

Comparação de desempenho entre TCP Flavors Reno, Cubic e BBR

Ryan B. de Leon¹

1

Abstract. *This article presents a comparative analysis of network bandwidth usage among different TCP flavors, which are responsible for controlling congestion and bandwidth utilization. The study involves tests conducted using specific TCP flavors, all connected to the same network and communicating with the same server. The methodology includes measuring metrics such as throughput, bandwidth usage efficiency, latency, and the impact of congestion in controlled scenarios. The results allow for an evaluation of the relative performance of each flavor under similar conditions, identifying their advantages and limitations in terms of stability, fairness, and adaptability to network variations.*

Resumo. *Este artigo apresenta uma análise comparativa do uso da banda de rede entre diferentes flavors do algoritmo TCP, responsáveis por controlar o congestionamento e a utilização da largura de banda. O estudo envolve testes realizados utilizando um flavor específico do TCP, todos conectados à mesma rede e estabelecendo comunicação com o mesmo servidor. A metodologia inclui a medição de métricas como taxa de transferência, eficiência no uso da banda, latência e impacto do congestionamento em cenários controlados. Os resultados obtidos permitem avaliar o desempenho relativo de cada flavor em condições semelhantes, identificando suas vantagens e limitações em termos de estabilidade, equidade e adaptação a variações na rede.*

1. Introdução

O controle de congestionamento é um dos pilares fundamentais para o desempenho eficiente das redes de computadores. No contexto do protocolo de controle de transmissão (TCP), diversos algoritmos, conhecidos como “flavors”, foram desenvolvidos ao longo dos anos para otimizar a utilização da largura de banda, minimizar a latência e garantir a estabilidade da rede. Este trabalho tem como objetivo analisar comparativamente o desempenho de três desses flavors – Reno, Cubic e BBR – em cenários controlados, buscando compreender suas vantagens, limitações e impactos em aplicações que dependem de comunicação em rede.

A escolha deste tema se justifica pela relevância crescente da eficiência no uso da banda de rede em aplicações modernas, como transmissão de vídeo, serviços em nuvem e sistemas distribuídos. Cada flavor de TCP possui características específicas que o tornam mais ou menos adequado a determinados cenários de rede. A comparação entre Reno, Cubic e BBR permite identificar como cada algoritmo lida com condições variáveis de congestionamento e como suas diferenças podem influenciar o resultado final de uma aplicação.

Os flavors de TCP são variantes do algoritmo de controle de congestionamento, cada um implementando estratégias distintas para gerenciar a janela de transmissão –

o mecanismo que regula a quantidade de dados enviados antes de receber confirmação (ACK) do destinatário. A janela de transmissão é ajustada dinamicamente, com base em fatores como perda de pacotes e tempo de resposta, sendo um componente crucial para o desempenho e a estabilidade do TCP.

O TCP Reno introduziu conceitos como "slow start" e redução exponencial da janela de congestionamento em caso de perdas. É eficiente em redes com baixa latência e congestionamento moderado, mas enfrenta limitações em redes de alta capacidade ou longa distância.

O TCP Cubic é otimizado para redes modernas com alta largura de banda e latência. Ele utiliza uma função cúbica para o crescimento da janela de congestionamento, melhorando a eficiência em ambientes com grandes capacidades.

O TCP BBR adota uma abordagem inovadora ao medir diretamente a largura de banda disponível e o RTT para ajustar sua taxa de envio. Isso o torna eficiente em redes congestionadas ou com condições variáveis.

A influência desses algoritmos no desempenho de aplicações pode ser significativa. O controle inadequado do congestionamento pode levar a perdas de pacotes, aumento de latência e ineficiência no uso da banda, impactando diretamente a experiência do usuário e a qualidade dos serviços. Este trabalho busca explorar como Reno, Cubic e BBR se comportam em condições similares, oferecendo uma base para a escolha do algoritmo mais adequado para diferentes aplicações e cenários de rede.

2. Revisão Sistemática da Literatura

A revisão sistemática da literatura (RSL) terá como foco compreender detalhadamente o funcionamento dos algoritmos de controle de congestionamento do protocolo TCP, analisando como cada algoritmo opera, suas características principais, e os fatores que influenciam seu desempenho. O objetivo é identificar as abordagens empregadas por cada algoritmo para lidar com problemas como detecção de congestionamento, ajuste dinâmico da janela de transmissão, e recuperação de perdas, além de explorar os cenários em que cada algoritmo apresenta melhor desempenho. Essa revisão busca sintetizar o conhecimento existente na literatura e possíveis aplicações práticas de cada abordagem.

