

GISANE APARECIDA MICHELON

**CENTRALIDADE PONDERADA POR
POTENCIAL APLICADA À REPLICAÇÃO DE
DADOS EM REDES *AD HOC* MÓVEIS**

Tese apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Doutor em Informática.

CURITIBA

2015

GISANE APARECIDA MICHELON

**CENTRALIDADE PONDERADA POR
POTENCIAL APLICADA À REPLICAÇÃO DE
DADOS EM REDES *AD HOC* MÓVEIS**

Tese apresentada ao Programa de Pós-Graduação em
Informática da Pontifícia Universidade Católica do Paraná
como requisito parcial para obtenção do título de Doutor
em Informática.

Área de Concentração: *Ciência da Computação*

Orientador: Prof. Dr. Luiz Augusto de Paula Lima Jr.

CURITIBA

2015

M623c
2015

Michelon, Gisane Aparecida
Centralidade ponderada por potencial aplicada à replicação de dados em
redes *AD HOC* móveis / Gisane Aparecida Michelon ; orientador, Luiz
Augusto
de Paula Lima Jr. – 2015.
viii, 102 f. : il. ; 30 cm

Tese (doutorado) – Pontifícia Universidade Católica do Paraná,
Curitiba,
2015
Bibliografia: f. 94-102

1. Computação. 2. Sistemas de comunicação móvel. 3. Banco de
dados -
Gerência. I. Lima Júnior, Luiz Augusto de Paula. II. Pontifícia
Universidade

Esta página deve ser reservada à ata de defesa e termo de aprovação que serão fornecidos pela secretaria após a defesa da dissertação e efetuadas as correções solicitadas.

Agradecimentos

A Deus, por iluminar meu caminho e ajudando-me a superar os obstáculos que surgiram no andamento deste trabalho.

Aos meus pais, pelo incentivo em continuar os estudos e conquistar esta etapa.

Ao meu marido, pela ajuda e compreensão, estando sempre presente nos momentos difíceis.

Ao orientador Prof. Dr. Luiz A. P. Lima Jr., pelo acompanhamento no desenvolver do trabalho e ao Prof. Dr. Alcides Calsavara pela ajuda na escrita e na apresentação de artigos.

A Fundação Araucária, pelo auxílio financeiro que contribuiu para a realização deste trabalho.

A todos que de alguma forma contribuíram para a realização e a divulgação deste trabalho.

Sumário

| | |
|---|-------------|
| Agradecimentos | i |
| Sumário | ii |
| Lista de Figuras | iv |
| Lista de Tabelas | v |
| Lista de Abreviaturas | vi |
| Resumo | vii |
| Abstract | viii |
| | |
| Capítulo 1 - Introdução | 1 |
| 1.1. Motivação..... | 4 |
| 1.2. Objetivos..... | 5 |
| 1.3. Organização | 7 |
| | |
| Capítulo 2 - Fundamentação Teórica | 8 |
| 2.1. MANETs | 9 |
| 2.2. Replicação de dados | 11 |
| 2.2.1. Alocação da réplica | 11 |
| 2.2.2. Realocação da réplica | 12 |
| 2.2.3. Atualização da réplica | 12 |
| 2.3. Centralidade em Grafos | 13 |
| 2.3.1. Centralidade de grau | 14 |
| 2.3.1.2. Centralidade de Autovetor..... | 15 |
| 2.3.2. <i>Eigenvector centrality</i> | 16 |
| 2.3.3. Centralidade de intermediação..... | 16 |
| 2.3.4. Centralidade de proximidade | 17 |
| 2.4. Campos magnéticos virtuais e sensibilidade ao contexto | 21 |
| 2.5. Discussão | 23 |
| | |
| Capítulo 3 - Trabalhos Relacionados | 25 |
| 3.1. Algoritmos de replicação Hara | 25 |
| 3.2. <i>Framework</i> SCALAR | 27 |

| | |
|---|-----------|
| 3.3. EARAM-Alocação de réplica baseada em estabilidade da vizinhança..... | 29 |
| 3.4. EVC– <i>Eigenvector centrality</i> (centralidade de autovetor)..... | 31 |
| 3.5. Centralidade de subgrafo..... | 32 |
| 3.6. Discussão | 34 |
| Capítulo 4 – O Método de Replicação de Dados | 36 |
| 4.1. Descrição do método proposto..... | 37 |
| 4.1.1. Alocação da réplica | 37 |
| 4.1.2. Realocação da réplica | 39 |
| 4.1.3. Centralidade ponderada por potencial – Formalização..... | 41 |
| 4.2. Demonstração do método de replicação de dados | 52 |
| 4.2.1. Alocação da réplica | 52 |
| 4.2.2. Realocação da réplica | 54 |
| 4.2.3. Análise do custo de comunicação..... | 56 |
| 4.3. Discussão | 57 |
| Capítulo 5 – Ambiente de Simulação e Resultados Obtidos | 59 |
| 5.1. Ambiente de simulação | 59 |
| 5.2. Parâmetros das simulações | 61 |
| 5.3. Métricas de avaliação..... | 64 |
| 5.4. Avaliação dos resultados obtidos..... | 66 |
| 5.4.1. Análise da viabilidade da centralidade <i>Vol</i> na alocação dos dados..... | 67 |
| 5.4.2. Efeito da variação no número de consultas (requisições)..... | 70 |
| 5.4.3. Efeito da variação do <i>alfa</i> α (Ajuste fino de α)..... | 77 |
| 5.4.4. Efeito do aumento da mobilidade dos nós | 79 |
| 5.4.5. Efeito da acessibilidade dos dados..... | 82 |
| 5.4.6. Efeito do aumento do número de nós – análise da escalabilidade | 84 |
| 5.4.7. Efeito da variação do período para recalculiar o potencial | 86 |
| 5.5. Discussão..... | 87 |
| Conclusão | 90 |
| Referências Bibliográficas | 94 |

Lista de Figuras

| | | |
|-------------|---|----|
| Figura 2.1 | Comunicação direta entre os nós..... | 9 |
| Figura 2.2 | Centralidade de grau..... | 15 |
| Figura 2.3 | Centralidade de proximidade..... | 18 |
| Figura 2.4 | Vizinhança do nó i | 19 |
| Figura 2.5 | Exemplo de campos magnéticos virtuais..... | 22 |
| Figura 3.1 | <i>Backbone</i> virtual..... | 28 |
| Figura 4.1 | Nó requerendo a réplica | 43 |
| Figura 4.2 | Coefficiente $\beta(i)$ | 45 |
| Figura 4.3 | Sobreposição de potenciais..... | 50 |
| Figura 4.4 | Eliminação de uma réplica..... | 51 |
| Figura 4.5 | Grafo de 8 nós e 10 enlaces..... | 52 |
| Figura 4.6 | Alocação de uma réplica através do algoritmo de <i>Vol</i> | 54 |
| Figura 5.1 | Exemplo do arquivo de configuração..... | 60 |
| Figura 5.2 | Exemplo de uma rede no INETMANET..... | 66 |
| Figura 5.3 | Volume de dados (em Kbytes) para acessar a réplica na fase de alocação dos dados..... | 68 |
| Figura 5.4 | Efeito da variação no número de requisições com 2 nós acessando.. | 72 |
| Figura 5.5 | Efeito da variação no número de requisições com vários nós acessando a réplica..... | 74 |
| Figura 5.6 | Área retangular 600m x 100m, com 2 nós em lados opostos acessando a réplica..... | 76 |
| Figura 5.7 | Efeito da variação no número de requisições com 2 nós acessando a réplica em uma área retangular 600m x 100m..... | 77 |
| Figura 5.8 | Efeito da variação do α na centralidade ponderada por potencial..... | 78 |
| Figura 5.9 | Aumento da mobilidade dos nós..... | 80 |
| Figura 5.10 | Acessibilidade dos dados com 20 nós..... | 83 |
| Figura 5.11 | Acessibilidade dos dados com 35 nós na rede..... | 84 |
| Figura 5.12 | Acessibilidade dos dados com 50 nós na rede..... | 85 |
| Figura 5.13 | Comparativo da acessibilidade dos dados com 20 e 50 nós na rede.. | 86 |
| Figura 5.14 | Ajuste fino do Potencial..... | 87 |

Lista de Tabelas

| | | |
|------------|---|----|
| Tabela 3.1 | Resumo das informações que cada método considera..... | 34 |
| Tabela 4.1 | Notações dos principais elementos da fase de realocação dos dados. | 49 |
| Tabela 4.2 | Matriz de adjacência do grafo da Figura 4.3 (<i>Vol</i>)..... | 53 |
| Tabela 4.3 | Cálculo da <i>centralidade baseada em volume localizado (Vol)</i> | 53 |
| Tabela 4.7 | Potencial da réplica 1..... | 55 |

Lista de Abreviaturas

| | |
|------------|--|
| AC | <i>Access counter</i> - contador de acessos |
| AODV | <i>Ad Hoc On-demand Distance Vector Routing</i> – roteamento de vetor de distância <i>ad hoc</i> sob demanda |
| AF | <i>Access Frequency</i> - frequência de acesso |
| CC | <i>Closeness Centrality</i> - centralidade de proximidade |
| CPP | Centralidade Ponderada por Potencial |
| DAFN | <i>Dynamic Access Frequency and Neighborhood</i> - frequência de acesso dinâmica e da vizinhança |
| DC | <i>Degree centrality</i> - centralidade de grau |
| DCG | <i>Dynamic Connectivity based Grouping</i> - conectividade dinâmica baseada em grupo |
| GPS | <i>Global Positioning System</i> – Sistema de posicionamento global |
| IN | Instabilidade do nó |
| MANETs | <i>Mobile Ad hoc Network</i> - redes <i>ad hoc</i> móveis |
| MIDS | <i>Minimum Independent Dominating Set</i> - conjunto dominante independente mínimo |
| MFT | <i>Magnetic Field Table</i> - tabela dos campos magnéticos |
| MU | Unidade magnetizável |
| NP | <i>Non-Deterministic Polynomial time</i> - tempo polinomial não determinístico |
| NS | Número de saltos |
| OLSR | <i>Optimized Link State Routing</i> - roteamento otimizado por estado do enlace |
| OMNET++ | <i>Objective Modular Network Testbed in C++</i> |
| ROWA | <i>Read One - Write All</i> - lê um - escreve todos |
| RWP | <i>Random WayPoint</i> - modelo de mobilidade de percurso aleatório |
| RWR | Read/Write Ratio - razão leitura/escrita |
| SAF | <i>Static Access Frequency</i> - frequência de acesso estática |
| TTL | <i>Time-to-Live</i> - tempo de vida |
| <i>Vol</i> | Centralidade baseada em volume localizado |

Resumo

Este trabalho aborda o problema de replicação de dados em redes *ad hoc* móveis (do inglês, *Mobile Ad hoc Network* - MANETs) visando trazer uma nova solução com o objetivo de tratar dos problemas inerentes a este contexto de forma a melhorar o desempenho no acesso aos dados nestas redes. As MANETs são formadas por nós móveis, que se comunicam sem nenhuma infraestrutura existente. Elas são muito dinâmicas em relação à tecnologia de comunicação, à conectividade, à largura de banda e aos recursos dos dispositivos. Para disponibilizar dados aos usuários em qualquer lugar e a todo tempo, diversas questões precisam ser consideradas e muitos problemas devem ser tratados, particularmente os relacionados à mobilidade e à limitação de recursos computacionais. Estes fatores fazem com que o desenvolvimento de métodos de replicação de dados se torne um grande desafio neste contexto, pois uma série de requisitos, dependentes das aplicações, devem ser considerados de forma transparente e adaptativa. Para tratar desses problemas, um algoritmo distribuído dinâmico foi construído aplicando o conceito de campos magnéticos virtuais para ponderar o cálculo da centralidade de um nó no grafo da rede. O modelo reativo definido, denominado centralidade ponderada por potencial, permite a realocação dos dados influenciada por fatores dinâmicos como a frequência de acesso às réplicas e a instabilidade dos nós, bem como define como se dá o ciclo de vida das réplicas. Assim, em sistemas onde a tolerância a faltas pode ser implementada através da replicação de dados, poderiam ser considerados fatores extremamente dinâmicos (e.g., mobilidade e limitação de recursos) que devem influenciar a decisão de quais nós da rede manterão as réplicas dos dados. O método proposto foi implementado e analisado em relação à abordagens que não consideram fatores dinâmicos. Os resultados são promissores principalmente no que diz respeito à média de volume de dados transmitidos na rede para obter a réplica.

Palavras-Chave: replicação de dados, centralidade em grafos, MANETs, campos magnéticos virtuais, centralidade ponderada por potencial.

Abstract

This work addresses the problem of data replication in Mobile Ad-hoc Networks (MANETs) by proposing a new approach based on replica attraction. MANETs are challenging because they are formed by nodes that move freely typically on a flat area, and they are, therefore, very dynamic in relation to communication technology, connectivity, bandwidth and the devices resources. In order to make data available to the user at any place and any time, issues related to the mobility of nodes and the limitation of computational resources need to be addressed. Moreover, there is a variety of applications-dependent requirements that need to be taken into account in a transparent and adaptive manner. In order to address these problems, a dynamic distributed algorithm was built applying the concept of virtual magnetic fields in the computation of the central node of the network graph. The reactive model defined, called weighted centrality by potential, allows the reallocation of replicas influenced by dynamic factors such as access frequency to replicas and the instability of nodes, without overlooking the graph centrality itself. A replica life-cycle mechanism was also defined in accordance to the potential model. Therefore, in systems where fault tolerance can be implemented by data replication, the use of virtual magnetic fields, allows to take into account extremely dynamic factors (e.g., mobility and limited resources) that should influence the decision of which network nodes should keep replicas of data. The proposed method was implemented and analyzed in comparison with other competing approaches. The results were promising especially regarding the average volume of data transmitted over the network to access the replica.

Keywords: data replication, graph centrality, MANETs, virtual magnetic fields, weighted centrality by potential.

Capítulo 1

Introdução

As redes *ad hoc* móveis (MANETs) são formadas por nós móveis que se comunicam sem nenhuma infraestrutura existente e encaminham pacotes através de múltiplos saltos [LIM09]. Elas são muito dinâmicas em relação às tecnologias de comunicação, a conectividade, a largura de banda e aos recursos dos dispositivos [HAD06]. Assim, permitem a comunicação entre dispositivos onde não é possível a estruturação de uma rede fixa, como por exemplo, em ambientes de difícil acesso ou em operações militares.

Essas redes têm se expandido rapidamente, tornando-se cada vez mais comuns no cotidiano das pessoas. Isso é devido ao crescimento da popularização de dispositivos móveis como *laptops* e *smartphones*, acompanhado da necessidade das pessoas se comunicarem em qualquer lugar e a todo tempo.

Contudo, para disponibilizar essas facilidades, diversas questões devem ser consideradas e muitos problemas devem ser tratados. Problemas como a mobilidade, tornam um desafio o acesso aos dados. Uma informação pode estar disponível em um nó num certo momento e no instante seguinte este nó pode se tornar inacessível. Assim, se o dado torna-se inacessível em um nó, uma réplica dele deverá estar disponível em outro nó.

Outro problema, além da mobilidade, é a limitação de recursos existentes nessas redes, como por exemplo, energia e processamento, e os problemas como heterogeneidade de dispositivos. Assim, a utilização dos recursos desses dispositivos deve ser otimizada e os dados devem ser disponibilizados a fim de satisfazer os requisitos de acordo com o contexto do ambiente. Também, os diferentes requisitos do ambiente requerem uma solução flexível para adaptar tais mudanças, que podem ocorrer muito rapidamente em um ambiente móvel. Uma maneira de lidar com os desafios é a utilização métodos de *replicação de dados*.

Pelo fato deste ambiente ser muito dinâmico e exigir aplicações adaptativas, muitas vezes a entrega de uma mensagem não é destinada para um endereço específico, mas encaminhada a um nó que possua determinadas características de acordo com fatores dependentes da aplicação. Uma forma de tratar isso é através do encaminhamento de mensagens em nível de aplicação, como por exemplo, através de campos magnéticos virtuais [LIM10]. Este conceito, inspirado da Física, permite o encaminhamento de mensagens em redes *overlay* sobre redes físicas, possibilitando a mobilidade dos nós [LIM10].

Ao utilizar campos magnéticos virtuais, a ideia principal é criar ligações dinâmicas modeladas entre componentes computacionais. Assim, mensagens que chegam de um certo nó são encaminhadas para o nó que possuir maior força de atração do campo magnético [LIM10] de acordo com relações de influência magnética preestabelecidas.

