

Universidade Federal de Minas Gerais

TP1 – Redes de Computadores

Alunos: Enzo Roiz e Juan Oliveira

1 – Introdução

Este trabalho visa a instalação do ambiente virtualizado que simula uma rede, o *Mininet*, onde serão executados programas para a avaliação de desempenho de protocolos. Para isto serão escritos dois pares cliente-servidor em *python*. Em um par, o cliente envia n mensagens ao servidor e aguarda que o servidor lhe envie uma resposta ao final. No outro par, a cada mensagem enviada pelo cliente, o servidor envia de volta uma resposta. A rede simulada para este trabalho terá a seguinte topologia: Um host conectado a um switch, por um link com largura de banda 10Mbps e latência de 10ms. Este switch é por sua vez conectado a outro switch, por, inicialmente, um link com largura de banda 1Mbps e latência de 100ms. Este último switch é então ligado a outro host com largura de banda 10Mbps e latência de 10ms. Será avaliado então o impacto de três parâmetros sobre o processo de comunicação: o tamanho das mensagens, a latência do link entre os dois switches e a banda do link entre os dois switches.

2 – Desenvolvimento

2.1 – Pares Cliente-Servidor e Topologia

Os dois pares de cliente e servidor, n mensagens e uma resposta, n mensagens e n respostas, foram escritos baseados nos sockets básicos mostrados em sala de aula, e encontrados em link descrito nas referências para este trabalho.

2.1.1 – Servidores

Em os ambos servidores é determinado um tamanho de buffer para recepção de mensagens, ou seja, quantos bytes o servidor é capaz de receber por vez. Para este trabalho o tamanho de buffer adotado foi 1024B ou 1KB, como utilizado nos exemplos em sala de aula e no livro texto adotado para a disciplina.

Ambos os servidores têm operação semelhante: O servidor cria o socket e “escuta” a rede, aguardando pela conexão do cliente. Após o cliente se conectar, o servidor envia uma mensagem informando o tamanho de buffer utilizado para recebimento de mensagens.

Em um servidor, a cada mensagem recebida é enviada uma resposta para o cliente, enquanto no outro servidor, todas as mensagens são recebidas antes de ser enviada a resposta ao cliente.

A conexão é encerrada quando o servidor encontra um “caractere de parada de transmissão”.

2.1.2 – Clientes

Assim como nos servidores, os clientes têm modo de operação semelhante. Cada cliente cria o socket e se conecta ao servidor. Após a conexão com o servidor, o cliente recebe uma mensagem contendo a informação do tamanho do buffer de recebimento de mensagens do servidor. No cliente, tem-se uma variável para definir quantos bytes serão enviados. Assim o cliente obtém o número de mensagens através da fórmula:

Fórmula 1 – Número de mensagens

$$\text{Número de mensagens} = \frac{\text{Bytes a enviar}}{\text{Tamanho do buffer do servidor}}$$

Sabendo o número de mensagens a serem enviadas, o cliente realiza um loop para mandar o número de mensagens especificado. Para cada iteração do loop, o cliente preenche uma string do mesmo tamanho do buffer de recepção do servidor com o caractere 'a', e envia a string para o servidor. Em um cliente para cada iteração do loop é esperada uma resposta do servidor, enquanto no outro a resposta do servidor é esperada após o envio de todas as mensagens. Em ambos os clientes a última mensagem contém o caractere 'b', entendido pelo servidor como o fim de transmissão por parte do cliente.

2.1.3 – Topologia

Como descrito na introdução, a topologia simulada pelo *Mininet*, tem dois hosts e dois switches. Cada host é ligado a um switch por um link com 10Mbps de banda e 10ms de latência. Os switches são interligados por, inicialmente, um link com 1Mbps e latência de 100ms.

No arquivo de topologia existem duas variáveis, banda e latência, através das quais se define a banda e latência do link entre os dois switches. Estas duas variáveis são as que serão alteradas para a realização das medições de tempo.

2.2 – Medições

Com o *Mininet* instalado em máquina virtual e devidamente configurado, as medições de tempo para o envio de mensagens se deram de duas maneiras:

Na primeira, foi definido um tamanho de mensagem para os clientes 1KB, e serão variadas a banda (nos valores 0.1Mbps, 1Mbps, 10Mbps e 100Mbps) e a latência (nos valores 1ms, 5 ms, 10ms, 50ms, 100ms, 500ms) do link entre os dois switches .

Na segunda, a banda e a latência do link entre os dois switches serão fixadas em 1Mbps e 100ms, respectivamente, e será variado o tamanho das mensagens (nos valores 100B, 1KB, 10KB, 100KB) enviadas pelos clientes. Para este último caso será efetuado ainda mais um teste, considerando uma taxa de erro (perda de pacotes) de 1%, no link entre os dois switches.

Para cada caso foi medido o tempo total de transmissão e colocado em uma planilha para a avaliação do desempenho, o próximo assunto abordado.

