



BALANCEAMENTO DE CARGA EM ARQUITETURA SDN USANDO SWITCH OPENFLOW PARA APLICAÇÕES DE ENTREGAS OTIMIZADAS

Jean P. O. da SILVA¹; Guilherme C. ARAUJO²; Kleber M. S. REZENDE³

RESUMO

Este trabalho apresenta uma proposta de arquitetura baseada em SDN (Software Defined Networking) voltada para o balanceamento de carga e controle de fluxo em aplicações de entrega otimizadas. A solução em desenvolvimento utiliza switches compatíveis com o protocolo OpenFlow e o controlador Ryu, com foco na atuação direta sobre as tabelas de fluxo dos dispositivos de rede. O objetivo é desenvolver um algoritmo próprio de redirecionamento de tráfego, inspirado nos princípios do PEWMA (Probabilistic Exponentially Weighted Moving Average), mas adaptado ao ambiente de emulação. O desenvolvimento é realizado no emulador Mininet, onde diferentes cenários são construídos para avaliar o comportamento da rede diante de variações no tráfego.

Palavras-chave: PEWMA; Controle de fluxo; Ryu;

1. INTRODUÇÃO

O crescimento exponencial do tráfego de rede, impulsionado por aplicações que exigem alta disponibilidade e baixa latência, como serviços de entrega de conteúdo e IoT, tem desafiado a eficiência das redes tradicionais (Cisco, 2020). Nessas arquiteturas, o plano de controle está rigidamente acoplado ao plano de dados, limitando a adaptação dinâmica às condições da rede e impactando negativamente o balanceamento de carga e a qualidade do serviço.

As Redes Definidas por Software (SDN) surgem como solução inovadora, separando logicamente o plano de controle do plano de dados. O controle é deslocado para um aplicativo externo, o controlador, que pode implementar decisões centralizadas para ajustar, em tempo real, rotas e regras de fluxo nos dispositivos de rede.

Este trabalho utiliza o controlador Ryu, que permite criar aplicações em Python com controle direto das tabelas de fluxo de switches OpenFlow. Inspirado no algoritmo PEWMA (Probabilistic Exponentially Weighted Moving Average), propõe-se desenvolver uma estratégia própria de balanceamento de carga que, embora não implemente diretamente o PEWMA, adota sua lógica adaptativa para atualizar dinamicamente as regras de fluxo, buscando distribuir melhor o tráfego, reduzir sobrecargas e melhorar o desempenho da rede (Odoh, 2022).

¹Discente de Tecnologia em redes de Computadores, IFSULDEMINAS – Campus Inconfidentes. E-mail: jean.pietro@alunos.ifsuldeminas.edu.br.

²Discente de Tecnologia em redes de Computadores, IFSULDEMINAS – Campus Inconfidentes. E-mail: guilherme.coelho@alunos.ifsuldeminas.edu.br.

³ Orientador, IFSULDEMINAS – Campus Inconfidentes. E-mail: kleber.rezende@ifsuldeminas.edu.br.

2. FUNDAMENTAÇÃO TEÓRICA

As Redes Definidas por Software (SDN) se baseiam na separação entre os planos de controle e dados, permitindo maior flexibilidade e controle centralizado (Valente, 2023). Essa arquitetura é útil em ambientes com demandas dinâmicas de tráfego e requisitos de QoS. O protocolo OpenFlow é o principal elo entre o controlador SDN e os dispositivos de rede, viabilizando o controle direto das tabelas de fluxo (Hussain, 2022).

O Ryu, controlador SDN em Python, é amplamente usado em pesquisas por oferecer uma API de alto nível para aplicações como monitoramento e balanceamento de carga (Montazerolghaem, 2025). O balanceamento de carga visa distribuir o tráfego entre múltiplos caminhos para melhorar o desempenho da rede, podendo ser feito por algoritmos estáticos ou dinâmicos (Islam, 2017). Este trabalho propõe uma solução própria inspirada no PEWMA, um método adaptativo de previsão de tráfego (Odoh, 2022).