Como na alocação inicial dos dados, geralmente não há ainda, informações do contexto, é interessante alocá-lo em um nó de forma a torná-la acessível de modo igual a

todos os nós. Entretanto, por causa da mobilidade e outras questões que tornam a rede dinâmica, o dado precisará, muito provavelmente, ser realocado.

Desta forma, para a realocação dos dados na rede, é interessante disponibilizar os dados de acordo com fatores que possam minimizar o uso de recursos e maximizar a sua disponibilidade na rede.

Este trabalho propõe um novo método de replicação de dados, considerando, que as duas etapas, de alocação e de realocação têm características diferentes e, portanto podem ser tratadas com métodos específicos para cada uma delas. O método definido usa o conceito de campos magnéticos virtuais para o posicionamento das réplicas nos nós mais apropriados e para o ciclo de vida (i.e., criação e destruição) das réplicas.

Para determinar a distribuição inicial das réplicas na rede foi utilizada uma medida de centralidade, conceito vindo da teoria de grafos. Na medição de *centralidade de proximidade*, o nó mais central, que está mais próximo de todos os outros nós é que receberá a réplica. Conseqüentemente, o nó que tem maior proximidade, muito provavelmente, é o que tem o melhor conhecimento da rede e maior poder de disseminação da informação. O objetivo disso é melhorar o desempenho na busca do dado de forma semelhante a todos os nós. Como o número de réplicas a serem alocadas na rede, no modelo proposto, são limitadas, a alocação das réplicas em um nó mais central não sobrecarregará este nó.

Na segunda etapa, de realocação das réplicas, estas serão realocadas considerando informações de contexto, tais como: a instabilidade de um nó e a frequência de acesso à réplica. Para isso, é realizado o encaminhamento de mensagens ao nível da aplicação baseado em campos magnéticos virtuais considerando sensibilidade ao contexto, conceito este foi utilizado para agregar a ponderação na centralidade.

1.1. Motivação

Devido aos inúmeros problemas encontrados em MANETs, a disponibilidade dos dados neste tipo de rede torna-se um grande desafio. A acessibilidade dos dados em ambientes dinâmicos é bem menor se comparada com as redes fixas. Isso ocorre porque os nós são móveis e podem desconectar-se da rede, tornando-os inacessíveis.

Métodos de replicação de dados são usados neste contexto para reduzir estes problemas. O acesso remoto aos dados é fundamental em MANETs, e a sua replicação incrementa sua disponibilidade. Mas, devido à limitação de recursos dos dispositivos móveis, é inviável armazenar uma réplica de cada item de dado em cada nó. Desta forma, uma estratégia de replicação que leve em conta as características dos nós, precisa ser empregada.

Assim, a escolha da localização da réplica tem um grande impacto no tráfego da rede. Uma das primeiras tarefas que um método de replicação de dados precisa implementar é de *alocação dos dados*. Entretanto, o problema de alocação de dados através de uma solução ótima, é NP-completo em redes estáticas [COO02]. Em MANETs, este problema é ainda mais difícil de ser tratado. Muitas vezes, não é viável utilizar algoritmos que produzem soluções exatas, sendo então propostos algoritmos aproximados, que encontram soluções próximas à alocação ótima para um determinado momento [JIN04].

Com a finalidade de tratar a complexidade do problema de alocação de dados e aumentar o desempenho do método, algumas métricas de teoria de grafos podem ser utilizadas para analisar a rede, com o propósito de obter as características e as propriedades estruturais da rede. Uma dessas métricas é a *centralidade de proximidade*. Devido ao fato da réplica estar mais centralizada, o volume de dados trafegando é reduzido, e conseqüentemente a sobrecarga da rede é atenuada.

Após a alocação dos dados, devido à dinamicidade das MANETs, as mudanças no ambiente podem exigir que as réplicas sejam realocadas em outros nós, para que o dado

continue acessível ou mais próximo dos nós que tem maior número de acessos. Para satisfazer os requisitos das aplicações é necessário que os dados sejam providos considerando mudanças de contexto, em um ambiente móvel, de forma transparente e adaptativa. O paradigma de *sensibilidade ao contexto* é uma área da computação móvel onde existem grandes oportunidades de pesquisa e tem evoluído proporcionalmente ao crescimento do uso de dispositivos móveis. Isso é devido ao fato do usuário, mesmo em movimento, ter o desejo de acessar informações em qualquer lugar, a qualquer momento e a partir de qualquer dispositivo [LOU09]. Assim, a replicação de dados em um ambiente móvel pode e deve se beneficiar de informações de contexto.

A ideia é, portanto, na fase de *realocação dos dados*, fazer uso do conceito de campos magnéticos virtuais para quantificar informações de contexto referentes à instabilidade da rede e à frequência de acesso à réplica. Além disso, o encaminhamento de mensagens em campos magnéticos ocorre no nível de aplicação e é independente dos protocolos das camadas inferiores. Desta forma, pretende-se tratar o problema principal de métodos de replicação de dados: maximizar a disponibilidade dos dados na rede minimizando o uso de recursos dos dispositivos e da rede. Essas questões ainda em aberto serão tratadas considerando fatores dinâmicos que não foram explorados em outros trabalhos e usando técnicas ainda não utilizadas.

1.2. Objetivos

O objetivo principal desta tese é o desenvolvimento e a avaliação de um novo método de replicação de dados, usando medidas de centralidade e campos magnéticos virtuais, a fim de tratar de problemas inerentes de MANETs. Este método considera somente o acesso aos dados para leitura, não permitindo escrita dos dados. Sincronização de réplicas está, portanto, fora do escopo deste trabalho.

Como objetivos específicos que o trabalho atingiu podemos mencionar:

- Definir um método para a alocação inicial de réplicas e avaliar o seu impacto na fase seguinte de realocação.
- Implementar o algoritmo de alocação dos dados.
- Desenvolver e implementar o algoritmo para realocação dos dados utilizando campos magnéticos virtuais para determinar a sua distribuição em função de características como frequência de uso e localização.
- Testar campos magnéticos virtuais em redes com mobilidade.
- Avaliar os resultados obtidos com o método de replicação de dados, o qual compreende os algoritmos distribuídos para a alocação e a realocação de dados através de métricas de desempenho predefinidas.

Sendo assim, a proposta deste trabalho foi especificar um novo método de replicação de dados em MANETs usando medidas de centralidade para alocação dos dados e campos magnéticos virtuais para a realocação dos dados. A métrica de análise da centralidade da rede encontra o nó mais central da rede. Assim, este nó tem um maior poder de difusão da informação e um melhor conhecimento da rede em relação aos outros nós. Desta forma, quando se aloca os dados neste nó, por ele ser mais central, diminui a distância em saltos e minimiza os custos de comunicação no acesso aos dados através dos outros nós.

Da mesma forma, a técnica de campos magnéticos virtuais aplicada a MANETs tem o potencial de diminuir os custos de comunicação e melhorar a disponibilidade dos dados. A escolha do local onde o dado é armazenado deve ser projetada dependendo do contexto em que o usuário e o dispositivo se encontram.

Para melhorar a disponibilidade dos dados, o monitoramento da instabilidade do nó melhora a possibilidade que estes sejam alocados em outros nós, antes do nó ficar indisponível. Por exemplo, se o nó estiver com o nível de bateria muito baixo existe uma grande probabilidade que logo o nó se torne indisponível. Assim, o conceito de campos

magnéticos é aplicado e cada nó terá um valor potencial para alocar o dado, de acordo com as informações: frequência de acesso ao dado, instabilidade do nó e distância em saltos. Os nós que mais acessam os dados e os mais estáveis têm um potencial maior e influenciam na decisão da escolha do nó que deverá possuir o dado ou sua réplica.

Apesar de existirem muitos trabalhos relacionados à replicação de dados, destaca-se que nenhum deles utiliza campos magnéticos virtuais, conceito este que inspirou a utilização da ponderação em medidas de centralidade. Neste trabalho foi utilizado o conceito de centralidade para a alocação, e acrescentar o conceito de campos magnéticos virtuais, para a realocação em um ambiente móvel. Esses conceitos permitem tratar das particularidades das duas fases de replicação: a alocação e a realocação, uma vez que elas têm características diferenciadas. Assim, este método de replicação traz flexibilidade e adaptação de acordo com o ambiente onde o sistema se encontra com o objetivo de tratar diferentes requisitos da rede e dos dispositivos. A utilização de campos magnéticos virtuais possibilita distribuir as réplicas com as mudanças ocorridas no ambiente.

1.3. Organização

Esta tese está organizada em 5 capítulos. O Capítulo 2 descreve a fundamentação teórica, abordando as redes *ad hoc* móveis, técnicas de replicação de dados, a métrica de análise da centralidade da rede e o conceito de campos magnéticos virtuais. O Capítulo 3 apresenta os trabalhos que são correlatos a esta tese. O Capítulo 4 descreve o método de replicação de dados. O Capítulo 5 discute os principais resultados obtidos. E por fim, são apresentadas as conclusões.

Capítulo 2

Fundamentação Teórica

Neste capítulo são apresentados os principais conceitos que foram utilizados no desenvolvimento deste trabalho. Na Seção 2.1 são apresentados os fundamentos sobre as redes *ad hoc* móveis, rede que foi utilizada nas simulações deste trabalho. A Seção 2.2 aborda sobre métodos de replicação de dados. A Seção 2.3 descreve a centralidade em grafos, que serve de base para a definição da proposta desta tese: centralidade ponderada por potencial. A Seção 2.4 trata de campos magnéticos virtuais, conceito utilizado para a definição da ponderação na centralidade. E a Seção 2.5 descreve a discussão deste capítulo.

2.1. MANETs

As MANETs são formadas por nós móveis, que se comunicam sem nenhuma infraestrutura existente, e encaminham pacotes através de múltiplos saltos [LIM09]. A Figura 2.1 mostra um exemplo dos nós que se conectam diretamente. Nesse tipo de comunicação, os nós intermediários devem repassar os pacotes e operar como roteadores das mensagens entre os nós [DER09].

Diferentemente das redes sem fio, as MANETs não necessitam de um ponto de acesso para realizar suas operações e seus nós movem-se livremente. Isso permite a comunicação entre dispositivos, onde não é possível a estruturação de uma rede fixa, como por exemplo, em ambientes de difícil acesso, como coordenação de operações militares [HAD06]. Realmente, no início essas redes foram desenvolvidas para serem utilizadas em cenários militares, desastres, incêndio, sensoriamento ambiental, entre outros. Mas, com o crescimento no uso de dispositivos móveis, estão sendo utilizadas também nas atividades diárias, onde as aplicações estão relacionadas a cenários que exigem uma inicialização rápida, tais como, a troca de informações em grupos.



Figura 2.1: Comunicação direta entre os nós [BAR11].

Em MANETs, um dos principais protocolos de roteamento utilizado é o OLSR (*Optimized Link State Routing*) [CLA03] e o AODV (*Ad Hoc On-Demand Distance Vector*) [PER03].

O protocolo OLSR é proativo, ou seja, periodicamente troca informações sobre a topologia da rede com os nós, de modo a atualizar constantemente suas tabelas de roteamento. Esse protocolo gerencia o tráfego de pacotes, de modo organizado, através do caminho mais curto [CLA03]. Ele é baseado no protocolo *Link State*, orientado a tabelas, sendo uma versão otimizada para dispositivos móveis gerando menos tráfego na rede. Ele tem a vantagem de disponibilizar, de imediato, as rotas quando necessário. Assim, cada nó utiliza a informação atualizada para enviar os pacotes e mesmo quando um nó está se deslocando seu pacote pode ser entregue com sucesso. Este protocolo usa nós chamados *Multipoint Relay* (MPR). Os MPRs servem de ponte para retransmitir a informação aos outros nós ao qual estes estão ligados. No OLSR somente os nós MPRs é que são responsáveis pelo controle de tráfego e retransmitem as atualizações, reduzindo a sobrecarga de mensagens que são trafegadas na rede. Cada nó seleciona seus MPRs, sendo que a informação deste nó atinge todos os nós que estejam a 2 saltos do nó em questão [MAG09].

O protocolo AODV é reativo, ele determina as rotas dinamicamente e não requer que os nós mantenham as rotas para todos os destinos [HER11]. Assim, ele tem baixa sobrecarga de mensagens na rede. Se a rota não é conhecida, quando um nó precisa enviar uma mensagem na rede, este envia para toda a rede uma mensagem denominada RREQ, para descobrir a rota. No processo de envio de RREQ um nó intermediário armazena em sua tabela de roteamento o endereço de seus vizinhos estabelecendo assim um caminho reverso. Quando um nó que possua uma rota recente para o destino ou o RREQ atinge o seu destino é enviado para a fonte um pacote de RREP. Cada nó no caminho também atualiza sua tabela com a nova informação da rota [HER11].

Em MANETs, um problema particularmente relevante é a replicação de dados que será descrita a seguir.

2.2. Replicação de Dados

A replicação de dados é uma técnica de gerenciamento de múltiplas réplicas de objetos de dados. Uma réplica é a cópia de um objeto de dados armazenado em um nó da rede. Um objeto é uma unidade mínima de replicação em um sistema, podendo ser um serviço, um dado, ou qualquer outra informação [SAI05]. Assim, considera-se uma réplica, a partir da primeira cópia realizada de um dado.

A técnica de replicação é utilizada com o objetivo de melhorar a disponibilidade dos dados, criando, para isso, a cópia local, ou mais próxima dos nós que possuem uma maior frequência de acesso [KAT12]. Isso também evita a sobrecarga na comunicação (sendo que o número de saltos para acessar a réplica torna-se menor) e a indisponibilidade dos dados devido à dinamicidade da rede [KAW06]. De maneira geral, replicação de dados é baseada em estatísticas de acessos dos nós e a distância em número de saltos [ATS08]. Contudo, na replicação de dados devem ser consideradas algumas questões importantes, que serão descritas a seguir.

2.2.1. Alocação da réplica

A escolha da localização da réplica tem um grande impacto no tráfego da rede. Definir a quantidade de réplicas e onde elas deverão ser colocadas é um importante aspecto em ambientes distribuídos. Um número insuficiente de réplicas pode aumentar o tempo de acesso aos dados quando comparado ao número suficiente de réplicas. Isso porque a réplica pode estar localizada distante do nó que fez a requisição, ou ainda, tornar os dados inacessíveis, por exemplo, devido ao particionamento da rede. Do contrário, muitas réplicas, podem sobrecarregar a rede, pelo fato da demanda de administração das réplicas na rede, ter um número maior de réplicas a serem atualizadas [ATS08]. Assim, o

método de replicação deverá considerar o *trade-off* entre o custo de alocação, de atualização e de acesso as réplicas [DER09].

2.2.2. Realocação da réplica

A realocação da réplica define quando, onde, quem e como a réplica é atribuída a outros nós na rede [OLI10]. Devido ao fato da mobilidade dos nós, não existe uma alocação estática que seja ótima [HAR05]. Desta forma é necessária a realocação da réplica dinamicamente, para melhorar a disponibilidade dos dados. Essa realocação pode ocorrer de forma: proativa, onde os dados são replicados periodicamente, e de forma reativa, onde a decisão de quando replicar os dados é dinâmica, baseada nas mudanças da topologia ou no padrão de acesso aos dados.

2.2.3. Atualização da réplica

A atualização das réplicas e o gerenciamento da consistência são necessários quando os dados sofrem modificações. Os itens de dados devem ser invalidados e as atualizações dos dados devem ser repassadas para outras réplicas existentes na rede.

Neste trabalho de tese é feita a suposição de que o acesso às réplicas é somente para leitura, pois nem sempre os dados exigem atualização. Muitas aplicações exigem somente acesso à leitura, como, por exemplo os cenários onde as pessoas usam seus dispositivos móveis em shopping e em um campus universitário acessando dados que não são atualizados (músicas ou material de aulas) [ATS09], [ATS13] e [HAR06]. Nestes cenários um mesmo nó teria a necessidade de acessar várias vezes a mesma réplica por motivos como: (a) *buffer* de memória com tamanho limitado, (b) o nó pode acessar

“partes” diferentes da mesma réplica e (c) um algoritmo de sincronização de réplicas poderia estar em execução.

2.3. Centralidade em Grafos

Grafos são utilizados para descrever a estrutura topológica de uma rede. Um grafo é a representação matemática constituída por um conjunto de pontos, denominados *nós* ou *vértices*, conectados por linhas que expressam a relação entre eles, denominadas *arestas* [FRE10]. Um grafo G é definido por dois conjuntos V e E , onde V é o conjunto de vértices e $E \subseteq V \times V$ é o conjunto de arestas.