3 – Avaliação de desempenho

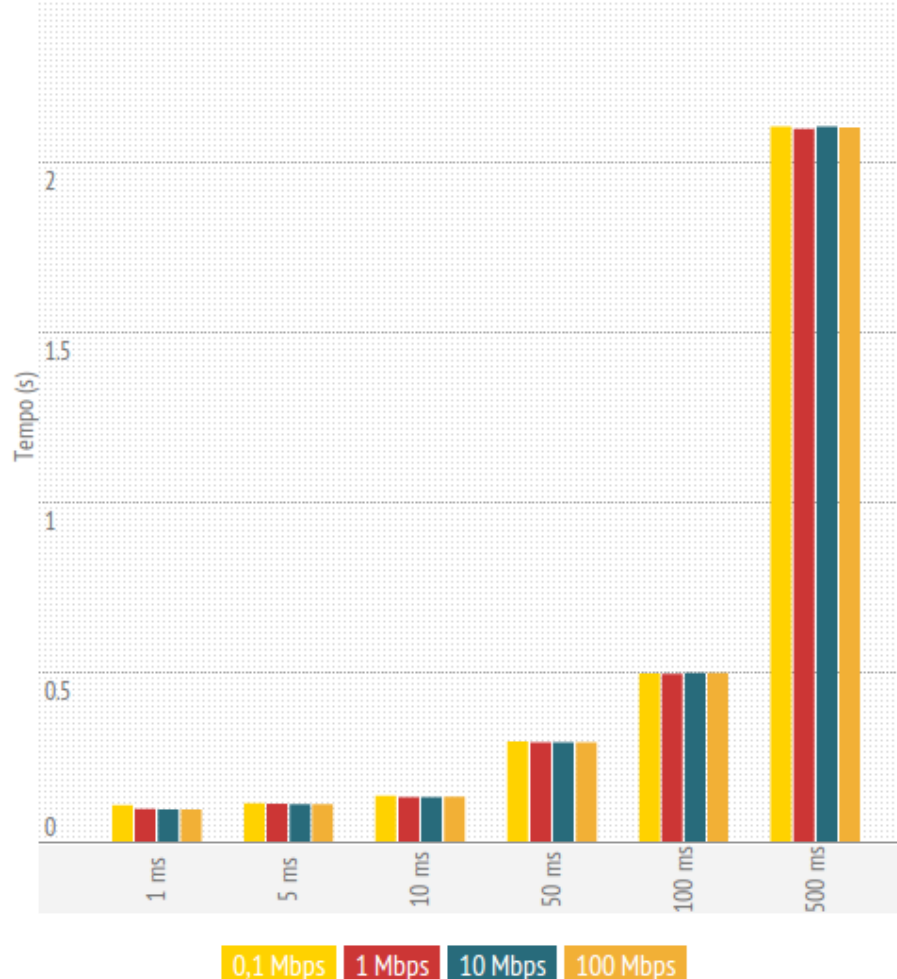
3.1 – Avaliação de desempenho: Mensagens de 1KB, variando Banda e Latência

3.1.1 – Par n mensagens, uma resposta

3.1.1.1 – Tempo total de transmissão

O gráfico a seguir representa a variação do tempo total de transmissão de mensagem, de acordo com a latência e largura de banda.

Figura 1 – (N-1) Comparativo de tempo de transmissão total para diferentes latências – agrupado por largura de banda



Podemos observar através do gráfico que quanto maior a latência, maior será o tempo total de transmissão de mensagem. Isto se dá devido ao fato de a mensagem enviada para a realização dos testes (1 KB) ser considerada pequena para todas as larguras de banda consideradas neste trabalho. Assim a latência é dominante em relação ao tempo total de transmissão. Podemos enxergar isto através da fórmula:

