



Dimensionamento da Capacidade de um Enlace de Dados com Serviços Elásticos e Stream Utilizando a linguagem Python

Palavras-Chaves: Qualidade de Serviço(QoS), Codificadores/Decodificadores, Simulação por eventos Discretos, Python.

Autores:

Gabriel Enrico Alves do Nascimento - Faculdade de Tecnologia

Edson Luiz Ursini (Orientador) - Faculdade de Tecnologia

1. Introdução

Com o surgimento da Internet no final da década de 1960, o propósito inicial era o envio de arquivos, ou melhor, tráfego com características elásticas no qual os problemas de atrasos são bem menos relevantes que os de perda de pacotes[2], já que ainda não se pensava em serviços que poderiam ser executados em tempo real, por exemplo, o serviço VoIP (voz sobre o protocolo IP), que requer atrasos muito limitados, característica dos serviços stream. Todavia, o iminente crescimento das redes móveis e do tráfego de informações, alinhado com os novos tipos de serviços, exigiu um planejamento e dimensionamento mais eficiente e eficaz dessas redes que começaram a incluir voz, vídeo e dados na mesma infraestrutura[1].

O fundamental propósito do planejamento de uma rede multiserviços IP é atender as condições de QoS (Qualidade de Serviço) para um determinado tipo de serviço e volume de tráfego[1]. Esse planejamento é feito de tal modo que possibilite um funcionamento adequado dessa rede, considerando as diferentes características e os requisitos estipulados para cada tipo de serviço. Outra característica dos serviços stream é que estes requerem uma determinada largura de banda que vai depender do tipo de serviço e do Codificador/Decodificador utilizado[2].

Com o crescimento tecnológico, houve uma melhora na transferência de dados, ocasionado pela velocidade de transmissão e capacidades de armazenamento

de dados, dando surgimento a novos tipos de serviços como videoconferência, jogos em tempo real, etc. Essas novas demandas de serviços têm, em sua grande maioria, características do tipo Stream.

Segundo o ITU-T (International Telecommunication Union, Standardization Sector), os requisitos percebidos pelos usuários das redes devem seguir o conceito de QoS (Qualidade de Serviço), que são exigências para cada tipo de serviço na rede[3].

Atualmente, a transmissão de dados pela Internet não tem requisitos explícitos e funciona pelo Best Effort (melhor esforço), ou seja, se tenta fazer o melhor que é possível, já que a Internet não tem aparato para prover qualidade de serviço (QoS)[4]. Contudo, essa forma de funcionamento não é condizente com os novos tipos de serviços que possuem requisitos severos de QoS, relacionados sobretudo ao atraso e à largura de banda. Há outras formas de prover QoS na Internet que não foram inicialmente concebidas, ou seja, utilizando DiffServ ou Integrated Services, mas mesmo estas dependem de análise do desempenho da rede [5]. Assim sendo, é fundamental a verificação dos requisitos de qualidade (atraso, perda e jitter), além de verificar se a largura de banda oferecida está de acordo com as necessidades do usuário em cada tipo diferente de aplicação.

O desenvolvimento ao longo dos anos de ferramentas matemáticas e de computação e o seu uso para simulações de redes de comunicação mostraram ser uma forma importante de analisá-las e de realizar futuros aprimoramentos da qualidade de serviço (QoS) desses sistemas. Isso é devido a que os modelos analíticos para modelagem das redes têm limitações para avaliar sistemas com distribuições de probabilidade gerais, que não as distribuições exponenciais devido à sua característica de falta de memória. Podemos dizer que atualmente não é possível a análise de desempenho de sistemas de comunicação sem o uso desse tipo de software científico.

Portanto, neste projeto serão abordados métodos de análises e simulações de redes de comunicação com a assistência dos softwares Arena e Python. O software Arena pode servir como referência para comparar e/ou validar o Python que vem ganhando destaque no meio acadêmico.

2. Metodologia

O estudo a seguir apresenta uma análise inicial de uma rede com serviços de streaming e elásticos, com o objetivo de avaliar a viabilidade da linguagem de

programação Python como ferramenta para a análise de redes mais complexas. A proposta é compará-la a softwares de alta performance, como o Arena. Ao longo do projeto serão apresentados modelos de política de serviço para tratamento dos diversos dados (stream, sensíveis a atrasos, e elásticos, sensíveis a perda) e do dimensionamento da rede, utilizando alguns CODECS (Codificadores-Decodificadores). Essa comparação visa analisar a possibilidade de utilizar o Python em ambientes de simulação, em substituição a softwares específicos de simulação.

2.1. Medidas Utilizadas na Rede

Para a simulação da Rede, foi considerado uma rede com 1 Gbps de banda, com um fluxo de dados em Mbps que circulam na rede, chegando a um valor máximo de 433,904 Mbps (340 Mbps para o serviço elástico e 93,904 Mbps para o stream), sendo equivalente a aproximadamente 43,4 % do total suportado pela rede.

Esses valores, juntamente com as taxas dos codificadores para os serviços *stream*, foram utilizados para a criação do modelo de simulação. Podendo ser analisados, por exemplo, os bloqueios e os atrasos de pacote (elástico).

2.2. Apresentação dos Codificadores

Entre os diversos codificadores presentes atualmente, foram selecionados dois codificadores (um codificador de voz e um codificador de vídeo) bastante utilizados no mercado: H.264 e G.711. Esses codificadores, juntamente com os dados coletados na rede, foram utilizados na criação de modelos de simulação nos ambientes ARENA e Python, nos quais foram analisadas variáveis como a quantidade de bloqueios, variação do atraso, entre outras características.

