

# Análise do impacto de um caminho degradado no desempenho da transmissão MPTCP

Mateus Alves Amorim, Conrado José Gusso Bozza e Eduardo Parente Ribeiro

**Resumo**—A heterogeneidade dos caminhos disponíveis nos dispositivos atuais, como redes WiFi, WiMAX e LTE, causa grande impacto no atraso e na taxa de transferência quando são empregados protocolos de múltiplos caminhos, como o *Multipath TCP*. Este artigo propõe uma análise do impacto na taxa de transferência de uma conexão MPTCP de um caminho degradado por perda de pacotes. Estuda-se um limiar da razão de perda de pacotes a partir do qual é melhor utilizar uma única conexão TCP pelo melhor caminho disponível. Este artigo descreve testes em um cenário de rede heterogênea comumente encontrado em dispositivos móveis, composto por uma conexão WiFi e outra de rede celular, no ambiente de simulação NS-3.

**Palavras-Chave**—Multi-caminhos, MPTCP, Taxa de transferência, Razão de perda de pacotes, NS-3.

**Abstract**—The heterogeneity of available paths in the current devices, such as Wi-Fi, WiMAX and LTE networks, has a great impact on delay and throughput when multipath protocols, such as *Multipath TCP*, are used. This paper proposes an analysis of the impact on the transfer rate of an MPTCP connection of a degraded path by the packet loss. A threshold of packet loss ratio from which it is best to use a single TCP connection with the best available path is studied. This paper describes tests in a heterogeneous network scenario commonly seen in mobile devices, consisting of a WiFi connection and a cellular network, in the NS-3 simulation environment.

**Keywords**—Multipath, MPTCP, Throughput, Packet loss ratio, NS-3.

## I. INTRODUÇÃO

A crescente facilidade de acesso à Internet por diferentes tecnologias, como por exemplo ADSL, fibra óptica, Wi-Fi, WiMax e redes de telefonia móvel, e o advento de dispositivos com múltiplas possibilidades de conexão à Internet, como por exemplo os *smartphones* e os *notebooks*, trouxeram uma gama de possibilidades de conectividade aos usuários finais. Essa característica é conhecida como *multihoming*. A Internet desde sua concepção vislumbrava a possibilidade de múltiplos caminhos para o envio dos dados, porém, do ponto de vista da camada de transporte, o modelo TCP/IP, que se tornou referência na Internet, estabelece um único fluxo de dados entre a origem e o destino, independente da disponibilidade de múltiplos caminhos.

O *Multipath TCP* (MPTCP) é uma extensão para o protocolo TCP proposta recentemente e atualmente implementada em muitos sistemas finais, como por exemplo os sistemas MAC OS X e IOS da empresa *Apple*. O seu desenvolvimento teve como base a compatibilidade com o TCP e com os

protocolos das camadas adjacentes. Para tanto, teve 3 (três) objetivos de compatibilidade estabelecidos [1]: 1) As aplicações devem utilizar a API de socket já existente para o TCP na utilização do MPTCP; 2) O protocolo deve ser compatível com a rede onde está sendo implantado, ou seja, com a Internet, podendo atravessar middleboxes de mercado, como *firewalls*, NATs e *proxies*; e 3) O protocolo deve manter a equidade entre os usuários da rede, coexistindo amigavelmente com outros fluxos TCP padrão em situações de gargalo.

Outro ponto importante dos protocolos de múltiplos caminhos, não só do MPTCP, é que a taxa de transferência agregada deve ser melhor do que a conseguida por um único fluxo TCP quando da utilização do melhor caminho dentre os disponíveis. Caso contrário, a sua utilização não apresenta benefício, a não ser a resiliência. Atualmente, esse é o maior desafio para os protocolos de múltiplos caminhos, pois a multiplicidade de caminhos disponíveis aos dispositivos finais também apresenta grande heterogeneidade nas suas características [2]. Redes heterogêneas, com diferentes valores de atraso, largura de banda e perda de pacotes impactam significativamente nas taxas de transferência e no atraso conseguidos pela aplicação que utiliza serviço de transporte confiável, pois necessita de reordenamento dos pacotes recebidos antes que sejam entregues à aplicação.