Fórmula 2 – Tempo de transmissão

$$\text{Tempo de transmissão} = \frac{\text{Volume de dados}}{\text{Banda}}$$

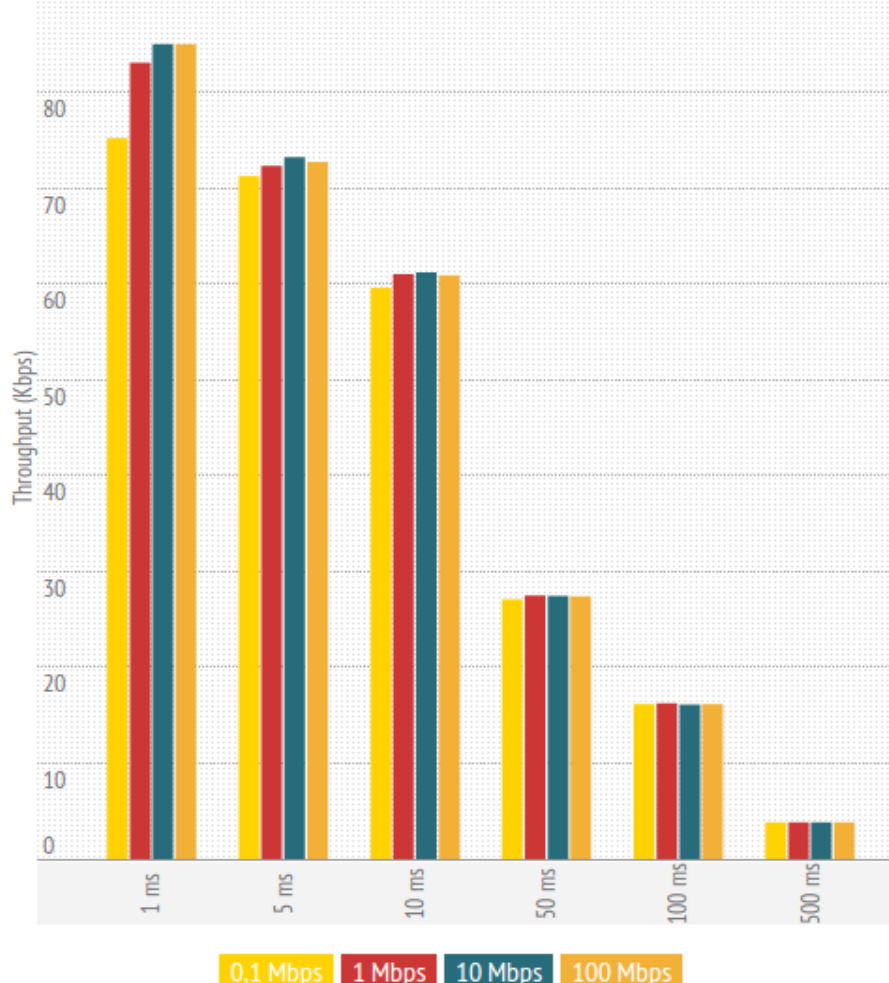
$$\text{Tempo de transmissão total} = \text{Tempo de transmissão} + \text{Tempo de Propagação}$$

Como o volume de dados é pequeno em relação a qualquer uma das bandas testadas, o tempo de transmissão será muito pequeno, assim, o tempo de transmissão total, mostrado na *Fórmula 2*, terá como seu maior responsável o tempo de propagação ou latência. Analisando o gráfico podemos perceber de maneira mais efetiva que a largura de banda não interfere no tempo total de transmissão, pois os resultados são muito próximos entre as larguras de banda testadas para uma mesma latência.

3.1.1.2 Throughput

O gráfico abaixo mostra como o throughput varia à medida que aumentamos a latência e a banda.

Figura 2 – (N-1) Comparativo de throughput para diferentes latências – agrupado por largura de banda



Sabe-se o throughput ser calculado sob a fórmula:

$$\text{Throughput} = \frac{\text{Volume de dados(bits)}}{\text{Latência}}$$

Como o servidor recebe 1KB por vez, e o volume de dados é fixo, mensagem de tamanho 1KB, pela fórmula acima podemos ver que, quanto maior a latência, menor o throughput. Analisando

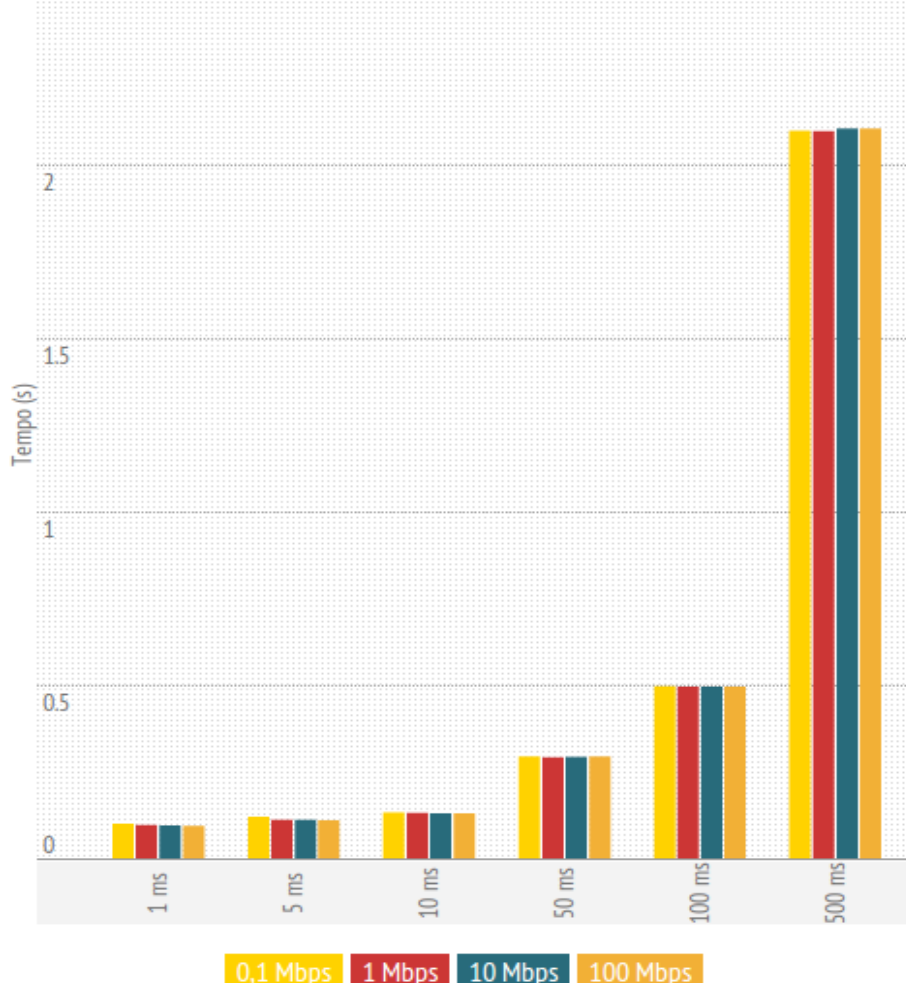
cada latência separadamente, podemos ver como a largura de banda interfere no throughput, pois quanto maior a banda disponível, maior o throughput. Porém ao aumentarmos a latência, podemos ver que o throughput fica cada vez mais igual para todas as larguras de banda, ficando imperceptível a diferença. Como explicado anteriormente, isto se deve ao fato de a latência dominar o tempo total de transmissão.