2.1. Etapas da RSL

A Revisão Sistemática da Literatura (RSL) é uma metodologia rigorosa e estruturada utilizada para identificar, avaliar e sintetizar de forma abrangente os estudos já realizados sobre um determinado tema. Esse método permite a compilação de evidências existentes de maneira organizada e imparcial, oferecendo uma visão clara e consolidada do estado da arte em relação à questão investigada. Além disso, a RSL garante a transparência do processo de busca e seleção de trabalhos, minimizando vieses e assegurando que as conclusões sejam baseadas em dados sólidos e representativos.

A seguir, cada etapa do processo de condução da RSL será explicada de forma detalhada, contemplando desde a formulação da pergunta de pesquisa até a extração e análise dos dados.

2.2. Etapa A

O passo A da Revisão Sistemática da Literatura (RSL) consistiu na definição da questão de pesquisa, estabelecendo os objetivos relacionados ao tema. A formulação da pergunta de pesquisa foi fundamental para orientar a seleção das palavras-chave utilizadas nas buscas nas bases de dados. A questão principal definida foi: Comparação de desempenho entre TCP Flavors.

O objetivo deste artigo é realizar uma análise comparativa do comportamento de diferentes algoritmos de controle de congestionamento do protocolo TCP (comumente denominados "Flavors") em um mesmo ambiente de hardware. Busca-se investigar como essas variações de algoritmos impactam o desempenho em cenários específicos, considerando diferentes tipos de aplicações e condições de rede. Por meio dessa análise, pretende-se identificar as vantagens e limitações de cada algoritmo em termos de eficiência no uso de banda, estabilidade, latência e adaptabilidade, fornecendo subsídios para a escolha adequada de algoritmos em aplicações específicas.

Como critérios de inclusão, foram considerados artigos de acesso aberto e que abordassem temas próximos ao objeto de estudo. Por outro lado, os critérios de exclusão adotados incluíram a ausência do arquivo em formato .pdf disponível para consulta.

2.3. Etapa B

O passo B da Revisão Sistemática da Literatura (RSL) consistiu na elaboração de uma planilha contendo as palavras-chave de busca, além da definição dos parâmetros de pesquisa. As buscas foram configuradas para incluir artigos de qualquer período ou aqueles publicados a partir de 2023, conforme ilustrado na Figura a seguir.

Chave de busca	Qualquer período	2023
Google Academico		
TCP Reno vs TCP Cubic	3780	225
"congestion control algorithm" AND "bandwidth"	14.200	650
"congestion control algorithm" AND "bandwidth utilization" AND ("reno" OR "cubic" or bbr")	374	66
ACM Digital Library		
"TCP congestion control"	1206	53
"TCP congestion control" AND ("Reno" OR "Cubic" OR "BBR")	519	38
"TCP algorithms" OR "comparision reno cubic bbr"	150	8
IEEE Xplore		
TCP Flavors	61	1
"TCP congestion control"	782	58
Performance Analysis of TCP Congestion Control Algorithms	412	27

Figura 1. Resultados das buscas genéricas.

Essas buscas desempenharam um papel fundamental na filtragem dos artigos relevantes, utilizando palavras-chave alinhadas ao tema de estudo. Os mecanismos de busca empregados, conforme ilustrado na Figura, incluíram o Google Acadêmico, a ACM Digital Library e a IEEE Xplore.

2.4. Etapa C

O passo C da Revisão Sistemática da Literatura (RSL) foi dividido em várias etapas. Inicialmente, foram selecionados dez títulos em cada um dos três buscadores

utilizados, totalizando trinta títulos escolhidos, conforme Figura 2.

Id	Títulos	Selecionado?
1	Fairness Improvement of High Speed TCP Congestion Control Algorithm	S
2	Improving performance of TCP over wireless networks	N
3	A Survey of TCP Congestion Control Algorithm	N
4	A Comparative Performance Analysis of TCP Congestion Control Algorithms: Newreno, Westwood, Veno, BIC, and Cubic	S
5	TCP congestion control algorithm research	S
6	TCP Fairness Among Modern TCP Congestion Control Algorithms Including TCP BBR	S
7	Mechanism of Cyclic Performance Fluctuation of TCP BBR and CUBIC TCP Communications	S
8	Performance Analysis of TCP Congestion Control Algorithms	S
9	TCP congestion control algorithms and a performance comparison	S
10	Cross-Layer Optimization to Maximize Fairness Among TCP Flows of Different TCP Flavors	S
11	INET framework extensions for TCP Vegas and TCP Westwood: poster abstract	N
12	Performance Evaluation of TCP Congestion Control Algorithms in Data Center Networks	N
13	Identification of TCP Congestion Control Algorithms from Unidirectional Packet Traces	N
14	Research on Adaptive Congestion Avoidance Algorithm Based on RTT Congestion Signal	N
15	Congestion Control for Cross-Datacenter Networks	N
16	A Low Latency Congestion Control That Can Compete	N
17	Simulation-based comparisons of Tahoe, Reno and SACK TCP	S
18	An analytical model for evaluating utilization of TCP Reno	S
19	Comparative analysis of TCP congestion control mechanisms	S
20	Performance evaluation and comparison of Westwood+, New Reno, and Vegas TCP congestion control	S
21	TCP congestion control algorithm for heterogeneous Internet	N
22	A BBR-based congestion control for delay-sensitive real-time applications	S
23	TCP BBR congestion control optimization algorithm for high fairness and low retransmission	S
24	Improvement of RTT Fairness Problem in BBR Congestion Control Algorithm by Gamma Correction	N
25	Optimized Congestion Control Algorithm for QUIC in Wireless Networks: CubicBytes-N Algorithm	S
26	Performance Study of Improved BBR Congestion Control Algorithm using QUIC in Wireless LAN	S
27	BBR: Congestion-Based Congestion Control: Measuring bottleneck bandwidth and round-trip propagation time	S
28	Performance analysis of modern TCP variants: A comparison of Cubic, Compound and New Reno	S
29	CUBIC: a new TCP-friendly high-speed TCP variant	S
30	A Comprehensive Study of Modern and High Speed TCP-Variant in Linux Kernel: TCP CUBIC	S

