

Avaliação de Técnicas de Descoberta de Conteúdo em Redes Peer-to-Peer sobre Redes Móveis Ad hoc^{*}

Leonardo B. Oliveira, Daniel F. Macedo, Isabela G. Siqueira,
Antonio A. F. Loureiro, Hao Chi Wong[†], José M. S. Nogueira[‡]

¹Departamento de ciência da computação
Universidade Federal de Minas Gerais
Belo Horizonte, Minas Gerais — Brasil

{leob,damacedo,isabela,loureiro,hcwong,jmarcos}@dcc.ufmg.br

Abstract. *Both Mobile Ad Hoc Networks (MANETs) and Peer-to-Peer (P2P) Overlay Networks are decentralized and self-organizing networks with dynamic topology and responsible for routing queries in a distributed environment. Nevertheless, MANETs are composed of resource-constrained devices susceptible to faults, whereas P2P networks are fault-tolerant. Thus, we argue that these systems are not only similar, but also complementary, and that P2P networks are the ideal data sharing system for MANETs. This paper tackles the problem of integrating both networks from the point of view of content discovery. For that, we confront two classes of strategies: the structured strategy – based on distributed indexing – with the unstructured – based on flooding searches. More specifically, we study the performance of two P2P content discovery protocols over a MANET: Chord (structured) and Gnutella (unstructured). We use simulation to evaluate both protocols under many aspects and conclude that the choice of the strategy to be used depends on the application characteristics and the environment where it would be employed.*

Resumo. *Ambas as Redes Móveis Ad hoc (MANETs) e as Redes Par-a-Par (P2P) são auto-organizáveis, de natureza descentralizada e responsáveis por rotear requisições em ambientes distribuídos de topologia dinâmica. Entretanto, enquanto as MANETs são compostas de dispositivos com poucos recursos e, portanto, suscetíveis a falhas, as redes P2P são conhecidas pela sua tolerância a falhas. Por esta razão, sustentamos que MANETs e redes P2P não são apenas similares, mas também complementares e que sistemas P2P são ideais para se compartilhar arquivos em MANETs. Este trabalho aborda a integração dessas duas redes do ponto de vista de localização de conteúdo. Para tal, confrontamos a estratégia estruturada – que utiliza indexação distribuída – com a não estruturada – que emprega flooding nas buscas. Mais especificamente, estudamos o desempenho sobre uma MANET de dois protocolos P2P de localização de conteúdo: Chord (estruturado) e Gnutella (não-estruturado). Utilizamos simulação para efetuar os estudos sob diversos cenários e os resultados indicam que a escolha da estratégia a empregar dependerá da aplicação e do ambiente em que será utilizada.*

1. Introdução

O advento da computação móvel e o avanço da Internet nos últimos anos possibilitou o aparecimento de novas tecnologias e a quebra de paradigmas. Dentre tais tecnologias duas têm obtido especial atenção devido à vasta gama de aplicações que oferecem e à facilidade de

^{*}Trabalho desenvolvido com apoio do CNPq/MCT e da CAPES/MEC.

[†]Também associada ao Palo Alto Research Center, Palo Alto, CA.

[‡]Em período sabático nas universidades de Evry e UPMC/Paris6/LIP6, França.

serem colocadas em operação: as Redes Móveis *Ad hoc* (MANETs – *Mobile Ad hoc Networks*) e as Redes Par-a-Par (P2P – *Peer-to-Peer*). Por não serem dependentes de infra-estrutura prévia, MANETs possibilitam aplicações de resgate em situações de desastres e troca de informações em campos de batalha [1–4]. Já as redes P2P são uma alternativa às redes baseadas no modelo cliente/servidor e representam uma solução eficiente para compartilhamento de dados e processamento de informação em ambientes distribuídos [3, 5] – razão pela qual foram amplamente adotadas na Internet.

O sucesso de ambas tecnologias em prover soluções deve-se às suas semelhanças [3, 6–12]. Tanto MANETs quanto redes P2P são descentralizadas, auto-organizáveis e responsáveis por rotear requisições em ambientes distribuídos de topologia dinâmica. Além disso, em cada uma delas os integrantes da rede possuem características e funcionalidades equivalentes e podem enviar e responder requisições uns aos outros.

Nós vamos além e sustentamos que MANETs e redes P2P não são apenas similares, mas também complementares. Devido ao fato dos dispositivos nas MANETs geralmente possuírem baixo poder computacional e, por conseguinte, serem incapazes de agir como servidores todo o tempo (ou mesmo atender vários clientes ao mesmo tempo), aplicações P2P prometem ser uma poderosa ferramenta para disseminar informação nessas redes. Em outras palavras, desde que um rede P2P não possua um único provedor de serviço em determinado momento, a distribuição de tarefas entre os nós da rede previne sobrecargas. Além disso, espera-se que algumas aplicações viabilizadas pelas MANETs requeiram que os dispositivos ad-hoc trabalhem em cooperação com os demais, encorajando o compartilhamento de informações. Na comunicação entre membros de um grupo de resgate, por exemplo, um integrante do grupo pode requerer a localização do vizinho (outro integrante) mais próximo. É verdade que em uma rede baseada no paradigma cliente/servidor um dispositivo central pode ser responsável por armazenar essa informação. No entanto, tal abordagem seria não apenas mais cara – a informação desejada poderia estar muito distante e seriam necessárias constantes atualizações de localização –, mas também menos robusta – um único ponto de falha não é desejável em situações de resgate e servidores seriam fáceis alvos fáceis de ataques em campos de batalha.

A integração de redes P2P sobre MANETs não é uma tarefa simples. Ambas as redes precisam lidar com a conexão e desconexão dos nós. Além disso, as propriedades das MANETs, como o roteamento [6, 8], influenciam consideravelmente o comportamento da rede P2P. Existem muitos trabalhos que conduzem avaliações de desempenho de localização de conteúdo em redes P2P [13–16]. Embora esses estudos apresentem informações valiosas a respeito de diferentes classes de protocolos P2P, eles tratam do problema em redes fixas e não consideram o desempenho desses protocolos em ambientes com muitas perdas e alta dinamicidade, cenários típicos para MANETs. Apenas alguns, conforme descritos na Seção 5, estudam a integração de redes P2P sobre MANETs e propõem soluções de localização de conteúdo. Contudo, diversos aspectos ainda precisam ser considerados para que a integração possa ser viabilizada na prática.