3.1.2 – Par n mensagens, n respostas

Como descrito nos métodos de avaliação de desempenho, para esta avaliação, a mensagem enviada teria tamanho fixo de 1KB e seriam variados, latência e banda. Como utilizado no exemplo do livro, e slides do professor passados em aula, o servidor recebe, por vez, mensagens de 1024 bytes, ou 1KB. Este tamanho de buffer foi então adotado para ambos os códigos de servidores escritos para este trabalho, tanto para n mensagens e uma resposta, quanto para n mensagens e n respostas. Assim, como a mensagem enviada pelo cliente é igual ao tamanho do buffer do servidor (1KB), apenas um pacote de dados será enviado. Desta maneira os dois pares cliente-servidor obtêm resultados semelhantes, pois neste caso fazem as mesmas operações. Ambos enviam apenas um pacote e recebem apenas uma resposta, sendo este o único caso em que os dois pares cliente-servidor agem de maneira igual. Já que ao enviar mais de um pacote de dados um deles receberá apenas uma resposta, enquanto o outro receberá tantas respostas quanto forem os pacotes de dados. Isto posto, a avaliação de desempenho, para tempo total de transmissão e throughput, para os dois pares cliente servidor têm resultados semelhantes.

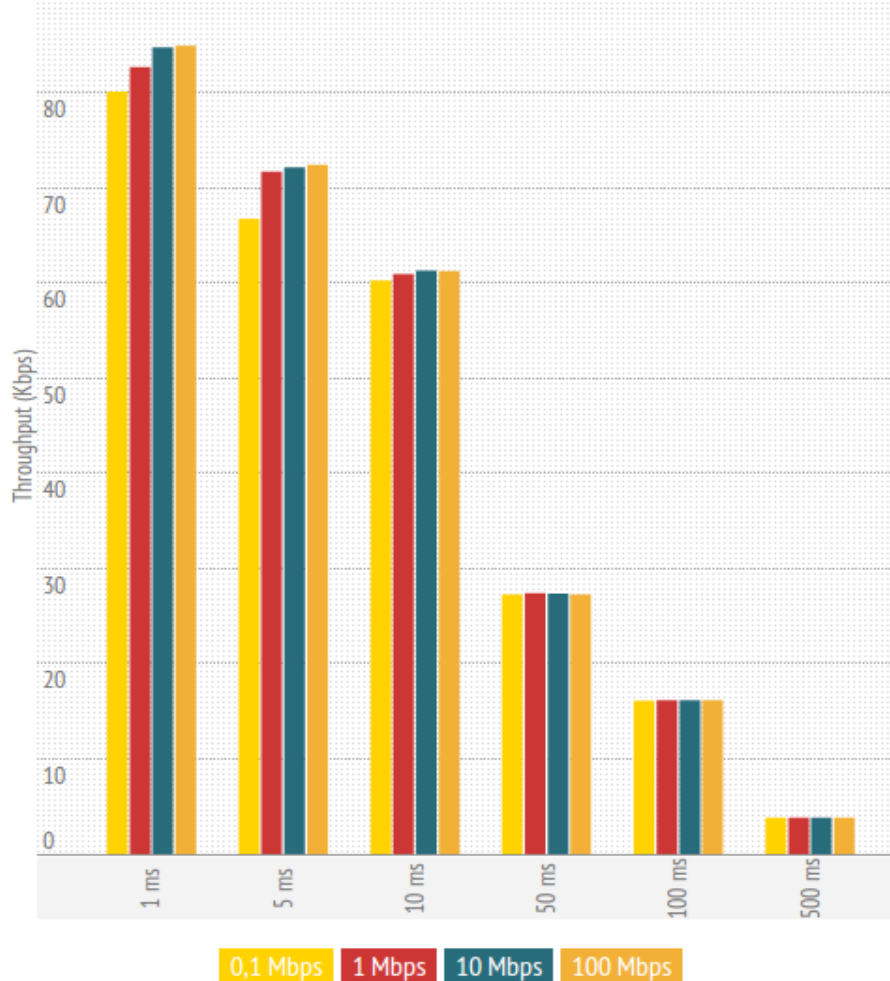
3.1.2.1 – Tempo total de transmissão

Figura 3 – (N-N) Comparativo de tempo de transmissão total para diferentes latências – agrupado por largura de banda



3.1.2.2 – Throughput

Figura 4 – (N-N) Comparativo de throughput para diferentes latências – agrupado por largura de banda



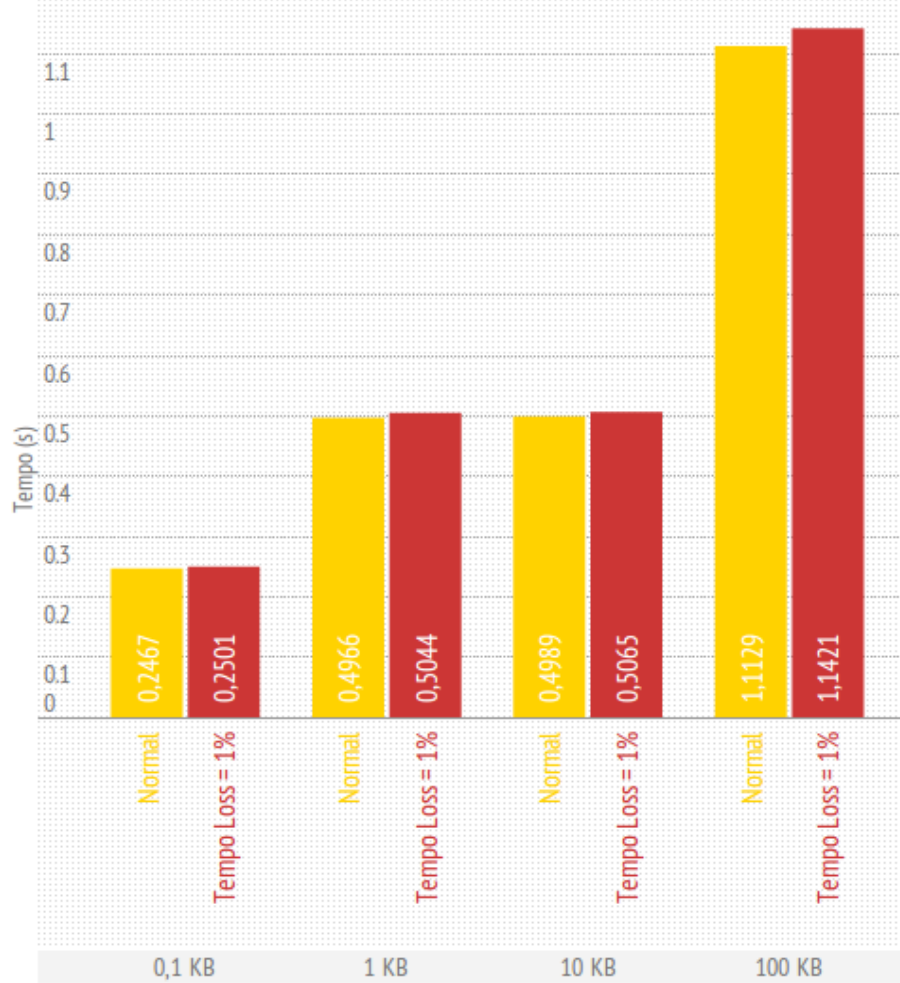
3.2 – Avaliação de desempenho: Banda = 1Mbps e Latência=100ms, variando Mensagens

3.2.1 – Par n mensagens, uma resposta

3.2.1.1 – Tempo total de transmissão

A seguir temos o gráfico construído a partir dos resultados obtidos para o envio de mensagens de diferentes tamanhos, sendo eles: 0.1KB, 1KB, 10KB e 100KB. Os tempos em amarelo, representam o envio sem perda de pacotes, já os tempo em vermelho representam uma perda de pacotes de 1%.