Neste contexto, nós propomos uma análise da influência de um caminho degradado pela razão de perda de pacotes (P) no desempenho da conexão MPTCP composta por dois caminhos, o melhor caminho e o caminho degradado, através de comparação da taxa de transferência de um fluxo TCP padrão pelo melhor caminho em relação a taxa de transferência de uma conexão MPTCP que utiliza os dois caminhos disponíveis. A razão de perda de pacotes do caminho degradado será variada para permitir definir o limiar de P a partir do qual é melhor utilizar somente o melhor caminho no cenário analisado.

O restante do artigo está organizado como segue. A seção II apresenta conceitos de protocolos de transporte. A seção III apresenta os protocolos de transporte multi-caminhos padronizados pelo IETF, apresenta o MPTCP, os principais problemas da transferência multi-caminhos e um estudo de alguns trabalhos relacionados. A seção IV descreve o ambiente de simulação do NS-3 utilizado, com os seus parâmetros de configuração, o cenário das simulações e o método utilizado para a realização da análise proposta. A seção V apresenta e discute os resultados das simulações. E por fim, a seção VI encerra com as conclusões.

## II. PROTOCOLOS DE TRANSPORTE

Os protocolos de transporte são responsáveis pela comunicação fim-a-fim de processos de hospedeiros diferentes, permitindo a comunicação inter-processos. Os principais protocolos da camada de transporte são o UDP e o TCP, definidos pelas RFC 768 [3] e RFC 793 [4], respectivamente.

O UDP é um protocolo simples não orientado à conexão, e por isso, não garante a entrega, a ordem ou a duplicidade dos dados transmitidos.

O TCP é um protocolo orientado à conexão que provê comunicação entre processos de hospedeiros diferentes com garantia de entrega em ordem e sem duplicidade dos dados transmitidos. Além disso, o TCP possui controle de congestionamento, o que permite que conexões TCP compartilhem com justiça um enlace congestionado. O algoritmo de controle de congestionamento mais tradicional e encontrado na maioria das implementações é o *New Reno*, especificado na RFC 6582 [5]. Este é um algoritmo de prevenção de congestionamento de Aumento Aditivo/Diminuição Multiplicativa (AIMD).

Considerando o comportamento da taxa de transferência da conexão TCP por consequência do controle de congestionamento, de acordo com a RFC 3819 [6], podemos calcular a taxa de transferência média da conexão TCP ( $R$ ) em função da razão de perda de pacotes ( $P$ ), do tempo de viagem de ida e volta (RTT), do *time-out* do pacote (RTO) e do tamanho máximo de segmento (MSS). Uma simplificação adicional é geralmente feita assumindo que RTO é aproximadamente  $5 \times$  RTT, como apresentado na equação da RFC 3819:

$$R = \frac{MSS}{RTT \cdot \sqrt{1.33P} + 5 \cdot RTT \cdot P \cdot (1 + 32P^2) \cdot \min(1, 3 \cdot \sqrt{0.75P})} \quad (1)$$

Para demonstrar o impacto da razão de perda de pacotes ( $P$ ) no desempenho da taxa de transferência média de uma conexão TCP, o gráfico da Fig. 1 apresenta o resultado da equação quando consideramos um cenário composto por 2 (dois) *host* conectados por um enlace com largura de banda de 10 Mbps e 10 ms de atraso (RTT igual a 20ms), quando variamos  $P$  de 0.001 até 0.1.

Pelo gráfico, quando o valor de  $P$  é de apenas 1%, o valor da taxa de transferência média é de aproximadamente 4.4 Mbps, o que representa apenas 44% da largura de banda disponível no enlace. Quando  $P$  é de 10%, o enlace está muito degradado, tornando a comunicação impraticável, demonstrando o grande impacto da razão de perda de pacotes na taxa de transferência efetiva da conexão TCP padrão.