Neste trabalho nós contribuimos para o desenvolvimento de compartilhamento eficiente de informações em MANETs através da avaliação do uso de técnicas de descoberta de conteúdo propostas para a Internet. Em particular, nós avaliamos duas técnicas de localização de redes P2P neste novo ambiente distribuído, uma estruturada e outra não estruturada. Para tal, empregamos dois protocolos utilizados na Internet, o Chord (estruturado) e o Gnutella (não-estruturado). Embora tais técnicas já tenham sido muito estudadas em redes fixas tradicionais, não temos conhecimento anterior de estudos voltados para redes P2P sobre MANETs. Com isso, também estamos contribuindo para identificar características dos protocolos – e de suas respectivas estratégias – que colaboram para o melhor desempenho do sistema, visando descobrir pontos fracos que ajudarão na adaptação e melhoria desses protocolos para MANETs.

Os resultados mostram que os protocolos não-estruturados, por utilizarem redundância de mensagens, são mais eficientes na localização do conteúdo, embora sejam mais caros que os protocolos estruturados. Estes, por outro lado, se mostraram mais apropriados para ambientes estáticos e permitem melhor economia de energia. Em geral, conclui-se que a escolha 7por adotar

uma ou outra solução depende do cenário e das características da aplicação e, em alguns casos, de uma união de ambas a fim de encontrar uma melhor relação custo-benefício.

O restante deste trabalho está organizado da seguinte maneira. A Seção 2 descreve brevemente os dois protocolos P2P utilizados. A Seção 3 apresenta o ambiente de simulação e o método de análise utilizado neste trabalho. A Seção 4.1 apresenta as avaliações e descreve os resultados de simulação, enquanto a Seção 5 mostra os trabalhos correlatos. Finalmente, a Seção 6 apresenta as conclusões.

2. Descrição dos Protocolos P2P Implementados

A fim de alcançar os objetivos deste trabalho, foi necessária a implementação de dois protocolos P2P, um estruturado e outro não. Optamos pelos protocolos estruturado Chord e não-estruturado Gnutella, respectivamente, pois são consolidados, difundidos e existe vasta documentação sobre os mesmos.

2.1. Protocolo Não-estruturado

O protocolo Gnutella propaga suas requisições por meio de *flooding* controlado. Sempre que um *peer* recebe uma requisição, ele tenta resolvê-la. Caso consiga, uma mensagem de resposta é enviada ao *peer* requerente diretamente. Caso contrário, o *peer* de posse da requisição a redireciona para todos os *peers* de sua lista de vizinhos. Para lidar com os problemas de propagação indefinida de requisições, o Gnutella utiliza uma estratégia semelhante à do protocolo IP, inserindo um campo de *Time-to-Live* (TTL) em suas mensagens de requisição, que é decrementado a cada *hop*. Além disso, cada um dos seus nós mantém uma *cache* que impede que uma mesma requisição seja considerada duplamente. Mensagens de entrada que apresentam TTL igual a zero ou que já foram registradas antes na *cache* são imediatamente descartadas.

Em nossa implementação do Gnutella, os vizinhos lógicos são escolhidos de forma aleatória. Para cada *peer* que entra na rede, é associado um número fixo de vizinhos, escolhidos dentre os *peers* que estão *online* no momento. Essa atribuição simula as redes Gnutella existentes na Internet, pois estas possuem servidores que servem como ponto de entrada na rede. Para manter atualizada a lista de vizinhos, os *peers* periodicamente enviam mensagens de *ping* para sua cada vizinho e ficam a espera de mensagens *pong* como resposta. Aqueles vizinhos que não respondem são substituídos por outros, utilizando a mesma política descrita anteriormente.

2.2. Protocolo Estruturado

No protocolo Chord [17], os *peers* são responsáveis por no máximo $\frac{1}{n}$ de todos os arquivos em uma rede de n nós. Isto faz com que a memória e largura de banda sejam balanceadas entre os nós e melhora o desempenho do sistema. O protocolo implementa buscas eficientes ao utilizar uma tabela chamada de *finger* (*finger table*), que implementa uma tabela *hash* distribuída (DHT), a qual armazena índices para viabilizar buscas em $O(\log n)$ tentativas.

De acordo com a especificação do protocolo [17], os *peers* são arranjados em um anel. A posição de um *peer* no anel é seu identificador módulo 2^m . Para manutenção das referências para vizinhos no anel, as quais podem se tornar inconsistentes devido à entrada e saída de *peers*, estes periodicamente executam a função *stabilize*.

Em relação ao Chord, nós implementamos seu conjunto completo de funcionalidades, incluindo os protocolos necessários para construção e manutenção dos índices distribuídos. Também implementamos as operações de inserção e exclusão, usando algoritmos similares ao de busca.

Os protocolos foram implementados sobre a plataforma de simulação do ns-2 (*Network Simulator*) [18] e seu módulo de comunicação sem fio e mobilidade [19]. A escolha pelo simulador deve-se ao fato do mesmo ser amplamente utilizado pela comunidade científica e já possuir implementação consolidada do protocolo de roteamento ad-hoc AODV.

3. Modelo de Rede

Em função do incipiente estudo de redes P2P sobre MANETs, não existem modelos reais para basearmos nossos experimentos. Optamos por modelar nossa rede a partir de uma aplicação voltada para grupos de busca e resgate em situações de desastre (terremotos, maremotos, tsunamis etc). Nesta aplicação, os integrantes dos grupos são equipados com dispositivos de comunicação sem fio (*handhelds*, *palmtops*, PDAs) e formam uma MANET, onde trocam conteúdo (arquivos com tamanho seguindo a distribuição uniforme com média de 10 KB) entre si através de uma rede P2P. Consideramos que um mesmo conteúdo pode ser disponibilizado por diferentes integrantes. Além disso, cada integrante possui inicialmente 10 arquivos distintos, de um total de 100. A distribuição dos arquivos entre os integrantes foi feita de maneira aleatória. Note-se que para a transmissão de voz apenas, uma estratégia mais eficiente seria o emprego de rádio, mas acreditamos que, em aplicações como essa, outras formas de conteúdo (como arquivos de fotos dos locais acidentados) também deverão ser compartilhados. Além disso, enquanto a comunicação via rádio demanda dos integrantes estar sempre atentos à comunicação, a aplicação P2P sobre MANET proporciona ao interessado a autonomia de buscar o conteúdo desejado quando bem entender.