Figura 5 - (N-1) Comparativo de tempos para diferentes tamanhos de mensagens – Com e sem perda de pacotes

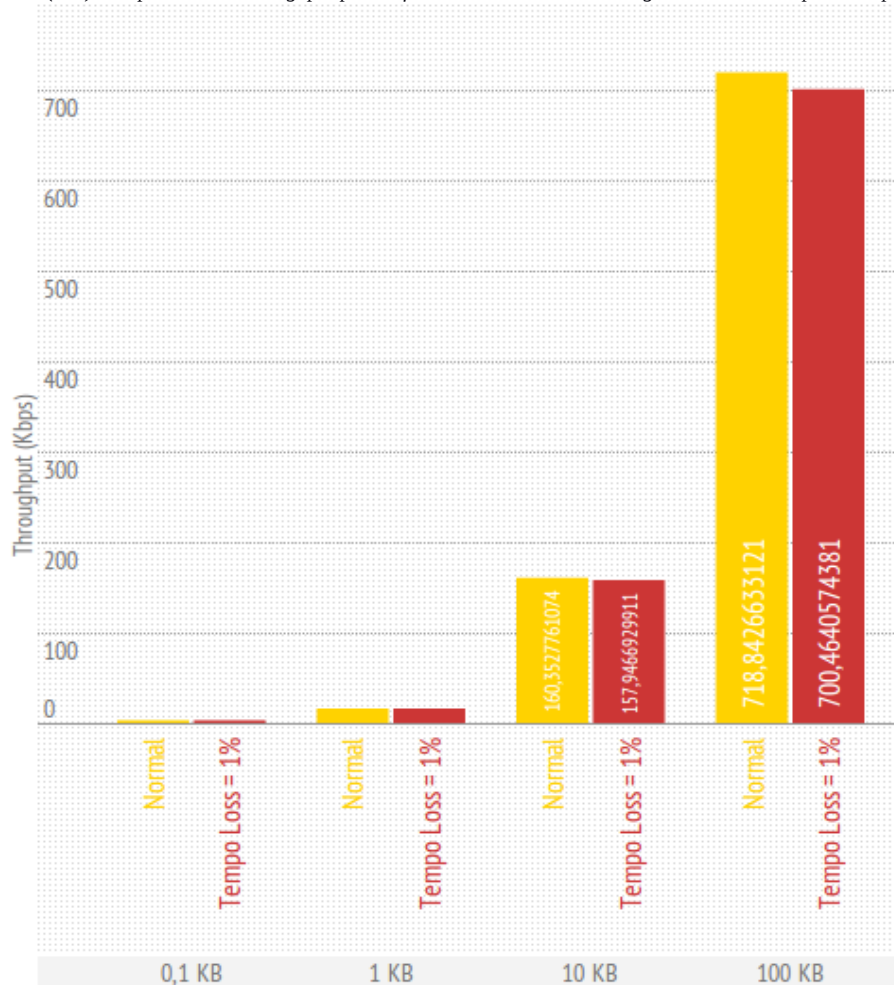


Naturalmente, quanto maior a mensagem, maior o seu tempo de envio, pois mais pacotes serão enviados pelo cliente pela rede. Através do gráfico percebemos também que a perda de pacotes afeta, mas não tão sensivelmente o desempenho do envio de mensagens, pois o envio com perda de pacotes para todos os tamanhos de mensagem foi mais lento, mas de uma forma não muito mais lenta do que o envio sem perda de pacotes.

3.2.1.2 – Throughput

No gráfico a seguir temos o throughput para mensagens enviadas de diferentes tamanhos, considerando os envios sem perda de pacotes e com uma perda de pacotes de 1%.

Figura 6 - (N-1) Comparativo de throughput para diferentes tamanhos de mensagens – Com e sem perda de pacotes