A *centralidade* de um grafo é a medida de quão central um vértice está em relação aos outros vértices do grafo. Em redes de computadores, essa medição determina a importância que um nó tem em uma rede. Desta forma, a medida de centralidade é um conceito importante para a análise de redes [ATS07]. A partir da medida de centralidade em cada nó na rede, os vértices são ordenados em função de sua importância, associada ao significado que cada medida considera ser um vértice mais central que outros [FRE10].

Algoritmos em grafos são muito utilizados em roteamento, agrupamento, entre outras funções em redes de computadores [GUE10]. Eles podem ser usados para determinar a localização para armazenamento dos dados, a fim de alcançar boa distribuição e garantir a melhor cobertura possível da rede. Com isso, diminui-se o número de mensagens trafegando e conseqüentemente a sobrecarga da rede, provendo o balanceamento de carga e a tolerância a falhas. Em MANETs muitos são os trabalhos que utilizam medidas de centralidade. Um trabalho que utiliza a centralidade de intermediação e grau em MANETs [KON14] desenvolveu uma técnica descentralizada para manter a conectividade da rede usando agentes inteligentes e autônomos. Diferentemente de algoritmos onde todos os nós têm a mesma importância, o algoritmo

proposto em [KON14] quantifica a importância de cada nó. Os resultados apresentados demonstraram o efeito do comportamento de tais agentes sobre uma rede de conectividade global. Outros trabalhos podem ser encontrados em [CHE14], [HUN13].

Segundo FREEMAN [FRE79], a centralidade de um nó pode ser determinada por referência a qualquer um dos três atributos estruturais diferentes desse ponto: a seu grau (*Degree Centrality*), a sua intermediação (*Betweenness Centrality*) ou a sua proximidade (*Closeness Centrality*). Cada um desses métodos de cálculo de centralidade e também algumas variações deles serão detalhados nas seções seguintes.

2.3.1. Centralidade de Grau

A centralidade de grau (do inglês, *Degree Centrality* – DC) é a medida de centralidade mais simples e de menor complexidade. O número de conexões que um nó tem diretamente com outros nós em uma rede é denominado grau de um nó, ou seja, é o número de nós adjacentes a ele. O grau de um nó pode variar de 0, caso o nó seja isolado, até número de nós menos 1, quando um nó está conectado com todos os outros nós da rede. A Equação 2.1 define o grau de um nó [FRE79].

$$C_G(p_k) = \sum_{j=1}^n g(p_j, p_k) \quad (2.1)$$

Onde:

$g(p_j, p_k) = 1$, se p_j e p_k são conectados por uma linha; 0, caso contrário.

Em grafos orientados, o grau de um nó é dividido em grau de entrada e grau de saída. A Figura 2.2 mostra um grafo com o nó mais central em relação ao grau. Neste exemplo, o nó 3 tem 4 vértices, sendo portanto o mais central.

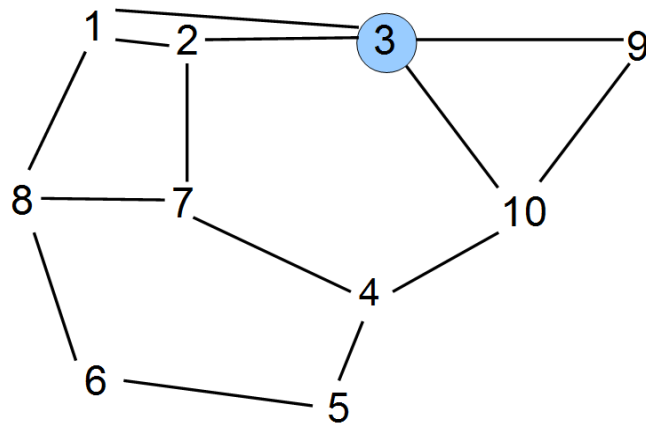


Figura 2.2: Centralidade de grau

2.3.1.2. Centralidade de Autovetor

Bonacich [BON87] propôs uma modificação da medida de centralidade de grau, que tem resultados superiores a medida original. Esta medida é conhecida como medição de autovetor (do inglês, *Eigenvector Centrality* – EC).

A ideia principal de Bonacich é bastante simples [BON87]. A abordagem original da centralidade de grau é que os vértices que têm mais conexões são mais propensos a serem poderosos, porque podem afetar diretamente um número maior de outros vértices. Entretanto, vértices com o mesmo grau não necessariamente são igualmente importantes.

Bonacich afirma que o poder de um ator é uma função de suas conexões e os atores em relação com este. Um ator conectado a muitos outros atores igualmente bem conectados é central, mas menos poderoso que um ator com muitas conexões a outros com poucas conexões, pois estes se tornam dependentes daquele que os conecta [HAN14].

Em [ATS07], foi utilizada a centralidade de autovetor com o objetivo de escolher quais nós da rede armazenarão uma réplica de dados a fim de verificar a escalabilidade de um sistema em MANETs. No capítulo de trabalhos correlatos este trabalho será descrito com mais detalhes.

2.3.2. Centralidade de Intermediação

A centralidade de intermediação (do inglês, *Betweenness Centrality* – BC) é a medida de quão localizado está um nó em relação aos possíveis caminhos da rede. Indica quem é o mais influente em redes sociais [GAN12].

O cálculo de intermediação calcula o número de vezes que um nó age como ponte ao longo do caminho mais curto entre dois nós. Este cálculo é demorado e custoso [BRA01]. Do ponto de vista de cálculo, tanto a centralidade de intermediação como a de proximidade de todos os vértices de um grafo envolvem o cálculo dos caminhos mais curtos entre todos os pares de vértices de um grafo.

A Equação 2.2 define a centralidade de intermediação de um nó [BRA01].

$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \quad (2.2)$$

Onde:

$\sigma_{st}(v)$: é o número de caminhos mais curtos através do nó v .

σ_{st} : é o número total de caminhos mais curtos do nó s para o nó t .

2.3.3. Centralidade de Proximidade

A centralidade de proximidade (do inglês, *Closeness Centrality* – CC) é o número de saltos requeridos, através do caminho mais curto, para um nó acessar todos os outros nós na rede. É uma dentre as várias métricas que analisa a rede através da teoria de grafos com o propósito de obter as características e as propriedades estruturais da rede em relação a centralidade. Os índices de centralidade de proximidade são atribuídos de tal forma que o nó que tem o valor mais alto (ou dependendo da equação, o valor mais baixo) é o nó mais central. Geralmente, esse nó mais central terá melhor conhecimento da rede e maior poder de disseminação da informação em comparação com os outros nós da rede. A centralidade de proximidade é calculada pela Equação 2.3 [SAB66].

$$CC(v) = \frac{1}{\sum_{u \in V} d(v, u)} \quad (2.3)$$

Onde:

$d(v, u)$: distância (caminho mais curto) entre o nó v e o nó u .

A métrica de centralidade de proximidade somente pode ser calculada em grafos conectados. Em grafos não conectados a distância entre os nós pode ser indefinida, já que os nós podem tornar-se inalcançáveis, pelo fato de não existir um caminho entre dois nós.

A Figura 2.3 mostra o grafo de conexão, de acordo com a Equação 2.3, os nós 3, 4 e 7 que tem o maior valor de centralidade de proximidade (CC), sendo qualquer um destes o nó mais central. Na figura foi escolhido o nó 4 para ser o mais central, mas os nós 3 e 7 também poderiam ser mais centrais.

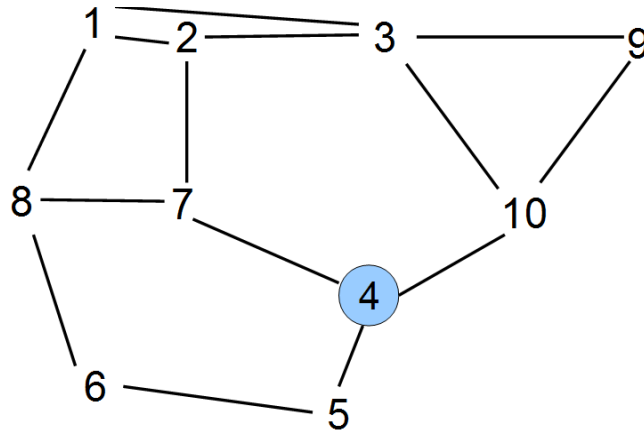


Figura 2.3: Centralidade de proximidade

Existem muitos trabalhos que tratam de centralidade de proximidade através de técnicas tradicionais. Entretanto essas abordagens geralmente exigem conhecimento global da rede e o custo computacional é muito elevado. Algumas técnicas calculam aproximações para o problema, minimizando o custo, e executando de modo distribuído, como é o caso do algoritmo de *centralidade baseada em volume localizado* (*Vol*) [WEH12]. Este trabalho usa uma aproximação para o problema de forma distribuída, requerendo somente o conhecimento do grau dos seus vizinhos que estão a uma distância predefinida em número de saltos. Assim não há exigência de conhecimento global da rede. O volume de um nó i é definido como a soma de todos os graus dos nós vizinhos (a h saltos) do nó i , inclusive o nó i , conforme a Equação 2.4 [WEH13]. Se $h = 0$, somente é calculado o grau do nó i , e então, essa métrica se iguala a centralidade de grau. A Figura 2.4 mostra uma ilustração da vizinhança do nó i , com $h=0$, $h=1$ e $h=2$.

$$Vol(H_h^i) = \sum_{j \in H_h^i} dg_j \quad (2.4)$$

Onde:

- h : é o número de h -saltos.
- H_h^i : é a h -vizinhança do nó i (inclusive o nó i).
- dg_j : é o grau do nó j .

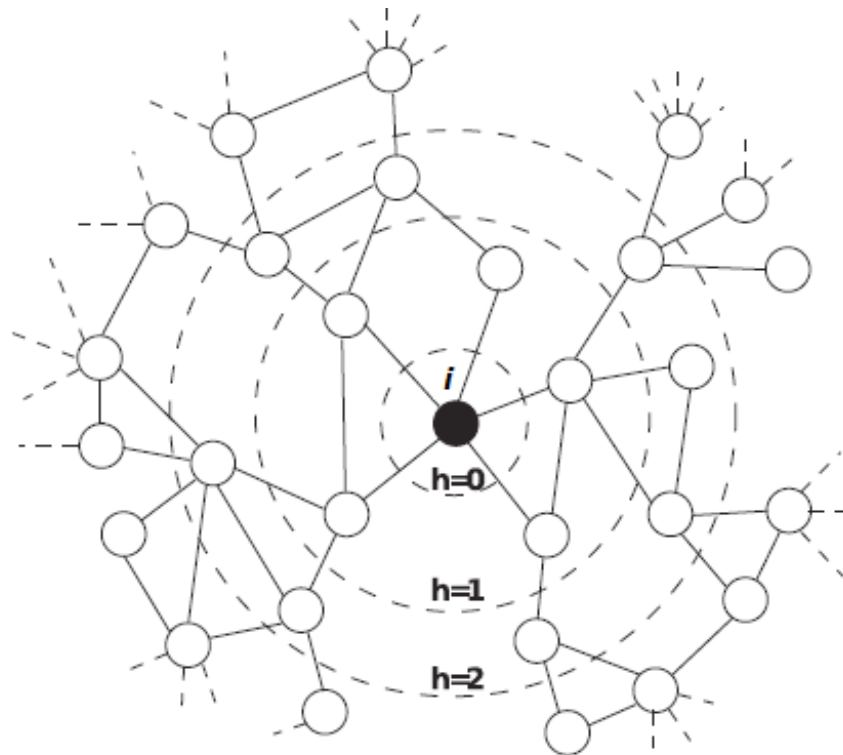


Figura 2.4: Vizinhança do nó i [WEH12].

Para o cálculo da vizinhança, o nó i envia uma mensagem com $TTL = h$ para todos os seus vizinhos com seu grau e uma identidade única para evitar mensagens repetidas. Ao receber a mensagem, cada nó verifica a identidade para checar se a mensagem já foi recebida. Caso contrário ele armazena as informações sobre o grau, para

depois calcular seu valor de volume, decrementa o TTL e retransmite a mensagem a seus vizinhos, se o TTL é diferente de zero. Se $TTL = 0$, não retransmite a mensagem. Este cálculo executa de forma paralela em cada nó e, no tempo definido por h , todos os nós conhecem sua h -vizinhança e seus graus.

Os resultados da *centralidade baseada em volume localizado* foram comparados com técnicas tradicionais de centralidade de proximidade. O *Vol* teve resultados muito semelhantes, mas com um custo computacional muito menor. Além disso, técnicas tradicionais de centralidade de proximidade não são facilmente paralelizáveis [WEH12].

A proximidade dos resultados de *Vol* com os resultados das técnicas tradicionais depende do raio da rede não ser muito maior que o valor definido em h . Conforme apresentado em [WEH12], a correlação usando *Vol* e centralidade de proximidade pode chegar a 0,99, em uma rede com 1000 nós.

Em relação a complexidade da mensagem, para $h=2$, a complexidade é de $n * dg_{avg}^2$, onde n é o número de nós na rede e dg_{avg} é o grau médio da rede [WEH12].

Essa complexidade é válida para o caso do valor de $h=2$, ou seja, no caso onde as mensagens são enviadas para todos os vizinhos de um nó e estes reenviam para seus vizinhos. Em relação à complexidade de tempo, esta é esperada para ser $O(1)$ passos, uma vez que h é escolhido.

O algoritmo RAND proposto em [OKA08] calcula a centralidade de proximidade combinando soluções exatas e aproximadas. Ele encontra os k vértices que resultam em um menor valor. Neste algoritmo, é gerado um *ranking* dos k nós que tiverem o menor valor, e então estes, serão os nós mais centrais. O valor de k é um parâmetro de entrada que determina o número de valores mais baixos de centralidade de proximidade que serão calculados. Eles utilizaram aproximações para encontrar os k vértices e depois uma solução exata para calcular a distância média para todos os k vértices. RAND calcula a

centralidade de proximidade de um vértice, através da média das distâncias pelo caminho mais curto para o vértice amostra, em vez de calcular para todos os vértices.

Das três medidas descritas por FREEMAN [FRE79] pode-se fazer a seguinte comparação: o grau é uma medida da influência direta que um vértice tem em relação a seus vizinhos diretos, a proximidade diz respeito com o tempo que uma informação leva para chegar a todos os vértices na rede, e a intermediação de um vértice está relacionada ao controle da comunicação entre os demais pares de vértices da rede [FRE10].

2.4. Campos Magnéticos Virtuais e Sensibilidade ao Contexto

O conceito de campos magnéticos virtuais foi extraído do conceito de campos magnéticos da Física. Uma mensagem, ao chegar a um nó, será atraída para o nó que exerce uma força maior de campo magnético sobre este nó. Cada nó tem um valor potencial associado a ele que representa a força de atração que tem este nó. Um cenário de campos magnéticos será descrito a seguir. Se uma mensagem chega ao nó A (neste exemplo a mensagem do tipo $c - m_c$), de acordo com a Figura 2.5, o nó B e C exercem influência sobre o nó A, e o nó D exerce influência sobre o nó C. Assim, a mensagem será encaminhada para o nó D (unidade magnetizável – MU) que tem mais força de atração ($\sigma = 50$, σ indica a força do campo magnético) [LIM10].

Como a força de atração pode ser alterada a qualquer tempo, cada nó possui uma tabela de nós e suas forças, cujos campos magnéticos de uma determinada classe é afetado, chamada Tabela de Campos Magnéticos (*Magnetic Field Table- MFT*). Assim, este conceito pode ser utilizado para refletir as mudanças do ambiente.

A sensibilidade ao contexto permite obter informações em um ambiente dinâmico para adicionar valor aos serviços, buscando otimizar as aplicações. Informações chamadas de contexto servem como entrada para o sistema caracterizar a situação de uma entidade, perceber mudanças ocorridas e, a partir delas, tomar decisões mais inteligentes e adaptáveis às necessidades reais do usuário da computação móvel em suas atividades diárias. Uma entidade pode ser uma pessoa (individual ou grupo), ambiente (sala, prédio, etc.) ou coisas (objetos físicos, computadores, etc.) que seja considerada relevante para a interação usuário-aplicação [DEL10], [BAL07]. Informações de contexto, por exemplo podem ser capturadas através das atividades dos usuários, recursos e infraestrutura disponíveis [CHE00]. Assim, o conceito de sensibilidade ao contexto pode ser utilizado para tratar com a dinamicidade de uma MANET. No método desenvolvido neste tese, informações sobre a instabilidade dos nós e a frequência de acessos às réplicas serão consideradas informações de contexto.

