

Reduzindo os Efeitos do *Bufferbloat* sobre Multi-Caminhos em Redes Sem Fio Heterogêneas

Benevid Felix, Aldri Santos, Michele Nogueira

¹Núcleo de Redes Sem-Fio e Redes Avançadas (NR2)
Universidade Federal do Paraná (UFPR)

{benevid,aldri,michele}@inf.ufpr.br

Abstract. *Low latency and higher throughput requirements on Internet access have driven advances in order to manage the high-density resulted from the heterogeneous wireless networks (HetNets). The Multi-Path Transmission Control Protocol (MPTCP) allows embedded mobile devices to transmit data over multiple networks by heterogeneous technologies. However, MPTCP yields performance degradation when applied over HetNets, even more under bufferbloat effects. This paper provides a detailed analysis of jointly the MPTCP congestion control mechanism and the AQM CoDel, which has offered promising results against bufferbloat effects. Further, a new discipline, LIFO-CoDel, is proposed aiming to reduce the dropped packets through the MPTCP congestion control mechanism and prioritizing the most recent packets. Simulation results indicate increases in throughput, keeping RTT low, and a reduction in the waiting time.*

Resumo. *A demanda por baixa latência e maior vazão no acesso à Internet tem impulsionado avanços na gerência da alta densidade existente das redes heterogêneas sem fio (HetNets). O protocolo MPTCP (Multi-Path Transmission Control Protocol) possibilita dispositivos móveis embarcados transmitir dados por múltiplas redes através de tecnologias heterogêneas. Entretanto, o MPTCP gera degradações no desempenho em HetNets, ainda mais diante do fenômeno bufferbloat. Este trabalho provê uma análise detalhada do controle de congestionamento do MPTCP em conjunto com a disciplina AQM CoDel, a qual se apresenta como promissora contra os efeitos do fenômeno bufferbloat. Além disso, uma nova disciplina, a LIFO-CoDel, é proposta com o objetivo de reduzir a taxa de pacotes descartados pelo controle de congestionamento do MPTCP em redes heterogêneas sem fio através do mecanismo de controle de congestionamento do MPTCP e da prioridade aos pacotes mais recentes. Resultados de simulações indicam a melhoria da vazão, mantendo o RTT baixo, e a redução do tempo de espera dos pacotes.*

1. Introdução

A demanda pelo acesso à Internet com requisitos de baixa latência e maior largura de banda é crescente, especialmente em redes celulares [Hossain and Hasan 2015]. Uma das soluções para atender a esta demanda consiste na introdução de pequenas células, densificando os pontos de acesso à rede. Isto resulta em uma rede heterogênea sem fio (Het-Net - *Heterogeneous Network*) combinando células grandes e pequenas com diferentes tecnologias, desempenho, cobertura e qualidade de serviço [Jeanette Wannstrom 2014].

Os dispositivos móveis embarcados com diferentes tecnologias de acesso sem fio (*Multihomed*) provêm acesso simultâneo a múltiplas redes (Celular, WiFi, etc). Este acesso múltiplo permite o encaminhamento dos dados por diferentes caminhos na tentativa de alcançar objetivos variados, tais como maior vazão, transições mais suaves, e outros.

Devido ao protocolo TCP original não suportar transmissões por múltiplos caminhos, tem-se explorado no contexto de redes celulares o MPTCP, uma versão multi-caminhos do protocolo TCP [Chen et al. 2013, Lee and Lee 2015]. O seu objetivo é manter compatibilidade com o TCP, provendo o envio por multi-caminhos apenas quando os envolvidos na conexão suportam o protocolo. O controle de congestionamento é executado fim-a-fim, como no TCP, e faz uso de sinais como estouro de temporizadores e reconhecimentos (ACKs) duplicados para identificar um possível congestionamento na rede e reduzir o tamanho de sua janela de congestionamento [Raiciu et al. 2012]. O uso do protocolo em servidores ainda é incipiente em ao TCP. Contudo, estudos realizados em [Mehani et al. 2015] demonstraram o interesse dos provedores de serviços em oferecer suporte a conexões multi-caminhos.

Apesar do MPTCP apresentar bom desempenho no contexto de redes de centros de dados [Raiciu et al. 2011, Singh et al. 2015], muitos problemas existem quando aplicado em HetNets [Ferlin et al. 2014, Khalili et al. 2013, Arzani et al. 2014]. O MPTCP sofre degradação no desempenho em cenários com caminhos heterogêneos, com variação na taxa de perda de pacotes, latência e taxa de transferência. Esta degradação aumenta quando os caminhos apresentam grandes variações do atraso causadas pelo fenômeno *bufferbloat* [Ferlin-Oliveira et al. 2014]. Este fenômeno, evidenciado por [Nichols and Jacobson 2012], ocorre devido à existência de grandes filas que absorvem enorme quantidade de tráfego em um canal congestionado, acarretando alta latência com baixa perda de pacotes. 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.

Estudos recentes acerca do *bufferbloat* atestam a existência do problema em redes sem fio [Alfredsson et al. 2013]. As abordagens empregadas para mitigar o problema envolvem algoritmos de controle de congestionamento (CC) ou disciplinas de controle ativo de filas (AQM - *Active Queue Management*). O primeiro depende do algoritmo de CC fim-a-fim, e o segundo, da implementação de uma disciplina AQM nos pontos de encaminhamento da rede. Estudos realizados em [Chen et al. 2013, Ferlin et al. 2014, Zhou et al. 2015] mostram que os algoritmos de CC do MPTCP sofrem degradações em um cenário de HetNets. No melhor de nosso conhecimento, a disciplina AQM *Controlled Delay* (CoDel) [K. Nichols 2014] e suas variações, como *Flow Queue Controlled Delay* (FQ-CoDel) [Hoeiland-Joergensen et al. 2015], possuem bons resultados frente ao fenômeno com redução expressiva do tempo de permanência na fila.