## III. PROTOCOLOS DE TRANSPORTE MULTI-CAMINHOS

Muito embora o protocolo TCP seja amplamente utilizado na Internet, ele não foi projetado para ter suporte a transferências por multi-caminhos. Na verdade, o TCP estabelece uma conexão ponto-a-ponto com somente um par de endereços IP de origem e destino, não podendo ser associado a um novo endereço IP sem que a conexão seja reestabelecida. Para realizar essa tarefa é necessário um protocolo de transporte projetado especificamente para esse fim, ou seja, não basta ter um sistema multi-abrigado com múltiplas interfaces de acesso

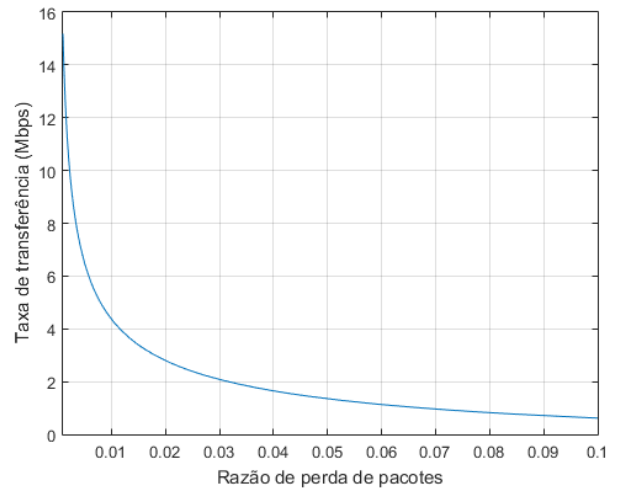


Fig. 1. Taxa de transferência média da conexão TCP em função da razão de perda de pacotes ( $P$ ), segundo a equação 1

à Internet disponíveis. Esse sistema multi-abrigado necessita de um protocolo para transferência por multi-caminhos.

Buscando esse objetivo, muitos foram os protocolos propostos para realizar a transferência por multi-caminhos, nas camadas de enlace, rede, transporte, aplicação ou *cross layers*, como podemos verificar no levantamento do estado da arte apresentado no artigo [2].

Porém, os únicos protocolos da camada de transporte que até o presente momento foram padronizados pelo IETF são o SCTP-CMT [7] e o MPTCP [8]. Os dois protocolos apresentam muitas diferenças nas suas arquiteturas, contudo enfrentam os mesmos problemas inerentes à transferência por multi-caminhos, como o reordenamento de pacotes, a justiça e outros. O SCTP surgiu muito antes do MPTCP, mas, ao contrário do SCTP, o MPTCP já desfruta de um grande alcance. Muito disso se deve do fato do SCTP ser originalmente um protocolo voltado para a sinalização em redes telefônicas, não tendo por isso preocupação em ser transparente para o usuário.

Neste artigo, utilizaremos o MPTCP como protocolo de transporte, porém muitas das conclusões decorrentes deste trabalho são válidas para o SCTP-CMT.

### A. Multipath TCP (MPTCP)

O protocolo MPTCP foi desenvolvido com o objetivo de ser uma extensão do protocolo TCP capaz de transferir dados através de múltiplos caminhos. O MPTCP é o esforço de padronização mais recente de um protocolo de transporte multi-caminhos confiável para a Internet. Resultado do grupo de trabalho MPTCP da IETF [9] que estuda o assunto desde 2009 e tem como responsabilidade a padronização do protocolo quanto a aspectos da arquitetura, dos modos de operação, do controle de congestionamento e da segurança.

A arquitetura do MPTCP, que se encontra na RFC 6182 [1], apresenta os objetivos funcionais, que definem os serviços e recursos que uma implementação do MPTCP deve possuir, e os objetivos de compatibilidade, que determinam como o protocolo deve interagir com outras entidades.