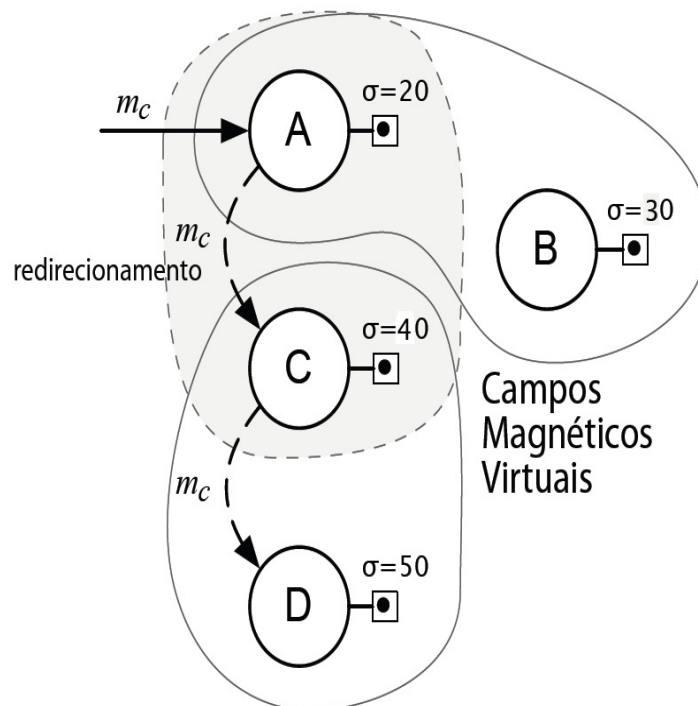


Figura 2.5: Exemplo de campos magnéticos virtuais [LIM10]

O método de Campos Magnéticos Virtuais foi avaliado, através de simulações, em relação ao número de mensagens que foram enviadas para nós errados, ou seja, que não tinham a maior força de campo magnético. Após o período de estabilização do algoritmo, que ficou em 3.25ms para uma rede de 100 nós, em uma área de 1.500m x 500m, nenhuma mensagem foi enviada para os nós errados [LIM10]. Os resultados das simulações também mostraram que a proposta é eficaz, no que diz respeito ao custo de armazenamento, processamento e comunicação.

Apesar do comportamento do sistema mudar de acordo com os requerimentos da aplicação, esse método ainda não foi avaliado em redes móveis, sendo somente simulado em redes estáticas, mesmo tendo sido inicialmente concebido para tal.

2.5. Discussão

Neste capítulo foram apresentados os conceitos e as tecnologias que foram utilizadas para o desenvolvimento deste trabalho. O problema a se resolver é a de alocação e a realocação dinâmica das réplicas. O contexto são as redes móveis conectadas (MANETs). Em MANETs, ocorre mobilidade, mas sempre existe um caminho até o destinatário [HAR08]. O método envolverá o cálculo dinâmico da centralidade (para determinar os nós mais propensos a receberem as réplicas) aplicando-se o paradigma de Campos Magnéticos Virtuais.

Existem diversos trabalhos na literatura, utilizados para o cálculo da centralidade em uma rede. Muitos deles resultam em soluções exatas, mas sob a penalidade de uma maior complexidade no algoritmo, elevando os custos com o seu processamento. Outros, no entanto, procuram técnicas que tornem o algoritmo mais rápido, mas que resultam em soluções aproximadas e mais eficientes em relação ao custo computacional. E ainda

existem os trabalhos que calculam a centralidade combinando soluções exatas e aproximadas, com o objetivo de diminuir o custo computacional.

Em MANETs, como a rede é dinâmica e os recursos escassos, um algoritmo deve ser eficiente e ao mesmo tempo viável, com o objetivo de não afetar o desempenho dos nós e da rede. Nem sempre um algoritmo com soluções exatas é necessário, visto que as mudanças nesta rede podem exigir que o cálculo seja executado várias vezes ao longo do tempo.

O próximo capítulo aborda os trabalhos existentes na literatura que mais se relacionam a esta proposta de tese, com suas vantagens e desvantagens.

Capítulo 3

Trabalhos Relacionados

Neste capítulo são descritos os trabalhos relacionados a esta tese. São considerados os trabalhos de replicação de dados em MANETs, os quais serviram de base para o estudo e o desenvolvimento do novo método. A Seção 3.1 aborda os algoritmos de Hara, a Seção 3.2 descreve o *Framework* SCALAR. A Seção 3.3 o método EARAM_SN, a Seção 3.4 EVC–*Eigenvector Centrality* e a Seção 3.5, a Centralidade de Subgrafo. A Seção 3.6 conclui este capítulo.

3.1. Algoritmos de replicação de dados Hara

Um dos primeiros trabalhos abordando replicação de dados com o objetivo de melhorar a acessibilidade dos dados em MANETs foi desenvolvido por Hara [HAR06].

Este método considera a frequência de acesso dos nós para cada item de dado e o estado da conexão da rede. Na alocação da réplica não existe um servidor central que é responsável pelas réplicas, os nós móveis determinam de forma autônoma e dinâmica a alocação [BLE10]. Hara desenvolveu três principais técnicas, as quais ficaram conhecidas como modelo Hara:

1- SAF (*Static Access Frequency*): é uma técnica que considera apenas as frequências de acessos aos itens de dados do nó, assim, os nós não necessitam trocar informações e o tráfego de mensagens na rede é baixo. Apesar disso, pode ocorrer o problema de dois nós próximos possuírem a mesma réplica. Neste método as réplicas são alocadas em ordem decrescente em relação à frequência de acesso, e de acordo com o espaço de memória disponível. Para tratar o problema de réplicas duplicadas, foi desenvolvido o segundo método, descrito a seguir.

2- DAFN (*Dynamic Access Frequency and Neighborhood*): esse método considera a frequência de acessos aos itens de dados do nó e também da vizinhança. No entanto, quando há uma duplicação de réplica entre quaisquer nós conectados, é escolhida uma das réplicas aleatoriamente para ser descartada.

3- DCG (*Dynamic Connectivity based Grouping*): é um método que leva em consideração a frequência de acessos aos itens de dados e a topologia da rede. O DCG agrupa os nós móveis que são componentes biconectados em uma rede. Agrupando os nós como componentes biconectados o grupo não fica dividido mesmo se um nó se desconecta da rede. Desta forma, os grupos são considerados estáveis e eles não se particionam com facilidade devido a mudanças na topologia da rede. Eles compartilham as réplicas e são coordenados por um dos nós membros. O coordenador do grupo é responsável por determinar a alocação de réplicas baseado nas frequências de acessos obtidas dos seus membros, e então a réplica é alocada no nó que tem maior número de acessos para o item de dados correspondente.

A realocação de dados é proativa, sendo que o período para realocação é fixo. Assim ele não considera as mudanças no ambiente para realizar a realocação e pode

tornar-se não adequado para redes com mobilidade. Além disso, isso gera outro problema: de definir o período de realocação. Se este for muito grande, pode prejudicar o acesso aos dados. Do contrário, o tráfego de dados pode ser muito elevado.

Posteriormente, com o objetivo de tratar essas questões, foram propostas algumas extensões. Uma delas foi desenvolvida para suportar realocação reativa dos dados de acordo com a razão leitura/escrita. Quando essa razão é alta, até um limite pré-definido, significa que existem mais operações de leitura para esses dados, portanto é interessante possuir mais cópias, para o processo de leitura ser mais ágil. Entretanto, se a razão é baixa, então é necessário realizar menos réplicas.

3.2. Framework SCALAR

Em [ATS13], foi desenvolvido um *framework* para replicação de dados distribuídos chamado SCALAR. Nesse *framework*, os nós da rede formam um *backbone*¹ virtual através de um *conjunto dominante conectado* baseado no grafo de topologia da rede. O objetivo disso é melhorar a acessibilidade dos dados e diminuir a sobrecarga de comunicação da rede.

Diferentemente de muitas técnicas existentes na literatura, SCALAR é uma solução completa para MANETs de média (aproximadamente 20 nós) e larga escala (acima de 400 nós). Por ser uma solução completa, o *framework* de replicação não depende das camadas inferiores da rede para a procura e a replicação dos dados.

O modelo do *framework* SCALAR é composto de três partes: o *backbone* virtual, o protocolo de busca de dados escalável e o esquema de replicação reativo. A primeira etapa é a construção do *backbone* virtual, onde a ideia é minimizar o número de nós envolvidos na busca dos dados e com isso diminuir o custo de acesso aos dados através da construção do *conjunto dominante conectado*, que é o menor subconjunto de vértices

¹ *Backbone*: a tradução para português significa espinha dorsal, são as ligações principais de um sistema amplo.

conectados de um grafo, sendo que cada nó que não pertence a este subconjunto, seja adjacente a, no mínimo, um vértice do subconjunto. Assim, cada nó é um nó dominante (*backbone*), ou está conectado diretamente com, ao menos, um nó dominante. Os nós dominantes são os responsáveis pela busca dos dados.

Na segunda etapa, foi desenvolvido um protocolo de busca de dados para localizar um item de dados, em vez de utilizar o *flooding*². Assim, no *framework* SCALAR, a pesquisa e o roteamento do item de dados para o nó requisitante, envolve uma quantidade menor de nós. Isso traz uma grande vantagem em redes com muitos nós, pois somente os nós *backbone* participam do encaminhamento de mensagens. A Figura 3.1 mostra a pesquisa de um item de dados. O nó 9 solicita um dado e para isso seleciona aleatoriamente na lista de nós vizinhos por um nó *backbone*. Após selecionar o nó *backbone* (neste caso o nó 3) encaminha a solicitação para este nó. Como o nó 3 não tem o item de dado ele encaminha a solicitação para um conjunto aleatório de nós *backbones* que sejam vizinhos: o nó 4 e o nó 7, fazendo isso até o dado ser encontrado.

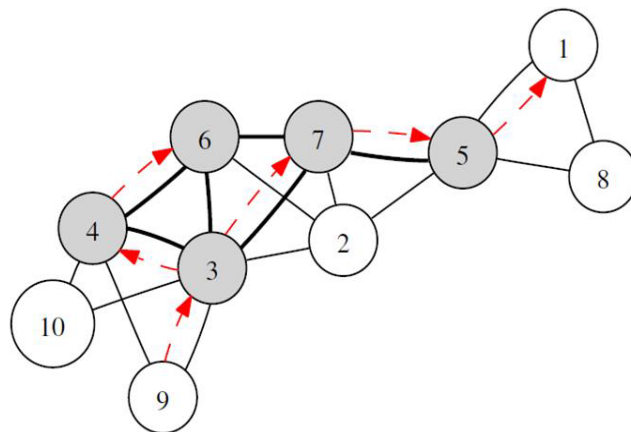


Figura 3.1: *Backbone* virtual [ATS13]

² *Flooding*: também chamado de inundação, um nó repassa a mensagem a todos os nós que estão conectados, exceto para o nó do qual ele recebeu a mensagem.

A última etapa compreende o método reativo de replicação de dados. Nesse método, a decisão de replicação dos dados é tomada depois de receber o item de dados, considerando a frequência de acesso e a distância em número de saltos de um item.

O desempenho do *framework* SCALAR foi simulado em cenários que diferem principalmente na densidade dos nós da rede. O modelo de mobilidade utilizado foi o percurso aleatório (do inglês, *Random WayPoint* – RWP) com velocidade de até 3m/s. Foram avaliados a sobrecarga que o protocolo acrescenta, a acessibilidade dos dados, entre outros fatores importantes para MANETs. Eles comparam o desempenho de SCALAR com outras duas técnicas bem conhecidas na literatura [HAR06], o SAF (*Static Access Frequency*) e DAFN (*Dynamic Access Frequency and Neighborhood*). O *framework* SCALAR executa melhor do que as outras técnicas em relação a acessibilidade dos dados, sobrecarga de mensagens e escalabilidade da rede, principalmente para redes com muitos nós. Mesmo assim, uma das desvantagens deste método é que a decisão para replicar os dados é baseada somente na frequência de acesso e a distância para o dado. SCALAR não considera a instabilidade dos nós, se, por exemplo, a velocidade de um nó é alta ou a bateria do dispositivo está baixa, ou ainda este nó tem capacidade de memória e processamento limitada, mesmo assim o nó poderá armazenar a réplica e fazer parte do *backbone*, o que é ainda pior.

3.3. EARAM_SN - Alocação de réplica baseada em estabilidade da vizinhança

Neste trabalho é proposto um algoritmo distribuído de alocação de réplicas adaptativo com o objetivo de alocar uma réplica próxima da solução ótima. Em redes fixas, a alocação da réplica depende do padrão de leitura/escrita, mas em MANETs depende não somente disso, mas também da mobilidade dos nós. Este método de

alocação de réplicas considera a frequência de acesso de leitura/escrita e a topologia da rede, a fim de minimizar os custos de acesso [JIN04].

Para obter as mudanças na topologia da rede eles sugerem o uso de GPS (*Global Positioning System*) ou um protocolo de roteamento que é capaz de analisar a topologia da rede, como por exemplo, o TORA (*Temporally-Ordered Routing Algorithm*) [PAR97].

As estatísticas de frequência de acesso são coletadas somente de vizinhos estáveis. A estabilidade da vizinhança é definida pela distância entre dois vizinhos (pode ser medida por GPS), medida em um intervalo de tempo. Desta forma, a oscilação de alocação de réplicas é reduzida quando a mobilidade dos nós é alta.

Para manter a consistência das réplicas é utilizada a política ROWA (*Read One - Write All*) onde a leitura é efetuada em um nó, mas as escritas são encaminhadas a todos os nós. O algoritmo é executado para cada nó que possui uma réplica e periodicamente em um tempo t , o qual é um parâmetro uniforme no sistema. O período é mais curto quando as mudanças de topologia da rede e o padrão leitura/escrita são mais frequentes.

A métrica de custo de comunicação dos acessos aos dados utilizada neste trabalho é o número de saltos. Em MANETs, o custo de comunicação entre dois nós está relacionado por exemplo à largura de banda, ao consumo de energia e ao atraso de comunicação. Estes fatores se relacionam ao número de saltos, sendo então utilizado como métrica de custo de comunicação.

Este método foi simulado em uma rede com área de 1000m x1000m, composta de 100 nós móveis, uma velocidade de 0-10m/s e o raio de alcance dos nós é de 200m. O modelo de mobilidade é *Random Waypoint* e cada experimento é executado 10 vezes para obter uma média dos valores.

ERAM_SN foi comparado com um algoritmo de alocação de réplicas estáticas, ou seja, quando as réplicas distribuídas sobre a rede não mudam de nó durante todo o tempo de simulação. Os resultados mostraram que o custo de acesso à réplica é reduzido quando utilizado ERAM_SN.

3.4. EVC – *EigenVector Centrality* (centralidade de autovetor)

Neste trabalho [ATS07], os autores utilizaram a centralidade de autovetor de Bonacich [BON87] (*Eigenvector Centrality*), já abordada no capítulo anterior, com o objetivo de determinar nós para armazenar a réplica. Medida de centralidade é um conceito importante para a análise de redes, ela aponta a importância de um nó na rede. Diferentemente da centralidade de grau, EVC não assume que cada conexão para um nó é igualmente importante, a ideia é que os nós mais importantes devem ter maior influência para o cálculo da centralidade. EVC é uma medição que indica a habilidade de um nó em disseminar dados em uma rede. A centralidade de um nó depende do número de conexões e da qualidade da conexão (peso de um enlace) em relação a frequência de conexões com outros nós. Um exemplo de uso de EVC é o *Google's Page Rank* [BRI98].

As simulações foram realizadas no ambiente JIST (*Java in Simulation Time*), com o protocolo de roteamento AODV. A área de simulação foi de 500 x 500m, com 20 nós em um raio de alcance dos nós de 100m. O modelo de mobilidade é *Random WayPoint* e a máxima velocidade de deslocamento dos nós 3m/s. As simulações foram realizadas em um tempo de 200s.

Os resultados foram favoráveis, colocando a réplica usando EVC em comparação com um nó aleatório, EVC teve resultados melhores em relação a acessibilidade dos dados. A taxa de acessibilidade foi medida pelo número de acessos requeridos com sucesso, dividido pelo número total de acessos requeridos. Desta forma, os resultados mostraram que, encontrando o nó mais central com EVC, pode-se apontar o nó que seja melhor para armazenar a réplica.

