

Implementação e Análise Prática de Desempenho do Mecanismo MARA em Redes em Malha Sem Fio

Diego Passos, Célio Albuquerque

Instituto de Computação – Universidade Federal Fluminense (UFF)
Niterói – RJ – Brasil

{dpassos, celio}@ic.uff.br

Abstract. *This paper provides a deeper analyses of the Metric Aware Rate Adaptation (MARA) mechanism, a joint approach for routing metrics and automatic rate adaptation on wireless mesh networks. The theoretical aspects of MARA and preliminary simulation results are presented in [18]. In this paper, we conduct an additional set of simulations, implement MARA on off the shelf routers and present results of experiments performed on a real wireless mesh network. We also discuss the practical issues of our implementation of MARA. Results confirm the good performance of MARA in comparison to other proposals, especially in scenarios with large number of hops.*

Resumo. *Este artigo apresenta uma análise mais profunda do mecanismo Metric Aware Rate Adaptation (MARA), uma abordagem unificada para os problemas de métrica de roteamento e adaptação automática de taxa em redes em malha sem fio. Os aspectos teóricos do MARA e resultados preliminares de simulação são apresentados em [18]. Neste artigo, são apresentados um conjunto adicional de simulações, uma implementação do MARA em roteadores comerciais, além de resultados de experimentos conduzidos em uma rede real. Também são discutidos os aspectos práticos da implementação do MARA. Os resultados obtidos confirmam o bom desempenho do MARA em relação a outras propostas, especialmente em cenários com grande número de saltos.*

1. Introdução

Redes em malha sem fio, ou *mesh*, são compostas por roteadores *mesh* e clientes *mesh*. Os roteadores *mesh* formam uma malha de enlaces sem fio que é utilizada para comunicação em múltiplos saltos pelos clientes *mesh*. Cada roteador *mesh* trabalha como um ponto de acesso, servindo aos clientes como um *gateway* para o restante da rede.

Nos últimos anos, as redes em malha sem fio têm apresentado um enorme aumento em popularidade, em grande parte devido ao seu potencial baixo custo de implantação e manutenção. Muitas empresas já contam com soluções *mesh*, embora seus custos ainda sejam, em geral, altos [23, 25]. No entanto, existem algumas soluções comerciais de baixo custo especificamente direcionadas aos usuários finais [21, 22]. Além disso, existem ainda soluções abertas bastante consolidadas geralmente aplicadas à projetos de inclusão digital [19, 24].

Para lidar com o requisito de baixo custo, as redes em malha sem fio geralmente empregam dispositivos baseados no padrão IEEE 802.11. Estes dispositivos são capazes de operar a múltiplas taxas de transmissão, variando de 1 Mb/s a 54 Mb/s. No entanto,

selecionar a taxa de transmissão mais adequada não é trivial, dado que existe um compromisso entre a capacidade de um enlace e a taxa de perda de pacotes. Tipicamente, a probabilidade de perda de pacotes aumenta com o aumento da taxa de transmissão.

Existem na literatura diversos algoritmos de adaptação automática de taxa [1, 2, 5, 6, 9, 11, 12, 14, 15, 20, 26], sendo que a grande maioria utiliza os próprios quadros de dados para gerar estatísticas sobre a qualidade dos enlaces. O problema com esta abordagem é que a existência de dados estatísticos depende da existência de tráfego na rede.

Além disso, os roteadores *mesh* precisam implementar um mecanismo dinâmico de descoberta de rotas. Um dos elementos principais deste mecanismo é a métrica de roteamento. Embora existam diversas métricas de roteamento com formulações bastante coerentes [3, 4, 7, 8, 13], todas elas enfrentam o obstáculo de obter informações consistentes e confiáveis sobre a qualidade dos enlaces sem fio.

Estes desafios motivaram a proposta do mecanismo MARA. O MARA [18] é um mecanismo unificado para a solução dos problemas de adaptação automática de taxa e métrica de roteamento. Através de uma abordagem *cross-layer*, o MARA realiza decisões conjuntas de roteamento e seleção de taxa, garantindo que ambas sejam coerentes entre si. Um mecanismo mais preciso para a inferência de informações acerca da qualidade dos enlaces também faz parte da proposta.

A avaliação preliminar realizada em [18] demonstrou o potencial do mecanismo. Nos dois cenários de simulação apresentados, o MARA se mostrou bastante superior a diversas combinações de algoritmos de adaptação de taxa e métricas de roteamento da literatura.

Neste artigo, são apresentados novos resultados de desempenho do MARA. Os cenários de simulação utilizados são estendidos para avaliar como o MARA se comporta em redes com um maior número de nós. Este artigo também apresenta uma implementação prática do MARA em uma rede real. Especificamente, neste trabalho são abordadas as soluções encontradas para os desafios da implementação do mecanismo. Tal implementação foi utilizada para a realização de experimentos práticos com o MARA, com a finalidade de validar os resultados de simulação obtidos anteriormente.

Além de implementar e avaliar o mecanismo MARA original, neste artigo são avaliadas as ideias de otimização propostas em [18]. Duas variações do MARA, denominadas MARA-P e MARA-RP são apresentadas e avaliadas. Estas variações levam em consideração o tamanho dos pacotes para efeito de seleção de taxa de transmissão (no caso do MARA-P e do MARA-RP) e para efeito de seleção de rotas (apenas no MARA-RP).

Os resultados da avaliação mostram uma grande melhora de desempenho em termos de vazão e atraso fim a fim com a utilização do MARA. Em alguns cenários, esta melhora se manifesta também em termos de redução na perda de pacotes. Os experimentos reais também mostram melhora com a utilização do MARA. Em relação ao desempenho das otimizações, os resultados não mostram ganhos consideráveis, quando comparadas à versão original do MARA.