Para permitir o uso de multi-caminhos, a implementação do MPTCP deve perseguir os seguintes objetivos funcionais: 1) Melhorar a taxa de transferência: o MPTCP deve suportar o uso concorrente de multi-caminhos. Para cumprir o mínimo de desempenho que justifique uma implantação, uma conexão MPTCP não deve ter menos desempenho do que uma única conexão TCP padrão no melhor caminho disponível; e 2) Melhorar a resiliência: o MPTCP deve suportar o uso de multi-caminhos intercambiáveis para fins de resiliência. Permitindo que segmentos possam ser enviados e reenviados por qualquer caminho disponível. Isso garante que, no pior dos casos, o protocolo não deve ser menos resiliente do que o TCP padrão por um único caminho.

No tocante a resiliência, esse objetivo é alcançado pelas implementações atuais. Porém, com relação à melhora da taxa de transferência, esse objetivo nem sempre é atingido quando em ambientes de rede heterogêneas. Isso ocorre principalmente como consequência dos problemas descritos abaixo.

### B. Principais problemas da transferência multi-caminhos

Os problemas na transferência multi-caminhos, em sua grande maioria, ocorrem em cenários de redes heterogêneas, causando perda de desempenho na taxa de transferência, aumento ou variação do atraso fim-a-fim e perda de pacotes. Dentre os problemas, os principais são o bloqueio do *buffer* de recebimento - HOL (*Head-Of-Line*), causado pelo reordenamento de pacotes no *buffer* de recebimento, e o congestionamento *Bufferbloat*.

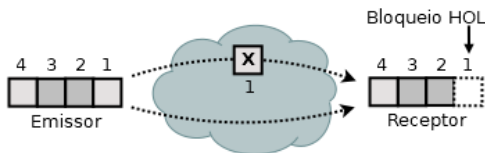


Fig. 2. Bloqueio do *buffer* de recebimento - HOL (*Head-Of-Line*)

O bloqueio HOL ocorre devido ao esgotamento do *buffer* de recebimento em decorrência do recebimento de segmentos fora de ordem, causado pela heterogeneidade dos caminhos, mais especificamente, por diferenças no RTT ou perda de pacotes, como podemos ver na Fig. 2. Enquanto os segmentos faltantes não forem recebidos, não é possível fazer a entrega ordenada dos dados, e os segmentos já recebidos ficam armazenados no *buffer*. Como o *buffer* possui um tamanho limitado, pode ser preenchido e não permitir que o emissor encaminhe mais segmentos até receber o segmento esperado e libere os dados para a aplicação.

Esse é um dos motivos da RFC 6182 recomendar que o tamanho do *buffer* de recebimento seja de no mínimo  $2 \times \sum_{i=1}^n (BW_i \times RTT_{max})$ . Sendo  $BW_i$  a capacidade do caminho  $i$ ,  $n$  o número de caminhos disponíveis e  $RTT_{max}$  o maior atraso entre os caminhos. Além disso, para compensar a heterogeneidade dos caminhos e evitar o bloqueio HOL, o MPTCP usa o mecanismo de Retransmissão e Penalização [10]. Este reencaminha o segmento causador do bloqueio HOL em um subfluxo que tenha espaço na CWND. O objetivo é

tornar mais rápida a recuperação das situações de bloqueio, compensando a diferença entre os RTT dos caminhos.

O fenômeno *Bufferbloat* [11] se refere à formação de longas filas de pacotes, congestionamentos, em equipamentos de rede com grande capacidade de memória, que apenas descartam os pacotes quando o *buffer* enche. Isto acarreta em longos atrasos para os protocolos de transporte que utilizam um controle de congestionamento baseado em perdas, degradando o desempenho da transmissão. Uma vez que o controle de congestionamento do MPTCP utiliza a perda de pacotes como sinal para identificar o congestionamento, o *bufferbloat* não percebido acaba interferindo no desempenho dos demais caminhos.