Também foi analisado como o método comportava-se em relação a escalabilidade, aumentando o número de nós para 100, 200 e 400 nós na rede. No entanto, quando analisando os resultados deste método em relação a escalabilidade do sistema, este não teve resultados positivos. Quando foi aumentado o número de nós na rede, utilizando EVC não melhorou a acessibilidade dos dados. Foi observado nos resultados iniciais que,

quando escalando, o tamanho do sistema tem um efeito negativo na acessibilidade dos dados. Assim, para manter o desempenho do método de replicação de dados quando o número de nós aumenta, mecanismos adicionais são necessários.

3.5. Centralidade de Subgrafo

Em [PUS14], foi desenvolvido um método para predição da mobilidade da réplica onde os dados podem ser acessados por qualquer nó em um número de saltos mínimo. Para isso, foi utilizado o conceito de conjunto dominante mínimo e centralidade de subgrafo, para decidir o número de réplicas existentes no sistema. A construção de um conjunto dominante mínimo garante que os nós acessem os dados a um salto de distância. A complexidade do algoritmo de conjunto dominante mínimo é NP-*hard*, entretanto foi utilizado um algoritmo aproximado com complexidade NP-completo. Subgrafo é uma medida de centralidade que analisa a rede e determina a importância que um nó tem no grafo. Medidas de centralidade ajudam a selecionar pontos de replicação que estão distribuídos na rede para melhorar a acessibilidade dos dados. A acessibilidade dos dados aumenta porque são selecionados nós na rede que podem cobrir um número maior de outros nós em um menor número de saltos. Neste trabalho são feitas várias suposições, entre elas duas são importantes:

- 1- Os nós são idênticos e não são feitas limitações na capacidade de hardware e de software.
- 2- A topologia de rede não é um grafo completo.
- 3- Não há atualização dos dados, pois os mesmos não sofrem modificações.

Um nó na rede é denominado como servidor se ele armazena uma réplica de um dado. Caso contrário ele é chamado como cliente.

Primeiramente é calculado o conjunto dominante mínimo. Os nós denominados como dominantes serão os pontos de replicação, onde as réplicas serão alocadas. Estes

nós enviam uma mensagem para seus vizinhos adjacentes com sua identificação e identificação da réplica. Os nós adjacentes aos nós dominantes armazenam em uma tabela estas informações e acessam a réplica deste servidor quando necessário. Se a mobilidade do servidor é percebida, então é calculada a centralidade de subgrafo para cada nó na região do servidor para realocar a réplica. Para calcular a centralidade de subgrafo foi construída a matriz de estabilidade que representa a estabilidade de cada nó. O nó com máxima centralidade de subgrafo será considerado mais estável e recebe a réplica.

Os resultados das simulações mostraram que o tempo de resposta e o atraso no acesso aos dados foi reduzido, entretanto os dois algoritmos usados neste trabalho são de complexidade considerável. A complexidade de tempo para calcular o conjunto dominante mínimo é de $O(2^n)$ e do algoritmo de subgrafo é de (n^3) . O ideal é utilizar algoritmos com complexidades lineares, já que estes delimitam o tamanho da saída em relação ao tamanho da entrada dos dados. Algoritmos de ordem de complexidade exponencial geralmente não são úteis sob o ponto de vista prático [CUN12].

Outra desvantagem é que este método não leva em consideração a frequência de acesso aos dados, e este parâmetro é um dos mais considerados em qualquer método de replicação de dados.

A Tabela 3.1 apresenta um resumo dos trabalhos correlatos com as informações que cada um considera.

Tabela 3.1: Resumo das informações que cada método considera

| Método | Considera | Não considera |
|--|--|--|
| Hara | Acessos | Instabilidade Número de saltos |
| SCALAR | Acessos Distância para o dado | Instabilidade |
| EARAM | Acessos Instabilidade (somente em relação à mobilidade) | Número de saltos |
| EVC - Centralidade de AutoVetor | Centralidade (em relação à conectividade) | Acessos Número de saltos |
| Centralidade de Subgrafo | Centralidade (em relação ao número de saltos) Instabilidade (somente em relação à mobilidade) | Acessos Complexidade dos algoritmos |

3.6. Discussão

Neste capítulo foram apresentados os trabalhos que se relacionam a esta tese. A etapa de estudo do material bibliográfico foi importante, pois muitos fundamentos foram pesquisados até encontrar a solução para o domínio do problema encontrado.

Dos estudos realizados a partir dos trabalhos citados nesta tese, percebeu-se que a realocação de dados de forma proativa, executada em períodos predefinidos, poderia ser inadequada, considerando que o ambiente em uma MANET é dinâmico. Um método reativo de realocação seria mais eficiente, pois adaptam-se às mudanças ocorridas no ambiente.

Outra característica dos métodos de replicação de dados aqui apresentados, é que nenhum deles considera o conjunto de informações dinâmicas, como proposto nesta tese, para a escolha da realocação da réplica. A maioria deles considera duas dentre as três informações aqui citadas. Por exemplo, o método de replicação de dados SCALAR [ATS13] não considera instabilidade dos nós. Em [JIN04], EARAM_SN considera estabilidade, mas somente em relação a mobilidade dos nós. E a centralidade de Autovetor - EVC [ATS07] não considera nem a frequência de acesso nem a instabilidade dos nós. O método desenvolvido nesta tese considera as três informações: acessos, instabilidade em relação a mobilidade, quantidade de recursos (bateria, memória, processamento, etc.) e número de saltos medido através da centralidade.

Além disso, é importante levar em consideração o custo dos algoritmos para as decisões de onde alocar as réplicas. Como em MANETs os recursos são escassos, o número de mensagens ou o seu tamanho, devem ser avaliados, a fim de tornar o método viável em relação ao seu custo-benefício. Muitos dos trabalhos aqui descritos, consideram o aumento da acessibilidade dos dados com a aplicação do seu método mas não considera a complexidade e o custo do algoritmo. Neste trabalho foi levado em consideração a complexidade e o custo dos algoritmos. Também foi adicionado o tamanho das mensagens (volume de dados) gastas com o cálculo dos algoritmos.

No capítulo seguinte, é descrito o método de replicação de dados, com as etapas do desenvolvimento e os algoritmos utilizados.

Capítulo 4

O Método de Replicação de Dados

Neste capítulo é apresentado o método de replicação de dados desenvolvido nesta tese, abordando basicamente a descrição e o funcionamento da alocação e da realocação de dados em MANETs. A Seção 4.1 trata das características do método proposto, primeiramente descrevendo a alocação das réplicas e posteriormente a realocação da réplica. A Seção 4.2 mostra um exemplo de um cenário de aplicação do método de replicação de dados. E a Seção 4.3 apresenta a discussão deste capítulo.

4.1. Descrição do Método Proposto

O método de replicação de dados proposto é distribuído, visto que uma arquitetura com algum grau de centralização pode não se adaptar à mobilidade das MANETs. Desta forma, não existe a distinção entre réplicas primárias e secundárias. Todas as réplicas são tratadas da mesma maneira. O método é dividido em duas fases principais: a fase de *alocação* e a fase de *realocação reativa*. A fase de alocação corresponde à distribuição inicial da réplica na rede. Esta fase pode afetar a fase de realocação dinâmica que pode eventualmente reposicionar a réplica para aproximá-la de nós com altos índices de acesso ou mesmo criar réplicas. As fases estão detalhadas nas Seções 4.1.1 e 4.1.2 a seguir. A sincronização de réplicas está fora do escopo deste trabalho que considera que os dados são somente para leitura.

4.1.1. Alocação da réplica

Na alocação inicial da réplica, considerando comunicação bidirecional entre cada par de nós conectados, e que não existe nenhuma informação sobre os acessos às réplicas e sobre a instabilidade dos nós na rede, foram realizados estudos teóricos das seguintes técnicas para a determinação de qual nó receberia a réplica: (a) centralidade de grau, (b) centralidade de proximidade e (c) centralidade de intermediação já descritas no Capítulo 2.

No método CPP foi utilizada a centralidade de proximidade porque ela considera o número de saltos requeridos para um nó acessar todos os outros nós na rede. Sendo esta uma das informações avaliadas para a escolha da localização da réplica e que atende os objetivos deste trabalho. Também pelo fato que foi encontrado um algoritmo para uma aproximação da solução exata com custo computacional menor. Apesar da complexidade

dos algoritmos de centralidade, o cálculo da centralidade de proximidade do grafo inteiro somente precisará ser executado em casos específicos: (1) quando a réplica é alocada pela primeira vez; (2) quando houver mudança abrupta na topologia da rede. Tipicamente a topologia muda gradativamente, então pode-se utilizar alternativas para que o cálculo da centralidade não sobrecarregue a rede com mensagens. Este cálculo poderá ser otimizado, por exemplo, enviando as informações com outras mensagens do algoritmo de centralidade ponderada por potencial, ou seja, vai de “carona”, não gerando tanta sobrecarga na rede com mensagens como ocorre no início, na fase de alocação dos dados.

O algoritmo de *centralidade de proximidade* aloca a réplica no nó mais central. Como quando ao inicializar a rede não estão disponíveis as informações sobre os acessos às réplicas e outros parâmetros dos nós e da rede, não é possível alocar as réplicas considerando, por exemplo, o número de acessos e a mobilidade. Assim, alocando uma réplica no nó mais central, ela estará no nó que se encontra mais próximo de todos os outros nós e que tem melhor posicionamento para disponibilizar a réplica entre os outros nós, com relação ao número de saltos. Essa métrica avalia a capacidade que cada nó tem de tornar a réplica acessível para todos os outros nós. Este nó central tem um poder de disseminação maior e um melhor conhecimento da rede em comparação com outros nós. Alocando a réplica neste nó, pode-se diminuir o número de saltos quando os outros nós precisam acessar a réplica. Além disso, como o número de réplicas para a alocação inicial será pequeno, este nó não será sobrecarregado com o armazenamento de muitas réplicas.

O algoritmo de centralidade de proximidade possibilita a escolha de alocar uma ou mais réplicas na rede, de acordo com a necessidade, dado que ele permite efetuar um *ranking* dos nós que têm maior centralidade de proximidade. Mas, se alocar mais que uma réplica, as mesmas serão alocadas muito próximas uma das outras, na região central do grafo. Assim, optou-se por alocar uma réplica inicialmente, e na sequência fazer cópias das réplicas de acordo com a necessidade.

Neste trabalho, para a alocação de uma réplica, foi utilizada uma aproximação para o cálculo de centralidade de proximidade, o qual foi chamada de *centralidade*

baseada em volume localizado (Vol) [WEH12]. Ela mostrou-se mais eficiente em relação ao tempo de execução em comparação com o algoritmo de centralidade de proximidade tradicional, existente na literatura, como já abordado no Capítulo 3. Esse algoritmo utiliza uma aproximação para a solução do problema.

Nas simulações realizadas (Seção 5.4.1) o impacto da fase de alocação na fase subsequente, de realocação, foi pequeno porque ocorre um gasto inicial no volume de dados que são necessários para calcular a centralidade *Vol*. Além disso, a fase de alocação tem um tempo curto de duração, em razão de que ela é executada enquanto o método de replicação de dados obtém as informações de acesso e instabilidade dos nós e então entra a fase de realocação da réplica que tem maior tempo de duração. Porém, esse gasto no volume de dados ocorre somente na distribuição inicial da réplica na rede. No caso da rede ser muito dinâmica em relação a mobilidade dos nós, esse cálculo da centralidade no início pode não ser interessante, visto que, muito rapidamente essa centralidade poderia ser alterada.

4.1.2 Realocação da réplica através do cálculo da centralidade ponderada por potencial

A segunda fase de realocação da réplica é dinâmica, pois são consideradas as mudanças no ambiente e as decisões do método de realocação adapta-se de forma autônoma ao contexto. A decisão de realocação das réplicas é tomada considerando as seguintes informações do contexto:

- 1- a frequência de acesso à réplica (*AF*),
- 2- a instabilidade do nó (*IN*) e
- 3- o número de saltos medido através da centralidade.

Estes parâmetros dinâmicos afetarão o cálculo da centralidade da rede, onde a ideia é deslocar a réplica a cada salto. Assim somente os nós vizinhos do nó que tem a réplica disparam o cálculo do potencial, como será visto na Seção 4.1.3.

O fato de realocar as réplicas mais próximas dos nós que as acessam com maior frequência, reduz os custos e o tempo de comunicação. *Protocolos reativos* (ou dinâmicos) alocam as réplicas de acordo com a necessidade e com os parâmetros obtidos dinamicamente do ambiente, a fim de gerar menos sobrecarga na rede [ATS09]. Esses protocolos são específicos para tratar das variações que ocorrem em um ambiente móvel.

A AF é o número de vezes que um determinado nó acessou uma réplica nas últimas Δt unidades de tempo (Δt é um intervalo de tempo fixo). Se um nó simplesmente encaminha uma requisição de outro nó, o número de acessos é também incluído neste nó. Para decisões de realocação de réplica é considerado o nó que tem mais alta frequência de acesso. Esta informação é muito importante em um método de replicação de dados, pois a ideia é que a réplica se aproxime do nó que mais a acessa com o objetivo de minimizar o uso de recursos da rede para este acesso.

A IN mede a "inadequação" do nó em receber réplicas. Tipicamente, pode ser medida pela sua velocidade de deslocamento, pela sua conectividade (intermitência de conexão, i.e. se o nó se conecta e desconecta em um curto período de tempo), pelo seu raio de alcance ou potência de transmissão (a qual caracteriza a abrangência de cobertura) e pela quantidade de recursos disponíveis (o nível de bateria restante, a capacidade de processamento e armazenamento). A instabilidade de um nó, por exemplo, deve ser diretamente proporcional à sua velocidade e inversamente proporcional ao seu nível de bateria. Sendo assim, nós estacionários devem ter um IN menor do que nós móveis, e quanto mais móvel for o nó, maior a sua IN . $IN(i)$ representa uma medição de instabilidade do nó i , que varia entre 1 (muito estável – tipicamente um nó que pertence a uma rede com infraestrutura fixa) até o limite máximo (um nó movendo-se rapidamente com baixo nível de recursos). Este valor é de conhecimento global aos nós da rede. Por exemplo, quando o nível da bateria de um nó está baixo, seu IN é alto, e este nó não pode

receber uma réplica. No caso dele já armazenar uma réplica, a réplica será transferida para outro nó mais estável.

Para a adaptação às mudanças ocorridas no ambiente foi utilizado o conceito de campos magnéticos virtuais, onde cada nó tem um valor potencial de acordo com estas informações de contexto: AF e IN , influenciando na decisão de criação de novas réplicas e de realocação das réplicas existentes.

As informações de contexto AF e IN , são armazenadas nos próprios nós. Para evitar que a réplica se desloque para os nós que acessaram a réplica anteriormente e não acessam mais, é considerada a ancianidade da informação (Δt é a janela de tempo para a contagem do número de acessos válidos usada no cálculo da AF). Se nas últimas Δt unidades de tempo, nenhuma solicitação de acesso passar pelo nó ou for feita pelo nó, o seu AF será reduzido ao valor zero. A IN também é considerada por Δt , após este período o valor de IN é reavaliado.

Para a decisão de onde alocar a nova réplica ou mover uma réplica existente, é levado em consideração a AF , a IN e o número de saltos (NS - através da centralidade Vol), a partir do qual é calculado o potencial $P(i)$.

A IN é utilizada tanto para escolher o nó que receberá a nova réplica quanto para escolher um nó para realocar uma réplica já existente em um nó mais estável. Assim, a ideia é que a réplica tende a ser deslocada para nós com a menor IN , maior AF e mais central (NS).

4.1.3. Centralidade ponderada por potencial (CPP) - Formalização

Seja $G_t=(V, E_t)$ um grafo conectado não direcionado representando a topologia da MANET no momento t . V é o conjunto de nós e E_t é o conjunto de arestas. Seja $n = |V|$ o número de nós no grafo e $m = |E_t|$, o número de enlaces. A distância (pelo caminho mais

curto) calculada em número de saltos do nó i ao nó j ($i, j \in V$) é denominada $d(i, j)$. $N(i)$ é o conjunto de nós adjacentes a i (vizinhos diretos do nó i). Na sequência, é necessário distinguir um nó que armazena a réplica de um nó que não armazena a réplica. Seja $R \subseteq V$ conjunto de nós que armazenam a réplica dos dados.

A ideia da centralidade ponderada por potencial é deslocar a centralidade considerando fatores dinâmicos de uma MANET como frequência de acesso à réplica e instabilidade dos nós. Para tanto, um valor de "potencial" do nó (que tem a réplica e dos vizinhos diretos dele) é calculado para representar uma medida de centralidade ponderada por potencial que, além da *centralidade baseada em volume localizado* (*Vol*), também considera a frequência de acesso (*AF*) e a instabilidade (*IN*). O princípio é que quanto maior o potencial de um nó, maior a força com que o nó atrai a réplica em sua direção.