Figura 2. Artigos selecionados pelo título.

A etapa seguinte consiste na seleção de pelo menos vinte artigos, cujos resumos foram lidos e avaliados para a próxima etapa.

Id	Títulos	Revista	Conferência	Ano de publicação	Primeiro autor
1	Fairness Improvement of High Speed TCP Congestion Control Algorithm		International Conference on Computational and Information Sciences	2013	Zhou Chi-Min
2	A Comparative Performance Analysis of TCP Congestion Control Algorithms: Newreno, Westwood, Veno, BIC, and Cubic		ICSC	2020	Sanjeev Patel
3	TCP congestion control algorithm research		ICIT2012	2012	Xiaoling Sun
4	TCP Fairness Among Modern TCP Congestion Control Algorithms Including TCP BBR		CloudNet	2018	Kanon Sasaki
5	Mechanism of Cyclic Performance Fluctuation of TCP BBR and CUBIC TCP Communications		COMPSAC	2020	Kouto Miyazawa
6	Performance Analysis of TCP Congestion Control Algorithms	INTERNATIONAL JOURNAL OF COMPUTERS AND COMMUNICATIONS			
7	TCP congestion control algorithms and a performance comparison	Proceedings: Tenth International Conference on Computer C		--	Yuan-Cheng Lai
8	Cross-Layer Optimization to Maximize Fairness Among TCP Flows of Different TCP Flavors		GLOBECOM	2008	Toktam Mahmoodi
9	Simulation-based comparisons of Tahoe, Reno and SACK TCP	ACM SIGCOMM		1996	K. Fall.
10	An analytical model for evaluating utilization of TCP Reno		International Conference on Computer Systems and Technologies	2007	Mehdi hassani
11	Comparative analysis of TCP congestion control mechanisms		International Conference on Networking, Information Systems & Security	2020	Bazj, K
12	Performance evaluation and comparison of Westwood+, New Reno, and Vegas TCP congestion control	ACM SIGCOMM		2004	Grieco, L
13	A BBR-based congestion control for delay-sensitive real-time applications	Computing		2020	Najmuddin, S.
14	TCP BBR congestion control optimization algorithm for high fairness and low retransmission		--	2013	Wansu Pan
15	Optimized Congestion Control Algorithm for QUIC in Wireless Networks: CubicBytes-N Algorithm	IEEE Internet of Things Journal		2024	Xiang Xiao
16	Performance Study of Improved BBR Congestion Control Algorithm using QUIC in Wireless LAN	International Conference on Ubiquitous Computing and Commun		2021	Mengjie Zuo
17	BBR: Congestion-Based Congestion Control: Measuring bottleneck bandwidth and round-trip propagation time	ACM SIGCOMM		2016	Neal Cardwell
18	Performance analysis of modern TCP variants: A comparison of Cubic, Compound and New Reno		Biennial Symposium on Communica	2010	Abdeljaouad, I
19	CUBIC: a new TCP-friendly high-speed TCP variant	ACM SIGCOMM		2008	Sangtae Ha
20	A Comprehensive Study of Modern and High Speed TCP-Variant in Linux Kernel: TCP CUBIC		The International Arab Journal of Information Technology	2019	Abbar Khan

Figura 3. Títulos selecionados.

Após a análise dos resumos, foram inicialmente selecionados dez artigos para uma leitura diagonal. Esse tipo de leitura consiste em revisar novamente o resumo, já avaliado na etapa anterior, seguida pela leitura da introdução, análise de imagens e tabelas, e, por fim, a leitura da conclusão. Com base nessa leitura, cinco artigos foram selecionados para

uma análise mais aprofundada. No entanto, ao realizar a leitura diagonal, constatou-se que os artigos escolhidos não eram suficientemente específicos ao tema proposto, o que exigiu a busca por novos artigos nas bases de dados. Dessa forma, os artigos selecionados para a leitura completa não haviam sido incluídos nas etapas anteriores, conforme ilustrado na Figura 4.

