

Um Algoritmo Eficiente de Disseminação dos Estados de Enlace para Redes em Malha Sem Fio

Miguel Elias M. Campista, Luís Henrique M. K. Costa e
Otto Carlos M. B. Duarte *

¹ Grupo de Teleinformática e Automação
PEE/COPPE - DEL/POLI
Universidade Federal do Rio de Janeiro

{miguel, luish, otto}@gta.ufrj.br

Abstract. *Currently, wireless mesh networks (WMN) are considered the most cost-effective solution for access networks. WMNs are characterized by a stationary backbone, composed by wireless routers, and the presence of an Internet gateway. This work proposes a new algorithm for link state dissemination, which profits from the main WMN characteristics. The proposed algorithm reduces the network routing overhead while keeping updated the metrics for the frequently used links. Thus, we avoid latency in route discovery, and we also save network resources. The proposed algorithm limits the range of routing updates towards the network gateway, since the most usual application in WMN is Internet access. The results obtained show a considerable decrease in network routing overhead.*

Resumo. *Atualmente, as redes em malha sem fio (RMSF) são consideradas a solução com a melhor relação custo-benefício para as redes de acesso. As RMSF são caracterizadas pela presença de um backbone estacionário, composto por roteadores sem fio, e a presença de um gateway para a Internet. Este trabalho propõe um novo algoritmo de disseminação de estados do enlace, que explora as particularidades das RMSF. O algoritmo proposto diminui a sobrecarga de controle da rede, mantendo porém as métricas dos enlaces mais utilizados sempre atualizadas. Assim, evita-se latência no descobrimento de rotas e ainda economiza-se recursos da rede. O algoritmo proposto concentra as atualizações dos estados do enlace na direção do gateway da rede, visto que a aplicação mais comum em RMSF é o acesso à Internet. Os resultados obtidos mostram uma diminuição significativa da sobrecarga de controle da rede.*

1. Introdução

Atualmente, as Redes em Malha Sem Fio (RMSF) vêm sendo consideradas a solução com a melhor relação custo-benefício para as redes de acesso [Akyildiz et al., 2005]. A ausência da infra-estrutura cabeada e a ubiquidade de acesso reduzem o custo de implementação da rede, ao mesmo tempo em que provêm conectividade aos usuários onde quer que eles estejam. Essas características são devidas, principalmente, à presença de um *backbone* estacionário composto por roteadores sem fio

*Este trabalho foi realizado com recursos do CNPq, CAPES, FAPERJ, RNP/FINEP/FUNTEL e FUJB.

e à transmissão por rádio-difusão. Dentre os principais cenários de aplicação das RMSF estão a interconexão de redes ad hoc isoladas e de redes par-a-par, e a construção de redes comunitárias [Campista et al., 2007].

A conectividade da rede em malha sem fio é mantida através de comunicações em múltiplos saltos entre qualquer par origem-destino da rede. A comunicação por múltiplos saltos exige a presença de uma rota composta por roteadores encarregados de encaminhar o tráfego até o destinatário. Antes mesmo das RMSF, esse tipo de comunicação já era utilizado pelo modo de operação ad hoc do padrão IEEE 802.11. Nas redes ad hoc, há dois tipos de protocolos de roteamento: reativos e pró-ativos. A diferença principal entre eles é o tipo de procedimento pelo qual as rotas são formadas e mantidas. Nos protocolos reativos, as rotas são requisitadas sob-demanda, sempre que um nó fonte possui dados a enviar. Essa estratégia, por um lado, evita sobrecarga de controle na rede, mas por outro, aumenta a latência da descoberta de rotas. Em oposição, a estratégia pró-ativa mantém as informações necessárias para o cálculo de rotas até o destinatário constantemente atualizadas. Isso elimina a latência da descoberta de rotas, mas produz maior sobrecarga de controle na rede. Nas redes ad hoc, a estratégia reativa pode ser justificada pela mobilidade dos nós, pela alta variação das condições dos enlaces e pelos recursos de banda passante e bateria escassos. Em contrapartida, os protocolos reativos incorrem em uma latência inicial para descoberta de rotas. Considerando o problema da sobrecarga de controle, alguns protocolos pró-ativos como o OLSR [Clausen et al., 2001] e o Fisheye [Pei et al., 2000] reduzem o número de mensagens de controle utilizando técnicas de controle de difusão.

O emprego dos protocolos de roteamento ad hoc nas redes em malha sem fio representa a solução mais simples, visto que as comunicações também são realizadas por múltiplos saltos sem fio. Entretanto, características relacionadas à arquitetura da rede em malha e à sua matriz de tráfego motivam adaptações aos protocolos legados das redes ad hoc. Essas adaptações têm por objetivo aumentar a eficiência do encaminhamento de pacotes nas redes em malha sem fio. Dentre as características da arquitetura e da matriz de tráfego estão o posicionamento fixo dos nós do *backbone* e a presença de um *gateway* para a rede cabeada. Os protocolos de roteamento para redes em malha atuais lidam com essas características com novas métricas de roteamento e com novas formas de descobrimento e manutenção de rotas. Porém, essas novas métricas introduzem uma sobrecarga de controle extra, o que incentiva a maioria dos protocolos de roteamento das redes em malha visarem reduzir essa carga. Alguns protocolos utilizam uma abordagem híbrida, reativa e pró-ativa, não eliminando o problema da latência na descoberta de rotas. Outros protocolos continuam utilizando uma abordagem completamente pró-ativa, empregando mecanismos de controle de difusão não adaptados às redes em malha.