### C. Demais trabalhos relacionados

Um dos objetivos funcionais da arquitetura do MPTCP é a melhora do desempenho da conexão através da taxa de transferência agregada dos caminhos. Porém, em redes heterogêneas, muitas vezes esse objetivo não é atingido [12]. O principal motivo disso é o problema do reordenamento no *buffer* de recebimento causado pela diferença de atraso entre os caminhos. Isso se agrava em fluxos de curta duração, ou seja, de poucos KB, pois nesse caso a largura de banda é menos importante do que o atraso para a taxa de transferência. Os trabalhos apresentados nesta subseção propõem escalonadores com algoritmos de filtragem dos caminhos disponíveis baseada em suas características visando retirar os caminhos que causam degradação no desempenho da conexão do universo de caminhos disponíveis para escalonamento.

O artigo [13] propõe o algoritmo Freeze Packet Scheduling (FzPS), baseado na diferença de atraso dos caminhos. Se a diferença for significativa, só escalona pacotes para o caminho mais rápido, congelando o caminho mais lento. Essa opção decorre do fato do MPTCP ser geralmente benéfico para fluxos de longa duração, mas apresentar um desempenho pior que o TCP explorando o melhor caminho para fluxos de tamanho pequeno, apenas centenas de KB. Neste caso, os autores observam que seria melhor usar apenas o caminho mais rápido, uma vez que o atraso é mais importante do que a largura de banda da rede em tão pequeno fornecimento de dados.

Em [14], os autores propõem o algoritmo escalonador Multi-level Correlation Scheduling (MLCS) que apresenta um modelo para calcular a taxa de transferência e a correlação dos caminhos. O MLCS classifica os caminhos podendo remover um subfluxo quando o desempenho do subfluxo está abaixo dos critérios, diminuindo os efeitos de subfluxos degradados e melhorando a taxa de transferência agregada. Além disso, o MLCS escalona os segmentos com base nas correlações entre os caminhos e suas capacidades de transmissão, visando a melhor utilização dos recursos da rede.

A principal diferença deste artigo para os citados anteriormente é o foco na análise do limiar da razão de perda de pacotes a partir da qual a taxa de transferência de um fluxo TCP padrão pelo melhor caminho disponível é superior ao da conexão MPTCP.

## IV. METODOLOGIA

Para a realização do experimento utilizamos o simulador de eventos discretos NS-3 (*Network Simulator*, versão 3) [15]. O MPTCP não está implementado no NS-3 em sua versão oficial. Para a realização das simulações no NS-3 versão 3.19 utilizamos o módulo do MPTCP proposto em [16], que está em conformidade com a RFC 6824 [8] e segue de perto o *design* do kernel Linux MPTCP [17].

O controle de congestionamento do TCP padrão adotado foi o *New Reno* e para o MPTCP utilizamos o *Linked Increase Algorithm* (LIA), definido na RFC 6356 [18], que provê um desempenho similar ao conseguido pelo *New Reno* para um fluxo padrão TCP. O tipo de fila nos dispositivos de rede adotado foi o *Droptail* com limite da fila de 50 pacotes. O tamanho do pacote foi configurado em 1400 *Bytes*. E o escalonador do MPTCP utilizado foi o padrão do módulo, *Round-Robin*.

No *host A* foram instaladas aplicações geradoras de fluxo de dados TCP e MPTCP do tipo *bulk*. No *host B* instalamos aplicações consumidoras de fluxo de dados, *sink*. Não realizamos simulações com tráfego de dados do tipo *web*, por serem fluxos de dados pequenos e de curta duração, não indicados para análise de taxa de transferência, objetivo principal deste artigo.