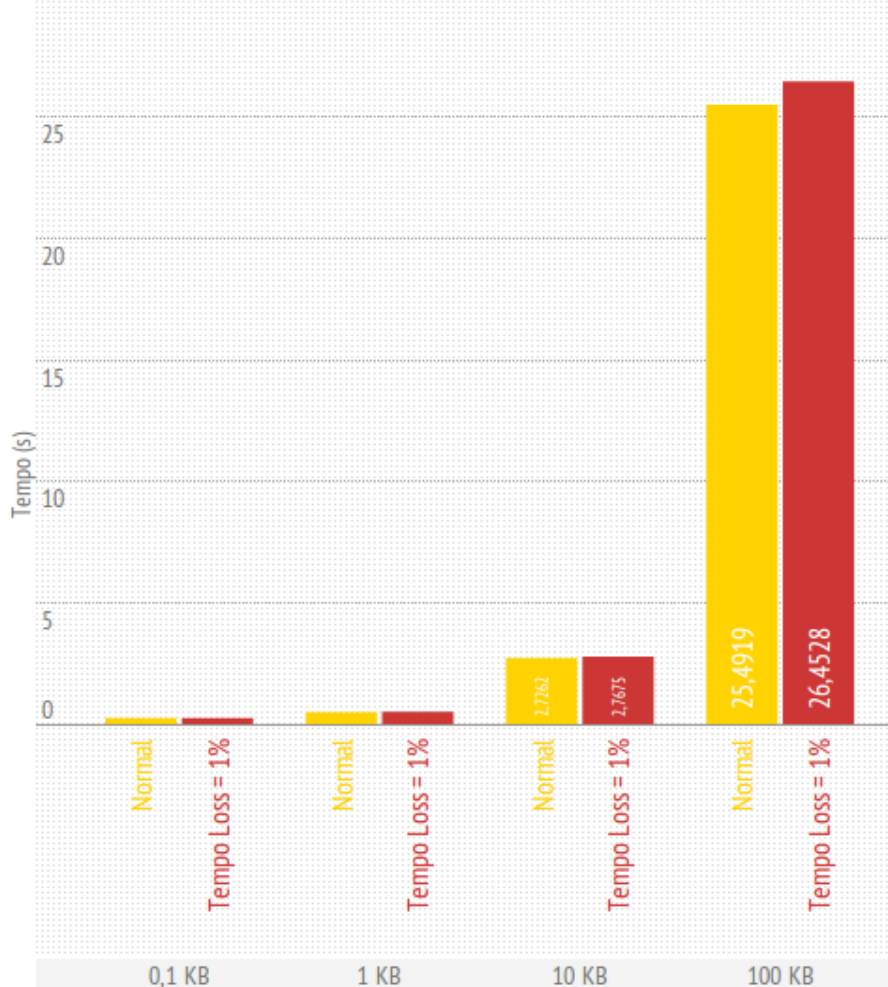
Como na análise anterior, de tempo total de transmissão, o throughput aumenta à medida que aumentamos o tamanho da mensagem enviada, porém o tempo de envio da mensagem não aumenta tão significativamente quanto o throughput. Esse aumento significativo do throughput é devido ao fato de, neste método de avaliação de desempenho, o cliente mandar n pacotes sem que seja interrompido, sem envio de resposta por parte do servidor. Assim, pela *Fórmula 3*, de throughput, se aumentarmos consideravelmente o tamanho da mensagem (volume de dados), mas não ocorrer um aumento tão considerável na latência (tempo de envio), como acontece neste exemplo, o throughput aumentará também de maneira considerável. Em outras palavras, o cliente envia mensagens cada vez maiores, com tempos totais de transmissão semelhantes. O que significa que o cliente é capaz de colocar na rede uma grande quantidade de bits, até que o servidor lhe envie a resposta. De forma análoga à seção anterior, a perda de pacotes afeta o throughput, mas seu impacto, pelo menos nos casos abordados, não é tão significativo.

3.2.2 – Par n mensagens, n respostas

3.2.1.1 – Tempo total de transmissão

O gráfico a seguir, nos permite observar como o tempo aumenta, neste caso significativamente, à medida que as mensagens enviadas aumentam de tamanho. Enquanto para enviar 1KB, o tempo de envio é de cerca de 0.5 segundo, para enviarmos 100KB, o tempo necessário é de aproximadamente 25.5 segundos, considerando que não há perda de pacotes, os mesmos tempos de envio para o par cliente-servidor que envia n mensagens e recebe apenas resposta ao final, são de, respectivamente, 0.5 segundo e 1.1 segundo. O aumento vertiginoso do tempo total de transmissão então se deve ao fato de o cliente esperar resposta do servidor a cada envio de pacote.

Figura 7 - (N-N) Comparativo de tempos para diferentes tamanhos de mensagens – Com e sem perda de pacotes