A seguir será definido como este potencial é calculado.

Equações para o cálculo distribuído do potencial de um nó

Para o cálculo do potencial de um nó foram desenvolvidas fórmulas diferenciadas para o nó que tem e que não tem a réplica. Desta forma, a *AF* do nó que tem a réplica é atenuado, pois se fosse utilizado a mesma fórmula para calcular o potencial do nó que tem a réplica, os outros nós teriam uma chance muito menor de “atrair” a réplica mais próximas a eles, pois o nó que armazena a réplica, se computado o *AF* de todos os nós, sempre terá *AF* maior (e um potencial maior que os outros nós também). E, se fossem contados somente os acessos “locais” do nó que tem a réplica, este ficaria com seu potencial menor, tendo o risco da réplica ser movida desnecessariamente.

O contador de acessos (*AC*) do nó que acabou de perder a réplica precisa ser atualizado para que o nó não sofra uma variação brusca no seu potencial. Isso pode ocorrer porque ele representava os valores de acessos dos seus vizinhos e ao perder a

réplica estes acessos também serão perdidos. Então este nó que perdeu a réplica não pode ter poucos acessos a ponto de não representar os valores de acessos de seus vizinhos.

1- Para o nó que não tem a réplica ($i \notin R$) a Equação 4.1 é utilizada para calcular a $AF(i)$.

$$AF(i) = \frac{AC(i)}{MaxAC} (i \notin R) \quad (4.1)$$

Onde:

- $AC(i)$: contador de acessos, é o número de acessos à réplica que um nó i realizou ou que o nó repassou nos últimos Δt unidades de tempo. Se $i \in R$, $AC(i)$ não é incrementado quando a réplica contida no nó for acessado por outro nó $j \neq i$. Essa restrição ocorre para que o nó que armazena a réplica não concentre todos os ACs de todos os nós, ficando com uma força potencial maior e os outros nós nunca teriam chance de atrair a réplica próxima a eles. A Figura 4.1 mostra um nó requerendo a réplica e o AC sendo computado nos nós do caminho.

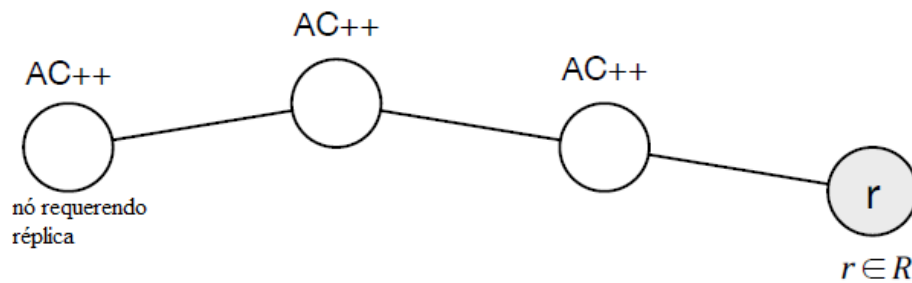


Figura 4.1: Nó requerendo a réplica

- $MaxAC$: é o número máximo de acessos por Δt que um nó é capaz de tratar ($MaxAC$ é considerado um valor global e de conhecimento comum de todos os nós na rede).

2- Para o nó com réplica ($i \in R$) a AF é função do $AC_N(i)$ ponderado pela diferença entre ele e a contagem de acessos “locais” ($AC(i)$ feitos pelo nó). Essa ponderação ($\beta(i)$) é necessária para compensar o fato do nó que tem a réplica concentrar naturalmente um maior número de acessos. A Equação 4.2 é utilizada para calcular $AF(i)$ do nó que tem a réplica.

$$AF(i) = \frac{\beta(i) \times AC_N(i)}{MaxAC} \quad (i \in R) \quad (4.2)$$

Onde:

- $AC_N(i)$: é o máximo contador de acessos de um vizinho direto, e é definido pela Equação 4.3.

$$AC_N(i) = Max \{ AC(j) : j \in N(i) \} \quad (4.3)$$

- $\beta(i)$: é o coeficiente de ponderação da AF do nó que tem a réplica descrito pela Equação 4.4. O nó que armazena a réplica pode não ter nenhum acesso local. Se utilizado o maior valor de AC dentre os seus vizinhos diretos, este nó sempre teria $AF(i)$ maior e conseqüentemente um potencial ($P(i)$) maior que os seus vizinhos. Se considerado somente o AC “local”, a réplica poderia ser movida para outro nó desnecessariamente, ou seja, qualquer nó que efetuasse um acesso já teria $AF(i)$ maior e o potencial maior de “atrair” a réplica para si. Assim, utilizando $\beta(i)$ o valor de $AF(i)$ é ajustado conforme a Equação 4.4.

$$\beta(i) = 1 - \frac{k \delta(i)}{(k-1)|\delta(i)| + MaxAC} \quad (4.4)$$

Onde:

- $\delta(i)$: é a diferença entre os acessos do vizinho (com mais acessos) e os acessos locais. A Equação 4.5 descreve $\delta(i)$. É o número de acessos do vizinho com mais acessos menos os acessos locais.

$$\delta(i) = AC_N(i) - AC(i) \quad (4.5)$$

Note que $\delta(i) < MaxAC$ porque $MaxAC \geq AC_N(i)$ e $MaxAC \geq AC(i)$. Também note que, se AC_N for igual a AC , então $\delta(i)$ é zero, $\beta(i)$ valerá 1 e AF será calculado como no caso de não ter a réplica.

O gráfico de $\beta(i)$ e $\delta(i)$ está apresentado na Figura 4.2. A título de exemplo, $MaxAC$ foi fixado em 100. Se o máximo contador de acessos de um vizinho direto é maior que o contador de acesso local, $AC_N(i) > AC(i)$, $\delta(i)$ será positivo. Caso contrário, negativo.

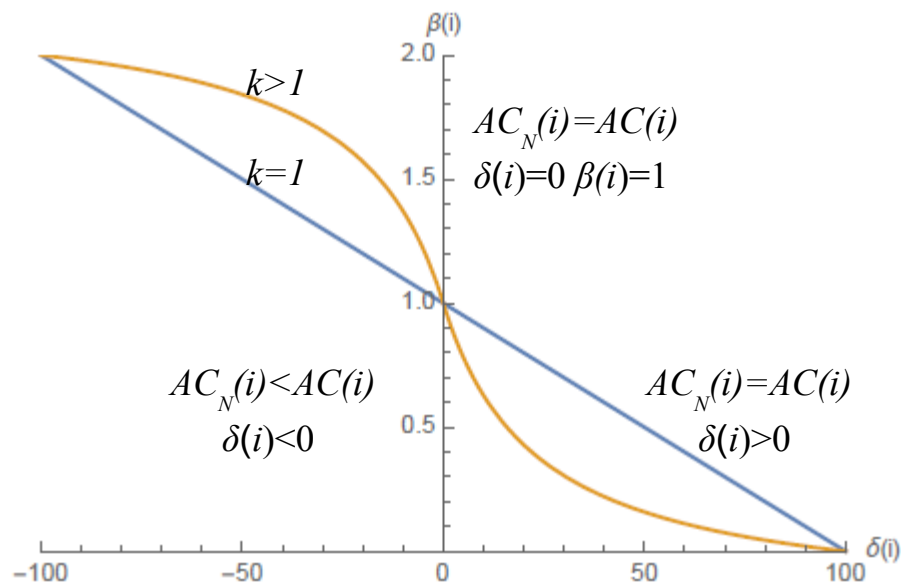


Figura 4.2: Coeficiente $\beta(i)$

- k : é uma constante que afeta a velocidade de decaimento inicial da curva em função de $\delta(i)$. Quando $k = 1$, o decaimento da curva é linear. O valor de k influencia a proporção que $\delta(i)$ terá no cálculo de $\beta(i)$. $\beta(i)$ afetará o cálculo do potencial. Quanto maior o valor de k , mais impacto este terá em $\delta(i)$ e isso influencia na velocidade com que a réplica se desloca de um nó para outro. Na Seção 4.2.2 será mostrada a influência de k no cálculo do $P(i)$.

A Contagem de Acessos (AC) de um nó que possui a réplica precisa ser recalculada quando o nó perde a réplica. Enquanto ele tinha a réplica, o cálculo era baseado nos acessos de seus vizinhos diretos. Mas, ao perder a réplica, um novo valor precisa ser atribuído de forma que o nó não tenha tantos acessos ao ponto de atrair a réplica imediatamente novamente para si, nem que tenha tão poucos acessos a ponto de não representar os valores de acessos de seus vizinhos. Sendo i o nó que armazenava a réplica que a perdeu para um vizinho, e sendo $P(i)$ o potencial de i quando tinha a réplica e $P'(i)$ o seu novo potencial ao perder a réplica, então $P'(i) < P(i)$. O seu novo potencial deve ser menor que o anterior sob pena da réplica voltar imediatamente ao nó i (que aumentaria o seu potencial ao perder a réplica). Com base nisso, a dedução do novo valor da contagem de acessos do nó i , designado por $AC'(i)$:

Sendo $P(i) = \frac{AF(i)}{IN(i)}$, para $i \in R$. Quando o nó i perde a réplica (i.e., $i \notin R$), o seu novo

potencial é dado por $P'(i) = \frac{AF'(i)}{IN(i)}$ (supondo que não houve mudança de sua instabilidade). Para que i não aumente a sua força ao perder a réplica (e, portanto, possa receber imediatamente de volta a réplica que acabou de perder), o valor de $AC'(i)$ será conforme Equação 4.6.

$$AC'(i) = \left[\left(1 - \frac{\delta(i)}{MaxAC} \right) AC_N \right] \quad (4.6)$$

Pois:

Como $P'(i) \leq P(i) \Rightarrow AF'(i) \leq AF(i) \Rightarrow AC'(i) \leq \beta(i) * AC_N(i)$ e considerando $k=1$,

assim, $AC'(i) \leq \left(1 - \frac{\delta(i)}{MaxAC}\right) AC_N$.

O peso aplicado ao nó i pelos fatores dinâmicos da rede (notadamente, AF e IN) é denominado $W(i)$. Este peso é proporcional à frequência de acesso à réplica (AF) e inversamente proporcional à instabilidade (IN) de i . Ou seja, quanto maior o número de requisições para uma determinada réplica (que trafegam através do nó i) e quanto menor sua instabilidade, maior seu $W(i)$. O peso do nó i é definido pela Equação 4.7.

$$W(i) = \frac{AF(i)}{IN(i)} \quad (4.7)$$

- $IN(i)$: é a medição de instabilidade do nó i . $IN > 0$ e positivo, ou seja, IN é um valor inteiro e ≥ 1 .

O potencial de um nó corresponde a força que um nó tem em “atrair” a réplica para si, levando em consideração a *centralidade baseada em volume localizado* (Vol), AF e IN . A fim de poder ajustar o grau de influência que a centralidade de um nó (calculando $Vol(i)$) e o seu peso W exercem sobre o valor do potencial, o fator α precisou ser definido. α pertence ao intervalo $[0,1]$. O valor do α é definido de acordo com a aplicação, mas é o mesmo para todos os nós, sendo de conhecimento global aos nós na rede.

- $Vol(H^i_h)$: é a soma de todos os graus dos nós vizinhos do nó i a h -saltos (vide Equação 2.4). Para que $Vol \in [0,1]$, ele foi normalizado, assim a Equação 4.8 mostra o Vol normalizado no intervalo fechado $[0,1]$:

$$Vol(i)_{Norm} = \frac{Vol(H_h^i)}{maxVol} \quad (4.8)$$

O máximo valor de Vol ($maxVol$) corresponde a $n(n-1)$, onde n é o número de nós na rede. Como $n-1$ é o número máximo de enlaces que nó pode ter, então o $maxVol$ é $n(n-1)$, considerando um grafo completo.

O potencial do nó i , $P(i)$ para $i \in V$ é calculado através da Equação 4.9. Como Vol e $W(i)$ são valores no intervalo fechado $[0,1]$, $P(i) \in [0,1]$. Este potencial é calculado somente para o nó r (que tem a réplica) e para os nós vizinhos diretos dele ($i \in N(r)$). Isto evita que a réplica mova ou se afaste do centro da rede muito rapidamente e caso outros nós iniciam o acesso à réplica, não corre o risco da réplica estar muito distante do nó solicitante.

$$P(i) = (\alpha \cdot Vol(i)_{Norm}) + ((1 - \alpha) \cdot W(i)) \quad (4.9)$$

Após o cálculo $P(i)$ é executado um algoritmo para avaliar se a réplica deve ser deslocada para outro nó ou mesmo duplicada. O Algoritmo de Criação de Réplicas é executado apenas para os nós que têm as réplicas. Um nó r que armazena a réplica moverá esta réplica para outro nó vizinho, somente se $P(i) - P(r)$ for maior do que um limiar (*threshold* τ), conforme definido no Algoritmo de Criação de réplicas.

Caso existam dois nós com $P(i) - P(r) > \tau$ é feita uma cópia da réplica que é movida para o segundo nó que tem maior $P(i)$. Para que não ocorra um aumento demasiado de réplicas na rede, há um número máximo de réplicas que podem ser criadas na rede.

O τ é um limiar calculado a partir do custo de transferência da réplica para um salto na rede. Este valor pode mudar dependendo das características específicas do

ambiente da aplicação. Se existe um nó i onde a diferença do potencial é maior ou igual que o τ , este nó vai atrair a réplica em sua direção. A partir disso, o potencial é recalculado e o algoritmo é executado novamente, mas agora um outro nó pertencendo a R .

Algoritmo de Criação de Réplicas

```

1 Begin
2 Let  $r \in R$ :
3 maxCopy=0;
4 limit=2; //número de cópias da réplica permitidas
4 for all  $i \in N(r)$  and  $i \notin R$  do://nós que não tem a réplica, mas estão a um salto do nó que tem
5     if  $P(i) - P(r) \geq \tau$  then //potencial de um nó a um salto é  $\geq$  potencial do nó que tem a réplica
6         send replica to  $i$ ;
7     else if maxCopy $\leq$ limit //se não ultrapassou o número de cópias da réplica permitidas
8         copy replica to  $i$ ;
9         maxCopy+1;
10    endif
11    endfor
12 End

```

A Tabela 4.1 sumariza os valores envolvidos no cálculo do potencial.

Tabela 4.1: Notações dos principais elementos da fase de realocação dos dados

| | |
|--------------|---|
| AF | Frequência de acesso |
| AC | Contador de acesso |
| $MaxAC$ | Número máximo de acessos por Δt |
| IN | Medição de instabilidade |
| W | Peso aplicado ao nó |
| P | Potencial do nó |
| α | Fator de ponderação da centralidade e do W |
| $Vol(H_h^i)$ | Soma de todos os graus dos nós vizinhos do nó i a h -saltos |

| | |
|----------|--|
| β | Coefficiente de ponderação da AF do nó que tem a réplica |
| k | Constante que afeta a velocidade de movimentação da réplica |
| δ | Diferença entre AC e AC_N |
| AC_N | Contador de acessos de um vizinho direto |
| τ | Limiar calculado a partir do custo de transferência da réplica |
| AC' | Contador de acessos de um nó quando perde a réplica |

Em relação ao ciclo de vida das réplicas, a utilização da centralidade ponderada por potencial possibilita algumas vantagens no seu uso, tais como:

1- Na criação da réplica: impede a sobreposição de potenciais, pois quando dois nós acessam uma mesma réplica em lados opostos com alta frequência, uma cópia da réplica é gerada, evitando a sobreposição dos campos magnéticos. A Figura 4.3 mostra como ocorre a sobreposição de potenciais.