## A. Cenário de simulação

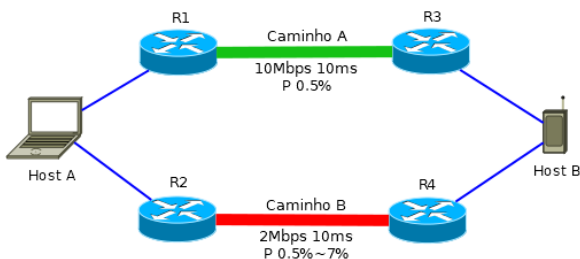


Fig. 3. Topologia de rede utilizada nas simulações

A Fig. 3 apresenta a topologia utilizada nas simulações. O cenário é composto por dois *hosts*, conectados por dois caminhos com características diferentes. O caminho A, identificado como o melhor caminho, possui largura de banda de 10Mbps, atraso de 10ms e razão de perda de pacotes de 0.005 (0.5%). Valores próximos aos comumente encontrados em um enlace com tecnologia WiFi/IEEE 802.11b. Já o caminho B, identificado como o caminho degradado, possui largura de banda de 2Mbps, atraso de 10ms e razão de perda de pacotes variando de 0.005 (0.5%) até 0.7 (7%). Esses valores são próximos aos encontrados em redes móveis 3G. A tabela I resume as configurações dos caminhos usadas nas simulações.

Os valores de  $P$  para o caminho B simulados foram de 0.5%, 1% e daí por diante incrementando em 1% até alcançar 7%. Para cada valor de  $P$  foram realizadas 10 (dez) simulações de 60s de duração, com diferentes sementes geradoras de números aleatórios a cada repetição. Como indicador de desempenho utilizamos o *Goodput* médio, tanto para o TCP padrão pelo

TABELA I  
PARÂMETROS DAS SIMULAÇÕES

Parâmetro	Caminho A	Caminho B
Tecnologia	WiFi/IEEE 802.11b	UMTS (3G)
Largura de banda	10Mbps	2Mbps
Atraso	10ms	10ms
Limite da fila	50	50
Tipo de fila	Droptail	Droptail

melhor caminho como para o MPTCP. O limiar será o valor de  $P$  do caminho degradado (caminho B) imediatamente menor ao do  $P$  cujo o *Goodput* médio da conexão MPTCP obteve um desempenho inferior ao conseguido pelo TCP padrão pelo melhor caminho (caminho A).

Para análise dos resultados alcançados pelo MPTCP agregado e por cada subfluxo em comparação ao desempenho de uma única conexão TCP pelo melhor caminho disponível, simulamos um fluxo TCP pelo caminho A, com razão de perda  $P = 0.5\%$  e com os mesmos parâmetros apresentados na tabela I para o caminho A. Foram realizadas 10 (dez) simulações de 60s de duração, com diferentes sementes geradoras de números aleatórios a cada repetição. O valor da média foi utilizado como linha de referência para a realização das análises.

## V. RESULTADOS

A Fig. 4 apresenta o *Goodput* alcançado pela conexão MPTCP para os diferentes valores de  $P$ . Foi traçado uma linha de referência que corresponde ao *Goodput* do fluxo TCP padrão pelo melhor caminho (caminho A) e com  $P=0.5\%$ . O objetivo da linha de referência é verificar o momento de interseção entre as curvas, ou seja, onde o *Goodput* do MPTCP cruza o limiar a partir do qual é melhor utilizar somente o melhor caminho. Pelo gráfico percebemos que entre 5% e 6% de  $P$  ocorre a interseção, logo, o limiar para o cenário proposto pode ser estabelecido como 5%. Qualquer valor de  $P$  acima de 5% para o caminho degradado irá gerar perda de desempenho na conexão MPTCP a ponto de não compensar a sua utilização, contrariando o objetivo funcional nº 1 da arquitetura do MPTCP [1].