Este trabalho propõe um novo algoritmo para disseminação de estados do enlace para as redes em malha sem fio. Diferente das propostas anteriores, o algoritmo proposto reduz a carga de controle da rede levando em consideração a matriz comum de tráfego nesse tipo de rede, que é concentrado na direção do *gateway* da rede cabeada. Além disso, o algoritmo proposto considera a atualização periódica das métricas dos enlaces da rede, sendo portanto por natureza um algoritmo baseado em estado do enlace. O algoritmo evita portanto a latência na descoberta de rotas, ao mesmo tempo em que mantém as métricas de interesse constantemente atualizadas. O algoritmo proposto trata a topologia

da rede em malha como uma árvore, onde o *gateway* é a raiz. Cada nó somente encaminha mensagens de controle originadas em um nó que pertença a sua rota até o *gateway* ou em um nó que o utiliza para alcançar o *gateway*. Comparado ao OLSR [Clausen et al., 2001], o algoritmo proposto reduz em até 78% e em até 73% o número de mensagens de controle de roteamento em cenários densos e esparsos, respectivamente. Além disso, considerando variação na qualidade dos enlaces, essa redução é ainda maior. Ela pode atingir 87% e 85% também em cenários densos e esparsos, respectivamente.

Este trabalho está organizado da seguinte forma. A Seção 2 analisa o estado da arte em roteamento em redes em malha sem fio, destacando as novas métricas e os principais protocolos existentes. A Seção 3 apresenta o problema da sobrecarga de controle em redes em malha sem fio. A Seção 4 identifica as premissas consideradas neste trabalho e apresenta o algoritmo proposto. Os resultados obtidos são analisados na Seção 5. Finalmente, a Seção 6 conclui este trabalho e identifica as direções futuras.

2. Trabalhos Relacionados

As características da arquitetura da rede, da matriz de tráfego e da qualidade dos enlaces das redes em malha sem fio são consideradas tanto nas métricas propostas quanto nos protocolos de roteamento.

2.1. Métricas de roteamento

Nas redes ad hoc, a métrica mais utilizada é o número de saltos. Essa métrica simples é conveniente devido à mobilidade e às limitações de energia dos nós. Porém, nas redes em malha, os nós do *backbone* são geralmente fixos e não possuem restrições de energia. Logo, o foco dos protocolos de roteamento é desviado para outros fatores como a qualidade dos enlaces, que é muito variável nos enlaces sem fio. A rota com o menor número de saltos pode não ser a melhor, pois normalmente enlaces longos possuem taxas de transmissão inferiores e não necessariamente possuem a melhor qualidade.

Dentre as novas métricas utilizadas em redes em malha estão: o ETX, o ETT e o WCETT. O ETX (*Expected Transmission Count*) [Draves et al., 2004a] calcula o número esperado de transmissões que um nó leva para enviar com sucesso uma sonda na taxa básica da rede em um dado enlace. O ETT (*Expected Transmission Time*) [Draves et al., 2004b], por outro lado, foi proposto porque a maioria dos pacotes de dados possui um tamanho superior ao de uma sonda e são enviados em taxas que podem variar de acordo com as condições do meio. O ETT considera o intervalo de tempo esperado por enlace, para um pacote de dados ser transmitido com sucesso. Assim, tanto o tamanho do pacote quanto a taxa de transmissão são levados em conta. O WCETT (*Weighted Cumulative ETT*) [Draves et al., 2004b] é uma evolução do ETT que considera múltiplos canais operando em múltiplas interfaces de rádio.

2.2. Protocolos de roteamento

Apesar de muitos protocolos de roteamento para redes em malha serem adaptações de protocolos legados das redes ad hoc, eles otimizam a descoberta e a manutenção de rotas, bem como adotam as novas métricas descritas na Seção 2.1.

O *Link Quality Source Routing* (LQSR) [Draves et al., 2004a] combina roteamento pró-ativo baseado em estado do enlace com o roteamento reativo das redes ad hoc.

O objetivo de usar protocolos reativos é reduzir a carga de controle da rede, que pode ser acentuada devido às variações na qualidade dos enlaces. O LQSR utiliza mecanismos de descoberta de rota como nos protocolos reativos, embora a arquitetura estacionária do *backbone* das redes em malha favoreça o uso de protocolos pró-ativos. Durante a descoberta de rotas, o LQSR obtém informações de roteamento, como os estados dos enlaces atravessados. Assim, é possível reduzir a frequência de inundação da rede com informações sobre estados dos enlaces. O protocolo SrcRR [Aguayo et al., 2005] utiliza apenas o mecanismo reativo de descoberta de rotas para atualizar as métricas dos enlaces usados. Assim, o SrcRR diminui ainda mais a carga de controle, porém calcula as suas rotas com um mapa da topologia reduzido. Tanto o LQSR quanto o SrcRR possuem procedimentos de descoberta de rotas baseados no protocolo de roteamento reativo DSR (*Dynamic Source Routing*) [David B. Johnson e Broch, 2001] das redes ad hoc. O protocolo *Mesh Distance Vector* (MeshDV) [Iannone e Fdida, 2005] trata a mobilidade dos clientes das redes em malha e não somente considera os nós do *backbone*, como o LQSR e o SrcRR. Para isso, todos os nós do *backbone* mantêm uma tabela com os endereços IP dos clientes diretamente conectados e outra com os endereços IP dos clientes conectados a outros nós do *backbone*. O *Optimized Link State Routing* (OLSR) [Clausen et al., 2001] é um protocolo de roteamento baseado em estado do enlace originalmente proposto para as redes ad hoc. Para redes em malha, o OLSR foi adaptado para utilizar o ETX.

O *Multi Radio LQSR* (MR-LQSR) [Draves et al., 2004b] e o *Directional OLSR* (DOLSR) [Das et al., 2006] são dois exemplos de protocolos de roteamento que empregam novas técnicas para aumento de eficiência. O MR-LQSR é uma adaptação do LQSR para utilizar múltiplos canais e múltiplas interfaces e o DOLSR é uma adaptação do OLSR para utilizar antenas direcionais.

A inundação da rede com anúncios de estados de enlace pode gerar problemas de escalabilidade, principalmente considerando que os estados de enlace mudam frequentemente. Uma maneira de reduzir a carga de controle nas redes em malha é limitar a transmissão de mensagens de controle a um escopo local. Alguns protocolos assumem que não é eficiente inundar toda a rede, visto que as comunicações são normalmente limitadas a nós próximos. Portanto, não é necessário disseminar pacotes de controle para todos os nós com a mesma frequência. Outra maneira de diminuir a carga de controle da rede é reduzir o número de nós que encaminham informações de controle. Alguns protocolos elegem o menor número de nós necessários para cobrir toda a rede, evitando assim o envio de mensagens redundantes.