3. MATERIAL E MÉTODOS

Este trabalho utiliza o emulador Mininet para criar topologias virtuais e testar três cenários distintos: (1) rede tradicional com balanceamento estático, (2) rede SDN com algoritmo simples (ex: round-robin no Ryu) e (3) rede SDN com algoritmo baseado nos princípios do PEWMA.

No primeiro cenário, foi necessário configurar os hosts como roteadores Linux com `ip_forward=1`, possibilitando uma topologia redundante com múltiplos caminhos, o que não seria viável com switches tradicionais devido ao STP, que desativa enlaces redundantes. A topologia, composta por quatro roteadores e quatro hosts finais (dois clientes e dois servidores — FTP e Web), foi configurada com rotas manuais e mesma métrica de prioridade para balanceamento, conforme ilustrado na Figura 1.

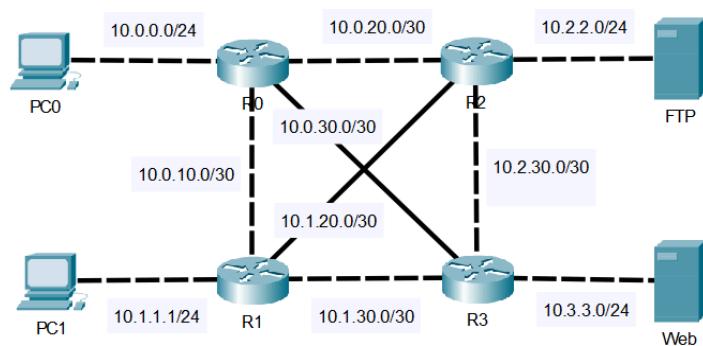


Figura 1 - Topologia Cenário 1 - Rede tradicional com balanceamento estático.

Os testes empregaram as ferramentas `iperf` para gerar tráfego TCP intenso e `ipstat` para monitorar o uso dos enlaces. O script de configuração foi desenvolvido em Python com a API do Mininet e realiza toda a configuração da topologia, endereços IP e rotas. Os serviços FTP e HTTP

são ativados com *pyftplib* e *http.server*, respectivamente. O próximo passo será a implementação dos cenários SDN usando o controlador Ryu e switches OpenFlow, onde será testado o algoritmo proposto inspirado no PEWMA.

4. RESULTADOS E DISCUSSÃO

Para a análise de desempenho da rede, foram consideradas três métricas principais: **taxa de transferência (MBps)**, que indica o volume de dados transferidos em cada conexão; **bitrate (Mbit/s)**, que representa a taxa de transmissão efetiva e a utilização da largura de banda; e **retransmissões TCP**, que evidenciam perdas e possíveis congestionamentos nos enlaces.

Os testes de desempenho realizados nos dois clientes (PC0 – Figura 2 e PC1 – Figura 3), acessando simultaneamente o servidor Web, evidenciaram diferenças marcantes na distribuição de tráfego entre as conexões.

Transferência PC0 para Servidor WEB Transferência PC1 para Servidor WEB

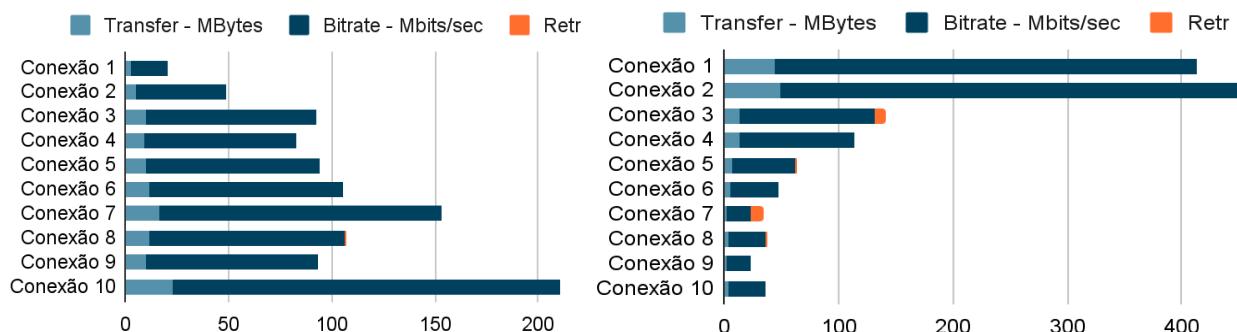


Figura 2 - Transferência PC0 para Servidor Web

Figura 3 - Transferência PC1 para Servidor Web

Nos testes realizados no ambiente tradicional com rotas estáticas e métricas iguais, observou-se um comportamento distinto entre os dois clientes. O **PC0** apresentou distribuição relativamente equilibrada do tráfego entre as conexões, com taxas de transferência entre **8 e 12 MBps**, bitrate com média de **83,5 Mbit/s** e **com apenas uma retransmissão**, indicando bom uso dos enlaces disponíveis.