Id	Títulos	Revista	Conferência	Ano de publicação	Primeiro autor
1	Fairness Improvement of High Speed TCP Congestion Control Algorithm	-	nce on Computational and	2013	Zhou Chi-Min
2	A Comparative Performance Analysis of TCP Congestion Control Algorithms: Newren	-	ICSC	2020	Sanjeev Patel
3	TCP Fairness Among Modern TCP Congestion Control Algorithms Including TCP BBR	-	CloudNet	2018	Kanon Sasaki
4	Mechanism of Cyclic Performance Fluctuation of TCP BBR and CUBIC TCP Communications	-	COMPSAC	2020	Kouto Miyazawa
5	TCP congestion control algorithms and a performance comparison	-	Conference on Computer	--	Yuan-Cheng Lai
6	A BBR-based congestion control for delay-sensitive real-time applications	Computing	-	2020	Najmuddin, S.
7	Comparative analysis of TCP congestion control mechanisms	-	ce on Networking, Informat	2020	Bazi, K
8	Simulation-based comparisons of Tahoe, Reno and SACK TCP	ACM SIGCOMM	-	1996	K. Fall.
9	An analytical model for evaluating utilization of TCP Reno	-	rence on Computer System	2007	Mehdi hassani
10	Performance analysis of modern TCP variants: A comparison of Cubic, Compound an	-	pl Symposium on Commun	2010	Abdeljaouad, I

Figura 4. Artigos lidos de forma diagonal.

Com base nesse processo rigoroso de seleção, a Figura 5 apresenta de forma detalhada os artigos que foram considerados relevantes e, portanto, selecionados para a leitura completa.

Id	Títulos	Revista	Conferência	Ano de publicação	Primeiro autor
1	Fairness Improvement of High Speed TCP Congestion Control Algorithm	-	on Computational	2013	Zhou Chi-Min
2	A Comparative Performance Analysis of TCP Congestion Control Algorithms: Newreno, Westwood	-	ICSC	2020	Sanjeev Patel
3	TCP Fairness Among Modern TCP Congestion Control Algorithms Including TCP BBR	-	CloudNet	2018	Kanon Sasaki
4	Mechanism of Cyclic Performance Fluctuation of TCP BBR and CUBIC TCP Communications	-	COMPSAC	2020	Kouto Miyazawa
5	A BBR-based congestion control for delay-sensitive real-time applications	Computing	-	2020	Najmuddin, S.

Figura 5. Artigos selecionados para leitura completa.

2.5. Etapa D

O passo D da Revisão Sistemática da Literatura (RSL) consiste em ler integralmente todos os cinco artigos previamente selecionados. A partir dessa leitura aprofundada, deve-se desenvolver o trabalho com base no tema proposto, utilizando as informações e análises encontradas nas publicações. Este processo exige uma compreensão crítica dos estudos, buscando identificar as contribuições relevantes, metodologias empregadas, lacunas existentes e possíveis direções futuras para a pesquisa. O objetivo é elaborar uma discussão fundamentada e coerente, alinhada com a temática central da RSL, contribuindo para o avanço do conhecimento na área de estudo.

2.6. Resultados da RSL

Conforme o artigo [4], os autores avaliaram a justiça entre o TCP BBR e o TCP CUBIC em redes com elementos práticos e implementações reais de TCP. Eles identificaram que o TCP BBR apresenta baixa justiça em cenários de alta latência e que os métodos tradicionais de controle de congestionamento, como o descarte de pacotes, não são eficazes para lidar com o comportamento do TCP BBR, já que este não reduz o tamanho da janela de congestionamento ao detectar perdas de pacotes.

Os resultados do estudo indicaram que o TCP BBR consome uma parcela significativa da largura de banda, reduzindo a taxa de transferência de outros fluxos TCP. Como alternativa, os autores sugerem que métodos que afetam diretamente os parâmetros do TCP BBR, como a largura de banda do gargalo (BtlBw) e o tempo de propagação mínimo

(RTprop), podem ser mais eficazes para melhorar a justiça entre o TCP BBR e outros algoritmos TCP.

Segundo o artigo [3], o algoritmo TCP Reno foi avaliado em comparação com outros algoritmos de controle de congestionamento, como CUBIC e BIC. Os resultados mostraram que, embora o Reno seja eficaz em cenários de largura de banda moderada, ele não apresenta o mesmo desempenho em redes de alta largura de banda, como as redes longas e rápidas (long fat networks). Nessas condições, o Reno é superado por algoritmos mais avançados, como o CUBIC, que são projetados para otimizar a utilização da largura de banda disponível.