O Fisheye [Pei et al., 2000] é o primeiro protocolo proposto para redes ad hoc que limita a disseminação dos pacotes de controle somente aos vizinhos. Para isso, o TTL (*Time-To-Live*) das mensagens de controle é ajustado conforme o número de nós da rede que se pretende alcançar. Quanto menor o TTL, menor é o número de nós que recebe a mensagem. Alguns protocolos foram desenvolvidos para as redes em malha seguindo o mesmo princípio do Fisheye. O *Localized On-demand Link State* (LOLS) [Nelakuditi et al., 2005] atribui um custo de longo-prazo e um custo de curto-prazo aos enlaces. O custo de longo-prazo representa o custo habitual do enlace e o custo de curto-prazo representa o custo atual. Para reduzir a carga de controle, os custos de curto-prazo são transmitidos aos vizinhos e os de longo-prazo são difundidos na rede com uma periodicidade menor. Outro protocolo, o *Mobile Mesh Routing Protocol*

(MMRP) [MITRE Corporation, 2006], atribui idade aos seus estados de enlace. Toda vez que um nó vai encaminhar uma mensagem de controle, ele diminui a idade do estado do enlace do tempo estimado para encaminhá-lo. Quando a idade do estado do enlace expira, ele é descartado, evitando a disseminação para todos os nós da rede. O OLSR limita o número de nós que difundem mensagens de controle, evitando redundâncias. Para tal, cada roteador OLSR elege o seu conjunto de MPRs (*MultiPoint Relays*), que são os nós responsáveis por encaminhar as mensagens de controle. Nenhum outro nó, exceto os MPRs, pode encaminhar mensagens de controle. O conjunto MPR é formado pelo menor número de nós vizinhos de um salto capaz de alcançar todos os nós vizinhos de dois saltos.

Dentre os protocolos de roteamento das redes em malha, há ainda aqueles que aproveitam a matriz de tráfego comum da rede. Nesses protocolos, assume-se que a aplicação mais utilizada é o acesso ao *gateway* da rede cabeada. Logo, um protocolo desse tipo considera que a topologia da rede é semelhante a uma árvore, otimizando o seu funcionamento. O *Ad hoc On-demand Distance Vector - Spanning Tree* (AODV-ST) [Ramachandran et al., 2005] é uma adaptação do protocolo reativo de redes ad hoc AODV [Perkins e Royer, 1999]. No AODV-ST, o *gateway* periodicamente faz um pedido de rota a todos os outros nós da rede para manter a sua tabela de roteamento atualizada. Outro protocolo desta categoria é o proposto por Raniwala *et al.* baseado no algoritmo *spanning tree* utilizado em alguns protocolos para redes cabeadas. A manutenção da árvore é realizada através de requisições de entrada e saída [Raniwala e Chiueh, 2005].

Neste artigo é proposto um algoritmo que combina os três tipos de protocolos descritos. Além de ser baseado em estados de enlace como o OLSR, o algoritmo proposto limita a disseminação de mensagens de controle baseado na matriz de tráfego comum da rede.

3. Sobrecarga de Controle em Redes em Malha Sem Fio

Um dos maiores desafios das redes sem fio é lidar com a alta variação das condições do meio de transmissão. Especialmente em redes sem fio de múltiplos saltos, o desafio é manter as informações da qualidade dos enlaces atualizadas, sem que isso resulte em aumento da sobrecarga de controle. É importante também que essas informações estejam sincronizadas para evitar decisões incoerentes e formação de *loops*. Surge então um compromisso entre frequência de atualização de informações de qualidade dos enlaces e sobrecarga de controle. Se por um lado a frequência de atualizações necessita ser alta para reproduzir mais precisamente o estado atual da rede, por outro lado, a quantidade de sobrecarga de controle deve ser limitada.

As métricas descritas na Seção 2.1, ETX, ETT e WCETT, estimam a qualidade do enlace através do envio de sondas aos nós vizinhos. A partir do número de sondas recebidas dos vizinhos e do número de sondas enviadas com sucesso, é possível calcular tais métricas. Logo, em redes em malha sem fio, o cálculo da qualidade dos enlaces já incorre em sobrecarga de controle. Protocolos como o LQSR e o SrcRR reduzem essa sobrecarga adotando uma abordagem híbrida, reativa e pró-ativa, para a descoberta e manutenção de rotas. Ao utilizar procedimentos de protocolos reativos, a carga de controle é reduzida, pois é possível utilizar a própria mensagem de requisição de rota para atualizar as métricas. Apesar dessa redução, a atualização só se faz sob demanda.

Isso favorece a perda do sincronismo entre os mapas da topologia dos nós, além de não evitar a latência inicial da descoberta de rotas.

Os protocolos pró-ativos não possuem latência inicial para encontrar rotas. Porém, a sobrecarga de controle pode ser muito alta, especialmente nas redes sem fio onde há alta variação da qualidade dos enlaces. Para contornar esse problema, algoritmos de controle de difusão são utilizados para reduzir o número de nós que reenviam mensagens de controle. Entretanto, há espaço para uma redução maior da sobrecarga de controle, pois esses algoritmos foram projetados para redes ad hoc. Nas redes ad hoc, todo par origem-destino da rede tem a mesma probabilidade de se comunicar. Em redes em malha, ao contrário, a maior parte do tráfego é enviada somente na direção do *gateway*.

4. Algoritmo Proposto

Nesta seção, primeiramente são identificadas as premissas deste trabalho. Em seguida, a operação do algoritmo proposto é detalhada.

4.1. Premissas

O algoritmo proposto neste trabalho parte de algumas premissas. A primeira é considerar que a topologia da rede é aproximadamente uma árvore. Isso ocorre porque a maior parte do tráfego da rede é direcionada ao *gateway* da rede cabeada, conforme diversos trabalhos da área [Ramachandran et al., 2005, Raniwala e Chiueh, 2005, Cheng et al., 2006, Gambiroza et al., 2004, Dong et al., 2006].