Falhas de enlace podem acontecer em nosso modelo, visto que a qualidade da comunicação é altamente dependente das condições do ambiente [20]. O modelo assume qualidade de enlace homogênea, em que todos os enlaces da rede possuem taxas de erro equivalentes. Entretanto, em um dos cenários avaliamos o impacto da qualidade do canal. Não consideramos falhas por falta de energia nas simulações.

Os integrantes da rede se movem em uma área de 200×200 m, com seu movimento regido pelo modelo *random way-point* [21], que é usualmente empregado para modelar mobilidade de indivíduos [21]. Este modelo alterna o estado do nó entre períodos de repouso e de mobilidade. Após esperar um tempo (tempo de repouso) no local em que se encontra, o nó escolhe aleatoriamente uma posição e se desloca para ela em velocidade uniforme escolhida a partir de um intervalo dado. Os integrantes ainda podem entrar e sair da rede, como em um revezamento de trabalhadores. Essa dinâmica seguiu a distribuição uniforme para escolha de nós que entravam e saíam da rede.

A aplicação que instancia a rede P2P é executada sobre o protocolo UDP [22], não havendo ACKs [22] e/ou retransmissões de mensagens perdidas. O modelo de comunicação entre os nós foi baseado no padrão IEEE 802.11 [22], com raio de alcance de 50 m, como encontrado em dispositivos portáteis. Os gastos de energia com envio e recepção de mensagens foram 330 mW e 230 mW, respectivamente [23].

4. Resultados

Nesta seção, apresentamos os resultados da avaliação dos protocolos P2P de localização de conteúdo sobre uma MANET em que o AODV atuou como protocolo de roteamento. Optamos pelo AODV porque foi o protocolo que, no geral, apresentou o melhor desempenho na avaliação descrita em [8]. A configuração do conjunto de simulações para essa avaliação é descrita a seguir.

4.1. Simulação

Os resultados da simulação foram divididos por propriedade avaliada. São elas: carga da rede, tamanho da rede, erro do canal, mobilidade e dinâmica da rede. Para alcançar tais propriedades, variamos um ou mais parâmetros de um conjunto. Este conjunto foi guiado por um padrão que segue o modelo descrito na Seção 3. A rede é composta de 50 nós, todos eles executando uma instância da aplicação P2P. O tempo de repouso e a velocidade máxima de cada nó são 0.1 s e 1.0 m/s, respectivamente.

Os *peers* do Gnutella tiveram no máximo 4 vizinhos (vide Apêndice) e o TTL das mensagens foi de 4 saltos. O Chord utilizou uma função *hash* de 10 bits e as tabelas *finger* eram atualizadas a cada 5 s. Mensagens de PING eram enviadas a cada 10 s e o *timeout* de um *pong* era de 3 s. As mensagens de ambos os protocolos possuíam o tamanho fixo de 64 bytes.

O tempo de simulação foi 500 s. Os resultados apresentados representam a média das 33 execuções, cada uma utilizando uma semente distinta para o gerador de números aleatórios. As métricas consideradas durante a simulação foram:

- **Taxa de sucesso:** o percentual de requisições atendidas com sucesso pela rede P2P.
- **Atraso:** o atraso percebido pelo usuário ao requerer conteúdo. Ele inclui o tempo de transmissão, localização e envio de resposta a uma requisição.
- **Energia por sucesso:** percentual de energia consumida por um 1% do total de *sucessos*. Essa métrica indica a eficiência da busca em termos de energia.
- **Consumo de energia:** consumo total de energia da rede.
- **Mensagens enviadas:** número total de mensagens P2P enviadas durante a simulação.

4.2. Carga da Rede

Analisamos o efeito da carga da rede sobre o desempenho dos protocolos P2P. Para tal, variamos o número de arquivos distintos na rede e o número de requisições efetuadas como um todo. Os resultados obtidos, apresentados em função do número de requisições efetuadas ao longo da simulação, são apresentados nas Figuras 1, 2, 3 e 4.

As Figuras 1, 2, 3 e 4 apresentam os resultados em função do número total de requisições efetuadas ao longo da simulação. Como pode ser observado na Figura 1, o Gnutella apresenta maiores taxas de sucesso (entre 60% e 70%, aproximadamente) do que o Chord (entre 10% e 20%, aproximadamente). Essa diferença deve-se à redundância de mensagens do Gnutella, em que cada *peer* redireciona uma requisição a toda sua lista de vizinhos. O Chord, por outro lado, depende apenas de uma única cópia da requisição, que pode ser perdida devida à baixa confiabilidade dos enlaces sem fio. Contudo, o custo do bom resultado do Gnutella é alto tanto em termos de atraso como em *overhead* de mensagens (Figuras 2 e 3, respectivamente). Quando comparado com o Chord, o Gnutella acarretou em atrasos de 200% a 1570% maiores e envios de mensagens de 111% a 851% maiores.

O Gnutella apresentou o maior consumo total de energia (Figura 4). Paradoxalmente, ele apresentou o menor consumo por sucesso. Tal resultado deveu-se à energia que o Chord gastou para manter as informações sobre os estado do anel consistentes. Embora o Chord tenha apresentado baixas taxas de sucesso (um pouco maior que 10%), o protocolo apresentou melhor escalabilidade e sofreu menor impacto ao variarmos a carga da rede.

4.3. Tamanho da Rede

Em seguida, avaliamos o impacto do número de *peers* em cada protocolo. A fim de mantermos a densidade da rede fixa (1.0 peer/m^2), ajustamos o tamanho da área de acordo com o tamanho da rede (número de *peers*). Os resultados obtidos são apresentados nas Figuras 5, 6, 7 e 8. Para uma rede com 100 *peers*, o Gnutella obteve a maior taxa de sucesso – 370%, 580%, e novamente 370% para 25, 50 e 75 *peers*, respectivamente (Figura 5). Por outro lado o protocolo Chord foi mais eficiente no tempo de resposta e número de mensagens enviadas. Ele apresentou atraso 77% menor e enviou de 25% a 70% menos mensagens (Figuras 6 e 7, respectivamente) que o Gnutella.

É importante notar que ambos os protocolos foram muito sensíveis à variação do tamanho da rede e apresentaram os melhores desempenhos em redes de tamanho médio (de 20 a 50 *peers*). Também verificamos que o consumo de energia por sucesso foi similar para os dois protocolos (Figura 8). Finalmente, nem Chord nem Gnutella lidaram bem com o crescimento da rede. No Gnutella, o uso de TTL acarretou em uma diminuição da taxa de sucesso do protocolo decresce à medida que a rede crescia.