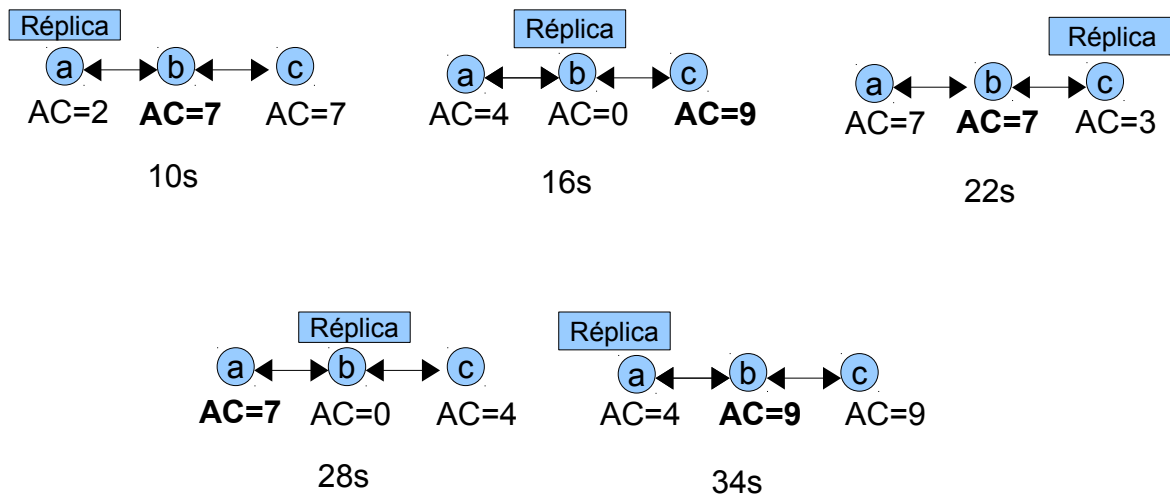


Figura 4.3: Sobreposição de potenciais

2- Na eliminação da réplica: ocorrendo dos nós diminuírem ou até mesmo cessarem o acesso as réplicas, essas tendem a voltar para o centro da rede, podendo localizarem-se até no mesmo nó (pois o AF ficará zerado ou com valor muito baixo e a centralidade Vol que terá uma influência maior no cálculo da força potencial). Ocorrendo esta situação, uma delas seria eliminada, fechando desta forma o ciclo de vida da réplica. Como a cada período é executado o $P(i)$, mesmo não tendo nenhum acesso o nó que tem a réplica dispara o cálculo do potencial. A Figura 4.4 mostra como ocorre a eliminação de uma réplica.

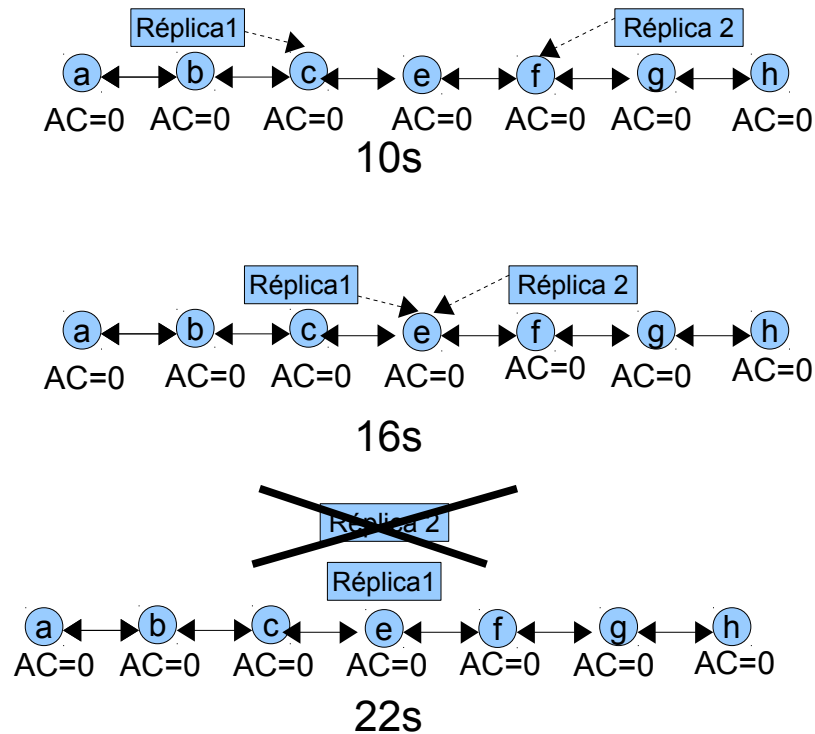


Figura 4.4: Eliminação de uma réplica

4.2. Demonstração do Método de Replicação de Dados

Com o objetivo de mostrar o método definido, esta seção considera um exemplo da aplicação do método de replicação de dados, considerando o grafo não-direcionado G da Figura 4.5, composto de 8 nós e de 10 enlases. Este cenário pode ser aplicado a uma escola onde os alunos têm computadores portáteis ou outros dispositivos móveis e movimentam-se na velocidade de até 3m/s [ATS08]. Eles podem trocar informações sobre a disciplina ou entretenimento. Primeiramente, será apresentada a fase de alocação da réplica e em seguida a de realocação da réplica.

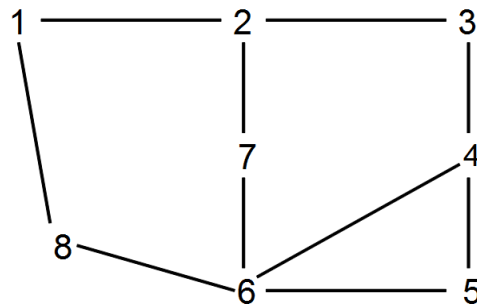


Figura 4.5: Grafo de 8 nós e 10 enlases

4.2.1. Alocação da réplica

A Tabela 4.2 apresenta a matriz de adjacência utilizada para o cálculo da centralidade de proximidade através do algoritmo com solução aproximada chamado de *centralidade baseada em volume localizado (Vol)*. Esta tabela apresenta uma visão global da rede. Entretanto, cada nó tem conhecimento somente dos vizinhos adjacentes, não tendo conhecimento global da rede.

Tabela 4.2: Matriz de adjacência do grafo da Figura 4.5 (*Vol*)

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 2 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 3 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 5 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 6 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 7 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 8 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |

A Tabela 4.3 mostra como foi realizado o cálculo do *Vol* da matriz de adjacência (Tabela 4.2), de acordo com a Equação 2.4. Neste trabalho foi definido o valor de $h=1$.

Tabela 4.3: Cálculo da *centralidade baseada em volume localizado* (*Vol*)

| Nó (<i>i</i>) | $H(i,h^*)$ | $dg(j^{**})$ | $Vol(i,h^*)$ |
|----------------------|------------------------------|--------------------------------|--------------------------------|
| 1 | 1,2,8 | 2+3+2 | 7 |
| 2 | 2,1,3,7 | 3+2+2+2 | 9 |
| 3 | 3,2,4 | 2+3+3 | 8 |
| 4 | 4,3,5,6 | 3+2+2+4 | 11 |
| 5 | 5,4,6 | 2+3+4 | 9 |
| 6 | 6,4,5,7,8 | 4+3+2+2+2 | 13 |
| 7 | 7,2,6 | 2+3+4 | 9 |
| 8 | 8,1,6 | 2+2+4 | 8 |

* $h=1$

** $j \in H(i,h)$

A Figura 4.6 mostra como ficou o grafo após o cálculo da *centralidade baseada em volume localizado*.

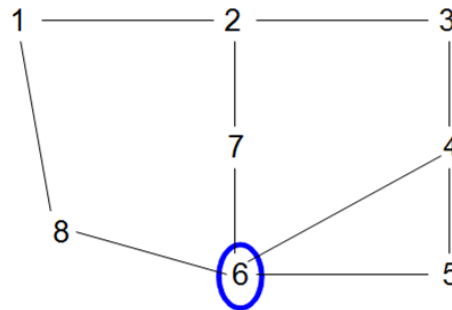


Figura 4.6: Alocação de uma réplica através do algoritmo *Vol*.

4.2.2. Realocação da réplica

A segunda etapa do método de replicação de dados compreende a realocação da réplica na rede dinâmica. De acordo com a primeira etapa de alocação e utilizando a *centralidade baseada em volume localizado*, a réplica estaria armazenada no nó 6.

A partir da segunda etapa serão consideradas além da distância em número de saltos (*NS*) também as informações de contexto (frequência de acesso (*AF*), instabilidade do nó (*IN*)) para decidir sobre a criação de uma nova réplica, realocação de uma já existente e a localização da réplica.

Neste exemplo foram utilizados os seguintes valores para o cálculo do potencial $P(i)$:

$$MaxAC = 12;$$

$$\alpha = 0,40;$$

$$\Delta t = 12s;$$

$k=1$;

$\tau = 0,03$;

$maxVol = n(n-1) = 8(8-1) = 56$, sendo n = número de nós.

Como a réplica inicialmente é armazenada no nó 6, o $\beta(i)$, o $AC_N(i)$ e o $\delta(i)$ somente precisa ser calculado para este nó ($6 \in R$). A Tabela 4.7 mostra as informações de contexto utilizadas para o cálculo do potencial dos nós vizinhos diretos do nó 6 e a força de atração que esses nós têm sobre a réplica 1.

Tabela 4.7: Potencial da réplica 1

| Node (i) | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|--------------|---|---|---|----------------------|------|-------------|---|----------------------|
| AC | 2 | 0 | 2 | 4 (2-local e 2-nó 3) | 1 | 1 | 0 | 6 (4-local e 2-nó 1) |
| AC_N | - | - | - | - | - | 6 | - | - |
| AC' | - | - | - | - | - | - | - | - |
| δ | - | - | - | - | - | 5 | - | - |
| β | - | - | - | - | - | 0,58 | - | - |
| AF | | | | 0,33 | 0,08 | 0,29 | | 0,5 |
| IN | | | | 1 | 4 | 2 | | 2 |
| W | | | | 0,33 | 0,02 | 0,14 | | 0,25 |
| Vol | | | | 11 | 9 | 13 | | 8 |
| Vol_{Norm} | | | | 0,19 | 0,16 | 0,23 | | 0,14 |
| P | | | | 0,26 | 0,07 | 0,17 | | 0,20 |

Se fosse utilizado um $k=2$, o valor de $\beta(6) = 0,41$, $AF(6) = 0,20$, $W(6) = 0,10$ e $P(6) = 0,15$. Desta forma, quanto maior o valor de k , menor o valor de $P(6)$.

De acordo com a Tabela 4.7, o nó com o maior valor de $P(i)$, (maior força de atração) é o nó 4. Como $P(i) - P(r)$ é maior do que o *threshold* (τ), de acordo com Algoritmo de Criação de Réplicas, a frequência de acesso e a estabilidade atrai o "centro"

do grafo em direção ao nó 4. O nó 6, que tem os dados iniciais, tem um valor potencial $P(6)= 0,17$. No entanto, o nó 4 tem valor potencial de $P(4)= 0,26$. Como $(0,26 - 0,17) \geq 0,03$, assim, o nó 4 vai atrair a réplica em sua direção. Como existe um segundo nó (8) que tem $P(i) - P(r)$ maior do que o *threshold* (τ), será feita uma cópia da réplica e enviada para este nó. A partir deste momento, a réplica 1 ficará armazenada no nó 4 e a réplica 2 no nó 8. O potencial é recalculado e o algoritmo é executado novamente, mas agora com o nó 4 e 8 pertencendo a R .

Cada nó calcula o seu potencial, pois ele conhece a sua frequência de acesso, seu *Vol* e sua instabilidade. A Tabela 4.7 não é, portanto, armazenada em um nó específico, ela está distribuída pela rede.

4.2.3. Análise do custo de comunicação

Uma estimativa do custo de comunicação do método proposto para a realocação da réplica pode ser assim apresentada:

Para o acesso à réplica, quando o nó que deseja realizar o acesso sabe onde a réplica está armazenada, o número de mensagens é $2 * d$, no pior caso, sendo d o diâmetro do grafo [MIC14]. Para o caso onde o nó não sabe a localização da réplica, ou seja, não acessou essa réplica até o momento, um *flooding* é necessário, para encontrar em que nó a réplica está alocada. O número de mensagens no pior dos casos é $(2m - n + 1) + d$, sendo m , o número de arestas e n , o número de nós do grafo.

Assim, a complexidade do algoritmo é de $O(n^2)$, onde m no pior caso (um grafo completo) é $n(n-1)/2$. Já, no melhor caso (quando o grafo é uma árvore), a complexidade é de $O(n)$. Como em uma árvore o número de arestas é $n-1$ (número de nós - 1), assim a complexidade ficaria, no melhor caso, n .

Como em uma topologia de MANETs dificilmente um grafo será completo (para ser completo cada nó teria que estar conectado a todos os outros nós da rede) a

complexidade será linear. O coeficiente de crescimento do número de mensagens será tanto maior quanto maior for o número médio de conexões dos nós. Em MANETs, no entanto, tipicamente este número é pequeno. Em um cenário típico existe um número máximo de conexões por nó e este valor não é grande o suficiente ao ponto de ser um grafo completo.

De acordo com Cunha [CUN12], é esperado que um algoritmo execute com complexidade linear, já que ele executa um número fixo de operações sobre cada elemento de entrada. Algoritmos de complexidade exponencial normalmente são inviáveis na prática.

4.3. Discussão

Neste capítulo foi apresentado o método de replicação de dados e as suas duas etapas: de alocação e de realocação da réplica com o cálculo do potencial dos nós. Na fase de realocação foi descrito o algoritmo que calcula, através do limiar para custo de transferência das réplicas, a viabilidade de copiar ou mover uma réplica. Para um melhor entendimento e validação do método foi descrito um exemplo do cálculo dos algoritmos aqui propostos e utilizados. Também foi discutida a complexidade dos algoritmos envolvidos. Com os esforços despendidos para a definição e a formalização do método pode-se constatar que esta fase é importante para encontrar falhas e problemas que muitas vezes só serão vistos na fase posterior: fase de validação do método. Também esta formalização é importante para o entendimento do problema e procura de uma solução viável.

Através do modelo proposto pode-se concluir que quanto menor o grau dos nós (menor número de conexões na rede), o desempenho do algoritmo é melhor. Isto porque se os nós estão conectados na sua maioria a um salto de distância de todos os nós, não

haverá diferença significativa em utilizar a centralidade ponderada por potencial. Entretanto, como já comentado na seção anterior, em MANETs tipicamente isso não ocorre.

O modelo aqui proposto considerou vários coeficientes, tais como o $\beta(i)$, o α e o k , que permitem uma implementação do método realizar um ajuste nos parâmetros do modelo proposto. Esta possibilidade de ajuste faz com que o método possa ser empregado em diversos cenários de aplicações.

No capítulo seguinte, é descrito o ambiente de simulação e os resultados obtidos com os testes realizados em MANETs.

Capítulo 5

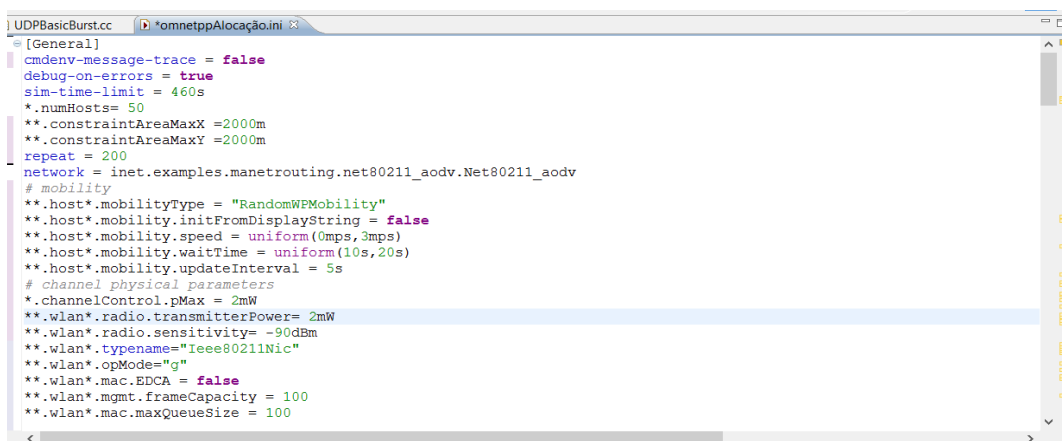
Ambiente de Simulação e Resultados Obtidos

Neste capítulo são apresentados: (a) o ambiente de simulação utilizado para executar os testes do método de replicação de dados e (b) a avaliação dos resultados obtidos. A Seção 5.1 apresenta o ambiente de simulação e as ferramentas utilizadas. A Seção 5.2 descreve os parâmetros utilizados para simular o algoritmo proposto. Na Seção 5.3 são apresentadas as métricas de avaliação. A Seção 5.4 discute a avaliação do método de acordo com os resultados obtidos. Por fim, são realizadas as últimas considerações do trabalho e as conclusões deste.