A fim de compreender o funcionamento destas duas abordagens realizamos simulações utilizando o NS-3 [Kheirkhah 2015] comparando os algoritmos de CC e as disciplinas de fila. Os resultados obtidos indicam que o mecanismo de controle de fila do CoDel causa muitos descartes de pacotes na fila. Isto impacta no CC do MPTCP que acaba não alcançando a capacidade de transferência do canal. A partir dos resultados desta análise, a disciplina LIFO-CoDel foi proposta, a qual altera a lógica da fila, originalmente FIFO para LIFO. O objetivo da alteração é permitir o encaminhamento dos

pacotes com menor tempo de espera. O tempo máximo do pacote na pilha é controlado com um parâmetro θ auto-ajustável. Com isto, o número de descartes é reduzido e conseqüentemente uma melhora no desempenho da transmissão multi-caminhos. Os resultados das simulações indicam uma melhora na vazão, com redução do número de descartes, e mantendo uma equidade no *Round Trip Time* (RTT) com o FIFO-CoDel no cenário de HetNets abordado neste trabalho.

O restante do trabalho está organizado da seguinte forma. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 detalha as características do protocolo MPTCP e o funcionamento do CoDel e FQ-CoDel. A Seção 4 apresenta os resultados da análise do impacto do CoDel e FQ-CoDel no CC do MPTCP em um cenário de redes heterogêneas. A Seção 5 descreve a proposta do LIFO-CoDel e os resultados obtidos. A Seção 6 apresenta as conclusões e direções futuras.

2. Trabalhos Relacionados

Diferentes trabalhos como [Ferlin-Oliveira et al. 2014], [Gómez et al. 2014] e [Chen and Towsley 2014] têm avaliado o impacto do *bufferbloat* em conexões MPTCP. Em [Gómez et al. 2014] foram realizadas simulações comparando diferentes algoritmos de CC do MPTCP e diferentes abordagens de roteamento multi-caminho. Em relação ao CC, os resultados mostram que o MPTCP sofre degradação do desempenho sob o aumento na taxa de perda do caminho, sendo o impacto maior quando o protocolo utiliza múltiplas interfaces, cenário comum em redes heterogêneas. O atraso em um dos caminhos impacta no desempenho dos demais e em alguns cenários o desempenho somado dos caminhos pode ser inferior ao de um único fluxo TCP, não justificando o uso do protocolo [Arzani et al. 2014]. Em [Arzani et al. 2014], os autores apresentam resultados obtidos através de experimentos com MPTCP em cenários heterogêneos. Um dos objetivos era compreender o papel da seleção de caminhos no desempenho do MPTCP. Os resultados obtidos demonstraram que o mecanismo de CC do MPTCP não precisa gerenciar a seleção de sub-fluxos e que a escolha do escalonador de pacotes tem grande impacto na vazão final. A solução proposta em [Ferlin et al. 2014] avalia dinamicamente o desempenho dos caminhos para adaptar o escalonamento dos dados. Os resultados apresentados mostraram que a solução é melhor que o escalonador padrão do MPTCP quando os caminhos possuem diferentes características, como redes 2G e WiFi, 2G e 3G. Porém, quando a heterogeneidade do caminho é menor, o desempenho é igual ou inferior ao escalonador padrão.

O trabalho de [Lee and Lee 2015] investiga como diferentes características das redes heterogêneas podem afetar o desempenho do MPTCP. Para melhorar o desempenho do protocolo quando o mesmo apresenta problemas como o bloqueio da fila de recebimento (HOL -*Head of Line*) e *bufferbloat*, um algoritmo de ajuste dinâmico da janela de recebimento do MPTCP com base nos valores dos RTTs é proposto. Contudo, os autores consideram um cenário em que a taxa de perda da rede celular é menor que a rede usando WiFi e dependem de um tempo mínimo de observação do RTT. A variação entre os RTTs dificulta estimar o tamanho da janela de recebimento. Em [Chen and Towsley 2014], os autores analisaram o bloqueio HOL causado pelo impacto do fenômeno *bufferbloat* e demonstraram que o caminho mais rápido passa por diversos períodos de ociosidade. Isto acarreta na degradação do desempenho porque o mecanismo de *idle restart* do MPTCP reinicia a janela em modo de partida lenta (*slow start*). Os autores sugeriram desativar o

mecanismo para que a janela continue crescendo após o período de ociosidade. Apesar de melhorar o desempenho, este ajuste torna os fluxos do MPTCP mais agressivos e podem interferir no objetivo do protocolo de manter o balanceamento com demais fluxos. Em [Ferlin-Oliveira et al. 2014], os autores avaliam os desafios da transmissão de dados através do MPTCP em cenários com caminhos heterogêneos e com a presença do *bufferbloat*. Os resultados mostram que em determinados cenários o MPTCP não consegue ultrapassar o desempenho do melhor caminho.

Apesar de não estar diretamente relacionado à transmissão multi-caminhos, o trabalho realizado em [Kulatunga et al. 2015] avalia o desempenho do CoDel e FQ-CoDel em redes com capacidade de transmissão limitada e com altos valores de RTTs. Os resultados apresentados mostram que o descarte na fila causados pelo CoDel impacta de forma crescente na taxa de transmissão conforme o RTT do caminho aumenta. A proposta do trabalho foi de otimizar os parâmetros do CoDel para equilibrar a taxa de transmissão e reduzir o impacto no CC s efeitos do *bufferbloat*. Apesar dos resultados justificarem a abordagem, é importante destacar que o ajuste de parâmetros pode ser impraticável em cenários de HetNets, nos quais as condições da rede mudam constantemente, e que isto deve ser feito de forma dinâmica.

3. O TCP Multi-caminhos e o Controle de Filas

O protocolo MPTCP [Barré et al. 2011] tem como base o protocolo TCP e possibilita a transmissão de dados por multi-caminhos. Como ilustrado na Figura 1, cada sub-fluxo MPTCP se comporta como um fluxo TCP. Cada sub-fluxo se associa a um endereço de rede e uma interface diferente (Figura 1.a) ou ainda, múltiplos fluxos podem ser criados sobre uma única interface existente (Figura 1.b). Entretanto, uma única conexão fim-a-fim é estabelecida. O protocolo possibilita adicionar ou remover caminhos enquanto a conexão estiver ativa. Alguns dos objetivos do protocolo é obter um desempenho melhor ao oferecido por um único fluxo e aumentar a resiliência utilizando diferentes caminhos [Ford et al. 2013].