4.4. Erro do Canal

Redes sem fio comumente apresentam taxas de erro bem maiores do que as redes cabeadas, o que leva a um número considerável de pacotes descartados. A qualidade do enlace é altamente

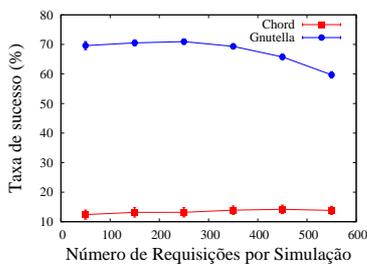


Figura 1: Taxa de sucesso

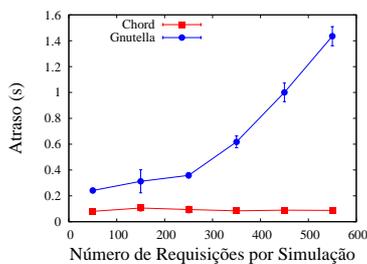


Figura 2: Atraso

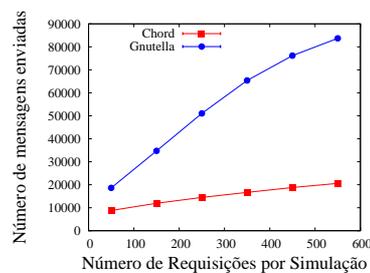


Figura 3: Mensagens enviadas

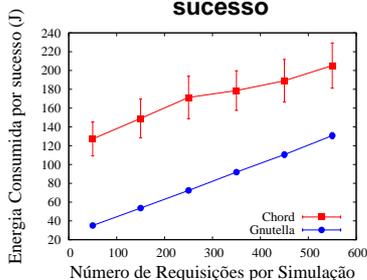


Figura 4: Energia por sucesso

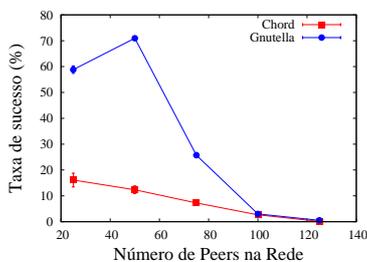


Figura 5: Taxa de sucesso

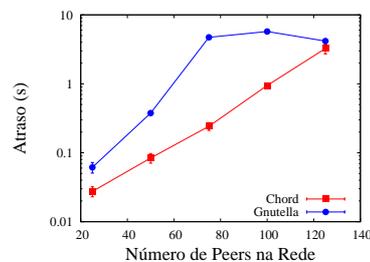


Figura 6: Atraso

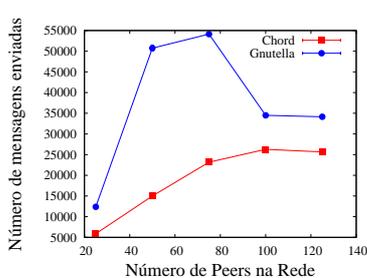


Figura 7: Mensagens enviadas

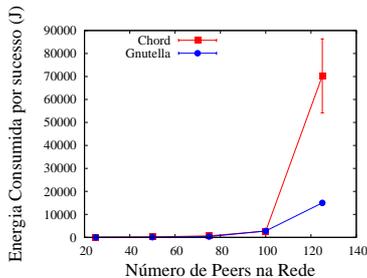


Figura 8: Energia por sucesso

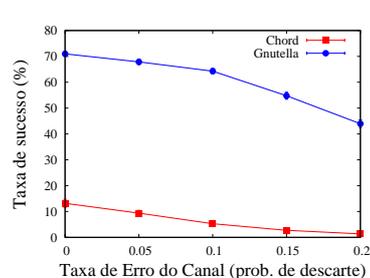


Figura 9: Taxa de sucesso

dependente do ambiente e da localização do nó, variando significativamente de uma região para outra [20, 24].

Como mostra a Figura 9, o Chord apresentou taxas de sucesso menores que 10%. Esse resultado deve-se à dependência do protocolo à confiabilidade do meio, visto que o protocolo não possui mensagens redundantes. O Gnutella, por outro lado, mostrou um bom desempenho até mesmo em ambientes cujas taxas de erros foram medianas, isto é, 0.05 e 0.1 %. Contudo, nem mesmo a redundância de mensagens do Gnutella foi capaz de mitigar satisfatoriamente descartes em cenários sob alta taxa de erro, pois nem todas as mensagens do protocolo não-estruturado possuem réplicas, aumentando a vulnerabilidade do protocolo a perdas de pacotes.

O tempo de resposta do Gnutella apresentou um aumento quase exponencial (600%) com o aumento do erro do canal (Figura 10). O Chord, entretanto, mostrou-se mais estável e variou seu tempo de resposta no máximo em 12%. Ele também envia cerca de 72% menos mensagens que o Gnutella (Figura 11).

A energia consumida por sucesso do Gnutella foi a menor entre os protocolos. Quando a taxa de erro foi 20%, o consumo de energia por sucesso do Gnutella foi, aproximadamente, 8× menor do que o do Chord (Figura 12). Essa mesma métrica apresentou crescimento exponencial para o Chord. Como menos requisições foram completadas, o *overhead* para manter o anel dominou o consumo de energia, incrementando o consumo de energia por sucesso (Figura 13). Entretanto, o Chord ainda consumiu menos energia por nó que o Gnutella, mesmo em cenários em que o Chord consome mais energia por sucesso.