O restante do artigo está organizado da seguinte forma. A Seção 2 apresenta brevemente o funcionamento do mecanismo MARA e de suas otimizações, MARA-P e

MARA-RP. Na Seção 3 são apresentados resultados de cenários adicionais de simulação. A Seção 4 discute os aspectos práticos da implementação do MARA realizada para este trabalho. A Seção 5 apresenta os resultados de desempenho em uma rede real. Esta seção é dividida em duas partes: os resultados dos experimentos com o MARA e dos experimentos com suas otimizações. Finalmente, na Seção 6 são apresentadas as considerações finais do artigo.

2. O Mecanismo MARA

O MARA modela o custo de cada enlace da rede como o tempo total necessário para a transmissão bem sucedida de um quadro. Para isso, o peso atribuído a cada enlace $a \rightarrow b$ da rede é dado pela expressão:

$$MARA_{ab} = \min_i \left(\frac{ETX_{ab}^i \cdot ps}{\lambda_i} \right), \quad (1)$$

onde λ_i representa a i -ésima taxa de transmissão disponível e ETX_{ab}^i representa o número médio de retransmissões necessárias até o sucesso no envio de um pacote de tamanho ps pelo enlace $a \rightarrow b$ utilizando a taxa λ_i .

Esta expressão de custo é uma versão simplificada da expressão apresentada na proposta original do MARA [18]. Nesta formulação, os tempos de *backoff* e *timeout* do protocolo de acesso ao meio são desconsiderados. Resultados preliminares de simulações executadas nos mesmos cenários considerados em [18] mostram que esta simplificação não afetou o comportamento do MARA na prática.

O número médio de retransmissões é dado pela métrica ETX do enlace [7], com base na probabilidade de sucesso na transmissão de um pacote por este enlace. No entanto, em sua formulação original, a métrica ETX infere esta probabilidade apenas na taxa de transmissão básica (em geral, uma taxa baixa, quando comparada à taxa utilizada para transmissão de dados). Um problema com a métrica ETX e com várias outras derivadas a partir dela [8, 13] é justamente o fato de que a avaliação da qualidade do enlace é realizada na taxa básica, enquanto os pacotes de dados são, em geral, enviados a uma taxa de transmissão superior. Esta incoerência leva a uma avaliação demasiadamente otimista dos enlaces, já que, em geral, quanto mais baixa a taxa de transmissão, mais robusta ela é e menor é a probabilidade de perda de pacotes. Como resultado, tais métricas apresentam resultados sub-ótimos.

Alguns trabalhos tentam contornar este problema através da inferência da qualidade dos enlaces em todas as taxas de transmissão disponíveis [3]. No entanto, como esta inferência é realizada através do envio periódico de pacotes de *probe*, realizá-la em todas as taxas de transmissão significa um aumento considerável no *overhead* do protocolo de roteamento.

Para solucionar este problema de maneira escalável, o MARA utiliza um método de conversão das probabilidades de sucesso dos enlaces para as diferentes taxas de transmissão. São utilizados pacotes de *probe* em 4 taxas de transmissão diferentes (1, 18, 36 e 54 Mb/s). Para inferir a probabilidade de sucesso do enlace nas demais taxas, o MARA utiliza uma função que relaciona a taxa de erro de pacotes (ou PER, *Packet Error Rate*) e o SNR do canal:

$$PER(SNR) = \frac{1 - \operatorname{erf}\left(\frac{SNR-a}{b\sqrt{2}}\right)}{2}, \quad (2)$$

onde erf representa a *função erro*.

A expressão, derivada a partir de dados empíricos, depende dos parâmetros a e b . Estes parâmetros são dependentes da taxa de transmissão escolhida e do tamanho do pacote considerado. Em [18] são apresentados os valores correspondentes a cada taxa de transmissão dos padrões IEEE 802.11b e IEEE 802.11g para diversos tamanhos de pacote.

2.1. Otimizações

Além da proposta original do MARA, em [18] também é sugerida a avaliação de duas possíveis otimizações do mecanismo. A ideia é levar em consideração o tamanho dos pacotes a serem transmitidos para realizar escolhas mais apropriadas de taxa de transmissão e rotas. Embora [18] não apresente uma definição formal do funcionamento destas otimizações, neste artigo elas foram implementadas para efeito de avaliação.

A otimização MARA-P leva em consideração o tamanho do pacote a ser transmitido para a escolha da taxa de transmissão. A ideia é definir classes de tamanhos de pacotes (*i.e.*, intervalos de tamanhos de pacote) e utilizar a Equação 1 para obter os valores de taxa de transmissão para cada classe. Neste caso, o valor da métrica ETX utilizado na expressão seria computado de acordo com o tamanho de pacote considerado.

No momento em que o protocolo de roteamento dispara o cálculo das métricas dos enlaces, o MARA-P verifica quais são as classes de tamanho definidas. Para as implementações realizadas neste trabalho foram utilizadas 4 classes: até 350 bytes, entre 351 e 750 bytes, entre 751 e 1300 bytes e acima de 1300 bytes. Para cada classe, o MARA-P escolhe o limite superior como valor representante do intervalo (no caso do último intervalo, optou-se por escolher o valor de 1500 bytes) e calcula o valor da Equação 1, como na versão original do MARA.

O resultado desta computação é uma taxa de transmissão para cada uma das classes de tamanho de pacote. Estas taxas são armazenadas em uma tabela para uso nas transmissões de dados futuras. Quando um novo pacote deve ser enviado, o nó consulta esta tabela verificando a qual classe o pacote pertence. A taxa de transmissão adequada é lida e informada à interface de rede, que prossegue com o envio do pacote.