Figura 1. Arquitetura MPTCP

O CC executado pelo MPTCP tem como base os algoritmos do TCP, como a variante *NewReno*. Todos eles modificam apenas a fase de prevenção de congestionamento (*Congestion Avoidance*), ficando as fases de inicialização lenta (*Slow Start*), retransmissão rápida (*Fast Retransmit*) e recuperação rápida (*Fast Recovery*) inalteradas [Singh et al. 2013]. O CC é executado em nível de sub-fluxo, utilizando uma janela de congestionamento (CWND) individual e uma janela de recebimento (RCVW) compartilhada para facilitar a entrega ordenada dos dados [Zhou et al. 2015]. Seus requisitos principais incluem aumentar a utilização vazão, não prejudicar fluxos concorrentes

e balancear o congestionamento. A vazão dos sub-fluxos em uma transferência multi-caminhos deve ser pelo menos melhor que um único fluxo através do melhor caminho. Os sub-fluxos precisam se comportar como um único fluxo TCP na existência de um gargalo comum. Por fim, o tráfego deve ser direcionado para os caminhos menos congestionados.

A maioria dos algoritmos de CC propostos para o MPTCP aplicam diferentes abordagens às fases de incremento aditivo e decremento multiplicativo. O algoritmo LIA (*Linked Increase Algorithm*), adotado como padrão pelo IETF [Ford et al. 2013], especifica apenas como deve ocorrer o incremento da janela de congestionamento ao receber um ACK, mantendo o decremento padrão como no TCP. O crescimento das janelas dos sub-fluxos são acoplados (Eq. 1). O parâmetro α , Eq. 2, controla a agressividade dos sub-fluxos de modo que o incremento da janela não seja superior ao de um fluxo de caminho único com o mesmo tamanho de janela. Nas Eqs. 1 e 2, w_i e RTT_i referem-se ao tamanho da janela de congestionamento e o tempo de ida e volta (RTT) de um sub-fluxo i , respectivamente. O parâmetro W representa o tamanho total das janelas de congestionamento de todos os sub-fluxos.

$$w_i = \begin{cases} \min(\alpha/W, 1/w_i) & \text{incremento} \\ w_i/2 & \text{decremento} \end{cases} \quad (1)$$

$$\alpha = W * \frac{\max(w_i/RTT_i^2)}{(\sum_i w_i/RTT_i)^2} \quad (2)$$

Em HetNets, o MPTCP tem apresentado diversos problemas que degradam seu desempenho. Isto ocorre porque os caminhos possuem diferentes características, como variação da largura de banda, atraso, taxa de perda, entre outras [Ferlin et al. 2014]. Apesar dos problemas, o cenário das HetNets com o MPTCP é muito promissor [Kurosaka and Bandai 2015]. As HetNets têm sido uma das propostas do *3rd Generation Partnership Project* (3GPP) [Jeanette Wannstrom 2014] para atender aos requisitos de serviços cada vez mais restritos, e.g., a 5^o geração das redes celulares requerem latência entre *2ms* a *5ms* e vazão aproximada de 10 Gbbs [Hossain and Hasan 2015]. Esta proposta inclui a densificação da rede, com a inserção de células de curto alcance (e.g. micro, pico, metro) para ampliar a cobertura, reduzir a latência e aumentar a vazão. Estas pequenas células implementam diferentes tecnologias de acesso, possibilitando que os dispositivos móveis, com suporte a múltiplas interfaces de rede (*multihomed*), possam explorar os benefícios da transmissão multi-caminhos utilizando o protocolo MPTCP.

A maioria dos roteadores atuais empregam um mecanismo FIFO (*First In, First Out*) para encaminhamento dos pacotes, e por simplicidade, descartam aqueles que chegam quando a fila está cheia (*DropTail*). A disciplina *DropTail* frente às grandes filas presentes nos roteadores e a um congestionamento persistente tem acarretado grandes variações no atraso. Sobre a perspectiva deste problema, a disciplina FIFO é ineficiente. O problema das grandes filas resulta no aumento excessivo da latência e degradação da vazão que acabam afetando a disponibilidade da rede. Em [Nichols and Jacobson 2012], este problema é evidenciado como fenômeno *bufferbloat* e uma disciplina denominada *Controlled Delay* (CoDel) é proposta para mitigar o problema. A disciplina CoDel guarda o instante enq_i que cada pacote p_i é colocado na fila. Ao retirar o pacote da fila no tempo

deq_i , ele calcula o tempo de permanência do pacote, conforme Eq. 3).

$$\delta_i = deq_i - enq_i \quad (3)$$

Este intervalo é utilizado pelo mecanismo empregado pelo CoDel para alternar entre dois estados: descarte e não-descarte. A disciplina CoDel inicia no estado de não-descarte e pode mudar seu estado conforme valor do intervalo (μ). O valor inicial de μ é λ , porém o valor é ajustado conforme o número de descartes consecutivos (n_{drop}). Se entre o tempo $t_1 = t$ e $t_2 = t + \mu$ um pacote p_i é desenfileirado tal que, $deq_i \in [t_1, t_2]$ e $\delta_i > \tau$, a CoDel entra em estado de descarte e os próximos pacotes desenfileirados são descartados enquanto $\delta_i > \tau$. A variável que controla do número de descartes consecutivos é incrementada em $n_{drop} + 1$ e o valor de μ é atualizado para $\mu\sqrt{n_{drop}}$. Caso $\delta_i < \tau$, a disciplina CoDel permanece ou entra no estado de não-descarte e a variável n_{drop} é reinicializada com 1 e μ para λ [Kulatunga et al. 2015].

A variação FQ-CoDel emprega a mesma abordagem do CoDel para controlar o atraso do pacote na fila. A diferença é que ele separa os fluxos em n filas para aplicar o mecanismo do CoDel de forma independente. Para tornar a utilização justa, os pacotes são escalonados com um algoritmo *Deficit Round Robin* (DDR) [Hoeiland-Joergensen et al. 2015]. O algoritmo DDR configura uma quantidade de Q_i (*quantum*) bytes para ser retirado da fila n_i a cada ciclo. Caso a fila n_i não tenha Q_i bytes, este déficit é acumulado para o próximo ciclo. O CoDel e FQ-CoDel sugerem valores $\tau = 5ms$ e $\lambda = 100ms$ como padrão para maioria das situações [Ali et al. 2015].