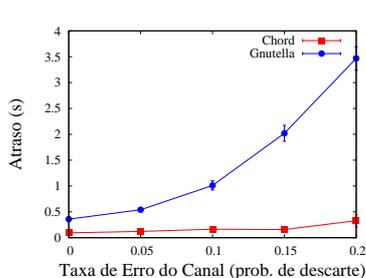


Figura 10: Atraso

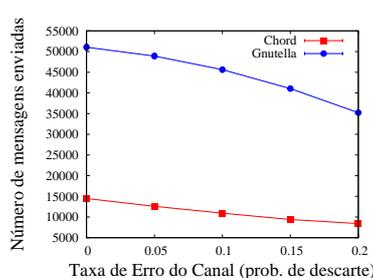


Figura 11: Mensagens enviadas

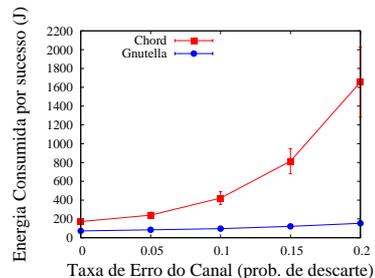


Figura 12: Energia por sucesso

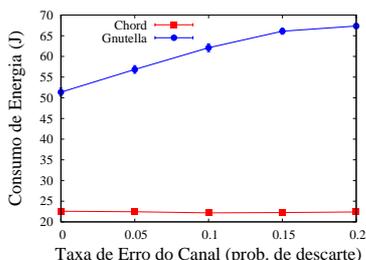


Figura 13: Energia consumida

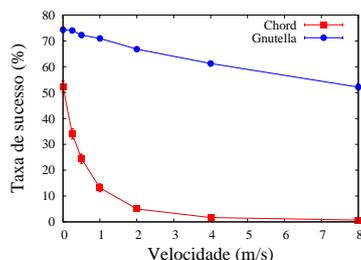


Figura 14: Taxa de sucesso

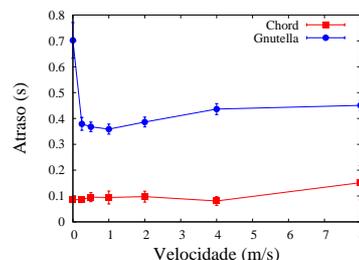


Figura 15: Atraso

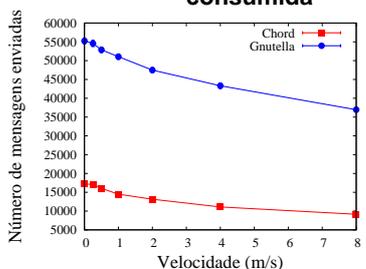


Figura 16: Mensagens enviadas

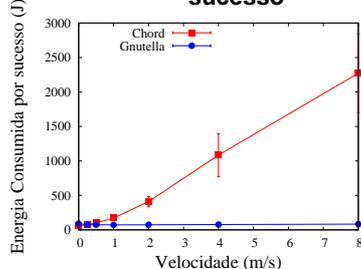


Figura 17: Energia por sucesso

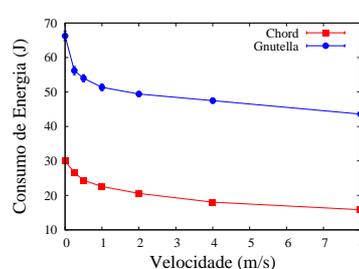


Figura 18: Energia consumida

4.5. Mobilidade

Apesar de redes P2P serem tolerantes a falhas, essas redes não foram projetadas para executar em ambientes móveis, onde desconexões ocorrem com frequência. A seguir, avaliamos o impacto de mudanças de topologia, devido à mobilidade, nos protocolos P2P. Apresentamos resultados em função das seguintes velocidades de nós: 0, 0.25, 0.5, 1, 2, 4 e 8 m/s.

Em cenários de baixa mobilidade, o Gnutella foi o que transmitiu o maior número de mensagens (Figura 16) e, por conseguinte, seu tempo de resposta foi também maior (Figura 15). Pela Figura 15, verificamos que a mobilidade é benéfica ao Gnutella, uma vez que menos pacotes foram corretamente entregues, e assim um menor tráfego foi imposto à rede. Contudo, sob alta mobilidade, ambos os protocolos sofreram degradação, aumentando o tempo de resposta e diminuindo a taxa de sucesso dos protocolos. Ao contrário dos outros cenários, o Chord foi o menos estável dos protocolos, apresentando alta variação na maior parte das métricas avaliadas. À medida que a velocidade foi acrescida de 0 a 2 m/s, a taxa de sucesso do Chord (Figura 14) foi decrescida de 50% para 5% e, continuando esse crescimento, a taxa chegou próxima de zero. O Gnutella foi menos afetado pela mobilidade, possuindo taxas de sucesso superiores a 60% durante todas as simulações.

A Figura 18 mostra o consumo médio de um nó durante toda a simulação. Curiosamente, o consumo de energia decresceu à medida que a mobilidade aumentou. Em relação à energia por sucesso (Figura 17), o Gnutella se manteve estável, enquanto o Chord aumentou o consumo e atingiu o pico de 2300 J, aproximadamente, a 8 m/s. Note que, quanto maior a mobilidade, menor o número de mensagens descartadas durante a simulação (Figura 19). Esses descartes

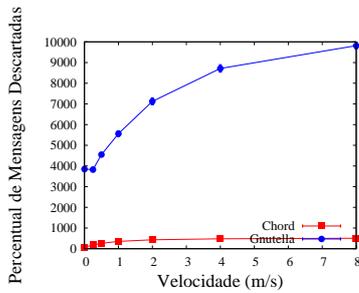


Figura 19: Mensagens descartadas

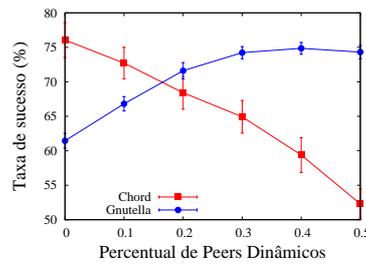


Figura 20: Taxa de sucesso

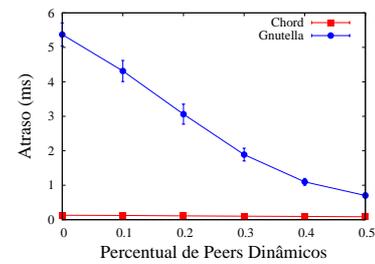


Figura 21: Atraso

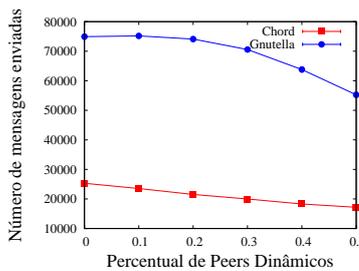


Figura 22: Mensagens enviadas

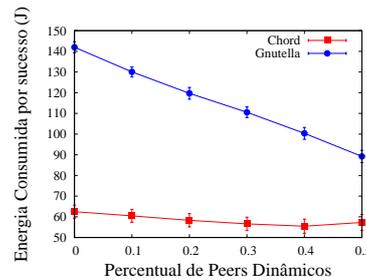


Figura 23: Energia por sucesso

foram causados por colisões e/ou vizinhos que estavam fora de alcance. Para velocidades acima de 4 m/s, no entanto, o número de mensagens descartadas estabiliza em torno de 500, para o Chord, e 10.000, para o Gnutella.