De acordo com o artigo [5], melhorias no algoritmo de TCP de alta velocidade reduziram a desigualdade no uso de largura de banda com o TCP Reno, promovendo maior equidade. Esses resultados reforçam a importância de adaptar algoritmos para equilibrar desempenho e justiça em redes compartilhadas.

Os artigos [1] e [2] ajudaram a compreender melhor o comportamento dos algoritmos de controle de congestionamento em diferentes cenários. O estudo de [1] destacou ajustes necessários no TCP BBR para melhorar a justiça na alocação de largura de banda.

Com isso, conclui-se que aprimorar algoritmos de controle de congestionamento é essencial para atender às demandas de equidade e eficiência em diversos cenários de rede.

3. Metodologia

O objetivo deste trabalho é analisar o comportamento dos algoritmos de controle de congestionamento TCP (Reno, Cubic e BBR) em diferentes cenários de tráfego, considerando métricas como throughput, latência e perda de pacotes. Para alcançar esse objetivo, foram realizadas simulações controladas em um ambiente experimental cuidadosamente configurado.

O experimento foi conduzido utilizando um hardware com especificações uniformes para garantir a comparabilidade dos resultados. O sistema operacional escolhido foi o Ubuntu 24.04, devido à sua ampla compatibilidade com ferramentas de análise de redes e suporte para a manipulação de algoritmos TCP.

A configuração de rede foi projetada para simular diferentes condições de tráfego, incluindo redes de baixa, média e alta latência, bem como cenários com diferentes níveis de largura de banda. Para isso, foi utilizado o comando tc (Traffic Control) do Linux, permitindo a aplicação de atrasos e controle de banda diretamente nas interfaces de rede.

3.1. Algoritmos TCP selecionados

TCP Reno: Um dos algoritmos mais antigos, amplamente utilizado como base para comparação. TCP Cubic: Algoritmo padrão em sistemas Linux modernos, otimizado para redes de alta largura de banda e alta latência. TCP BBR (Bottleneck Bandwidth and RTT): Um algoritmo mais recente que utiliza uma abordagem baseada em taxas para otimizar o throughput e minimizar a latência.

3.2. Ferramentas Utilizadas

Para a geração e controle do tráfego de rede, foi utilizada a ferramenta iPerf3, que permite a criação de fluxos de dados controlados e a medição precisa de métricas

como throughput e jitter. Além disso, os dados capturados durante os experimentos foram analisados utilizando o Wireshark, uma ferramenta robusta para a inspeção de pacotes e avaliação detalhada do comportamento dos algoritmos. Configuração dos Experimentos

3.3. Cenários de testes

Os testes foram realizados em diferentes cenários, ajustando os seguintes parâmetros:

Latência: Simulação de atrasos de 10 ms, 50 ms e 100 ms para representar redes locais, metropolitanas e de longa distância.

Largura de Banda: Configuração de limites de 10 Mbps, 50 Mbps e 100 Mbps para avaliar o desempenho em redes com capacidades variadas.

Cada teste foi repetido cinco vezes para garantir a consistência dos resultados e minimizar a influência de variações aleatórias. Métricas de Avaliação

A principal métrica analisada durante os experimentos foi o Throughput, ou seja, quantidade de dados transmitidos com sucesso por segundo.

Os dados obtidos foram armazenados em arquivos .csv e processados utilizando scripts em Python, que geraram gráficos e tabelas para a análise dos resultados.

3.4. Justificativas

A escolha dos algoritmos TCP avaliados se baseia na sua relevância e ampla utilização em redes modernas. Além disso, o uso de ferramentas como iPerf3 e Wireshark foi motivado por sua precisão e confiabilidade para a análise de redes.

3.5. Limitações

Embora os testes tenham sido realizados em um ambiente controlado, é importante destacar que a ausência de variáveis reais, como interferências externas e mobilidade, pode limitar a generalização dos resultados para cenários de produção.

4. Resultados

Os resultados obtidos nos experimentos demonstram o comportamento dos algoritmos TCP (Reno, Cubic e BBR) em diferentes cenários de largura de banda, considerando métrica como throughput. Os testes foram realizados utilizando o servidor de IP 41.226.22.119 (localizado na Tunísia e disponível na lista de servidores do iPerf3) para gerar tráfego. Cada experimento teve duração de 30 segundos, com coleta de dados suficiente para garantir a consistência das análises.

Os gráficos a seguir apresentam o comportamento dos três algoritmos nos cenários de largura de banda média de 10 Mbps, 50 Mbps e 100 Mbps. Cada gráfico compara os valores médios de throughput registrados para os algoritmos:

Figura 6: Comportamento dos algoritmos com banda média de 10 Mbps.