A segunda otimização implementada, denominada MARA-RP, estende o conceito utilizado pelo MARA-P também para seleção das melhores rotas. O procedimento do MARA-RP é idêntico ao do MARA-P, exceto pelo resultado do cálculo das métricas. Além de obter uma taxa de transmissão para cada classe de tamanho, são armazenados também os custos associados a cada uma delas. De posse destes custos para cada enlace, o MARA-RP computa uma tabela de roteamento para cada classe de tamanho. No momento de decidir qual é o próximo salto de um determinado pacote, o nó determina qual é a tabela de roteamento adequada àquele tamanho de pacote e faz a consulta. A seleção de taxa de transmissão é realizada de maneira idêntica à descrita para o MARA-P.

3. Simulações

Nas simulações apresentadas em [18], o MARA obteve resultados consideravelmente melhores que diversas combinações de algoritmos de adaptação de taxa e métricas de rotea-

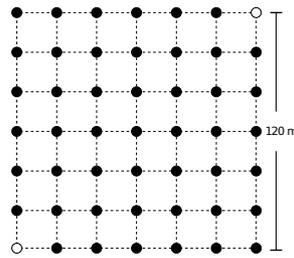


Figura 1. Representação da topologia em Grade.

mento da literatura. Em especial, com o aumento da distância geográfica entre a origem e o destino, os resultados sugerem que o ganho percentual no desempenho do MARA em relação às demais combinações aumenta.

Para confirmar esta hipótese neste artigo, foram realizadas simulações nos mesmos moldes das apresentadas em [18], porém em cenários diferentes. Todas as simulações consistem em um fluxo TCP enviado entre dois pares de nós da rede. As simulações têm duração total de 430 segundos, sendo que os 130 primeiros são utilizados para convergência do protocolo de roteamento (*i.e.*, o fluxo TCP só é iniciado após este período). O modelo de propagação utilizado para todos os cenários foi o *Shadowing*. Embora este modelo seja probabilístico, com os parâmetros utilizados, pode-se estimar que o raio de transmissão resultante a taxa de 1 Mbps varie em torno de 56 metros.

Um dos cenários avaliados é composto por 49 nós colocados nos vértices de uma grade quadrada de 120 metros de lado, como representado na Figura 1. Cada nó, portanto, fica distante 20 metros dos seus vizinhos imediatos de linha ou coluna. Em cada rodada de simulação foi gerado um fluxo de dados entre os nós 0 e 48, os mais distantes geograficamente (destacados em branco na Figura 1).

O gráfico da Figura 2 mostra os resultados de vazão média obtidos na topologia em grade, bem como os respectivos intervalos de 95% de confiança. Claramente o MARA foi bastante superior às demais propostas. De fato, o MARA obteve uma vazão quase 9 vezes maior que a da segunda melhor combinação (métrica ML [4] e algoritmo SNR [18]). A maior parte das combinações, alias, obteve resultados muito ruins. Exceto pelo MARA, apenas 4 combinações atingiram uma vazão maior que 10 Kbps.

Este resultado corrobora com a hipótese de que o desempenho do MARA em

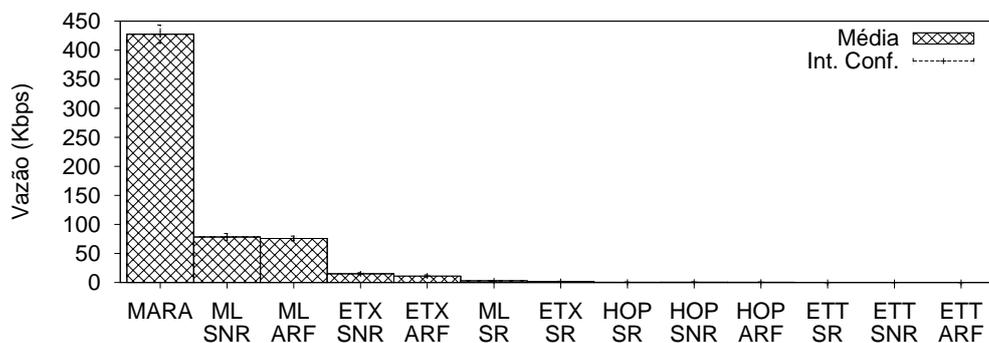


Figura 2. Resultados de vazão TCP na topologia em grade.

relação às demais propostas é especialmente relevante nos cenários em que os nós de origem e destino do fluxo de dados estão distantes geograficamente. Para efeito de comparação, nos resultados apresentados em [18], a maior diferença entre o MARA e as demais combinações foi de cerca de 4 vezes, enquanto neste cenário a diferença chegou a 9 vezes.

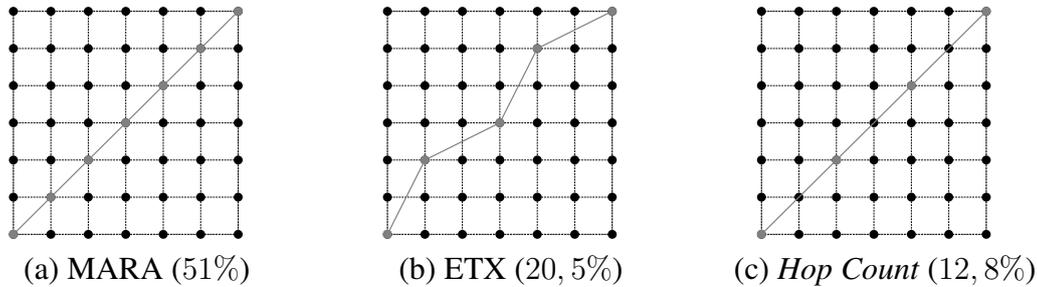


Figura 3. Rotas preferenciais de cada métrica na topologia em grade.

Ao analisar mais profundamente as simulações realizadas para compreender melhor a disparidade entre os resultados obtidos pelo MARA e pelas demais propostas, chamaram atenção as rotas preferenciais escolhidas por cada métrica de roteamento e pelo MARA. A Figura 3 mostra as rotas que foram utilizadas na maior parte do tempo de simulação pelas métricas ETX, *Hop Count* e pelo MARA. As legendas mostram também o percentual de tempo em que tais rotas foram utilizadas.