Por outro lado, o **PC1** demonstrou **desequilíbrio significativo**, com as duas primeiras conexões consumindo quase toda a largura de banda disponível (**acima de 300 Mbit/s**), enquanto as demais tiveram tráfego mínimo, criando uma média de **48,25 Mbit/s**. A partir da terceira conexão surgiram **retransmissões significativas**, sugerindo **congestionamento** em alguns enlaces.

Esses resultados evidenciam que, mesmo com rotas de mesma prioridade, a rede convencional **não realiza um balanceamento eficiente**, especialmente sob múltiplas conexões simultâneas. Isso reforça a necessidade de soluções mais inteligentes, como o uso de **SDN com algoritmos adaptativos**, capazes de reagir em tempo real ao comportamento da rede, otimizando o uso dos caminhos e evitando sobrecargas.

5. CONCLUSÃO

Os testes no ambiente tradicional mostraram que o balanceamento de carga estático não consegue lidar bem com o tráfego dinâmico, resultando em congestionamentos e uso desigual dos enlaces. Isso destaca a limitação das redes legadas e a necessidade de soluções mais inteligentes.

Nosso projeto, em desenvolvimento, usa SDN com controlador Ryu e switches OpenFlow para criar um algoritmo adaptativo inspirado no PEWMA, que promete distribuir melhor o tráfego e evitar sobrecargas. A implementação dos cenários SDN ainda está em andamento, mas espera-se que a abordagem traga melhorias significativas no desempenho da rede.

REFERÊNCIAS

CISCO. *Cisco Annual Internet Report (2018–2023) White Paper*. San Jose: Cisco, 2020. Disponível em:

<https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>. Acesso em: 16 jul. 2025.

HUSSAIN, Imran S.; YESVANTH, R.; YUVARAJAPATHI, V. *Traffic analysis in software defined network using OpenFlow protocol*. In: 2022 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI), Chennai, India. Anais [...]. IEEE, 2022. DOI: 10.1109/ACCAI53970.2022.10370398. Disponível em:

<https://ieeexplore.ieee.org/document/10370398>. Acesso em: 17 jul. 2025.

ISLAM, Shafinaz; WANG, Y.; GUO, X.; LIU, Y. A Load Balancing Strategy of SDN Controller Based on Round Robin Algorithm. *arXiv preprint arXiv:1710.06957*, 2017. Disponível em: <https://arxiv.org/pdf/1710.06957>. Acesso em: 17 jul. 2025.

MONTAZEROLGHAEM, Ahmadreza; IMANPOUR, Somaye. *Evaluation and Performance Analysis of the Ryu Controller in Various Network Scenarios*. arXiv, 25 maio 2025. Disponível em: <https://arxiv.org/abs/2505.19290> . Acesso em: 17 jul. 2025.

ODOH, K. *Real-time anomaly detection for multivariate data streams*. arXiv preprint arXiv:2209.12398, 2022. Disponível em: <https://arxiv.org/pdf/2209.12398.pdf>. Acesso em: 17 jul. 2025.

VALENTE, F. F. A.; SILVA JÚNIOR, I. W. Da. *Redes Definidas por Software (SDN): Uma Análise sobre as Tecnologias de SDN, suas Vantagens e Desafios, e sua Aplicação em Redes de Computadores*. Revista FT, v. 27, n. 127, out. 2023. Disponível em: <https://revistaft.com.br/redes-definidas-por-software-sdn-uma-analise-sobre-as-tecnologias-de-sdn-suas-vantagens-e-desafios-e-sua-aplicacao-em-redes-de-computadores/> . Acesso em: 17 jul. 2025.