4.6. Dinâmica da Rede

A seguir, apresentamos os resultados da avaliação da dinâmica da rede, que identificam como a entrada e saída de *peers* altera o comportamento da rede. *A priori*, sabemos que o Gnutella se auto-organiza mais rapidamente, visto que o Chord, ao contrário, demanda dos *peers* várias operações para que possam entrar e/ou sair da rede. Neste cenário variamos o percentual de *peers* deixando a rede entre 50% a 0%. Inicialmente, variamos o número de *peers* dinâmicos da rede – *peers* que deixavam a rede em algum momento da simulação – em cenários com mobilidade. Nesses cenários, os resultados relativos ao Chord foram negativos. Decidimos, então, repetir as simulações para uma rede estacionária.

Quando o número de *peers* dinâmicos diminuiu, a disponibilidade dos *peers* aumentou, e a rede Gnutella foi saturada com requisições. Isso é verificado no tempo de resposta (Figura 21) e no número de mensagens enviadas (Figura 22), que são inversamente proporcionais ao número de *peers* dinâmicos.

O Chord não apresentou bom desempenho para topologias muito dinâmicas, mas foi o que atuou melhor para até 20% de *peers* dinâmicos. Além de atingir taxas de sucesso bem próximas às do Gnutella, o Chord ainda obteve tempo de resposta menor (Figura 21), e menor consumo de energia por sucesso (Figura 23). Quando a rede não está saturada, o aumento do tráfego devido a um número maior de *peers* na rede Chord é mais que compensada pelo aumento da taxa de sucesso.

Note-se que, embora a maioria das aplicações possua alto índice de desconexões e de rotatividade de *peers*, isso não é uma regra. Existem aplicações que são executadas em ambientes de maior confiabilidade e por um número restrito de *peers*, e, por conseguinte, são mais “bem comportadas”. A análise da propriedade de dinâmica da rede mostrou que em contextos menos dinâmicos, o Chord pode apresentar menor atraso e consumo de energia que o Gnutella.

4.7. Discussão

Verificamos pelos resultados que protocolos estruturados não são apropriados para redes que apresentam alta dinâmica de nós ou baixa confiabilidade do canal. Entretanto, para redes estáticas e com baixa interferência, estes se mostram superiores aos não estruturados, alcançando taxas de sucesso superiores, e ao mesmo tempo, consumindo menos energia. A variação no desempenho dos protocolos estruturados se dá pelo comportamento altamente dinâmico da rede, que modifica a taxa de perda de pacotes. Os resultados mostram que o principal fator no desempenho destes protocolos é a mobilidade dos nós, assim quando a rede é estática o desempenho é limitado pelo número de nós ativos. Os protocolos não estruturados, entretanto, apresentam um desempenho mais estável, mostrando bons resultados em todos os cenários.

A implementação do protocolo estruturado utilizada nas simulações não utiliza transmissões confiáveis, como no TCP. Acreditamos que o uso de protocolos de transporte poderia melhorar o desempenho dos protocolos estruturados, apesar do aumento da complexidade e do número de mensagens enviadas. Além disto, o Chord apresenta um overhead maior que o Gnutella nos resultados apresentados, pois implementamos todas as funcionalidades do Chord, enquanto o protocolo Gnutella avaliado não possui algoritmos de descoberta de vizinhos.

5. Trabalhos Relacionados

Recentemente a comunidade de pesquisa atentou para a sinergia entre MANETs e redes P2P. Os trabalhos pioneiros compararam as redes apresentando suas similaridades e diferenças. Já os mais recentes propuseram soluções para ambientes em que MANETs e redes P2P coexistem.

Schollmeier et al. [7] e Borg [3] levantaram características em comum e as diferenças entre os dois tipos de rede. O primeiro focou principalmente em aspectos de roteamento e levantou questões sobre como seria a localização de conteúdo em protocolos P2P inspirados em protocolos de roteamento ad-hoc. Analogamente, indagou se o uso de funções *hash* e/ou *bloom filters* em protocolos de roteamento ad-hoc poderiam diminuir o tráfego da rede. Já Borg [3] discute aspectos de descoberta de *peers*, roteamento, localização de conteúdo, segurança e qualidade de serviço de ambas as redes. Borg identificou três semelhanças entre redes P2P e MANETs: (natureza descentralizada, heterogeneidade de conteúdo entre *peers/nós*, e conectividade temporária entre os *peers*).

Kortuem et al. propuseram *Proem* [25, 26], uma plataforma de *middleware* para desenvolver e disponibilizar aplicações P2P voltadas para PANs (*Personal Area Networks*). O *middleware* permite que engenheiros de *software* aumentem a sua produtividade no desenvolvimento de soluções P2P para MANETs.

Papadopouli e Schulzrinne [27, 28] e Klemm et al. [12] apresentaram sistemas de compartilhamento de dados P2P voltados para MANETs, chamados de *Seven Degrees of Separation (7DS)* e *Optimized Routing Independent Overlay Network (ORION)*. O 7DS é uma arquitetura que, por meio de um conjunto de protocolos, possibilita troca de dados entre *peers* que não estão diretamente conectados à Internet ao explorar a mobilidade dos nós, para tanto empregando técnicas P2P de compartilhamento de dados. Já o ORION consiste de um algoritmo de construção e manutenção de uma rede *overlay* que proporciona requisições, respostas e transferência de arquivos utilizando o paradigma P2P. No ORION, as conexões são feitas sob demanda e mantidas apenas quando necessárias. Além disso, o ORION busca construir a rede *overlay* de forma a refletir a rede subjacente, dessa forma minimizando o atraso e o *overhead*.

Franciscani et al. [9] propuseram quatro algoritmos de configuração e reconfiguração de sistemas P2P sobre MANETs. Os algoritmos possuíam o foco na otimização da configuração e reconfiguração da topologia. No algoritmo *regular*, *peers* conectam-se a vizinhos próximos (físicos), no *aleatório* parte da atribuição de vizinhos é feita de forma aleatória, buscando o fenômeno *small-world* [29,30]. Finalmente, no algoritmo *híbrido*, as conexões são feitas seguindo uma hierarquia.