A rota preferencial escolhida pelo MARA utiliza 6 enlaces de mesmo comprimento, passando apenas por nós da diagonal da grade. Uma escolha similar foi realizada pela métrica *Hop Count*. No entanto, o caminho selecionado pela métrica *Hop Count* utiliza apenas três saltos, obviamente mais longos que os escolhidos pelo MARA. Esta opção por enlaces mais longos (e por isso, provavelmente, de pior qualidade) resultou no baixo desempenho das combinações que utilizam a *Hop Count* como métrica. Já a métrica ETX optou por um caminho de 4 saltos, ligeiramente mais curtos que os selecionados pela *Hop Count*. Mesmo assim, o desempenho da combinação em questão (ETX e SNR) ficou bastante aquém da vazão obtida pelo MARA.

Outros dois cenários também foram avaliados. No primeiro, uma variação do cenário em grade, 49 nós foram espalhados aleatoriamente em uma área quadrada de 120 metros de lado. Já no segundo, foi feita uma extrapolação da uma rede em malha real, totalizando 20 nós. Os resultados obtidos nestes cenários, omitidos por restrições de espaço, são similares aos da topologia em grade. A vazão média obtida pelo MARA foi de 3 a 5 vezes mais alta que a de qualquer outra proposta avaliada. Os resultados para todos os cenários avaliados podem ser encontrados em [17].

Para avaliar o desempenho das otimizações MARA-P e MARA-RP, foram utilizadas simulações ligeiramente diferentes. Ao invés de utilizar um único fluxo TCP, foram utilizados 3 fluxos UDP, com datagramas de tamanhos diferentes. A ideia é que MARA-P e MARA-RP pudessem ter desempenho melhor que a proposta original especialmente para os fluxos de pacotes menores.

Os resultados de simulação, no entanto, não mostraram uma diferença considerável de desempenho em relação à proposta original do mecanismo MARA. O gráfico

da Figura 4 ilustra o comportamento obtido na maior parte das simulações. Para quase todos os cenários e fluxos os resultados de atraso foram muito próximos para as três variações. Um resultado interessante, no entanto, é o desempenho exatamente igual das duas otimizações em todas as simulações. Este resultado sugere que, no cenário avaliado, o tamanho do pacote não tem influência na escolha de rotas, ao menos no modelo adotado pelo MARA.

4. Aspectos de Implementação

A implementação real do MARA utilizada para os testes de desempenho apresentados aqui foi realizada em uma das redes em malha do projeto ReMoTE [19]. Esta rede é formada por roteadores comerciais *Linksys* WRT54G, utilizando um *firmware* customizado baseado na distribuição *Linux* OpenWRT [16].

Como o MARA participa tanto da adaptação automática de taxas, quanto da camada de roteamento, a implementação do algoritmo nesta rede foi realizada em dois módulos: um módulo de roteamento e outro de seleção de taxa. O módulo de roteamento é responsável por coletar informações sobre os enlaces, atribuir pesos a cada um deles, selecionar as taxas de transmissão ótimas e informá-las ao módulo de seleção de taxas. Este segundo módulo, por sua vez, armazena as taxas ótimas em uma tabela e se encarrega de analisar cada pacote antes de sua transmissão para configurar a interface na taxa correta.

A porção métrica do mecanismo MARA pode, a princípio, ser implementada em qualquer protocolo de roteamento baseado em estado de enlaces ou vetor de distâncias. Embora existam muitos protocolos otimizados para as redes sem fio de múltiplos saltos, neste trabalho optou-se pela utilização de um protocolo simples, que apenas implementa a funcionalidade básica do algoritmo de estado de enlaces. Este protocolo daqui em diante será referenciado como SLSP (*Simple Link State Protocol*).

O módulo de seleção de taxa tem por objetivo criar uma interface de comunicação entre o protocolo de roteamento e a interface de rede sem fio, de modo a prover as funcionalidades de seleção de taxa necessárias ao funcionamento do MARA. Este módulo é necessário pela limitação das configurações de taxa de transmissão nos *drivers* das interfaces de rede sem fio disponíveis no mercado. Em geral, não é possível especificar taxas

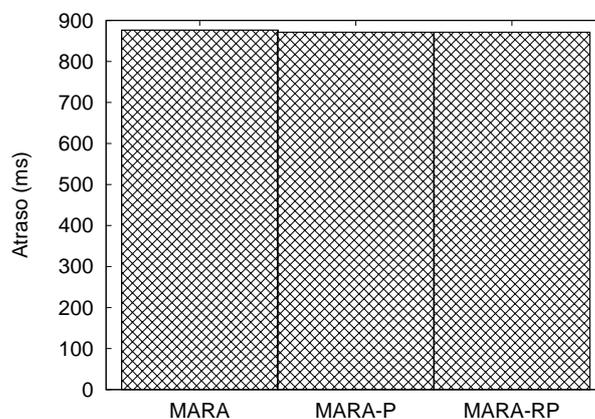


Figura 4. Comparativo de atraso UDP entre o MARA e suas otimizações.

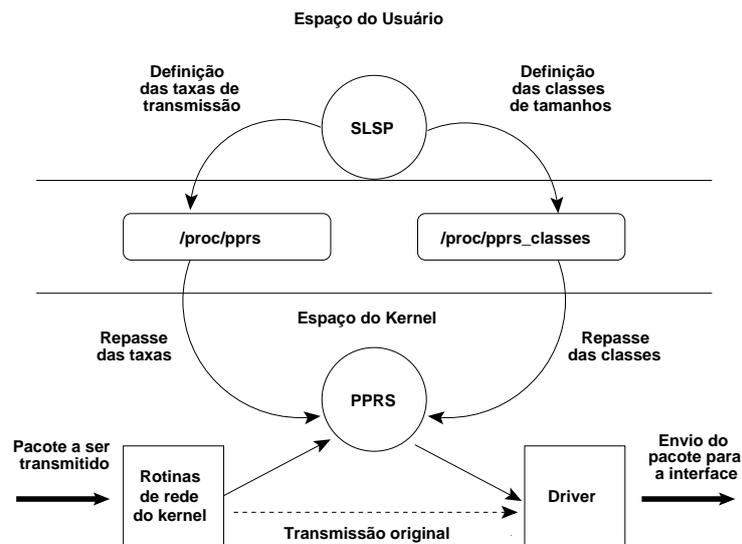


Figura 5. Esquema de funcionamento do módulo PPRS.

diferentes para cada vizinho.