A segunda premissa é considerar protocolos pró-ativos baseados em estado do enlace para eliminar a latência inicial de descoberta de rotas e evitar *loops* de roteamento. Os estados dos enlaces são enviados periodicamente. Caso haja uma atualização disparada em virtude da variação da qualidade de algum enlace da rede, a disseminação resultante será realizada, além das mensagens que já são periodicamente enviadas. Sempre que há variação da qualidade de um enlace por instabilidades do meio físico, os nós que detectaram essa variação disparam uma atualização para toda a rede. Essa mensagem é encaminhada conforme o algoritmo proposto, detalhado na Seção 4.2.

4.2. Operação do algoritmo

Este trabalho propõe um novo algoritmo para disseminação de estados do enlace com o objetivo de reduzir a carga de controle transmitida nas redes em malha sem fio. A idéia chave é concentrar o envio das mensagens de controle na direção do *gateway* da rede cabeada. Primeiro, o algoritmo é apresentado em sua versão básica. Em seguida, duas funções são adicionadas ao algoritmo básico, resultando em sua versão completa.

No algoritmo básico proposto, são definidos dois tipos de mensagens de controle de roteamento para anúncio dos estados dos enlaces. O primeiro tipo corresponde às *mensagens de disseminação controlada*, que são enviadas apenas para os nós que fazem parte dos enlaces na direção do *gateway*. O segundo tipo corresponde às *mensagens de inundação*, que são enviadas para toda a rede. Apesar de uma das premissas deste trabalho ser considerar que a maior parte do tráfego é direcionada ao *gateway*, é ainda possível que qualquer outro par fonte-destino da rede se comunique. Portanto, o emprego das mensagens de inundação tem por objetivo manter todos os nós cientes da topologia completa da rede e capazes de calcular a rota com o menor custo até qualquer destino.

O algoritmo proposto é caracterizado principalmente pela forma que as mensagens de disseminação controladas são encaminhadas. Assim, um nó n_i pertencente a uma rede em malha sem fio encaminha mensagens de disseminação controlada originada em nós que o utilizam como parte da rota até o *gateway* e em nós que pertencem à rota do próprio nó n_i até o *gateway*.

Semelhante a uma topologia em árvore, um nó n_i encaminha as mensagens de controle originadas em nós pais, que são os nós que pertencem à sua rota até o *gateway* (g), e encaminha mensagens de controle originadas em nós filhos, que são os nós que usam n_i para alcançar g . Formalmente, denota-se \mathcal{P}_i como o conjunto de nós pais do nó n_i e o conjunto \mathcal{F}_i o conjunto de nós filhos também de n_i . A Figura 1 ilustra os conjuntos \mathcal{P}_i e \mathcal{F}_i , onde $\mathcal{P}_i = \{p_1, p_2, \dots, g\}$ e $\mathcal{F}_i = \{f_1, f_2, \dots\}$.

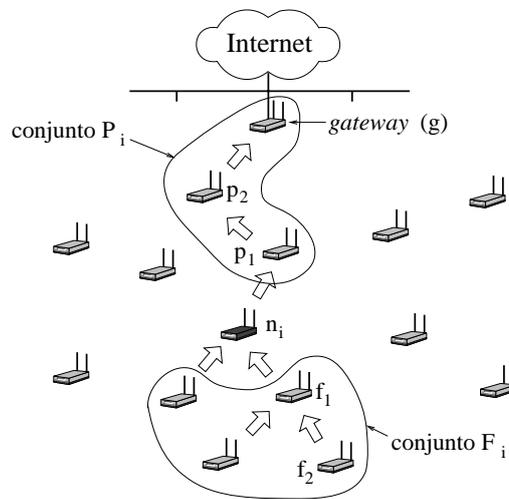


Figura 1. Rede em malha com topologia baseada em árvore.

Pode ser observado que as mensagens de disseminação controlada são encaminhadas no conjunto \mathcal{C}_i , onde $\mathcal{C}_i = \mathcal{P}_i \cup \mathcal{F}_i \cup n_i$.

As mensagens de disseminação controlada são geradas por cada nó periodicamente em intervalos T_{dc} . Já as mensagens de inundação são geradas em intervalos T_i . É importante notar que T_{dc} deve ser menor que T_i para que haja redução da carga de controle de roteamento como desejado.

Por hipótese, cada nó possui o mapa completo da topologia e tem condições de calcular a menor rota na direção do *gateway*, executando o algoritmo de Dijkstra, por exemplo. Após calcular a menor rota, ele notifica o vizinho utilizado como caminho para o *gateway* de que ele é seu filho. Assim, todos os nós conhecem os seus filhos. Essa notificação é feita da mesma forma que o OLSR notifica os seus vizinhos escolhidos para serem seus MPRs (*MultiPoint Relays*). A notificação dos MPRs no OLSR é realizada através das mensagens de HELLO, que listam os vizinhos de um nó e o tipo de relacionamento entre eles. Portanto, no algoritmo proposto, cada nó notifica o seu nó pai através das mensagens de HELLO. Após a convergência do algoritmo, cada nó terá conhecimento dos seus nós pais, através do algoritmo de Dijkstra, e dos seus nós filhos através das mensagens de HELLO. É importante destacar que cada nó conhece todos os nós pais até o *gateway* e apenas os seus nós filhos vizinhos de um salto. Porém, essa característica não

interfere no comportamento do algoritmo, que continua encaminhando pacotes de controle originados em nós pai e em qualquer nó filho, independente se é vizinho ou não. Por exemplo, o nó n_i da Figura 1 supostamente não sabe se uma mensagem de controle originada em f_2 deve ser encaminhada, visto que n_i não sabe diretamente se f_2 pertence ao conjunto \mathcal{F}_i . Porém, como f_1 encaminha mensagens originadas em f_2 e n_i sabe que f_1 é seu filho, garante-se também que f_2 pertence ao conjunto \mathcal{F}_i .

O pseudo-código abaixo ilustra o funcionamento do algoritmo proposto, considerando a ação realizada por n_i ao receber a mensagem de controle m . No pseudo-código, f é o roteador que originou a mensagem de controle m , v é o nó vizinho que encaminhou m por último, g é o *gateway* da rede e n_i é um roteador intermediário qualquer. A função `eh_pai(f, n_i, g)` verifica se f pertence ao conjunto \mathcal{P}_i . Já a função `eh_filho(v, n_i, g)` verifica se o nó v que encaminhou m é um filho de um salto do nó n_i .