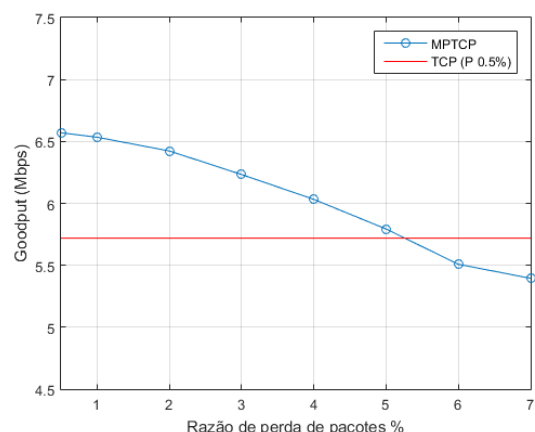


Fig. 4. Limiar - *Goodput* do MPTCP vs TCP de referência

No gráfico da Fig. 5 são apresentados o *Throughput* em função de  $P$  dos subfluxos da conexão MPTCP pelos caminhos A e B e a linha de referência correspondente ao *Throughput* do fluxo TCP padrão utilizando o caminho A. Podemos observar que, para todos os valores de  $P$  simulados, o desempenho do subfluxo da conexão MPTCP pelo caminho A é pior do que o do fluxo TCP padrão pelo mesmo caminho. Isso ocorre devido aos problemas da conexão MPTCP em cenários de rede heterogênea, já discutidos na seção III.B, decorrentes do reordenamento dos pacotes no *buffer* de recebimento causados pelo caminho degradado.

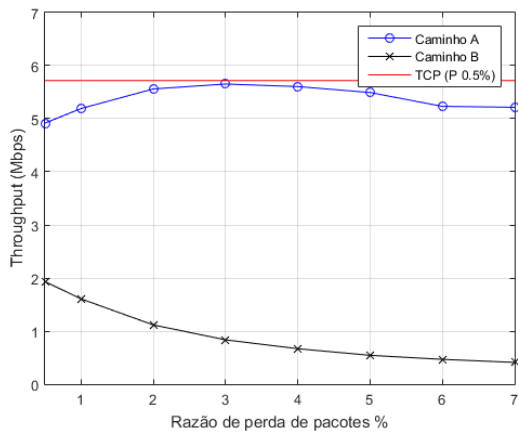


Fig. 5. Comparação do *Throughput* do melhor caminho (A) na conexão MPTCP e na conexão TCP

Além disso, o desempenho do subfluxo do caminho A aumenta no intervalo de 0 a 3% de  $P$ . No início do intervalo citado, o desempenho do subfluxo do caminho A não sofre com o efeito do reordenamento no *buffer* de recebimento decorrente das perdas de pacotes que ocorrem no caminho B. O escalonador do MPTCP seleciona um subfluxo para enviar os segmentos de acordo com o critério adotado em seu algoritmo, então envia os segmentos para a sua janela de congestionamento (CWND). Quando a CWND enche, o escalonador escolhe outro subfluxo entre os com CWND disponível para o envio. Com o aumento de  $P$  pelo caminho B, o controle de congestionamento reduz o tamanho da CWND do subfluxo pelo caminho B e, como consequência, o escalonador acaba por enviar mais segmentos pelo subfluxo do caminho A, pelo fato deste possuir uma maior CWND, refletindo na tendência de crescimento. Porém, quando o valor de  $P$  passa de 3%, o subfluxo do caminho A começa a sofrer com a perda de pacotes pelo caminho B, e o seu desempenho tende a diminuir.

## VI. CONCLUSÕES

Neste artigo, realizamos uma análise do impacto de um caminho degradado no desempenho da conexão MPTCP, através de simulações no NS-3, em um cenário comumente encontrado em dispositivos móveis com uma interface de rede WiFi e outra de rede móvel celular (3G).

Para o cenário proposto, o limiar de  $P$  a partir do qual é melhor utilizar o TCP padrão pelo melhor caminho ficou em

5%. Este resultado se refere ao cenário típico estudado, outros cenários devem ser estudados a fim de se obter um limiar para um caso mais geral.

Os resultados obtidos confirmaram que para grandes fluxos de dados, em ambiente de rede heterogênea onde um dos caminhos está degradado, é melhor em relação ao *Goodput* utilizar uma conexão TCP padrão com o fluxo de dados através do melhor caminho disponível.

Os trabalhos futuros incluem a proposta de um escalonador para o MPTCP, voltado para dispositivos móveis, com um algoritmo baseado na estimativa do *throughput* dos caminhos em função da razão de perdas, que seja capaz de remover um caminho degradado da conexão a partir de um limiar de perdas.