Especificamente, esse é o caso dos roteadores utilizados na implementação do MARA. Não existe um mecanismo no *driver* que permita que o SLSP informe à interface quais as taxas de transmissão para cada vizinho. De fato, o *driver* permite apenas a especificação de uma taxa de transmissão.

A solução encontrada foi a criação de um módulo para o *kernel* do *Linux* utilizado pelo OpenWRT que mantenha uma tabela de seleção de taxas, monitore os envios de pacotes, verifique a taxa mais apropriada (segundo a tabela) e requisiute uma alteração na taxa atual da interface. O módulo criado, denominado PPRS (*Per-Packet Rate Selection*), tem seu funcionamento ilustrado na Figura 5. Quando carregado, este módulo cria dois arquivos no diretório `/proc`. No *Linux*, os arquivos deste diretório servem como um canal de comunicação entre o *kernel* e o espaço do usuário. No caso específico do PPRS, estes dois arquivos permitem a definição de classes de tamanho de pacote e de regras de seleção de taxas de transmissão. Essas definições são realizadas através da simples escrita em um destes arquivos.

No instante do carregamento do módulo PPRS, ele interrompe o caminho normal de transmissão de pacotes pela interface de rede sem fio e se coloca como um intermediador. Usando desse expediente, o PPRS é capaz de interceptar os pacotes imediatamente antes deles chegarem ao *driver*. Neste momento, o pacote é analisado (especificamente, o endereço do próximo salto e o tamanho), a tabela de taxas é consultada e, se necessário, uma requisição de alteração da taxa é feita ao *driver*. Após esses passos, o pacote é passado para a função de transmissão do *driver*, que completará o processo.

A implementação da otimização MARA-RP necessita da definição de mais de uma tabela de roteamento (uma para cada classe de tamanho de pacote utilizada). Tal funcionalidade é disponibilizada pelo *kernel* do *Linux*, através do pacote *iproute2*. No entanto, para utilizar uma tabela de roteamento específica, é necessário criar *regras de encaminhamento* que relacionam características dos pacotes aos identificadores de cada

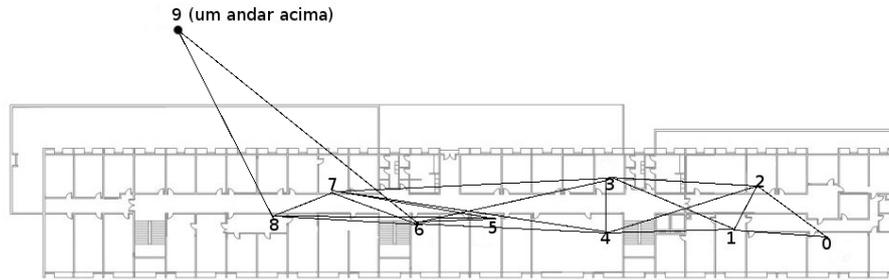


Figura 6. Representação da topologia do projeto ReMoTE.

tabela. Dentre as características permitidas, está uma marcação lógica do pacote (interna ao *kernel*) que pode ser criada através da ferramenta *iptables*.

De forma resumida, a implementação do MARA-RP funciona da seguinte forma. O protocolo SLSP computa as diversas tabelas de roteamento e as informa ao sistema operacional. Ao mesmo tempo, regras são criadas no *iptables* para, antes da decisão de roteamento, colocar a marcação lógica adequada em cada pacote, de acordo com o seu tamanho. São criadas ainda regras de encaminhamento, associando as marcações às suas respectivas tabelas de roteamento. Com estas configurações, o sistema operacional se encarrega de realizar o encaminhamento correto.

5. Avaliação de Desempenho

A topologia de rede utilizada para os experimentos realizados é ilustrada na Figura 6. Esta rede, composta por 10 roteadores *Linksys WRT54g* espalhados por um dos prédios da Escola de Engenharia da Universidade Federal Fluminense, é a atual rede de testes do projeto ReMoTE [19]. De fato, uma das topologias utilizadas nas simulações realizadas em [18] foi modelada a partir desta rede. Logo, os resultados dos experimentos descritos a seguir servem como uma validação do modelo de simulação adotado.

O protocolo SLSP e o módulo PPRS foram instalados em todos os nós da rede. Além do MARA, o SLSP implementa também as métricas ETX, ML e *Hop Count*. Em termos dos algoritmos de adaptação de taxa, infelizmente não foi possível implementar os mesmos utilizados no ambiente de simulação. O *driver* da interface de rede disponível nos roteadores não tem código fonte aberto. Logo, não foi possível implementar outros algoritmos de adaptação de taxa, além do já disponibilizado pelo *driver*.

Embora não haja documentação disponível sobre este algoritmo, analisando seu comportamento, ele se mostrou ser uma versão simplificada do ARF (*Auto Rate Fall-back*). A diferença entre ele e a versão original do ARF é a seleção de uma única taxa de transmissão para todos os vizinhos. Ou seja, ao invés de manter estado para todos os vizinhos, cada nó mantém apenas um conjunto de variáveis do ARF (limiares de sucessos e fracassos consecutivos e temporizadores).