Algoritmo proposto executado pelo nó n_i .

```

recebe_hellos();
...
recebe_estado_do_enlace() começar
  se ( $m$  é uma mensagem de disseminação controlada) então
    se ((eh_pai( $f, n_i, g$ )) ou (eh_filho( $v, n_i, g$ ))) então
      encaminha( $m$ );
    senão
      descarta( $m$ );
  fim do se
  senão # é uma mensagem de inundação
    encaminha( $m$ );
  fim do se
fim
...
dijkstra();

```

O algoritmo proposto possui ainda duas funções que podem ser adicionadas ao algoritmo básico apresentado. A versão completa do algoritmo utiliza os MPRs do OLSR para encaminhar as mensagens de inundação e regula o período em que as mensagens de inundação são originadas. A primeira função reduz a possibilidade de nós que não pertençam à um determinado conjunto \mathcal{C}_i recebam mensagens repetidas. É importante observar, entretanto, que não há uso de MPRs nas disseminações controladas. Nesse caso, é interessante manter a redundância para aumentar a probabilidade de recepção com sucesso dos estados dos enlaces mais utilizados. A segunda função parte do princípio de que quanto mais próximo um nó está do *gateway*, maior é o seu número de filhos. Portanto, não há necessidade dos nós mais próximos ao *gateway* enviarem mensagens de inundação com a mesma frequência que os nós mais distantes, visto que os nós mais próximos já pertencem a um número maior de árvores (conjuntos \mathcal{C}_i). O período de envio das mensagens de inundação (T_i) passa a ser então regulado de acordo com a função $f(d)$, chamada de função de ajuste da inundação, onde d é a distância em número de saltos do nó fonte até o *gateway*. A função $f(d)$ é dada por:

$$f(d) = T_i^{max} - a(d), \quad (1)$$

onde T_i^{max} é o período máximo de inundação e $a(d)$ é a função que ajusta T_i^{max} conforme a distância até o *gateway*. Na Seção 5, a influência de cada função adicional é avaliada, bem como o desempenho do algoritmo completo é comparado com outros algoritmos existentes na literatura.

5. Resultados de Simulação

O algoritmo proposto foi avaliado tendo como medida o número de mensagens de controle enviadas por segundo, uma vez que o objetivo é reduzir a carga de controle da rede sem afetar o seu desempenho. Os resultados foram obtidos utilizando programas escritos em C++. Preferiu-se utilizar um programa em C++ para isolar o problema analisado, e conseqüentemente, evitar a introdução de complexidade desnecessária.

A análise está dividida em duas partes. A primeira avalia a influência das funções adicionadas ao algoritmo básico e de alguns parâmetros importantes do algoritmo. As funções e os parâmetros são analisados isoladamente. Na segunda parte, o desempenho do algoritmo proposto é comparado ao desempenho do OLSR e do algoritmo de inundação. Nesta segunda parte, são considerados também cenários com e sem variação da qualidade do enlace. Nos cenários onde a qualidade do enlace varia, há o disparo de atualizações extras, aumentando ainda mais a carga de controle da rede. A escolha do OLSR para comparação se deve a importância do OLSR no padrão atualmente em desenvolvimento para redes em malha IEEE 802.11s. O padrão está definindo o OLSR como um protocolo base para o encaminhamento de pacotes nas redes em malha.

Na análise, considera-se que os nós recebem todos os pacotes de controle enviados pelos seus vizinhos e não há perda de mensagens de controle por falhas de transmissão. O encaminhamento, por sua vez, depende do algoritmo utilizado. No algoritmo proposto, o nó sempre encaminha o estado do enlace caso ele tenha sido originado por um nó pai ou filho pertencente ao seu conjunto C_i . Já no OLSR, o nó só encaminha caso ele seja um MPR do último nó que transmitiu a mensagem. No algoritmo de inundação, o nó sempre encaminha as mensagens recebidas. Independente do algoritmo utilizado, um mesmo pacote de controle só é transmitido uma única vez por cada nó, evitando a transmissão de pacotes duplicados.

Nos resultados, o número de nós na rede é variado. Os nós são posicionados em uma grade $n \times n$, onde o *gateway* está localizado em um dos vértices da grade. Todos os nós geram mensagens de controle. O período de inundação (T_i) do algoritmo básico é constante e igual a 100s e o período de difusão controlada (T_{dc}) é de 5s. O período T_{dc} utilizado é o mesmo definido pela RFC do OLSR para enviar seus estados de enlace. Já o valor de T_i foi assumido levando em consideração que T_i deve ser maior que T_{dc} . O número de pacotes de controle é calculado sobre um intervalo de 3600 segundos. É importante observar que as mensagens de controle são produzidas por todos os nós da rede, podendo haver transmissões simultâneas devido a reuso espacial. Porém, nos resultados não são considerados tempos de acesso ao meio ou colisões.

5.1. Avaliação de parâmetros

Nesta seção, são avaliados o desempenho do algoritmo com as funções adicionadas ao algoritmo básico e da relação entre os períodos de inundação (T_i) e de disseminação controlada (T_{dc}). Os nós estão separados de 10m e o raio de transmissão é igual a 21m.

Portanto, neste cenário, cada nó possui no máximo 12 vizinhos, sendo esse um valor intermediário para a densidade de nós da rede.

Relação entre períodos de transmissão das mensagens de controle

A Figura 2(a) mostra a carga de controle de roteamento enviada variando-se a relação entre o período de disseminação controlada (T_{dc}) e o período de inundação (T_i). A legenda da figura mostra $T_{dc} : T_i$. Pode-se notar que o período de disseminação controlada afeta mais o número de mensagens de controle da rede que o período de inundação. Mantendo $T_{dc}=4s$ e variando T_i de 100 para 500s, há uma redução da carga de controle de até 20%. Por outro lado, mantendo $T_i=500s$ e variando T_{dc} de 4 para 5s, há uma redução também de até 20%. Logo, uma menor variação em T_{dc} provoca o mesmo aumento na carga de controle que a variação de T_i .