## REFERÊNCIAS

- [1] A. Ford, C. Raiciu, M. Handley, S. Barre e J. Iyengar, "Architectural Guidelines for Multipath TCP Development," RFC 6182 (INFORMATIONAL), Internet Engineering Task Force, Março 2011.
- [2] M. Li, A. Lukyanenko, Z. Ou, A. Ylä-Jääski, S. Tarkoma, M. Coudron e S. Secci, "Multipath Transmission for the Internet: A Survey," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 4, pp. 2887-2925, 2016.
- [3] J. Postel, "User Datagram Protocol," RFC 768 (Internet STANDARD), Internet Engineering Task Force, Agosto 1980.
- [4] J. Postel, "Transmission Control Protocol," RFC 793 (Internet STANDARD), Internet Engineering Task Force, Setembro 1981.
- [5] T. Henderson, S. Floyd, A. Gurtov e Y. Nishida, "The NewReno Modification to TCP's Fast Recovery Algorithm," RFC 6582 (PROPOSED STANDARD), Internet Engineering Task Force, Abril 2012.
- [6] P. Karn, C. Bormann, G. Fairhurst, D. Grossman, R. Ludwig, J. Mahdavi, G. Montenegro, J. Touch e L. Wood, "Advice for Internet Subnetwork Designers," RFC 3819 (BEST CURRENT PRACTICE), IETF - Network Working Group, Julho 2004.
- [7] J. R. Iyengar, P. D. Amer e R. Stewart, "Concurrent Multipath Transfer Using SCTP Multihoming Over Independent End-to-End Paths," *IEEE/ACM Transactions on Networking*, vol. 14, no. 5, pp. 951-964, Outubro 2006.
- [8] A. Ford, C. Raiciu, M. Handley e O. Bonaventure, "TCP Extensions for Multipath Operation with Multiple Addresses," RFC 6824 (EXPERIMENTAL), Internet Engineering Task Force, Janeiro 2013.
- [9] IETF Working Group. Multipath TCP (Active WG). [Online]. Available: <https://tools.ietf.org/wg/mptcp/>
- [10] C. Paasch, S. Ferlin, O. Alay e O. Bonaventure, "Experimental evaluation of multipath TCP schedulers," in *2014 ACM SIGCOMM Capacity sharing workshop. CSWS 2014. ACM*, pp. 27-32, Agosto 2014.
- [11] B. Felix, A. Santos, B. Kantarci e M. Nogueira, "CD-ASM: A new queuing paradigm to overcome bufferbloat effects in HetNets," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC). IEEE, 2017*. pp. 1-6, Outubro 2017.
- [12] A. Nikraves, Y. Guo, F. Qian, Z. M. Mao e S. Sen, "An in-depth understanding of multipath TCP on mobile devices: measurement and system design," in *the 22nd Annual International Conference on Mobile Computing and Networking. MobiCom '16. ACM SIGMOBILE, 2016*. pp. 189-201, Outubro 2016.
- [13] J. Hwang e J. Yoo, "Packet Scheduling for Multipath TCP," in *2015 Seventh International Conference on Ubiquitous and Future Networks (ICUFN). IEEE, 2015*. pp. 177-179, Julho 2015.
- [14] B. Hu, L. Xing, Z. Wang e N. Liu, "MLCS: A multi-level correlation scheduling algorithm for multipath transport," in *2016 International Conference on Information Networking (ICOIN). IEEE, 2016*. pp. 166-171, Janeiro 2016.
- [15] NS-3 a discrete-event network simulator for Internet system. NS-3 Consortium. [Online]. Available: <https://www.nsnam.org>
- [16] M. Kheirkhah, I. Wakeman e G. Parisi, "Multipath-TCP in ns-3," in *2014 Workshop on ns-3. WNS3, 2014*. Maio 2014.
- [17] Multipath TCP in the Linux Kernel. C. Paasch, S. Barre e outros. [Online]. Available: <http://www.multipath-tcp.org>
- [18] C. Raiciu, M. Handley e D. Wischik, "Coupled Congestion Control for Multipath Transport Protocols," RFC 6356 (EXPERIMENTAL), Internet Engineering Task Force, Outubro 2011.