Ding e Bhargava [11] propuseram 5 abordagens de roteamento para redes P2P sobre MANETs. As abordagens são construídas através da combinação de estratégias de localização de conteúdo P2P e de roteamento ad-hoc, isto é, DHT e *broadcast*. Eles utilizaram as seguintes combinações para a camada P2P *overlay* e a de roteamento ad-hoc: *broadcast* (virtual) sobre *broadcast* (físico), DHT sobre *broadcast* (físico) e DHT sobre DHT. Além disso, também foram avaliados *broadcast* e DHT como protocolos *cross-layer*, isto é, um único protocolo para as duas camadas. Apesar dos importantes resultados, Ding e Bhargava não avaliaram protocolos P2P realmente adotados (Chord, Pastry, Gnutella, etc.), nem protocolos de roteamento *ad hoc* consolidados (DSR, DSDV, AODV, etc.). Além disso, o trabalho proposto não considera aspectos práticos (como mobilidade e erro do canal).

Outras propostas integraram conceitos de ambos os sistemas. Hu et al. [10], por exemplo, propuseram um protocolo de roteamento ad-hoc dinâmico chamado DSPR (*Dynamic P2P Source Routing*). Para melhorar a escalabilidade, o protocolo integra estratégias de roteamento do DSR e técnicas P2P de localização de conteúdo do Pastry [31].

6. Conclusão

O avanço da computação vêm quebrando paradigmas e possibilitando o surgimento de novas tecnologias. Dentre essas, MANETs e Redes P2P se destacam pela gama de aplicações que proporcionam e pela facilidade de serem colocadas em operação. Elas não são apenas semelhantes, mas também complementares. Logo, é necessário identificar estratégias de otimizar ambientes em que elas coexistem.

Neste trabalho contribuímos para o estado da arte ao efetuarmos uma avaliação de protocolos P2P de localização de conteúdo sobre uma MANET. Verificamos que nenhum protocolo sobressaiu-se em todos os cenários, mas sim em cenários específicos ou que apresentou melhor desempenho levando-se em conta uma determinada métrica. A estratégia estruturada, através do Chord, mostrou-se pouco viável em ambientes de topologia muito dinâmica e/ou de alta taxa de erro. O mau desempenho deveu-se ao consumo de largura banda pelas mensagens de controle, freqüentes em redes P2P estruturadas, e a ausência de redundância de pacotes. Por outro lado, em cenários estáticos com alta confiabilidade de canal, o Chord obteve melhor desempenho que o Gnutella, produziu maiores taxas de sucesso e menor consumo de energia (Seção 4.6) – entretanto esses cenários são a exceção, e não a regra em redes P2P sobre MANETs. A estratégia não-estruturada, por meio do Gnutella, mostrou-se a mais adequada (maior taxa de sucesso, consumo de energia mais eficiente) na maioria dos cenários. O bom desempenho deveu-se, principalmente, ao envio de requisições por *flooding*, que duplica mensagens e atenua descartes causados por falta de rota e erro do canal.

Um ramo de sistemas P2P sobre MANETs que ainda está à espera de ser pesquisado é o de modelos analíticos. Seja para construir a vizinhança lógica, seja para armazenar conteúdo, até onde sabemos, não existem modelos que representem esses sistemas sobrepostos. Verificamos a necessidade de uma análise do grau de redundância de mensagens tendo em vista o erro do canal, para determinar mais precisamente como a redundância afeta o desempenho dos protocolos.

Referências

- [1] Jiejun Kong and Mario Gerla. Providing real-time security support for multi-level ad-hoc networks. In *Proceedings of the IEEE Military Communications Conference (MILCOM'02)*, 2002.
- [2] Lidong Zhou and Zygmunt J. Haas. Securing ad hoc networks. *IEEE Network*, 13(6):24–30, November/December 1999. Special Issue on Network Security.
- [3] Joseph Borg. A comparative study of ad hoc & peer to peer networks. Master's thesis, University College London, 2003.
- [4] Z. J. Haas, J. Deng, B. Liang, P. Papadimitratos, and S. Sajama. Wireless ad hoc networks. In John G. Proakis, editor, *Wiley Encyclopedia of Telecommunications*. John Wiley & Sons, 2002.

- [5] Domenico Talia and Paolo Trunfio. Toward a synergy between P2P and grids. *IEEE Internet Computing*, 7(4):94–95, 2003.
- [6] Leonardo B. Oliveira, Isabela G. Siqueira, Antonio A. F. Loureiro. On the performance of ad hoc routing protocols under a peer-to-peer application. *Journal of Parallel and Distributed Computing (JPDC)*, 2005. Special Issue on the “Design and Performance of Networks for Super-, Cluster-, and Grid-Computing”. To appear.
- [7] Rüdiger Schollmeier, Ingo Gruber, and Michael Finkenzeller. Routing in peer-to-peer and mobile ad hoc networks: A comparison. In *Proceedings of International Workshop on Peer-to-Peer Computing*, Pisa, Italy, 2002. held in conjunction with IFIP Networking 2002.
- [8] Leonardo B. Oliveira, Isabela G. Siqueira, Antonio A. F. Loureiro. Evaluation of Ad Hoc Routing Protocols Under a Peer-to-Peer Application. In *IEEE Wireless Communications and Networking Conference (WCNC’03)*, pages 1143–1148, New Orleans, USA, March 2003.
- [9] Fernanda P. Franciscani, Marisa A. Vasconcelos, Rainer P. Couto, and Antonio A. F. Loureiro. Peer-to-peer over ad-hoc networks: (re)configuration algorithms. *Journal of Parallel and Distributed Computing (JPDC)*, 2005. Special Issue. Also appeared in 17th IEEE International Parallel and Distributed Processing Symposium 2003 (IPDPS’03).
- [10] Y. Charlie Hu, Saumitra M. Das, and Himabindu Pucha. Exploiting the Synergy between Peer-to-Peer and Mobile Ad Hoc Networks. In *HotOS-IX: Ninth Workshop on Hot Topics in Operating Systems*, Lihue, Kauai, Hawaii, May 18–21 2003.
- [11] Gang Ding and Bharat Bhargava. Peer-to-peer file-sharing over mobile ad hoc networks. In *2th IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pages 104–108, Orlando, Florida, March 2004.
- [12] Alexander Klemm, Christoph Lindemann, and Oliver P. Waldhorst. A special-purpose peer-to-peer file sharing system for mobile ad hoc networks. In *IEEE Semiannual Vehicular Technology Conference (VTC2003-Fall)*, October 2003.
- [13] Zihui Ge, Daniel R. Figueiredo, Sharad Jaiswal, Jim Kurose, and Don Towsley. Modeling peer-peer file sharing systems. In *Proceedings of the 22nd IEEE Conference on Computer Communications*, 2003.
- [14] Dimitrios Tsoumakos and Nick Roussopoulos. A comparison of peer-to-peer search methods. In *Sixth International Workshop on the Web and Databases*, pages 61–66, 2003.
- [15] Tsungnan Lin and Hsinping Wang. Search performance analysis in peer-to-peer networks. In *IEEE 3rd International Conference on Peer-to-Peer Computing (P2P’03)*, pages 204–205. IEEE Computer Society, 2003.
- [16] Beverly Yang and Hector Garcia-Molina. Improving search in peer-to-peer networks. In *22nd International Conference on Distributed Computing Systems (ICDCS’02)*, pages 5–14. IEEE Computer Society, 2002.
- [17] Ion Stoica, Robert Morris, David Liben-Nowell, David R. Karger, M. F. Kaashoek, Frank Dabek, and Hari Balakrishnan. Chord: a scalable peer-to-peer lookup protocol for internet applications. *IEEE/ACM Transactions on Networking*, 11(1):17–32, 2003. Also appeared in Proceedings of the ACM SIGCOMM ’01 Conference.
- [18] Kevin Fall and Kannan Varadhan. *Network Simulator Notes and Documentation*. The VINT Project, 2001.
- [19] The CMU Monarch Project. The CMU Monarch Projects Wireless and Mobility Extension to NS, September 2004. Work in Progress.
- [20] G. Gaertner and V. Cahill. Understanding link quality in 802.11 mobile ad hoc networks. *IEEE Internet Computing*, 8(1):55–60, 2004.
- [21] Josh Broch, David A. Maltz, David B. Johnson, Yih-Chun Hu, and Jorjeta Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Proceedings of the 4th annual ACM/IEEE International Conference on Mobile Computing and Networking*, pages 85–97, Dallas, Texas, United States, 1998. ACM Press.
- [22] Andrew S. Tanenbaum. *Computer Networks*. Prentice Hall, 4th edition, August 2002.
- [23] Juan-Carlos Cano and Pietro Manzoni. A performance comparison of energy consumption for mobile ad hoc network routing protocols. In *8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS’00)*, pages 57–64, San Francisco, CA, August 29–September 1 2000. IEEE Computer Society.