Função de ajuste da inundação

Dois tipos de funções para ajustar o período de inundação ($f(d)$) foram testados. A função $a(d)$, vista na Expressão 1, pode ser linear ou exponencial. A expressão linear utilizada é assumida igual a $a(d) = d + 20$ e a exponencial é $a(d) = d^2$. O T_i usado pelo algoritmo básico é igual a 100s e o T_{dc} é igual a 5s. Já usando a função de ajuste de inundação, $T_i^{max}=200s$ e $T_{dc}=5s$. O período T_i^{max} é igual a 200s porque para o maior cenário utilizado (100 nós), o menor valor de $f(d)$ é igual a 100s, assim como T_i do algoritmo básico.

A Figura 2(b) mostra que ambas as funções reduzem a carga de controle da rede em até 10%. Porém, a redução ao usar a função exponencial é ligeiramente maior, pois o valor de $f(d)$ retarda mais a crescer. Portanto, na média um maior número de nós utiliza um período de inundação maior.

Função de controle de inundação

A Figura 2(c) demonstra a eficiência do uso dos MPRs (*MultiPoint Relays*) no encaminhamento das mensagens de inundação. Utilizando MPRs observa-se uma redução de até 10% do número de mensagens de controle na rede.

5.2. Avaliação comparativa

Nesta seção, o algoritmo proposto é comparado ao algoritmo de disseminação do protocolo OLSR e ao algoritmo de inundação.

Na avaliação comparativa, o algoritmo proposto utiliza MPR, ajuste exponencial, $T_i^{max}=500s$ e $T_{dc}=4s$. Assim, reduz-se o número de mensagens de controle enviados a nós que não pertençam aos enlaces mais utilizados mas, em contrapartida, aumenta-se a frequência do envio das mensagens de disseminação controlada. Portanto, apesar da redução da carga de controle global da rede, os estados dos enlaces de interesse são atualizados mais frequentemente. Isso aumenta a precisão dos nós no cálculo de rotas. Uma maior precisão no cálculo de rotas permite que as melhores rotas sejam escolhidas, aprimorando as condições de entrega de pacotes de dados da rede. Tanto o OLSR quanto o algoritmo de inundação utilizam um período de inundação (T) igual a 5s.

Nesta seção, são utilizados dois cenários, um esparso e um denso, e cenários onde há ou não o disparo de atualizações em decorrência da variação das condições do meio. Em ambos os cenários, os nós estão posicionados em uma grade, separados entre si por 10m, como na Seção 5.1. Porém, no cenário esparso, o raio de transmissão diminui para

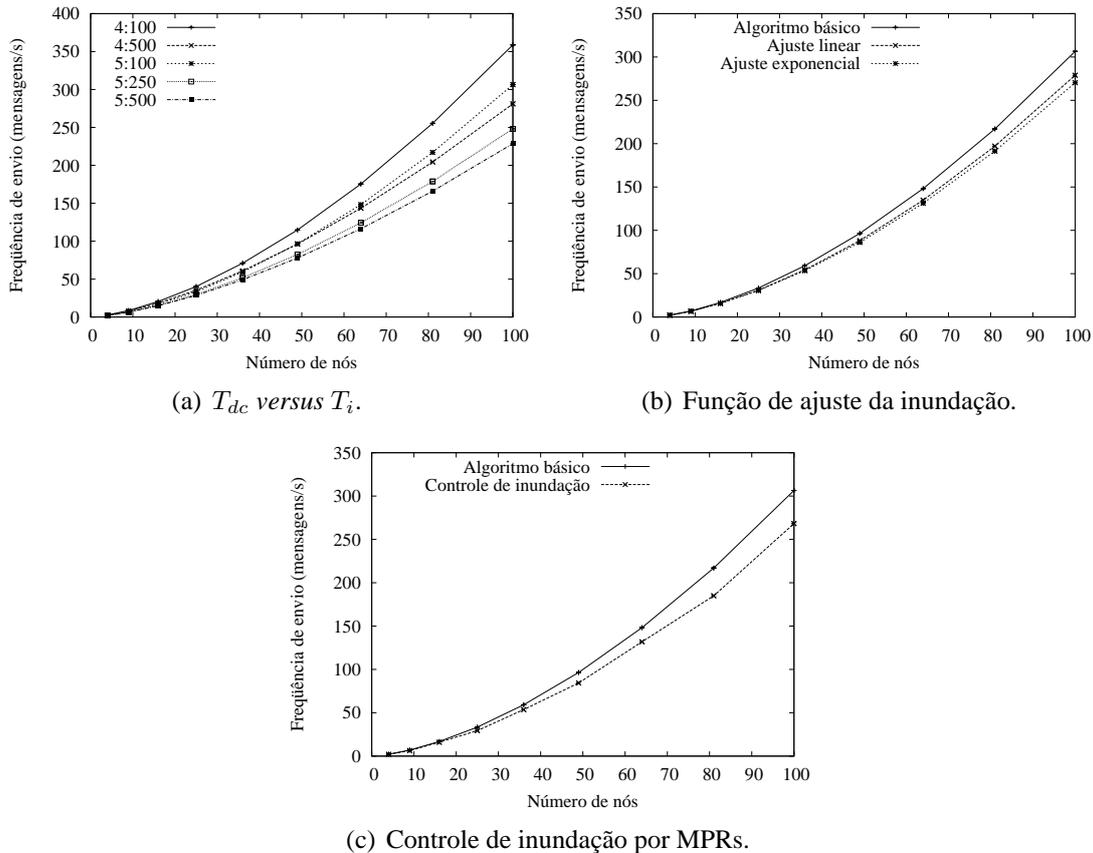


Figura 2. Avaliação dos parâmetros do algoritmo proposto.

12m. Assim, cada nó da grade possui no máximo 4 nós vizinhos. Já no cenário denso, o raio de transmissão é igual a 31m e, como consequência, cada nó possui no máximo 36 vizinhos. Nos cenários onde há disparos de atualizações de estados de enlace, considera-se a variação da qualidade dos enlaces. Sempre que ocorre um evento de variação, mensagens extras são enviadas. Tanto o OLSR quanto o algoritmo de inundação disseminam essas mensagens extras. Em oposição, o algoritmo proposto faz disseminação controlada. Neste trabalho, foram considerados disparos de atualizações a cada 0,5s em média.

