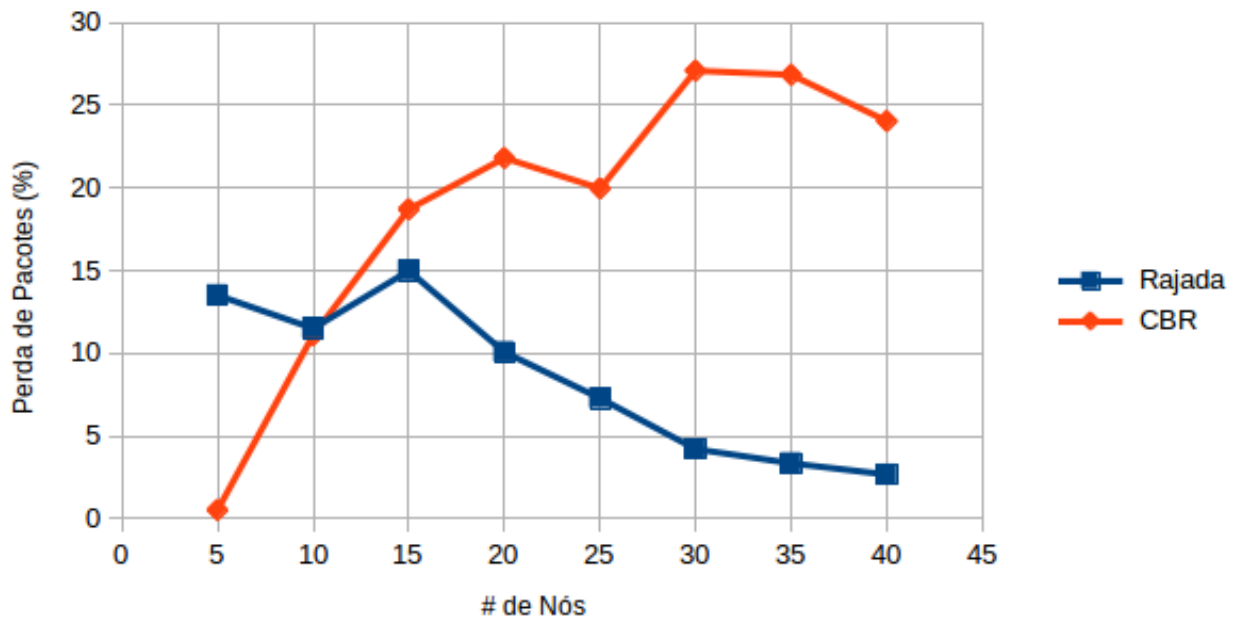


Figura 4.3: Perda média da rede

Como pode ser visto no gráfico da figura 4.1, a vazão para o fluxo em rajada se mantém constante e baixa de acordo com o número de clientes enquanto a vazão para o fluxo CBR é

alto para 5 cliente e diminui exponencialmente a medida que o número de nós da rede aumenta.

Isso acontece porque o tráfego em rajadas não envia pacotes o suficiente para congestionar a rede, já que o tempo em que os pacotes são enviados é muito menor do que o tempo de silêncio da rede. Já para o tráfego CBR, quando há poucos nós, a rede fica relativamente descongestionada e pacotes são enviados continuamente, por causa disso, vazões mais altas são atingidas em comparação ao tráfego em rajadas. Porém, ao se aumentar o número de clientes, a rede começa a ficar rapidamente congestionada, visto o número de pacotes enviados e a vazão cai exponencialmente.

O gráfico da figura 4.2 mostra que o atraso da rede aumenta de acordo com o número de clientes, porém, para o tráfego em rajadas, o aumento é pouco expressivo se comparado com o aumento do atraso na rede com tráfego CBR. Isso se dá pelo mesmo motivo já citado anteriormente, a rede com o tráfego em rajada não alcança o limiar necessário para congestionar a rede, ao contrário do tráfego CBR.

Vemos que este congestionamento pode causar um atraso grande na rede, visto que os resultados obtidos para uma rede com 40 clientes tem um atraso de quase 4s enquanto que na rede com tráfego em rajada, para os mesmos 40 clientes, a rede tem um atraso de aproximadamente 0.25s.

Para a perda de pacotes, como mostrado no gráfico da figura 4.3, vemos que o número de pacotes enviados influencia diretamente na porcentagem de perda. Para 5 clientes vemos que o tráfego CBR tem uma perda muito menor do que o tráfego em rajadas, isso se dá por causa do número de pacotes enviados e por causa do baixo nível de congestionamento da rede. Poucos pacotes são perdidos em ambos os tipos de tráfego, mas o número de pacotes enviados é muito maior no fluxo CBR, isso causa a diminuição da porcentagem da perda.