- [24] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris. A high-throughput path metric for multi-hop wireless routing. In *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MOBICOM'03)*, pages 134–146, San Diego, CA, USA, 2003. ACM Press.
- [25] G. Kortuem, J. Schneider, D. Preuitt, T. G. C. Thompson, S. Fickas, and Z. Segall. When peer-to-peer comes face-to-face: Collaborative peer-to-peer computing in mobile ad hoc networks. In *IEEE 1st International Conference on Peer-to-Peer Computing (P2P'02)*, pages 75–91, Linkopings, Suécia, August 2001.
- [26] Gerd Kortuem. Proem: a middleware platform for mobile peer-to-peer computing. *SIGMOBILE Mobile Computing and Communication Review*, 6(4):62–64, October 2002. Special Feature on Middleware for Mobile Computing.
- [27] Maria Papadopouli and Henning Schulzrinne. Effects of power conservation, wireless coverage and cooperation on data dissemination among mobile devices. In *2nd ACM international symposium on Mobile ad hoc networking and computing*, pages 117–127. ACM Press, March 2001.
- [28] Maria Papadopouli and Henning Schulzrinne. A performance analysis of 7DS a peer-to-peer data dissemination and prefetching tool for mobile users. In *Advances in wired and wireless communications, IEEE Sarnoff Symposium Diges*, Ewing, USA, March 2001.
- [29] S. Milgram. The small-world problem. *Psychology Today*, 1(1):60–67, 1967.
- [30] D.J. Watts and S. Strogatz. Collective dynamics of small-world networks. *Nature*, 393(6):440–442, June 1998.
- [31] Antony Rowstron and Peter Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, pages 329–350, 2001.

Apêndice: Número Máximo de Vizinhos Gnutella

No protocolo Gnutella, os *peers* enviam uma cópia de cada requisição para todos os seus vizinhos lógicos. Essa estratégia resulta no envio de diversas das mensagens de requisição, o que pode trazer mais sucesso nas buscas por conteúdo. Por outro lado, isso acarreta em maior tráfego, atraso e consumo de energia, que, paradoxalmente, pode levar ao decréscimo dos sucessos nas buscas. Existe, assim, um compromisso entre taxa de sucesso, atraso e consumo de energia. A fim de obter a melhor configuração do Gnutella para alcançar o melhor compromisso, fizemos um estudo preliminar às simulações descritas na seção 3. Em particular, variamos o número máximo de nós que um *peer* considera como seus vizinhos lógicos, dado que este parâmetro define a quantidade de cópias de requisições. As Figuras 25, 24, e 26, respectivamente, mostram os resultados de taxa de sucesso, energia consumida e atraso. A configuração da simulação foi a mesma descrita na Seção 4.1.

Os resultados mostram que as maiores taxas de sucesso foram alcançadas para 4 e 5 vizinhos. Além disso, para 5 ou mais vizinhos a energia consumida e o atraso cresceram exponencialmente, indicando que a rede começou a saturar. Concluímos então que o melhor compromisso é atingido quando o número de vizinhos máximo é 4.

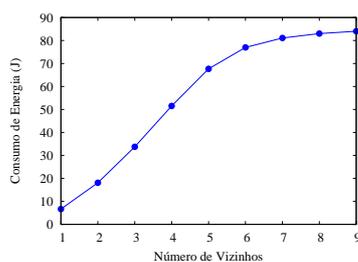


Figura 24: Energia consumida pela rede

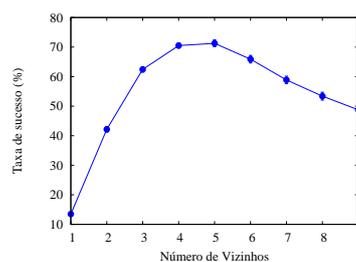


Figura 25: Taxa de sucesso

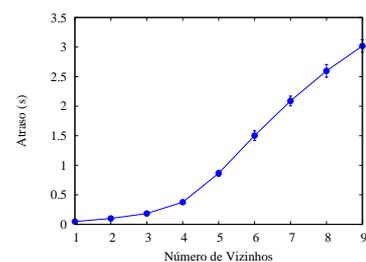


Figura 26: Atraso