A Figura 3(a) mostra o número de mensagens de controle enviadas pelo algoritmo proposto, pelo OLSR e pelo algoritmo de inundação em um cenário esparsos sem variação na qualidade dos enlaces. Sem variação de qualidade, as únicas mensagens de controle transmitidas são as periódicas. Na Figura 3(a), nota-se um desempenho do OLSR muito próximo ao do algoritmo de inundação. Isso ocorre porque em cenários esparsos o mecanismo MPR não é eficiente. Não há muitas transmissões redundantes e, por conseguinte, quase todos os nós devem ser eleitos MPR para que todos os outros recebam as mensagens de controle enviadas. Já no algoritmo proposto, o envio dos estados do enlace na direção do *gateway* é focado, reduzindo em até 73% o número de mensagens de controle em comparação ao OLSR.

Analisando o cenário esparsos onde há variação da qualidade dos enlaces, como visto na Figura 3(b), a diferença entre o OLSR e o algoritmo proposto é maior. Essa diferença é de até 78% em grades com 100 nós. A diferença aumenta, pois com variação

da qualidade dos enlaces, há o disparo de atualizações de controle, além das mensagens enviadas normalmente.

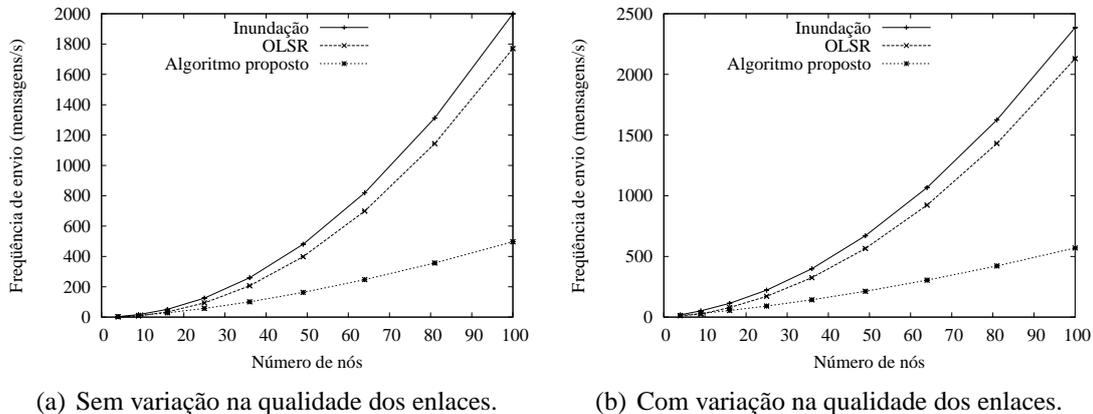
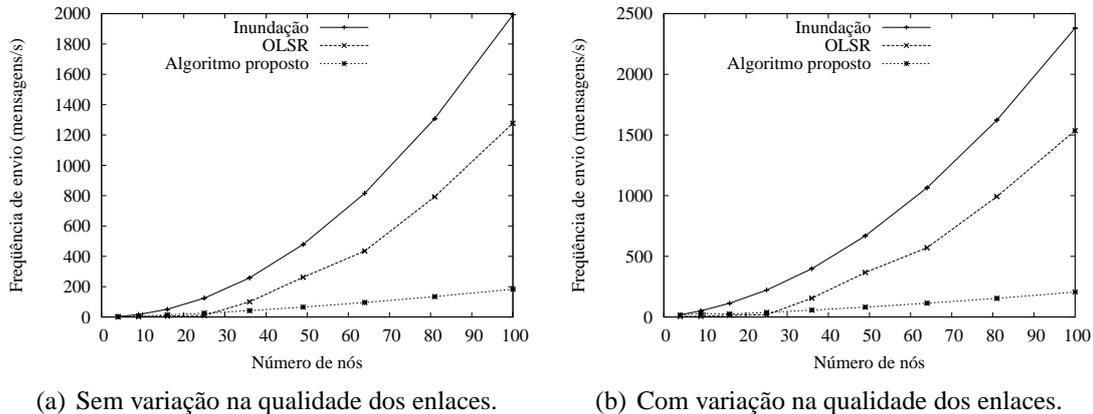


Figura 3. Cenário esparso.

A Figura 4(a) mostra o número de mensagens de controle enviadas em um cenário denso sem variação de qualidade dos enlaces. Nota-se na figura que a eficiência do OLSR aumenta com a densidade da rede, pois o número de mensagens de controle torna-se consideravelmente menor que o número enviado pelo algoritmo de inundação. Quanto mais nós estiverem dentro da mesma área de transmissão, maior será o número de transmissões redundantes. Conseqüentemente, a utilização do MPR é vantajosa. Por outro lado, o OLSR não superou o algoritmo proposto em número de mensagens de controle enviadas. Isso mostra que mesmo utilizando um período de disseminação controlada (T_{dc}) menor que o período de inundação (T) usado pelo OLSR e pelo algoritmo de inundação ($T_{dc}=4s$ e $T=5s$), o desempenho da proposta é melhor em até 85%, comparado ao OLSR. Uma maior frequência de envio de mensagens de controle mantém as métricas dos enlaces mais utilizados melhor atualizadas. Essa característica tem uma função importante, pois possibilita uma melhor escolha de rotas e, conseqüentemente, uma maior taxa de entrega de dados. Cenários mais densos favorecem mais o algoritmo proposto.

Na Figura 4(b), é mostrado o desempenho dos algoritmos em um cenário denso, onde a qualidade dos enlaces varia. Novamente, a eficiência do algoritmo proposto é melhor que a do OLSR e a do algoritmo de inundação. O algoritmo proposto reduz a carga de roteamento em até 87%. Observa-se que apesar da melhora de desempenho do OLSR em relação ao cenário esparso, a vantagem do algoritmo proposto é ainda mais acentuada. O elevado número de mensagens de controle enviadas por segundo realça a importância do algoritmo proposto. Em cenários reais, a carga de controle compete entre si e com os dados. Portanto, não enviar com sucesso todas as mensagens de controle prejudica o desempenho global da rede. Em vista disso, mais uma vez o algoritmo proposto é mais eficiente que o OLSR e o algoritmo de inundação.