4. Análise do Impacto do CoDel no CC do MPTCP em HetNets

A entre o fenômeno *bufferbloat* e o CC do MPTCP, embora realizada por alguns trabalhos citados na seção 2, não consideram o impacto das disciplinas de fila na transmissão multi-caminhos. Uma vez que o MPTCP utiliza um CC acoplado, se faz necessário avaliar a implicação das perdas de pacotes na fila, ocasionadas tanto por congestionamentos, quanto pelo mecanismo empregado pela disciplina de fila. Para avaliar este impacto, foram realizadas simulações com NS-3 [Kheirhah 2015] considerando um cenário com duas redes de acesso, celular e WiFi, comum para usuários de *smartphones*, ilustrado pela Figura 2. Neste cenário o equipamento do usuário (EU) estabelece a conexão MPTCP com o servidor pela sua interface 1 através da rede LTE (caminho A). Após o passo anterior, um sub-fluxo adicional é estabelecido pela interface 2 através da rede WiFi (caminho B). Ambos os caminhos, A e B, compartilham a fila de saída do roteador que dá acesso ao servidor. O caminho C é configurado intencionalmente com uma vazão inferior aos demais para representar um gargalo na rede. A partir destas definições, o cenário permite representar uma situação em que os sub-fluxos do MPTCP, originados da EU, compartilhem um ponto de congestionamento na (i.e. fila do roteador).

As simulações empregam os fatores listados na Tabela 1. Além desses, os pacotes possuem 1458 bytes, a fim de saturar mais rapidamente a janela de congestionamento, o tamanho máximo da fila é de 50 pacotes, e RCVWD e CWND são definidos com valor padrão do simulador de 64 Kilobytes. A carga de trabalho foi fixada em 4 Mbytes para facilitar o cálculo do *goodput*. E com intuito de não limitar os caminhos A e B até o roteador, a taxa de transferência foi definida com 1 Gbps e atraso padrão de 1ms, salvo

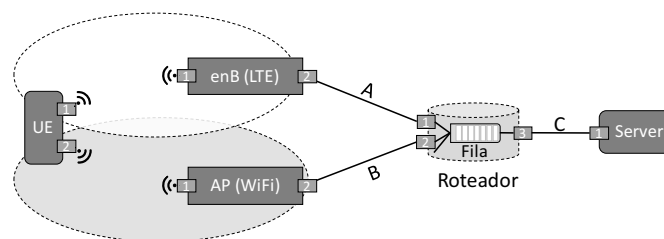


Figura 2. Cenário de Avaliação

os parâmetros definidos na tabela. Estes valores são definidos para que o caminho C funcione como um gargalo na transmissão de dados entre a EU e o servidor. O ponto de acesso (AP) emprega o padrão IEEE802.11a, deixando inalterados os valores padrões das camadas inferiores. Embora a EU tenha permanecido em uma posição fixa durante as simulações, o fator atraso do caminho, definido para o caminho A , tende representar as variações que ocorrem em função da mobilidade. Esta variação apenas foi definida para a rede LTE para representar a heterogeneidade entre os caminhos. E por fim, com objetivo de preencher a fila do roteador, todas as simulações foram realizadas com um terceiro nó transmitindo pacotes UDP para o servidor, ocupando 25% da banda do caminho C . Desta forma, os demais sub-fluxos se adequam aos outros 75% disponíveis.

Tabela 1. Fatores utilizados nas simulações

Fator	Objetivo	Valor
Atraso do caminho	Variar o RTT do caminho A	1ms, 10ms, 100ms
Disciplina AQM	Analisar o comportamento da fila	DropTail, CoDel
Algoritmo CC	Analisar o CC do MPTCP	Uncoupled, LIA e RTT_Compensator

Para avaliar o impacto da fila na transmissão de dados foram utilizadas métricas para medir o desempenho da transmissão fim-a-fim, do CC e da fila. Para análise da transmissão foi empregado o *goodput* ($bytes_recebidos/tempo_total$). O tamanho da janela de congestionamento e o RTT são considerados para analisar o CC do caminho. E por fim, o número de pacotes descartados, o tamanho e o atraso (tempo de permanência) foram aplicados para medir o desempenho da fila. Estas métricas são apresentadas com valores médios de 35 repetições e com intervalo de confiança de 95%.

4.1. Resultados

Os resultados apresentados referem-se à comparação entre os algoritmos LIA, *RTT_Compensator* e *Uncoupled* com as disciplinas CoDel e DropTail. Esta última foi empregada como base para comparações. A Figura 3 apresenta os resultados do tamanho médio da janela de congestionamento (CWND) e o valor médio do RTT para o CoDel e DropTail, provenientes de simulações realizadas com o algoritmo LIA. Em todos os gráficos, o eixo x representa o atraso atribuído ao caminho A . Com ambos os caminhos com atraso de 1ms, o RTT médio observado, tendo o CoDel como disciplina de fila do roteador, foi de 85ms para o caminho A (LTE) e 67ms para o B (WiFi). Os resultados indicam que a variação do atraso do caminho A influencia o RTT do caminho B . Com atraso de 100ms, o RTT médio do caminho A foi de 276ms e o caminho B variou entre

69ms a 71ms. Tendo como disciplina de fila o DropTail, o congestionamento tem um impacto considerável no RTT médio, atingindo 691ms para o caminho *A* e 478ms para o *B* com atraso de 100ms. Estes valores são quase 9 vezes maiores que os obtidos pelo CoDel. Por limitação de espaço, os gráficos com os resultados para *RTT_Compensator* e *Uncoupled* não foram apresentados, porém tiveram uma variação em relação ao LIA de aproximadamente 5% a 8%, respectivamente.