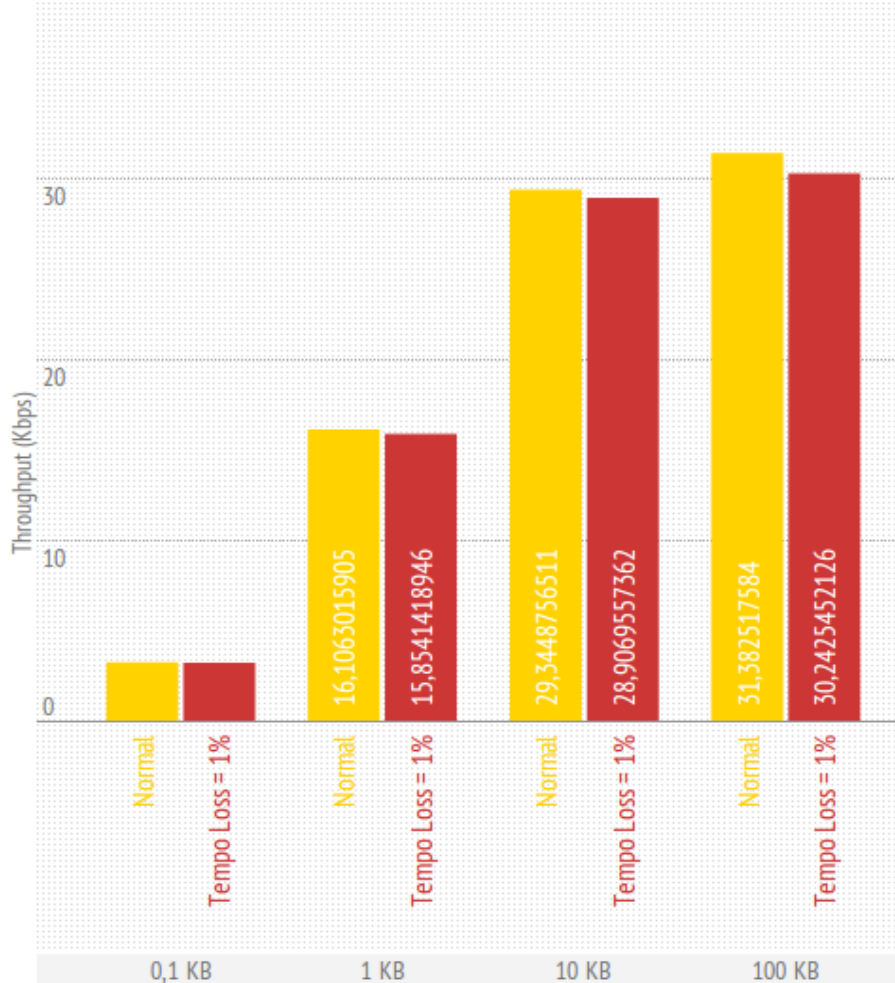


Para o par cliente-servidor n mensagens, n respostas, na rede transita apenas um pacote de cada vez, ou o pacote enviado pelo cliente, ou o pacote de resposta do servidor, enquanto para o outro de n mensagens, uma resposta, as mensagens do cliente são colocadas todas na rede no momento do envio, e recebidas pelo servidor, que após o recebimento das mensagens envia a resposta ao cliente. Para cada mensagem do cliente então, a rede é percorrida de ponta a ponta duas vezes. É fácil perceber assim que quanto maior a mensagem, mais vezes a rede será percorrida em sua totalidade e seu tempo de transmissão irá aumentar consideravelmente de acordo com o número de mensagens enviadas. Como no caso anterior, a perda de pacotes também afeta o desempenho do envio de mensagens mas de maneira não significativa, podendo-se observar que os tempos de envio com perdas de pacotes são novamente maiores que para envio sem perdas de pacote.

3.2.1.2 – Throughput

Abaixo, o gráfico nos permite perceber como o throughput é afetado à medida que aumentamos o tamanho das mensagens enviadas pelo cliente.

Figura 8 - (N-N) Comparativo de throughput para diferentes tamanhos de mensagens – Com e sem perda de pacotes



Neste caso, assim como no par cliente-servidor de n mensagens e uma resposta, o throughput cresce conforme aumentamos o tamanho das mensagens. A diferença, porém, é que neste caso o throughput não cresce vertiginosamente. O responsável por este fato é o tempo total de transmissão. Analisando novamente a *Fórmula 3*, de throughput, podemos perceber que o tamanho das mensagens (volume de dados) cresce consideravelmente, porém o tempo total de transmissão (latência) também cresce de maneira considerável, mantendo a proporção do quociente entre os dois. Por isso neste caso o throughput, é baixo, devido à ineficiência do par cliente-servidor. Da mesma forma da análise anterior, a perda de pacotes afeta o throughput, mas pouco significativamente.

4 – Conclusão

Este trabalho nos permitiu utilizar um ambiente virtualizado, o *Mininet*, para simular uma topologia de rede e executar dois pares cliente-servidor na rede simulada a fim de mensurar o desempenho no processo de comunicação utilizando diferentes abordagens. Em uma delas, utilizando mensagem de tamanho fixo, variando a banda e latência. Em outra, com banda e latência fixa, variando o tamanho das mensagens enviadas e também considerando taxa de erro (perda de pacotes). Foram escritos dois pares cliente-servidor: um para enviar n mensagens, com resposta do servidor ao final do envio e outro para enviar n mensagens, obtendo uma resposta a cada envio,

Através da análise de desempenho pudemos entender como cada um dos fatores analisados, banda, latência e tamanho de mensagem, afeta o tempo de envio dos pacotes e o throughput para cada par cliente-servidor escrito, nos mostrando a diferença de desempenho existente entre eles.

5 – Referências

Instalação do Mininet -

<http://mininet.org/vm-setup-notes/>

Passo a passo na utilização do Mininet -

<http://mininet.org/walkthrough/>

Sockets em Python -

<http://www.python.org.br/wiki/SocketBasico>

Configuração de interface Host-Only no Mininet -

<http://superuser.com/questions/429405/how-can-i-get-virtualbox-to-run-with-a-hosts-only-adapter>

Análise de Desempenho -

Larry L. Peterson and Bruce S. Davie (5th Edition). *Computer Networks: A Systems Approach*.

Geração de gráficos -

<http://www.infogr.am>