Os resultados obtidos mostram uma redução considerável na carga de controle de roteamento. Mesmo em cenários reais, o emprego do algoritmo proposto mostra-se promissor, contribuindo com a escalabilidade das redes em malha sem fio.



(a) Sem variação na qualidade dos enlaces.

(b) Com variação na qualidade dos enlaces.

Figura 4. Cenário denso.

6. Conclusão

Este artigo propôs um algoritmo para reduzir o número de mensagens de controle enviadas nas redes em malha sem fio por protocolos de roteamento baseados em estado do enlace. Para tanto, o algoritmo proposto explora duas características particulares dessas redes: sua arquitetura e a sua matriz de tráfego. O algoritmo privilegia o envio das mensagens de roteamento na direção do *gateway*, visto que a maior parte do tráfego é encaminhado nessa direção.

Os resultados mostraram que mesmo enviando mensagens de controle com uma maior frequência na direção do *gateway*, o número total de mensagens de controle enviadas na rede permanece menor que o enviado pelo algoritmo de inundação e pelo algoritmo de *MultiPoint Relays* utilizado pelo protocolo OLSR. Isso favorece a transmissão de dados, já que as métricas relacionadas aos enlaces mais utilizados são melhor atualizadas. Em comparação ao OLSR a redução no número de mensagens de controle chega a até 83% em cenários densos, onde não há variação na qualidade dos enlaces. Considerando a variação da qualidade dos enlaces, essa diminuição chega a até 87% também em cenários densos.

Futuramente, pretende-se implementar o algoritmo proposto em um protocolo de roteamento, com vistas à realização de uma prova de conceito em um protótipo de redes em malha, atualmente em implantação no laboratório.

Referências

- Aguayo, D., Bicket, J. e Morris, R. (2005). SrcRR: A high throughput routing protocol for 802.11 mesh networks (DRAFT). Relatório técnico, MIT.
- Akyildiz, I. F., Wang, X. e Wang, W. (2005). Wireless mesh networks: A survey. *Computer Networks*, 47(4):445–487.
- Campista, M. E. M., Moraes, I. M., Esposito, P. M., Amodei Jr., A., Cunha, D. O., Costa, L. H. M. K. e Duarte, O. C. M. B. (2007). The ad hoc return channel: A low-cost solution for Brazilian interactive digital TV. *IEEE Communications Magazine*, 45(1):136–143.

- Cheng, R.-G., Wang, C.-Y., Liao, L.-H. e Yang, J.-S. (2006). Ripple: A wireless token-passing protocol for multi-hop wireless mesh networks. *IEEE Communications Letters*, 10(2):123–125.
- Clausen, T., Jacquet, P., Laouiti, A., Muhlethaler, P., Qayyum, A. e Viennot, L. (2001). Optimized link state routing protocol. Em *IEEE International Multi Topic Conference (INMIC)*, p. 62–68.
- Das, S. M., Pucha, H., Koutsonikolas, D., Hu, Y. C. e Peroulis, D. (2006). Dmesh: Incorporating practical directional antennas in multi-channel wireless mesh networks. *IEEE Journal on Selected Areas in Communications*, 24(11):2028–2039.
- David B. Johnson, D. A. M. e Broch, J. (2001). *Ad Hoc Networking*, chapter 5, p. 139–172. DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks. Addison-Wesley.
- Dong, Q., Banerjee, S. e Liu, B. (2006). Throughput optimization and fair bandwidth allocation in multi-hop wireless LANs. Em *IEEE Conference on Computer Communications (INFOCOM)*.
- Draves, R., Padhye, J. e Zill, B. (2004a). Comparison of routing metrics for static multi-hop wireless networks. Em *ACM SIGCOMM*, p. 133–144.
- Draves, R., Padhye, J. e Zill, B. (2004b). Routing in multi-radio, multi-hop wireless mesh networks. Em *ACM International Conference on Mobile Computing and Networking (MobiCom)*, p. 114–128.
- Gambiroza, V., Sadeghi, B. e Knightly, E. W. (2004). End-to-end performance and fairness in multihop wireless backhaul networks. Em *ACM International Conference on Mobile Computing and Networking (MobiCom)*, p. 287–301.
- Iannone, L. e Fdida, S. (2005). MeshDV: A distance vector mobility-tolerant routing protocol for wireless mesh networks. Em *IEEE ICPS Workshop on Multi-hop Ad hoc Networks: from theory to reality (REALMAN)*, p. 103–110.
- MITRE Corporation (2006). Mobile mesh routing protocol. http://www.mitre.org/work/tech_transfer/mobilemesh/draft-grace-manet-mmrrp-00.txt.
- Nelakuditi, S., Lee, S., Yu, Y., Wang, J., Zhong, Z., Lu, G.-H. e Zhang, Z.-L. (2005). Blacklist-aided forwarding in static multihop wireless networks. Em *IEEE Conference on Sensor and Ad Hoc Communications and Networks (SECON'05)*, p. 252–262.
- Pei, G., Gerla, M. e Chen, T.-W. (2000). Fisheye state routing in mobile ad hoc networks. Em *ICDCS Workshop on Wireless Networks and Mobile Computing*, p. D71–D78.
- Perkins, C. E. e Royer, E. B. (1999). Ad-hoc on-demand distance vector routing. Em *IEEE Workshop on Mobile Computing Systems and Applications*, p. 90–100.
- Ramachandran, K. N., Buddhikot, M. M., Chandranmenon, G., Miller, S., Belding-Royer, E. M. e Almeroth, K. C. (2005). On the design and implementation of infrastructure mesh networks. Em *IEEE Workshop on Wireless Mesh Networks (WiMesh)*.
- Raniwala, A. e Chiueh, T.-C. (2005). Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network. Em *IEEE Conference on Computer Communications (INFOCOM)*, p. 2223–2234.