Figura 7: Comportamento dos algoritmos com banda média de 50 Mbps.

Figura 8: Comportamento dos algoritmos com banda média de 100 Mbps.

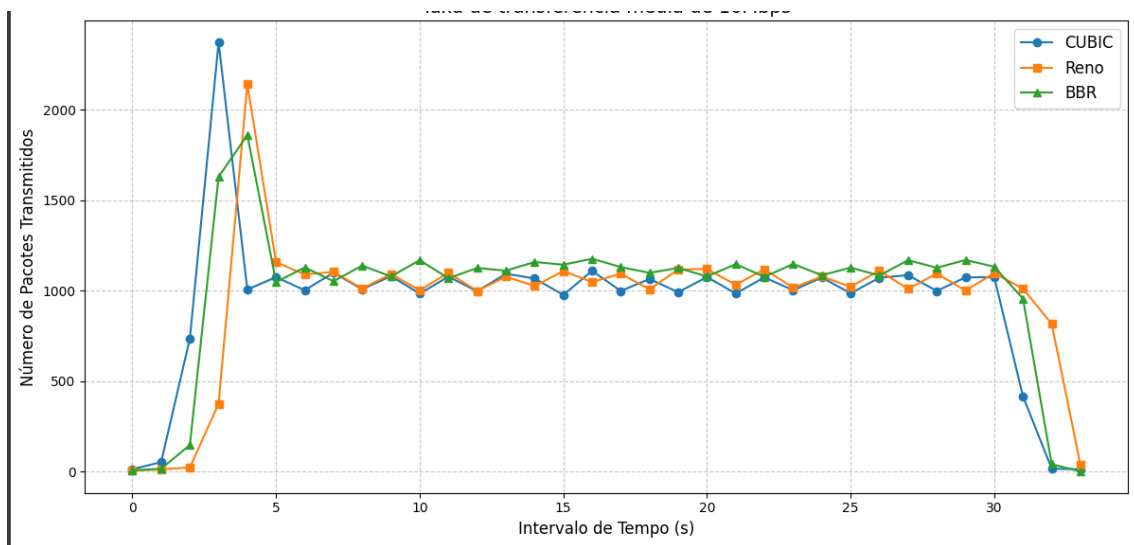


Figura 6. Comportamento dos algoritmos TCP (Reno, Cubic e BBR) em cenário de largura de banda média de 10 Mbps.

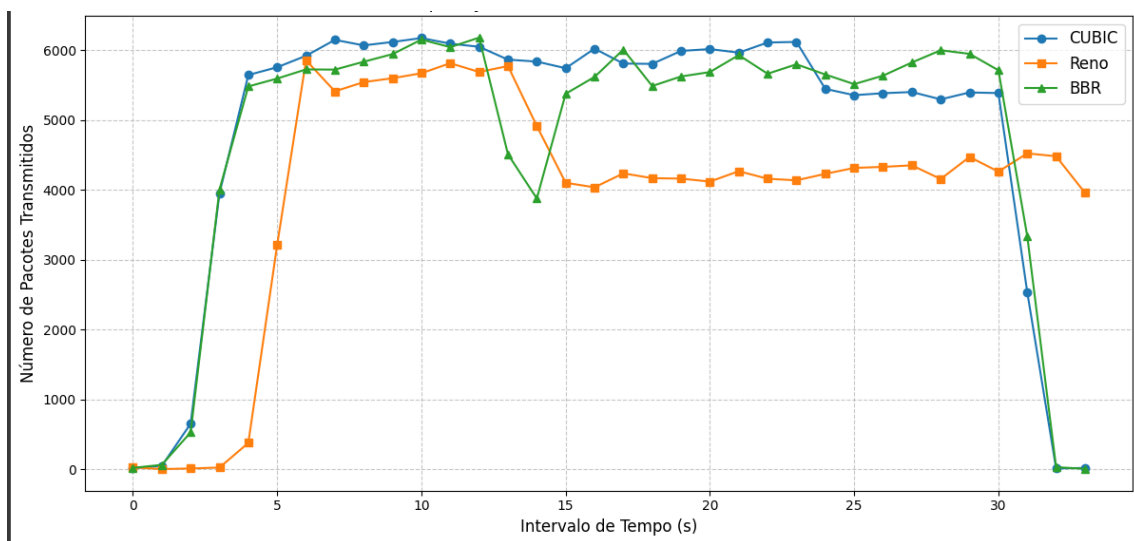


Figura 7. Comportamento dos algoritmos TCP (Reno, Cubic e BBR) em cenário de largura de banda média de 50 Mbps.