2.3. Cálculo da largura de Banda

Após os estudos sobre a rede, foi realizado o dimensionamento da mesma. Para ilustrar a representação do tráfego elástico, utilizou-se o serviços de vídeo sob demanda. Paralelamente, para representar o tráfego de streaming, foram utilizados os serviços de VoIP e videoconferência. Com base nos dados medidos e nas informações sobre os codificadores utilizados, foram realizados uma série de cálculos para a montagem do ambiente de simulação.

3. Resultado e Discussão

A Figura 1 mostra o modelo para o tráfego de streaming e elástico, representando a rede criada no software Arena.

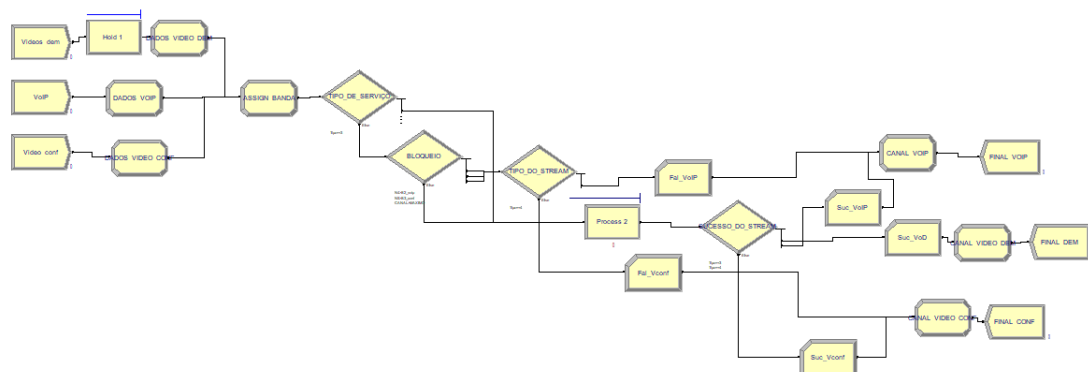


Figura 2. Simulação no software Arena

A Tabela 1 a seguir apresenta uma comparação dos resultados obtidos por meio do software Arena e da linguagem Python.

Tabela 1 - Comparação entre o python e o Arena

Nome	Arena	Python	Variação do Arena para o python (%)
Canal (média)	359000000	375246178	4,53
Canal (máximo)	492880000	493200000	0,06
Canal (mínimo)	0	10000000	-
Clientes que entraram (demanda)	20146	20358	1,05
Clientes atendidos (demanda)	20112	20332	1,09
Tempo médio de espera (demanda)	0,00000028579	0	-100
Tempo médio de permanência (demanda)	0,00027806	0,02	7093,44
Clientes que entraram (conf)	4	4	0
Clientes atendidos (conf)	1	1	0
Clientes bloqueados (conf)	0	0	0
Tempo médio de permanência (conf)	2,415	5,34	121,17

Clientes que entraram (voip)	161	160	-0,62
Clientes atendidos (voip)	118	111	-5,93
Clientes bloqueados (voip)	0	24	-
Tempo médio de permanência (voip)	1,947	2,38	22,23

Podemos observar que o resultado dos serviços em Python ficou relativamente próximo ao obtido no Arena. Essa pequena diferença em algumas variáveis ocorre porque, na simulação no Arena, há múltiplos serviços compartilhando o canal, o que influencia o comportamento global, afetando tanto a taxa de bloqueio quanto o tempo de permanência do serviço no sistema. Já no Python, a simulação considera apenas um serviço em execução, resultando em um cenário menos complexo.

4. Conclusões

Dessa forma, o Python demonstrou ser uma ferramenta robusta para a construção e análise de ambientes de simulação. Por ser uma linguagem em constante evolução e amplamente utilizada, é provável que, nos próximos anos, se torne ainda mais capaz de realizar simulações complexas, consolidando-se como uma alternativa poderosa a softwares tradicionais como o Arena.

5. Referências Bibliográficas

- [1] “Models for Dimensioning IP Wireless Links: Applying Analytical-Simulation with Incremental Validation”, URSINI, Edson Luiz.
- [2] L. M. da S, “Métodos para Avaliação das Técnicas de Policiamento de Tráfego e dos Codificadores/Decodificadores Visando o Dimensionamento de Redes de Dados Stream”, Iniciação científica realizada na UNICAMP, agosto/2016.
- [3] Parente, Eduardo. “Padrões ITU-T de QoS para Redes Baseadas em IP”. Universidade Federal do Paraná, 2003. Disponível em: <http://www.cricte2004.eletrica.ufpr.br/edu/anterior/cd03/trab1/gerson/ITU_T%20QoS%20em%20IP.htm>. Acessado em: 16 de outubro de 2024.
- [4] “Redes WiMAX: Qualidade de Serviço (QoS)”. Telecom, 2017. Disponível em: <https://www.teleco.com.br/tutoriais/tutorialredeswimax/pagina_2.asp>. Acessado em: 25 de novembro de 2024.
- [5] “Planejamento de enlaces IP multi-serviços, considerando requisitos de QoS”, URSINI, Trindade, Marcos.
- [6] Código para geração dos resultados em python: <<https://docs.google.com/document/d/1rBKEGkiUITtD98mSxjfoKsDklMFsKWCusg2rQ07cvzs/edit?tab=t.0>>.