Este cenário muda bastante quando se aumenta o número de clientes da rede. No caso da rede com tráfego em rajadas, o número de pacotes perdidos se mantém constante, uma vez que a rede não é congestionada, porém como o número de pacotes enviados aumenta, a perda diminui com o número de clientes. Para o tráfego em CBR o congestionamento da rede faz com que o número de pacotes perdidos aumente consideravelmente e o número de pacotes enviados também aumente, mas como a rede fica cada vez mais congestionada, muitos pacotes são perdidos.

Foram calculados também os intervalos de confiança de 95% para as execuções sobre os mesmos parâmetros, ou seja, existe a garantia que 95% das execuções possuem uma média dentro do intervalo calculado. Os intervalos são mostrados nas tabelas a seguir:

Tabela 4.1: Vazão média e intervalos de confiança de 95%

# de Nós	Tráfego em Rajada [min, média, máx] (Mbps)	Tráfego CBR [min, média, máx] (Mbps)
5	[0.173, 0.194, 0.216]	[2.814, 2.918, 3.022]
10	[0.208, 0.214, 0.220]	[2.012, 2.120, 2.228]
15	[0.197, 0.209, 0.220]	[1.519, 1.578, 1.638]
20	[0.215, 0.227, 0.240]	[1.108, 1.186, 1.263]
25	[0.220, 0.230, 0.241]	[1.032, 1.124, 1.216]
30	[0.243, 0.247, 0.251]	[0.886, 0.935, 0.984]
35	[0.231, 0.241, 0.250]	[0.726, 0.884, 1.043]
40	[0.243, 0.247, 0.252]	[0.859, 0.922, 0.986]

Tabela 4.2: Atraso médio e intervalos de confiança de 95%

# de Nós	Tráfego em Rajada [min, média, máx] (ms)	Tráfego CBR [min, média, máx] (ms)
5	[5.23, 6.92, 8.61]	[3.29, 18.16, 33.03]
10	[16.66, 18.88, 21.10]	[273.22, 400.78, 528.34]
15	[30.18, 35.44, 40.70]	[960.03, 1273.40, 1586.77]
20	[62.98, 69.84, 76.70]	[1384.90, 1849.98, 2315.06]
25	[90.90, 99.30, 107.69]	[1451.24, 1812.00, 2172.76]
30	[133.73, 140.74, 147.75]	[2409.36, 2762.57, 3115.77]
35	[149.16, 165.25, 181.34]	[2463.31, 3011.99, 3560.66]
40	[198.55, 205.35, 212.15]	[3403.81, 3705.55, 4007.28]

Tabela 4.3: Perda média e intervalos de confiança de 95%

# de Nós	Tráfego em Rajada [min, média, máx] (%)	Tráfego CBR [min, média, máx] (%)
5	[10.07, 13.51, 16.95]	[0.0, 0.52, 1.35]
10	[9.37, 11.50, 13.64]	[8.19, 11.09, 13.99]
15	[12.07, 14.99, 17.91]	[13.11, 18.71, 24.30]
20	[5.51, 10.07, 14.63]	[14.55, 21.80, 29.06]
25	[6.08, 7.28, 8.47]	[13.00, 19.96, 26.92]
30	[3.75, 4.21, 4.66]	[20.50, 27.08, 33.66]
35	[2.65, 3.35, 4.05]	[20.67, 26.83, 32.98]
40	[2.09, 2.67, 3.25]	[20.74, 24.03, 27.33]

## 5

# Conclusão

Este experimento mostrou que o tráfego em rajada, apesar de possuir uma vazão muito menor do que o tráfego CBR, se mostrou mais confiável para o número de clientes testados, porém, neste caso, a rede está sub-utilizada, visto que o tráfego em rajada está abaixo da assíntota gerada pelo tráfego CBR, na figura 4.1. Poderia ser feito o redimensionamento desta rede, a fim de aumentar a eficiência.

Poderia ser feito um aumento da taxa de dados gerados pelos clientes/servidor ou então, um aumento do tempo em que os clientes estão enviando rajadas de pacotes.

De um modo geral, para poucos clientes, a rede com tráfego CBR se mostrou muito mais eficiente, uma vez que a vazão é grande e o atraso e a perda são pequenos. Enquanto que, para muitos clientes, a rede com tráfego em rajadas se mostrou mais eficiente, uma vez que o atraso e a perda são pequenos, apesar da vazão não ser tão alta como no tráfego CBR para a mesma quantidade de clientes.

Para serviços que necessitam de alta disponibilidade/confiabilidade e baixa latência, mesmo com quantidade grande de outros nós conectados na mesma rede, o tráfego em rajadas se mostrou melhor, enquanto que para serviços que toleram uma alta taxa de perda de pacotes e um atraso grande, o tráfego CBR é uma opção melhor, já que a vazão é maior.

# 6

## Referências

- Manual de referência do NS-3

<https://www.nsnam.org/docs/release/3.8/manual.pdf>

- Documentação

<https://www.nsnam.org/doxygen/index.html>