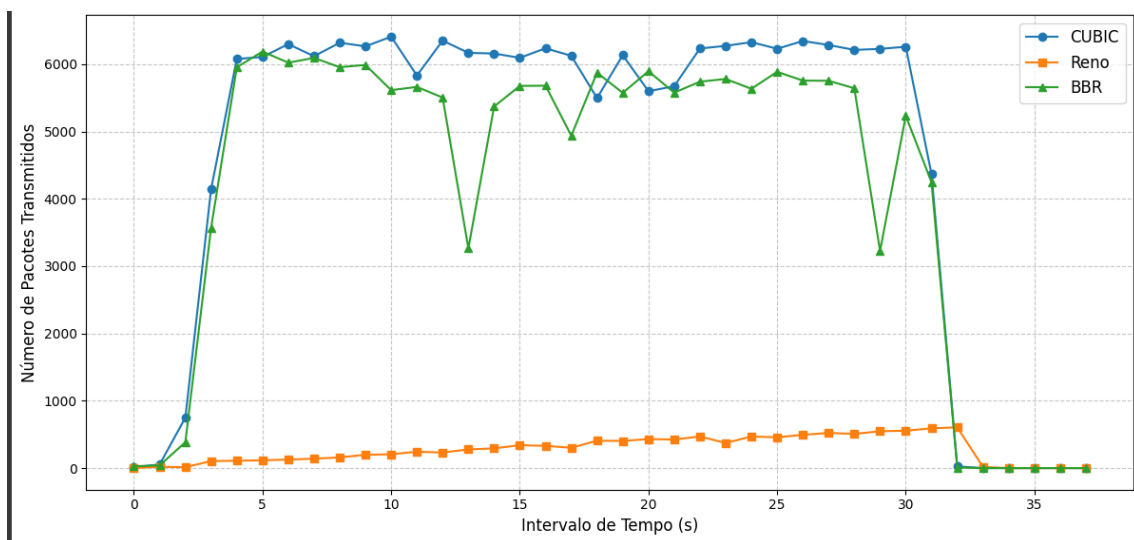


Figura 8. Comportamento dos algoritmos TCP (Reno, Cubic e BBR) em cenário de largura de banda média de 100 Mbps.

4.1. Análise dos Resultados

Nos cenários de baixa largura de banda (10 Mbps), o comportamento dos três algoritmos foi semelhante, com pequenas variações no throughput médio registrado. Isso indica que, em condições de valores em torno de 10Mbps, os algoritmos têm desempenho comparável.

À medida que a largura de banda aumenta (50 Mbps e 100 Mbps), o TCP Cubic e TCP BBR demonstraram um desempenho superior em termos de throughput, com vantagem do TCP Cubic, especialmente em redes com alta capacidade. Esse resultado está alinhado com o objetivo do algoritmo Cubic de otimizar a utilização de redes de alta largura de banda e alta latência.

Por outro lado, o TCP Reno apresentou limitações em cenários de maior largura de banda, refletindo sua abordagem mais conservadora ao controle de congestionamento. O TCP BBR apresentou resultados intermediários, com uma abordagem eficiente em redes de alta capacidade, mas sem superar o desempenho do Cubic.

4.2. Comportamento da Janela de Congestionamento (Cwnd)

Ao analisar o comportamento da janela de congestionamento (Cwnd) durante os experimentos, é possível observar que os algoritmos com melhor desempenho em termos de throughput, como o TCP Cubic e o TCP BBR, mantiveram a janela de congestionamento com valores significativamente mais elevados ao longo do tempo. Esse comportamento reflete a capacidade desses algoritmos de utilizar de forma mais agressiva a largura de banda disponível, ajustando a Cwnd de maneira dinâmica para otimizar a transferência de dados.

A Figura 9, com 100 Mbps, o TCP Cubic continuou a exibir a maior janela de congestionamento, refletindo sua agressividade em redes de alta capacidade. O TCP BBR, embora mais eficiente que o Reno, ainda não superou o Cubic em termos de Cwnd, embora tenha mostrado uma janela mais estável e adaptável ao longo do tempo.