5.1. Resultados do MARA Original

Nos experimentos realizados, foram considerados 4 parâmetros de desempenho: vazão TCP, taxa de perda de pacotes (na camada de aplicação), número médio de saltos e atraso

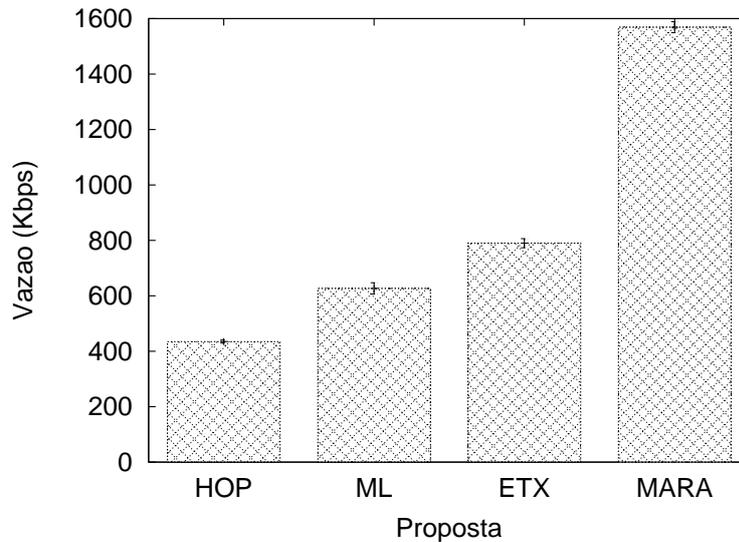


Figura 7. Resultados de vazão TCP nos testes na rede real.

fim a fim.

Para as medições de vazão, foi utilizada a ferramenta *Iperf* [10]. No entanto, dada a extensão da rede utilizada, é inviável obter informações de perda, atraso e número de saltos a partir dos próprios segmentos TCP. Para estimar este tipo de informação, optou-se pela utilização de um fluxo de dados ICMP enviado de forma concorrente através da ferramenta *ping*. Este fluxo fornece as informações de número saltos (a partir do campo TTL), RTT e perda de pacotes.

A Figura 7 ilustra os resultados de vazão média obtidos entre os nós extremos da rede. Assim como verificado nas simulações, o MARA obteve um resultado consideravelmente superior ao das demais propostas. Em relação ao desempenho da segunda melhor proposta neste quesito (da métrica ETX com a versão simplificada do ARF), o MARA obteve uma vazão quase duas vezes maior. Outro resultado similar obtido nestes experimentos foi o mau desempenho da métrica *Hop Count*. Assim como ocorria constantemente nas simulações, esta métrica obteve o pior resultado de vazão.

Do ponto de vista do RTT, o estimador de atraso utilizado nestes experimentos, o MARA também obteve os melhores resultados. A Figura 8 mostra o RTT médio no fluxo ICMP entre os nós 0 e 9 da topologia. De maneira curiosa, a métrica ML obteve um RTT médio mais baixo que a métrica ETX (embora, considerando-se as barras de erro, ambas as propostas tenham obtidos resultados estatisticamente equivalentes).

O gráfico da Figura 9 ilustra o número médio de saltos utilizados por cada métrica ao longo dos fluxos TCP entre os nós 0 e 9 da topologia. Como esperado, a métrica *Hop Count* utilizou caminhos bem mais curtos (em termos de saltos) que as demais. Os dados interessantes deste gráfico são relacionados às demais propostas.

Um resultado interessante foi a manutenção da escolha do MARA por caminhos compostos por 7 saltos. Ao longo de todas as repetições realizadas, o MARA jamais realizou uma escolha de rota com um número menor de saltos (embora caminhos diferentes possam ter sido escolhidos).

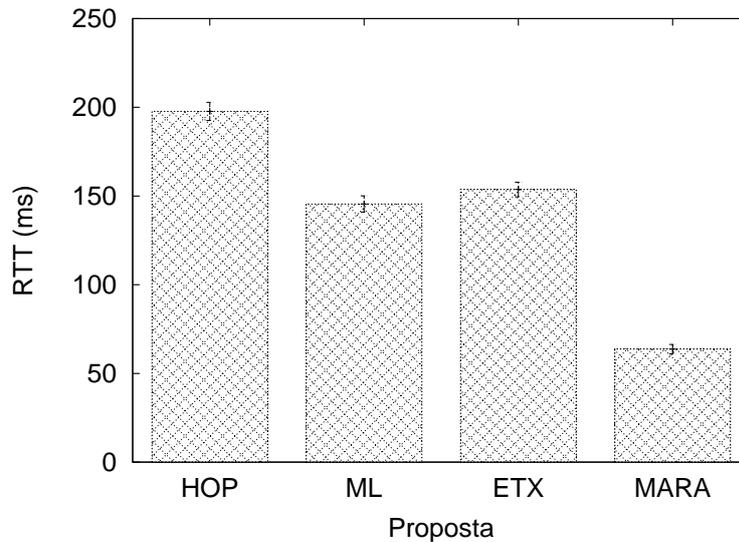


Figura 8. Resultados de RTT médio no fluxo de dados ICMP nos testes da rede real.

Este fato é interessante por dois aspectos. Primeiro, ele confirma a estabilidade das decisões de roteamento realizadas pelo MARA, já verificadas anteriormente em simulações (vide as frequências das rotas preferenciais na Figura 3). A estabilidade de escolhas é uma qualidade interessante em uma métrica, especialmente quando ela é utilizada em conjunto com um protocolo baseado em estado de enlaces, já que isso minimiza os problemas acarretados pelas inconsistências nas informações difundidas pela rede (como *loops*, por exemplo).

O segundo motivo de interesse neste resultado é o fato dos caminhos escolhidos pelo MARA serem, na média mais longos que os das demais métricas (para este cenário). Isto mostra um potencial caminho de otimização da formulação do MARA que, originalmente, não leva em consideração os efeitos da auto-interferência nos caminhos escolhidos. Caminhos mais longos, em termos de saltos, são estatisticamente mais propensos à auto-interferência já que, além de estarem em maior número, os enlaces são também entre nós mais próximos (em geral). Logo, uma formulação alternativa do MARA que leve em consideração também os efeitos da auto-interferência pode melhorar ainda mais o desempenho da proposta.