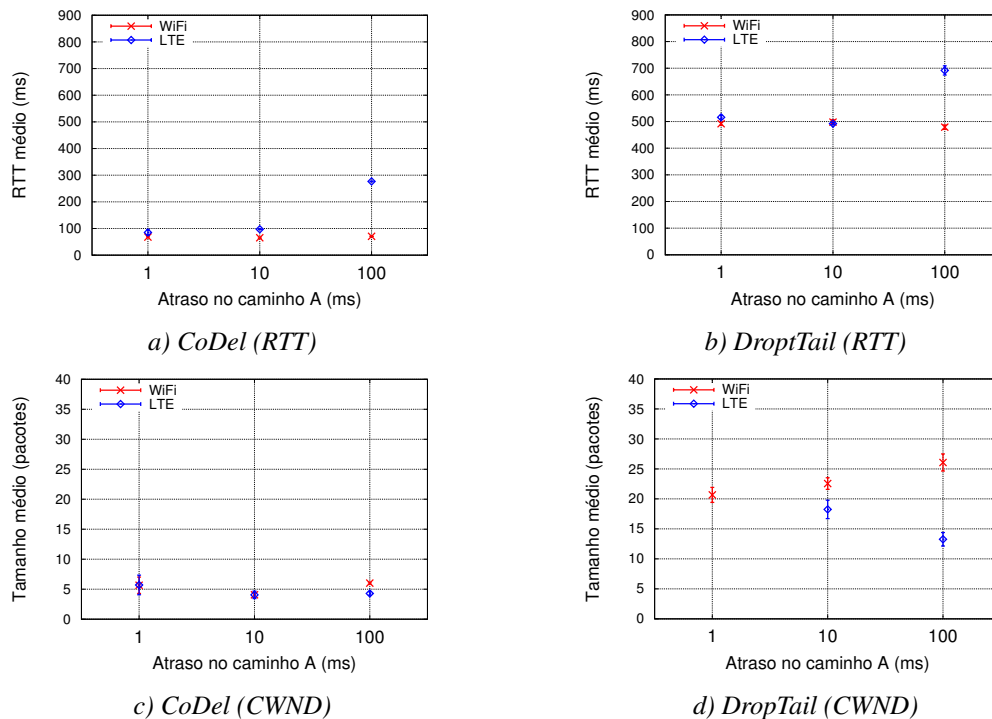


Figura 3. Desempenho da transmissão

Ainda na Figura 3, pode-se observar que o comportamento da janela de congestionamento (CWND) reflete o funcionamento do algoritmo de CC do MPTCP. Conforme o atraso aumenta no caminho *A*, a sua CWND é reduzida a fim de priorizar a CWND do caminho *B*, que possui um menor RTT. Contudo, este balanceamento do algoritmo não ocorre de forma adequada quando o CoDel é empregado, o que pode ser observado no gráfico quando o caminho *A* tem atraso de 10ms. Neste caso, ambas as janelas tiveram uma redução de tamanho e os resultados apresentados na Figura 4.a, referentes ao descarte médio na fila, ajudam a compreender o porquê. O CoDel (CoDel-FIFO) descarta um número muito maior de pacotes que o DropTail, impedindo a CWND de crescer. Os descartes tem grande impacto no desempenho da transmissão multi-caminhos, como pode ser observado no *goodput* apresentado pela Figura 4.b. O DropTail possibilita que toda a capacidade do canal disponível para as transmissões com MPTCP seja alcançada (i.e. 75%), enquanto o CoDel um pouco mais de 50%.

A variação do *goodput* do CoDel está diretamente relacionado ao número de descartes de pacotes na fila. Conforme o número de pacotes descartados aumenta, diminui o *goodput* da transmissão. Contudo, o RTT médio é expressivamente menor que o observado com o DropTail. Como o DropTail não descarta pacotes enquanto a fila é suficiente para absorver o tráfego, este atraso acarreta um RTT médio muito alto, no entanto, como

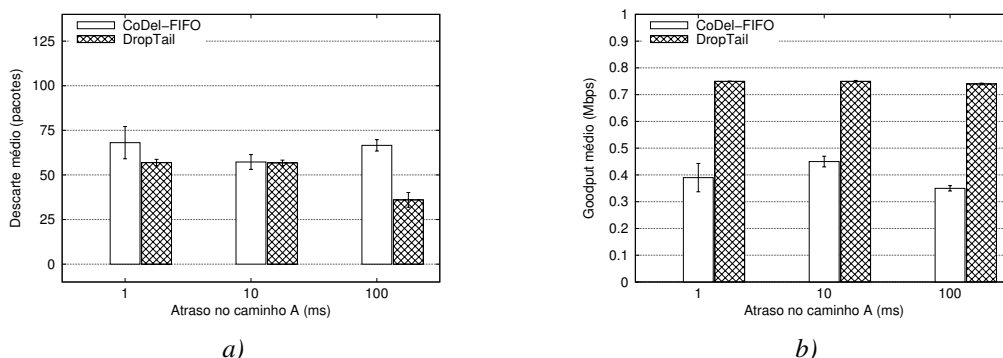


Figura 4. Descartes vs. Goodput

a fila está sempre cheia, a utilização do canal é mais eficiente. Com o CoDel, a fila não atinge 20% de sua capacidade. Isto pode ser observado nos resultados apresentados na Figura 5.a, referentes ao tamanho médio da fila alcançado com os diferentes algoritmos de CC. O algoritmo *Uncoupled* possui resultados mais divergentes com o algoritmo padrão (LIA) porque não acopla os sub-fluxos. Deste modo o atraso no caminho *A* interfere menos no caminho *B* e o mesmo consegue encaminhar mais pacotes. Mas isto também interfere no tempo de permanência do pacote na fila. Conforme Figura 5.b, este tempo é maior que os demais na maioria dos casos observados.

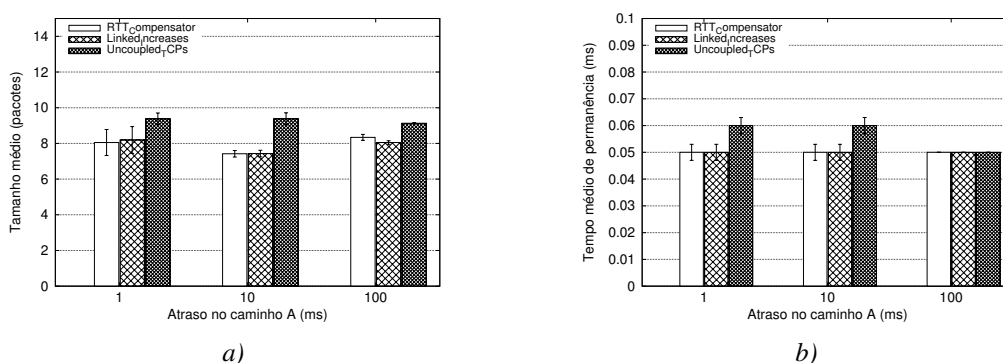


Figura 5. Estado da fila com CoDel vs. MPTCP CC

O CoDel, a partir deste ponto denominado FIFO-CoDel, reduz drasticamente o RTT dos caminhos em relação à disciplina padrão DropTail, contudo, penaliza o *goodput* da transmissão devido ao elevado número de descartes na fila. Este *trade-off* entre o *goodput* e a latência é percebido desde a proposta inicial da disciplina FIFO-CoDel em [Nichols and Jacobson 2012], porém com o MPTCP, se demonstra mais expressivo. Com intuito de reduzir este *tradeoff*, a próxima seção apresenta o LIFO-CoDel.