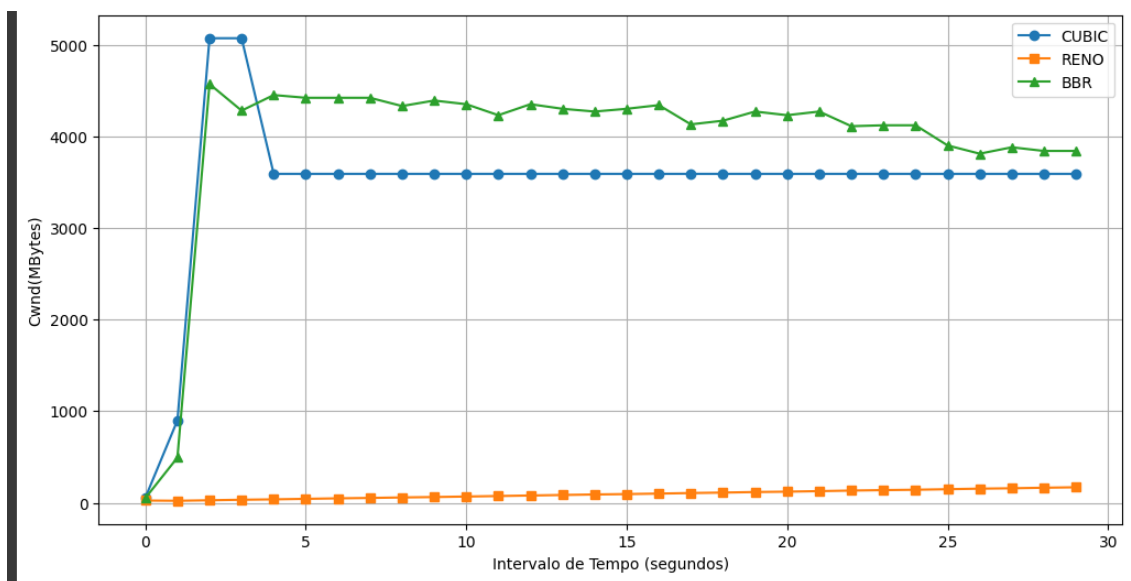


Figura 9. Janela de congestionamento em cenário de largura de banda média de 100 Mbps.

Esses comportamentos indicam que, em redes com maior largura de banda, algoritmos como o Cubic e o BBR têm um controle mais eficiente da janela de congestionamento, o que resulta em uma maior taxa de transferência de dados.

5. Conclusão

Os resultados obtidos neste estudo evidenciam as diferenças significativas no comportamento dos algoritmos TCP (Reno, Cubic e BBR) em diferentes cenários de largura de banda. Cada algoritmo apresentou características próprias em termos de throughput e comportamento da janela de congestionamento (Cwnd), refletindo suas respectivas abordagens para o controle de congestionamento.

Conforme [3], foi comprovado a eficácia do TCP Reno já que teve desempenho limitado em redes de alta largura de banda, com uma janela de congestionamento conservadora. É eficiente em redes de baixa a média largura de banda (10 Mbps e 50 Mbps) e em ambientes com alta latência, sendo simples e confiável. No entanto, não aproveita bem redes de alta capacidade.

Porém TCP Cubic se destacou em redes de alta largura de banda (50 Mbps e 100 Mbps), com uma janela de congestionamento que cresce rapidamente e maximiza a utilização da largura de banda. É ideal para ambientes de alta capacidade e alta latência, como datacenters, mas pode aumentar a variação no tempo de resposta, o que pode ser um problema em redes sensíveis à latência.

E por fim, O TCP BBR apresentou desempenho intermediário, com uma janela de congestionamento estável e adaptável, oferecendo um controle eficiente do congestionamento. Sua abordagem baseada em banda e RTT é vantajosa para redes modernas e de alta capacidade, mas ainda fica atrás do Cubic em redes de muito alta largura de banda, ainda mais, como mostra o artigo [4] que explica que apesar de bom desempenho pode ser injusto no consumo de banda com outros TCPs.

Em resumo, a escolha do algoritmo TCP ideal depende das características específicas da rede em questão, como a largura de banda disponível, a latência e a necessidade de eficiência na transferência de dados. O estudo reforça a importância de selecionar o algoritmo de controle de congestionamento mais adequado para garantir o melhor desempenho possível em diferentes cenários de rede.

Referências

- [1] Kouto Miyazawa, Saneyasu Yamaguchi, and Aki Kobayashi. Mechanism of cyclic performance fluctuation of tcp bbr and cubic tcp communications. In *2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 1–6, 2020.
- [2] Sayed Najmuddin, Muhammad Asim, Kashif Munir, Thar Baker, Zehua Guo, and Rajiv Ranjan. A bbr-based congestion control for delay-sensitive real-time applications. page 23, 2020.
- [3] Sanjeev Patel, Yash Shukla, Nikhil Kumar, Tejasv Sharma, and Kulgaurav Singh. A comparative performance analysis of tcp congestion control algorithms: Newreno, westwood, veno, bic, and cubic. In *2020 6th International Conference on Signal Processing and Communication (ICSC)*, pages 23–28, 2020.
- [4] Kanon Sasaki, Masato Hanai, Kouto Miyazawa, Aki Kobayashi, Naoki Oda, and Saneyasu Yamaguchi. Tcp fairness among modern tcp congestion control algorithms including tcp bbr. In *2018 IEEE 7th International Conference on Cloud Networking (CloudNet)*, pages 1–4, 2018.
- [5] Chi-Min Zhou. Fairness improvement of high speed tcp congestion control algorithm. In *Proceedings of the International Conference on Computational and Information Sciences (ICCIS)*, pages 1–4, Scottsdale, AZ, USA, 2013. IEEE.