Embora os resultados apresentados até aqui tenham sido relacionados ao fluxo TCP entre os nós 0 e 9 (nós mais distantes geograficamente), tendências semelhantes foram verificadas para todos os outros fluxos considerados (ou seja, para cada nó de destino da topologia). Os resultados obtidos nos experimentos com fluxos para outros nós de destino podem ser analisados em [17].

5.2. Otimizações MARA-P e MARA-RP

Assim como nas simulações, os experimentos para a avaliação do desempenho do MARA-P e do MARA-RP foram realizados com a utilização de 3 fluxos UDP com datagramas de tamanhos diferentes. Entretanto, nos experimentos realizados na rede real, foi utilizado também um fluxo de pacotes ICMP pequenos (64 bytes) para a medição do RTT (assim como feito para os experimentos com fluxos TCP).

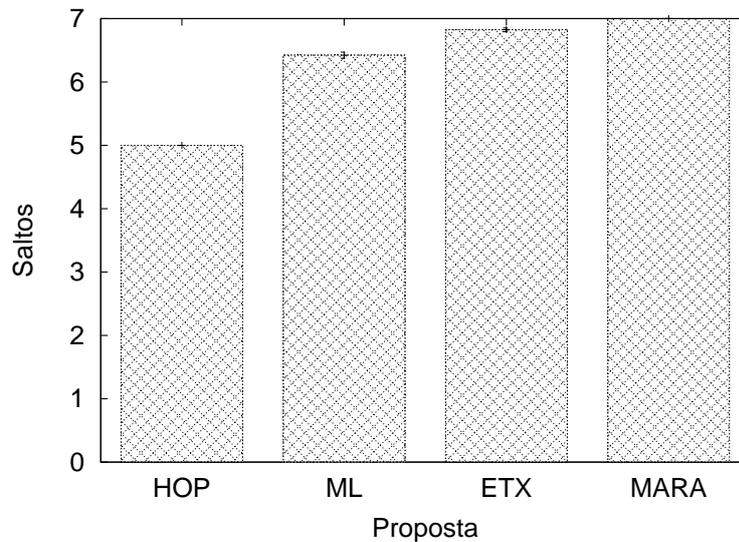


Figura 9. Número médio de saltos no fluxo de dados ICMP nos testes da rede real.

O gráfico da Figura 10 mostra o RTT médio obtido por cada uma das versões do MARA. Ao contrário do que havia ocorrido nas simulações, o MARA-P obteve uma redução percentual considerável do RTT médio, em relação à proposta original (cerca de 12%). No entanto, considerando os intervalos de confiança de 95% plotados no gráfico, pode-se notar que os resultados foram, de fato, muito próximos. Outro resultado surpreendente foi o mau desempenho do MARA-RP, que não apenas obteve um RTT médio bastante superior ao das outras variações, mas também apresentou uma alta variabilidade. Uma possibilidade é o alto *overhead* de processamento associado ao MARA-RP. Embora suas escolhas possam ser adequadas, a quantidade de recursos computacionais gastos é alta, considerando-se os recursos limitados de *hardware* disponíveis nos roteadores.

6. Conclusões

Neste artigo foi realizada uma avaliação mais profunda do mecanismo MARA. Além de simulações em cenários com um número maior de nós, o MARA foi implementado e avaliado em uma rede real. Foram apresentados também os detalhes da implementação realizada.

Os resultados de simulação apresentados mostram que a diferença percentual de desempenho entre o MARA e as demais propostas avaliadas se acentua com o aumento da distância geográfica dos nós e do aumento de caminhos alternativos.

Os resultados dos experimentos realizados na rede real confirmaram o bom desempenho do MARA. O MARA foi superior às demais propostas avaliadas tanto em termos de vazão, quanto de RTT. A vazão média obtida pelo MARA nos experimentos foi quase duas vezes mais alta que a da segunda melhor proposta. Em termos de atraso (representado nos resultados indiretamente pelo RTT), parâmetro de interesse do MARA, a diferença para a segunda melhor proposta foi de quase 3 vezes. Estes resultados são bastante similares aos encontrados nas simulações realizadas anteriormente em [18], reforçando a confiança no modelo de simulação adotado.

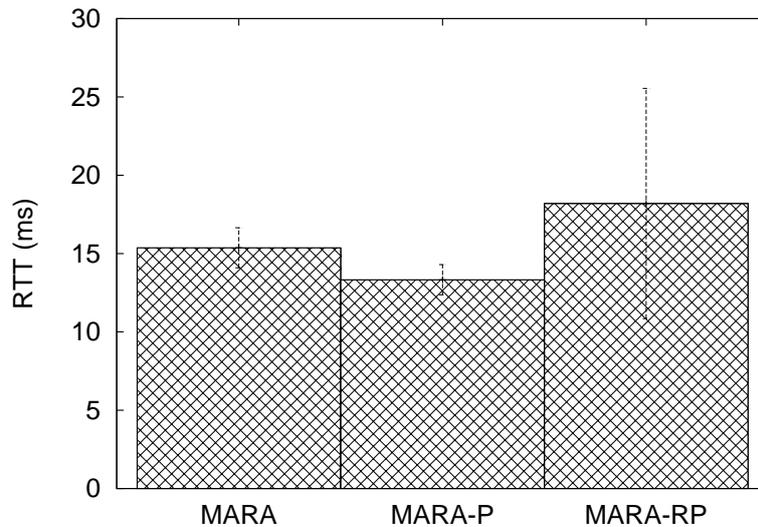


Figura 10. RTT médio obtido por cada uma das variações do mecanismo MARA na rede real.