5. LIFO-CoDel: Um Nova Disciplina de Fila

Esta seção apresenta LIFO-CoDel, uma nova proposta para reduzir o impacto causado pelo mecanismo de controle de fila no CC do MPTCP. A estratégia empregada pela proposta altera o escalonamento dos pacotes FIFO (*First In, First Out*), utilizada originalmente pelo FIFO-CoDel, para LIFO (*Last In, First Out*) além dos demais ajustes detalhados a seguir. Esta proposta tem seus fundamentos no trabalho de [Maurer 2015] que

utiliza uma estratégia denominada LIFO adaptativo com CoDel para controlar a sobrecarga de requisições nos servidores do Facebook. Neste trabalho, a proposta LIFO-CoDel é projetada para um contexto original de redes heterogêneas sem fio, as quais precisam de soluções específicas capazes de tratar suas características peculiares em conjunto com os requisitos de suas aplicações.

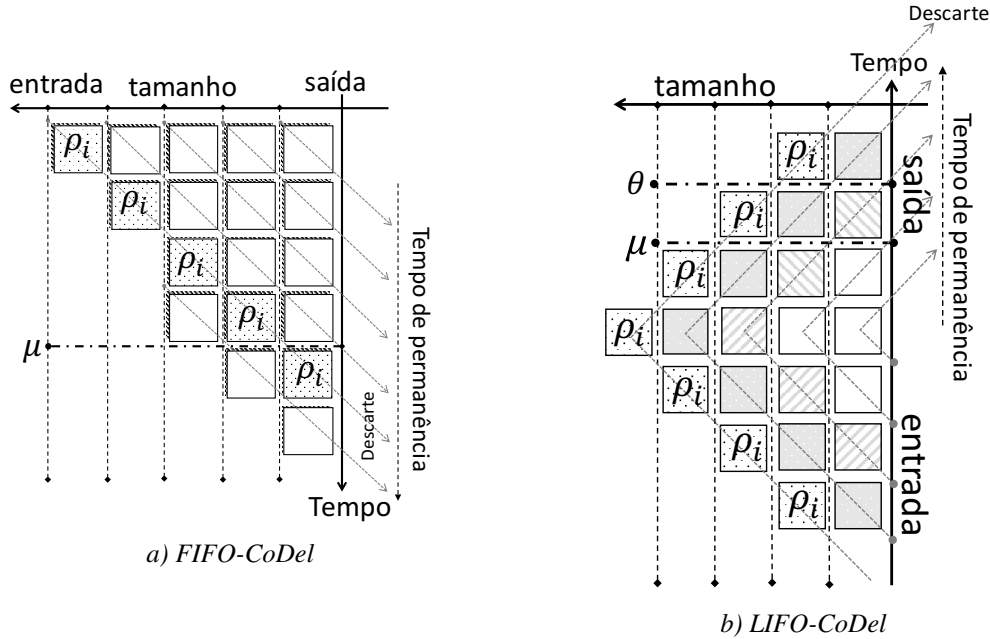


Figura 6. Atraso na fila vs. pilha

A fim de mostrar a diferença entre FIFO e LIFO, a Figura 6 ilustra em um plano cartesiano de tamanho da fila vs. tempo, a entrada, permanência e saída de pacotes das estruturas fila e pilha. Na disciplina FIFO-CoDel, conforme a fila cresce, o atraso do pacote aumenta. O parâmetro τ controla o atraso do pacote e o μ o tamanho da fila. Quando o pacote ρ_i é desenfileirado ele somente é descartado se δ_i é maior que τ e o instante $t = deq_i$ for maior que μ . Este último se ajusta dinamicamente conforme o número de descartes para controlar o tamanho da fila. Isto permite que pacotes com um tempo de permanência δ_i maiores que τ sejam encaminhados até que se entre novamente em estado de descarte. Contudo, na seção 4.1, os resultados indicam que este parâmetro não é efetivo. Este mecanismo não controla o tamanho da fila de modo que o canal seja ocupado de forma eficiente. Por este motivo LIFO-CoDel propõe duas principais alterações: alterar a fila para pilha, a fim de priorizar os pacotes com menor tempo de permanência; um novo parâmetro para controlar os descartes na fila.

A alteração da estrutura fila para pilha permite priorizar a retirada dos pacotes com menor tempo de permanência, ou seja os pacotes mais recentes. Considerando que o mecanismo esteja em estado de descarte, na Figura 6.b o pacote p_i seria descartado porque seu tempo de permanência é maior que τ e $deq_i > \mu$. Contudo, LIFO-CoDel utiliza o parâmetro θ para determinar se o pacote será descartado. O parâmetro θ representa a razão entre o tempo de permanência máximo δ_{max} (Eq. 4) e o tempo de permanência médio $\bar{\delta}$ (Eq. 5) para o n -ésimo pacote retirado da pilha, dado pela Eq. 6. Se $\bar{\delta} = 0$, então se assume $\theta = 0$. Juntamente com o θ , é calculado Γ_t , que se refere a diferença entre o

tempo de permanência atual e o anterior (Eq. 7). O valor de Γ_t é utilizado para incrementar ou reiniciar o valor da variável k (Eq. 8). Quando $\Gamma_t > 0$, assumindo a estrutura pilha, refere-se a um pacote já na pilha antes da última retirada, porém pode indicar também um atraso maior do sistema em relação à última retirada. No entanto, o valor de k e θ é reinicializado quando $\Gamma_t < 0$, ou seja, pacote atual possui um tempo de permanência menor que o anterior. Por fim, o pacote p_i somente é descartado se $k > \theta$.

$$\delta_{max} = \max(\delta_1, \dots, \delta_n) \quad (4)$$

$$\bar{\delta} = \frac{1}{n} \sum_{i=1}^n \delta_i \quad (5)$$

$$\theta = \frac{\delta_{max}}{\bar{\delta}}, \bar{\delta} \neq 0 \quad (6)$$

$$\Gamma_t = \delta_i - \delta_{i-1} \quad (7)$$

$$k = \begin{cases} k + 1, & \text{if } \Gamma_t > 0 \\ 0, & \text{if } \Gamma_t \leq 0 \end{cases} \quad (8)$$

5.1. Resultados

Nesta seção apresentamos os resultados da avaliação da LIFO-CoDel em comparação com os resultados do CoDel e FQ-CoDel. O cenário, as métricas e os parâmetros utilizados nas simulações são os mesmos definidos na Seção 4. Os resultados mostram que o LIFO-CoDel causa um impacto menor no CC do MPTCP, e com isto, melhora o desempenho da transmissão multi-caminhos. Um dos motivos desta melhoria é a redução no número de descartes na fila, ilustrado pela Figura 7.a. Esta redução contribui com o aumento do goodput, conforme Figura 7.b. O LIFO-CoDel tem um *goodput* cerca de 40% maior que FIFO-CoDel, para atraso de 1ms no caminho A. O aumento do atraso ocasiona um descarte maior de pacotes, e isto reduz o *goodput* do LIFO-CoDel. Contudo, com atraso de 100ms os resultados são discretamente melhores que o FIFO-CoDel.

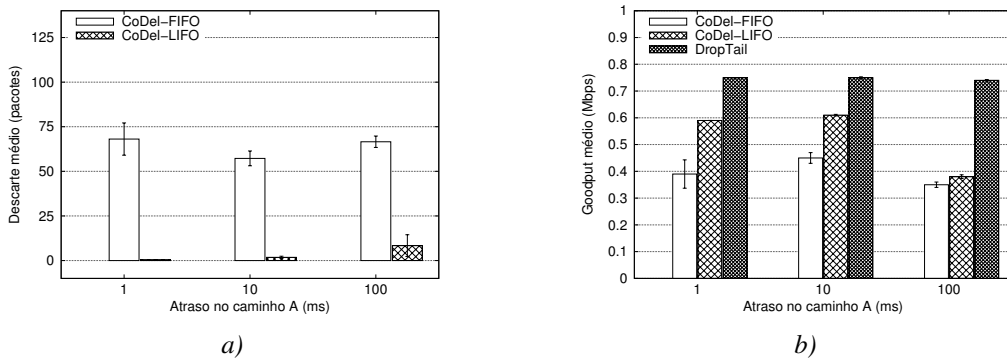


Figura 7. Descartes vs. Goodput

A Figura 8 apresenta os resultados para a variação do tamanho da CWND e RTT do LIFO-CoDel. Assim como CoDel, a janela de congestionamento se manteve em torno de 5 pacotes. Porém, com atraso de 100ms. O algoritmo CC não priorizou o caminho com menor RTT, como observado nos resultados diante do CoDel e DropTail.

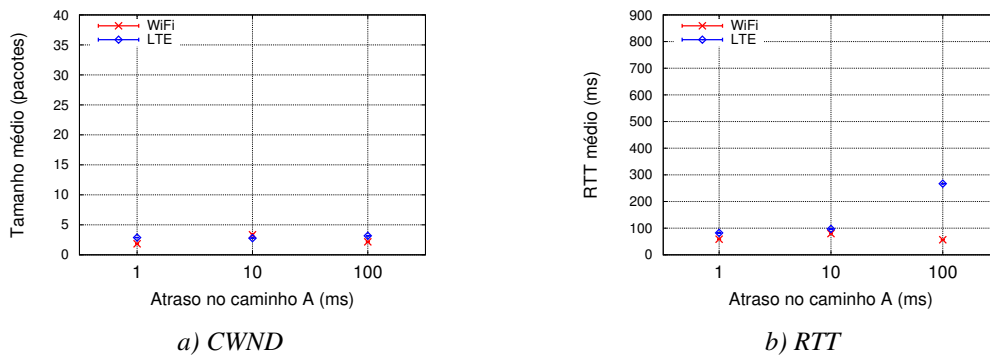


Figura 8. LIFO-CoDEL com LIA

As Figuras 9.a e 9.b apresentam respectivamente o tamanho médio e o tempo médio de permanência na fila sob as disciplinas FIFO-CoDel e LIFO-CoDel. O LIFO-CoDel reduz o tamanho da fila e consequentemente o tempo de permanência. Este resultados se referem às simulações com o algoritmo LIA. Com o algoritmo *RT_Compensator* os resultados variam de 2% a 5%. Com o algoritmo *Uncoupled*, a variação é mais expressiva para o LIFO-CoDel, com valores até 50% acima dos apresentados com *LIA*.

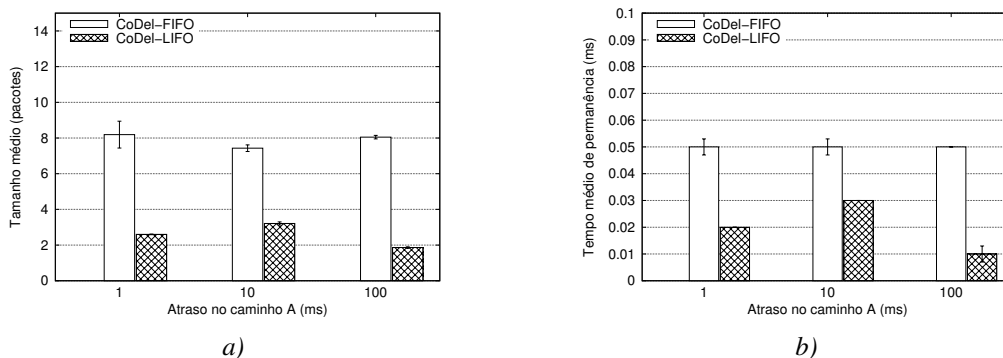


Figura 9. Estado da fila

A fim de compreender o impacto do FIFO e LIFO em cada caminho, realizamos simulações com o FQ-CoDel, que separa os sub-fluxos em filas independentes. Os resultados obtidos para ambas as disciplinas tiveram uma variação muito discreta, para o *goodput* ficou entre 2% a 3%. Este resultados indicam que a separação em filas distintas não muda o fato que ambos os sub-fluxos compartilham o mesmo gargalo.

6. Conclusões

Este trabalho analisou o impacto de disciplinas AQM no CC do MPTCP. Os resultados comparando as disciplinas DropTail, CoDel e FQ-CoDel em relação aos algoritmos *Uncoupled*, *LIA* e *RTT_Compensator* demonstram que existe um *tradeoff* entre RTT e vazão. Diante de *buffers* de grande capacidade, o DropTail não descarta pacotes e a fila cresce acarretando um alto valor de RTT (fenômeno *bufferbloat*). Porém a transmissão alcança a utilização máxima do canal. O CoDel resolve o problema dos atrasos, porém, os descartes impactam no CC do MPTCP e isto reduz a utilização do canal de transmissão. No cenário de redes heterogêneas sem fio, este aspecto tem impacto ainda maior. Ao aumentar o atraso em um dos caminhos se observou uma redução na vazão e aumento

do RTT em ambos os caminhos. Com base nestas observações, este trabalho apresentou a proposta de uma nova disciplina de filas FIFO-CoDel, qual reduz o impacto causado pelo mecanismo de controle de fila no CC do MPTCP com base na disciplina CoDel em cenários de redes heterogêneas sem fio. Os resultados da sua avaliação mostram que a redução no número de descartes na fila contribuiu para uma melhoria em métricas como *goodput*, RTT e tempo de permanência na fila.