Em relação às otimizações, os resultados de simulação mostram que as escolhas de MARA-P e MARA-RP foram equivalentes. Isto levanta a hipótese de que, ao menos no modelo de custo utilizado pelo MARA, o tamanho do pacote não tem influência na escolha das melhores rotas. Por outro lado, nos experimentos na rede real, o MARA-P obteve uma ligeira vantagem em relação ao desempenho da proposta original do MARA, em termos de atraso, enquanto o resultado do MARA-RP foi consideravelmente pior. Este resultado, provavelmente, se deve ao *overhead* de processamento desta solução.

Como o MARA objetiva minimizar o atraso considerando um único pacote, neste trabalho consideramos apenas cenários com um único fluxo. Como trabalhos futuros, pretende-se realizar uma avaliação do MARA em ambientes com múltiplos fluxos concorrentes.

Referências

- [1] P. Aravinda, K. Acharya, A. Sharma, E. M. Belding, K. C. Almeroth, e K. Papagiannaki. Congestion-Aware Rate Adaptation in Wireless Networks: A Measurement-Driven Approach. Em *SECON '08*, pp. 1–9, Jun. 2008.
- [2] J. Bicket. Bit-rate selection in wireless networks. Dissertação de Mestrado, M.I.T., Cambridge, MA, Fev. 2005.
- [3] J. Bicket, D. Aguayo, S. Biswas, e R. Morris. Architecture and evaluation of an unplanned 802.11b mesh network. Em *Proceedings of MobiCom '05*, pp. 31–42, 2005.
- [4] M. E. M. Campista, D. Passos, P. M. Esposito, I. M. Moraes, C. V. N. Albuquerque, D. C. M. Saade, M. G. Rubinstein, L. H. M. K. Costa, e O. C. M. B. Duarte. Routing metrics and protocols for wireless mesh networks. *IEEE Network*, 22(1):6–12, Jan./Fev. 2008.
- [5] K. V. Cardoso. Controle Automático de Taxa em Redes IEEE 802.11. Tese de Doutorado, Universidade Federal do Rio de Janeiro, Fev. 2009.

- [6] C. Chen, E. Seo, H. Luo, N. H. Vaidya, e X. Wang. Rate-adaptive framing for interfered wireless networks. Em *INFOCOM 2007*, pp. 1325–1333, Mai. 2007.
- [7] D. S. J. D. Couto, D. Aguayo, J. Bicket, e R. Morris. A high-throughput path metric for multi-hop wireless routing. Em *Proceedings of MobiCom '03*, pp. 134–146, 2003.
- [8] R. Draves, J. Padhye, e B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. Em *Proceedings of MobiCom '04*, pp. 114–128, 2004.
- [9] J. Ha, K. Lee, H. Kim, e I. Kang. A Snooping Rate Adaptation Algorithm for IEEE 802.11 WLANs. Em *ISWPC*, pp. 606–609, 2008.
- [10] Iperf. <http://iperf.sourceforge.net/>, Dez. 2009. Acessado em 14-12-2009.
- [11] A. Kamerman e L. Monteban. WaveLAN-II: A High-performance Wireless LAN for the Unlicensed Band. *Bell System Technical Journal*, pp. 118–133, 1997.
- [12] J. Kim, S. Kim, S. Choi, e D. Qiao. CARA: Collision-aware rate adaptation for IEEE 802.11 WLANs. Em *Proceedings of INFOCOM 2006*, pp. 1–11, Abr. 2006.
- [13] C. E. Koksal, K. Jamieson, E. Telatar, e P. Thiran. Impacts of channel variability on linklevel throughput in wireless networks. Em *Joint International Conference on Measurement and Modeling of Computer Systems*, pp. 51–62, 2006.
- [14] M. Lacage, M. H. Manshaei, e T. Turletti. IEEE 802.11 rate adaptation: a practical approach. Em *Proceedings of MSWiM '04*, pp. 126–134, 2004.
- [15] MadWifi. <http://madwifi-project.org/>, Dez. 2009. Acessado em 14-12-2009.
- [16] OpenWrt. <http://openwrt.org/>, Dez. 2009. Acessado em 14-12-2009.
- [17] D. Passos. Uma abordagem unificada para métricas de roteamento e adaptação automática de taxa em redes em malha sem fio. Dissertação de Mestrado, Universidade Federal Fluminense, Jun. 2009.
- [18] D. Passos e C. Albuquerque. Uma abordagem unificada para métricas de roteamento e adaptação automática de taxa em redes em malha sem fio. Em *SBRC 2009*, Mai. 2009.
- [19] Projeto ReMoTE. <http://mesh.ic.uff.br/>, Dez. 2009. Acessado em 14-12-2009.
- [20] B. Sadeghi, V. Kanodia, A. Sabharwal, e E. Knightly. OAR: an opportunistic auto-rate media access protocol for ad hoc networks. *Wireless Networks*, 11(1):39–53, 2005.
- [21] Solução La Fonera. <http://www.fon.com/>, Dez. 2009. Acessado em 14-12-2009.
- [22] Solução Meraki. <http://meraki.com/>, Dez. 2009. Acessado em 14-12-2009.
- [23] Solução Mesh Cisco. <http://www.cisco.com/go/wirelessmesh>, Dez. 2009. Acessado em 14-12-2009.
- [24] Solução Mesh do Projeto OLPC. http://wiki.laptop.org/go/mesh_network_details, Dez. 2009. Acessado em 14-12-2009.
- [25] Solução Mesh Siemens. <http://www.siemens.ie/carrier/topics/wireless-mesh/>, Dez. 2009. Acessado em 14-12-2009.
- [26] S. H. Y. Wong, H. Yang, S. Lu, e V. Bharghavan. Robust rate adaptation for 802.11 wireless networks. Em *Proceedings of MobiCom '06*, pp. 146–157, 2006.