Como trabalhos futuros, pretende-se avaliar novos algoritmos de CC propostos para o MPTCP no cenário de redes heterogêneas sem fio. Adicionalmente, incluir avaliações que considerem as perdas em decorrência da mobilidade e transição vertical entre as redes. Por fim, ajustar e formalizar matematicamente o parâmetro θ , utilizado pelo LIFO-CoDel, para obter uma melhor resposta em cenários com caminhos heterogêneos que apresentem um atraso acima de $100ms$.

Referências

- Alfredsson, S., Del Giudice, G., Garcia, J., Brunstrom, A., De Cicco, L., and Mascolo, S. (2013). Impact of TCP congestion control on bufferbloat in cellular networks. In *IEEE WoWMoM*, páginas 1–7.
- Ali, H., Khan, S., and Quaid, M. (2015). Comparative analysis of controlled delay (CoDel) with Deficit Round Robin (DRR) to overcome bufferbloat problem in wired network. *International Journal of Current Engineering and Technology*, 5(5):3378–3386.
- Arzani, B., Gurney, A., Cheng, S., Guerin, R., and Loo, B. T. (2014). Impact of path characteristics and scheduling policies on MPTCP performance. In *IEEE WAINA*, páginas 743–748.
- Barré, S., Bonaventure, O., Raiciu, C., and Handley, M. (2011). Experimenting with multipath TCP. *ACM SIGCOMM*, 41(4):443–444.
- Chen, Y.-C., Lim, Y.-s., Gibbens, R. J., Nahum, E. M., Khalili, R., and Towsley, D. (2013). A measurement-based study of multipath TCP performance over wireless networks. In *ACM IMC*, páginas 455–468.
- Chen, Y.-C. and Towsley, D. (2014). On bufferbloat and delay analysis of multipath TCP in wireless networks. In *IEEE IFIP*, páginas 1–9.
- Ferlin, S., Dreibholz, T., and Alay, O. (2014). Multi-path transport over heterogeneous wireless networks: Does it really pay off? In *IEEE GLOBECOM*, páginas 4807–4813.
- Ferlin-Oliveira, S., Dreibholz, T., and Alay, O. (2014). Tackling the challenge of bufferbloat in multi-path transport over heterogeneous wireless networks. In *IEEE IWQoS*, páginas 123–128.
- Ford, A., Raiciu, C., Handley, M., and Bonaventure, O. (2013). TCP extensions for multipath operation with multiple addresses. RFC 6824, IETF.
- Gómez, D., Garrido, P., Rabadan, C., Agüero, R., and Muñoz, L. (2014). TCP performance enhancement over wireless mesh networks by means of the combination of multi-RAT devices and the MPTCP protocol. *Network Protocols and Algorithms*, 6(3):56–81.

- Hoeiland-Joergensen, T., McKeeney, P., Taht, D., Gettys, J., and Dumazet, E. (2015). FlowQueue-CoDel. Internet-Draft draft-hoeiland-joergensen-aqm-fq-codel-00.txt, IETF.
- Hossain, E. and Hasan, M. (2015). 5G cellular: key enabling technologies and research challenges. *IEEE Instrumentation & Measurement Magazine*, 18(3):11–21.
- Jeanette Wannstrom, K. M. (2014). HetNet, Small Cells. TS 1576, 3rd Generation Partnership Project (3GPP).
- K. Nichols, V. J. (2014). Controlled Delay Active Queue management. RFC draft 6824, IETF.
- Khalili, R., Gast, N., Popovic, M., and Le Boudec, J.-Y. (2013). MPTCP is not pareto-optimal: Performance issues and a possible solution. *IEEE/ACM Networking*, 21(5):1651–1665.
- Kheirkhah, M. (2015). Multipath TCP in NS-3. Código fonte do projeto disponível em <http://dx.doi.org/10.5281/zenodo.32691>.
- Kulatunga, C., Kuhn, N., Fairhurst, G., and Ros, D. (2015). Tackling bufferbloat in capacity-limited networks. In *IEEE EuCNC*, páginas 381–385.
- Kurosaka, T. and Bandai, M. (2015). Multipath TCP with multiple ACKs for heterogeneous communication links. In *IEEE CCNC*, páginas 613–614.
- Lee, J. S. and Lee, J. (2015). Multipath TCP performance improvement in mobile network. In *IEEE ICUFN*, páginas 710–714.
- Maurer, B. (2015). Fail at scale. *ACM Queue*, 13(8):30:30–30:46.
- Mehani, O., Holz, R., Ferlin, S., and Boreli, R. (2015). An early look at multipath TCP deployment in the wild. In *ACM HotPlanet*, páginas 7–12.
- Nichols, K. and Jacobson, V. (2012). Controlling Queue Delay. *ACM Queue*, 10(5):20:20–20:34.
- Raiciu, C., Barre, S., Pluntke, C., Greenhalgh, A., Wischik, D., and Handley, M. (2011). Improving datacenter performance and robustness with multipath TCP. *ACM SIGCOMM*, 41(4):266–277.
- Raiciu, C., Paasch, C., Barre, S., Ford, A., Honda, M., Duchene, F., Bonaventure, O., and Handley, M. (2012). How hard can it be? designing and implementing a deployable multipath TCP. In *USENIX*, páginas 29–29.
- Singh, A., Xiang, M., Konsgen, A., Goerg, C., and Zaki, Y. (2013). Enhancing fairness and congestion control in multipath TCP. In *IEEE WMNC*, páginas 1–8.
- Singh, S. K., Das, T., and Jukan, A. (2015). A survey on internet multipath routing and provisioning. *IEEE Communications Surveys & Tutorials*, 17(4):2157–2175.
- Zhou, D., Song, W., Wang, P., and Zhuang, W. (2015). Multipath TCP for user cooperation in LTE networks. *IEEE Network*, 29(1):18–24.