5.1. Ambiente de Simulação

O método de replicação de dados, definido no Capítulo 4, foi implementado na linguagem de programação C++ com o *framework* de simulação OMNet++ (*Objective*

Modular Network Testbed in C++) [VAR15]. Este *framework* é de propósito geral, apesar de inicialmente ter sido desenvolvido para redes de computadores [VAR08]. O OMNeT++ possui licença livre para uso acadêmico, suporta diversos tipos de redes, incluindo redes *ad hoc* móveis e executa em vários sistemas operacionais, como Linux, Windows, Mac OS X, etc. O OMNeT++ foi desenvolvido em linguagem C++ e seus componentes são modulares. A arquitetura modular diferencia o núcleo das simulações da interface gráfica, que é baseada na plataforma Eclipse. Para a implementação das simulações, os módulos se agrupam mediante uma linguagem de descrição de topologia chamada NED. Os parâmetros de configuração das simulações são descritos em arquivos com um formato específico da plataforma (extensão ".ini"). A Figura 5.1 mostra um trecho de um arquivo de configuração. Neste arquivo é possível especificar protocolos utilizados, formato de saída dos resultados, padrão de mobilidade, potência de transmissão, entre outros.



```
UDPBasicBurst.cc  *omnetppAlocação.ini
[General]
cmdenv-message-trace = false
debug-on-errors = true
sim-time-limit = 460s
*.numHosts= 50
**.constraintAreaMaxX =2000m
**.constraintAreaMaxY =2000m
repeat = 200
network = inet.examples.manetrouting.net80211_aodv.Net80211_aodv
# mobility
**.host*.mobilityType = "RandomWPMobility"
**.host*.mobility.initFromDisplayString = false
**.host*.mobility.speed = uniform(0mps,3mps)
**.host*.mobility.waitTime = uniform(10s,20s)
**.host*.mobility.updateInterval = 5s
# channel physical parameters
*.channelControl.pMax = 2mW
**.wlan*.radio.transmitterPower= 2mW
**.wlan*.radio.sensitivity= -90dBm
**.wlan*.typename="Ieee80211Nic"
**.wlan*.opMode="g"
**.wlan*.mac.EDCA = false
**.wlan*.mgmt.frameCapacity = 100
**.wlan*.mac.maxQueueSize = 100
```

Figura 5.1: Exemplo do arquivo de configuração

O *framework* OMNeT++ permite a utilização de outras ferramentas de simulação, de acordo com a necessidade da aplicação e para facilitar o desenvolvimento destas. Para este trabalho foi utilizado o INETMANET, uma extensão do INET, pacote de simulações de rede do OMNeT++, mas que suporta MANETs. O INET é a biblioteca padrão do OMNeT++ que contém modelos da pilha de protocolos da Internet (TCP, UDP, IPv4,

IPv6, OSPF, BGP, etc.), protocolos da camada de enlace (Ethernet, PPP, IEEE 802.11, etc), suporta DiffServ, MPLS, e muitos outros protocolos e componentes. O INET segue a mesma filosofia de licença livre do OMNeT++. Além disso, ele permite modificar ou até mesmo desenvolver novos protocolos e modelos. O INETMANET basicamente provê as mesmas funcionalidades do INET, com protocolos e componentes adicionais para as MANETs, como por exemplo, protocolos de roteamento específicos para estas redes [DRE12].

5.2. Parâmetros das Simulações

As simulações do algoritmo distribuído para o cálculo da centralidade ponderada foram executadas em MANETs de médio porte [ATS13], de acordo com os parâmetros abaixo, assumindo que na rede todos os nós têm ao menos uma conexão, ou seja, a rede não tem nós isolados.

- Número de nós: 20 nós móveis [ATS13], [ATS07].
- Área para a simulação: 300m x 300m [ATS13], 350m x 350m (quando as simulações são para 35 nós), 400m x 400m (50 nós) e 600m x 100m (área retangular para demonstrar a força de atração da réplica).
- Alcance de comunicação: 100m [ATS13], [ATS07].
- Velocidade dos nós: na faixa de 0-3m/s, velocidade média de um pedestre caminhando [ATS13], [ATS07] e 0-6m/s.
- Modelo de mobilidade: *Random WayPoint* (ou "percurso aleatório") [BAI04], [JIN04], [ATS13], [ATS07].
- Tempo de simulação: no máximo 100s. Este foi o tempo utilizado aproximadamente, pois uma leitura ocorre a cada 1s, e foram testadas 40

requisições. O tempo restante foi utilizado para a inicialização da rede e para o cálculo dos algoritmos.

- Número de simulações: aproximadamente 10, para conseguir uma média nos resultados das simulações [JIN04]. Com este número de simulações os resultados se equiparavam não necessitando de um número maior de simulações. Isso ocorre porque as simulações foram dirigidas em melhor caso, caso médio e pior caso. Nestes casos, a réplica não é armazenada aleatoriamente em um nó, mas sim escolhida a sua localização de forma que ela represente o melhor caso, o caso médio e o pior caso.
- Tamanho da réplica (somente *payload*): 7,81 Kbytes ou 8.000 bytes referente ao tamanho aproximado de 4 páginas de texto.
- Tamanho da mensagem (somente *payload*): 4 bytes. São as mensagens gastas para calcular o algoritmo da centralidade *Vol* e centralidade ponderada por potencial (*CPP*). Como o tamanho dos pacotes em redes sem fio (MTU) é de 2.304 bytes (equivalente a 2,25 Kbytes) [PLA11], [SIL10], cada mensagem é enviada em um único pacote.

Não foi incluído, no tamanho da mensagem e no tamanho da réplica, os cabeçalhos dos pacotes de uma rede sem fio, já que o cabeçalho IP e MAC pode ser de tamanho variável [BRI15], [PAU15]. Também porque o objetivo deste trabalho é a camada de aplicação da rede e não os protocolos das camadas inferiores.

- Frequência de requisição de dados (taxa de leitura): a cada 1s.
- Potencial do nó - $P(i)$: calculado a cada 6s (para não sobrecarregar a rede mas, ao mesmo tempo para obter uma atualização constante). Nas simulações este foi o valor que levou a resultados bons levando-se em consideração o custo para calcular $P(i)$, conforme será mostrado na Seção 5.4.7.
- Δt : 12s (período de validade do *AC* e *IN*).

- α : 0,40 (este valor é usado para ajustar a intensidade de influência do *Vol* and *W* na fórmula $P(i)$. Nas simulações este foi o valor encontrado para que haja um equilíbrio adequado entre a centralidade (*Vol*) e o peso (*W*), conforme será discutido na Seção 5.4.3.
- *MaxAC*: 12 (número máximo de acessos).
- τ (custo de transferir a réplica): aproximadamente 0,05 (se este valor é muito menor, o custo de mover a réplica pode ser mais alto que o benefício da réplica ser movida para outro nó, no entanto se for muito maior, a réplica poderia nunca se mover para outro nó).
- k : 1
- número de réplicas: 2 (limite do número de réplicas a serem criadas no sistema).
- Protocolo de roteamento: OLSR [PUS14], descrito na Seção 2.1.
- Protocolo de transporte: UDP [ATS13].

Para comparação com o método de centralidade ponderada por potencial foram utilizadas duas estratégias para definir a localização da réplica. Na primeira, chamada “centralidade *Vol*”, que já é uma estratégia que faz parte da centralidade ponderada por potencial, a réplica permanece no centro do grafo durante todo o tempo de simulação. Na segunda estratégia, chamada “sem centralidade”, a réplica é colocada em um nó escolhido levando-se em consideração o número de saltos que o mesmo está em relação ao nó mais central, no que diz respeito ao melhor caso, caso médio e pior caso. Esses casos serão descritos na Seção 5.3. Esses métodos foram utilizados para comparação, com o objetivo de demonstrar que um método estático, mesmo que utilizando a centralidade, mas sem considerar fatores dinâmicos, não é adequado em um ambiente dinâmico, como nas MANETs.

A escolha pelo modelo de mobilidade utilizado (percurso aleatório do inglês, *Random WayPoint* - RWP) é pelo fato de ser um dos modelos mais utilizados [CAM02]. Neste modelo a localização, a velocidade e a direção de um nó são escolhidas

aleatoriamente e mudam cada vez que o nó atinge o seu destino. Também inclui um tempo de pausa entre as mudanças de direção e velocidade [BAI04]. Seu funcionamento pode ser descrito a seguir: um nó móvel permanece sem mobilidade durante um determinado período de tempo, chamado de *tempo de pausa*. Quando este tempo expira, um destino aleatório (dentro da área de simulação) é escolhido e uma velocidade é selecionada entre a velocidade mínima e máxima permitidas. Ao chegar ao destino, o nó móvel permanece imóvel pelo tempo de pausa e então novamente inicia o processo [CAM02].

5.3. Métricas de Avaliação

Os experimentos do método aqui proposto, foram realizados considerando as seguintes métricas de avaliação:

1- **Distância média da réplica:** é a média do número de saltos que necessitam ser atravessados por uma solicitação para obter a réplica [ATS09], [HAR09]. Essencialmente é a distância do nó que tem a réplica até o nó que solicita a réplica. Basicamente esta métrica determina o número de saltos que são necessários atravessar, utilizando o algoritmo de centralidade ponderada por potencial (solução aqui proposta), centralidade *Vol* (i.e., a réplica permanece no centro do grafo) e sem centralidade (i.e., a réplica é colocada em um nó dentre os casos que serão descritos na sequência: o pior, o melhor e o caso médio), em relação ao diâmetro do grafo (maior distância entre dois vértices do grafo através do caminho mais curto). Distância menor, em relação à localização da réplica, resultam em menor atraso para as solicitações de dados [ATS09].

2- **Volume de dados transmitidos (em Kbytes) por requisição:** é a soma dos tamanhos das mensagens enviadas (em Kbytes) do nó que tem a réplica até o nó que solicita a réplica. Essa métrica expressa a sobrecarga de mensagens na rede gerada pelo algoritmo de centralidade ponderada por potencial e centralidade *Vol*.

Os resultados das simulações utilizando as duas métricas acima descritas foram obtidos variando-se a topologia de rede e o nó onde a réplica estaria armazenada inicialmente (fase da alocação). Assim foi considerado o pior, o médio e o melhor caso em relação ao nó onde a réplica será armazenada para cálculo do número de saltos necessários. O nó que acessa esta réplica é sorteado, tanto no caso em que somente dois nós requerem a réplica e também quando vários nós requerem a réplica. Abaixo segue a explicação com um exemplo demonstrando o melhor caso, caso médio e pior caso.

- Melhor caso: a réplica está no mesmo nó que é mais central. Para se chegar a este nó foi calculado a centralidade *Vol*. Na topologia de rede cujos os nós estão representados na Figura 5.2, um exemplo seria a réplica estar armazenada no nó 8, que é o mais central.
- Pior caso: a réplica está na borda do grafo, ou seja, provavelmente os nós que acessarão, precisarão, na média, um maior número de saltos. Para se chegar a este nó buscou-se o nó que está mais distante do nó mais central do grafo. Na topologia de rede cujos os nós estão representados na Figura 5.2, um exemplo seria a réplica estar armazenada no nó 10.
- Caso médio: a réplica está a meia distância entre o nó mais central e o menos central (do pior caso). Para se chegar a este nó foi calculado o nó que necessita um maior número de saltos para chegar ao nó mais central do grafo (pior caso), dividido por 2. O nó que tivesse o número de saltos para chegar ao nó central mais próximo a este valor, seria escolhido. Na topologia de rede cujos os nós estão representados na Figura 5.2, um exemplo seria a réplica estar armazenada no nó 1. Por exemplo, se o nó 10 está a 4 saltos de distância do nó 8, o nó 1 está a 2 saltos de distância do nó 8.

Um quarto caso, o "aleatório", seria o mais próximo do real, porém tem-se menos controle do efeito da alocação inicial. Uma alocação inicial aleatória não traria muitos benefícios para a análise, uma vez que não teria como prever a distância do nó e assim um número maior de simulações precisariam ser realizadas para garantir os resultados.

A Figura 5.2 mostra um exemplo de uma rede no INETMANET com 20 nós, sendo que a rede é um grafo conexo, o alcance de cada nó é de 100m e a área da rede é de 300m x 300m.

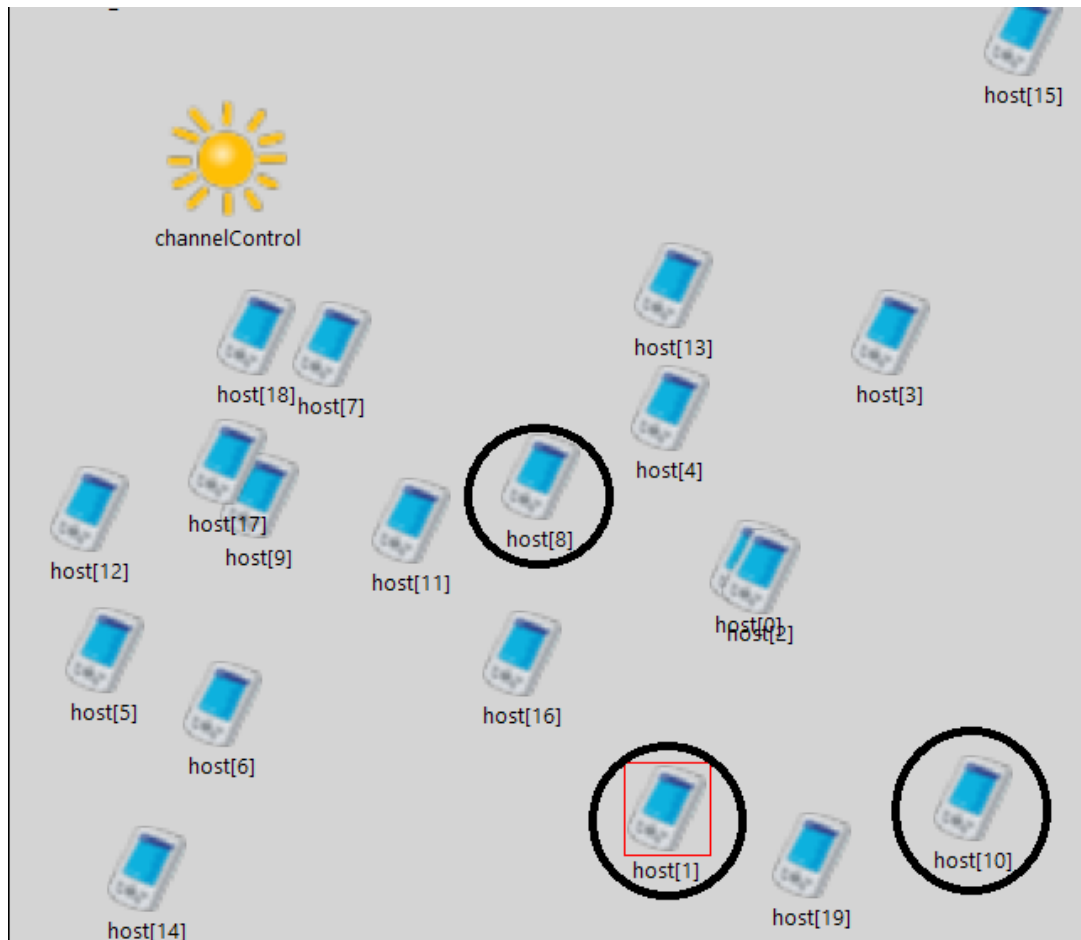


Figura 5.2: Exemplo de uma rede no INETMANET

5.4. Avaliação dos Resultados Obtidos

Primeiramente foram realizados testes para a verificação da viabilidade em utilizar centralidade na fase de alocação dos dados. Após, foram executados experimentos

para testar a realocação dos dados através do algoritmo de centralidade ponderada por potencial (*CPP*).

5.4.1. Análise da viabilidade da centralidade *Vol* na alocação dos dados

Como este projeto propõe que o método de replicação de dados utilize métodos diferenciados para alocar e realocar os dados, foi analisada a viabilidade em utilizar métodos específicos para cada uma delas. Para isso, foram executadas simulações para comparar a utilização de um método diferenciado para a alocação inicial dos dados na rede em relação a distribuição aleatória dos dados. Isso porque as informações de contexto (por exemplo, frequência de acesso à réplica) na alocação inicial não estão disponíveis para utilização da centralidade ponderada por potencial, então ela é utilizada na etapa de realocação.

A Figura 5.3 mostra o volume dos dados (em Kbytes) que são necessários para obter a réplica no método de alocação pelo algoritmo de centralidade *Vol* em comparação ao algoritmo sem centralidade (não utilização de centralidade). No grafo, o eixo horizontal indica os métodos utilizados na comparação e o eixo vertical indica a média do volume de dados (em Kbytes) utilizados. Neste cenário, 2 nós (de um total de 20 nós existentes na rede) aleatoriamente requerem o acesso à réplica. Observa-se na Figura 5.3 que a centralidade *Vol* necessita de um menor volume de dados para obter a réplica mesmo somando-se as mensagens que são gastas para o cálculo da centralidade.

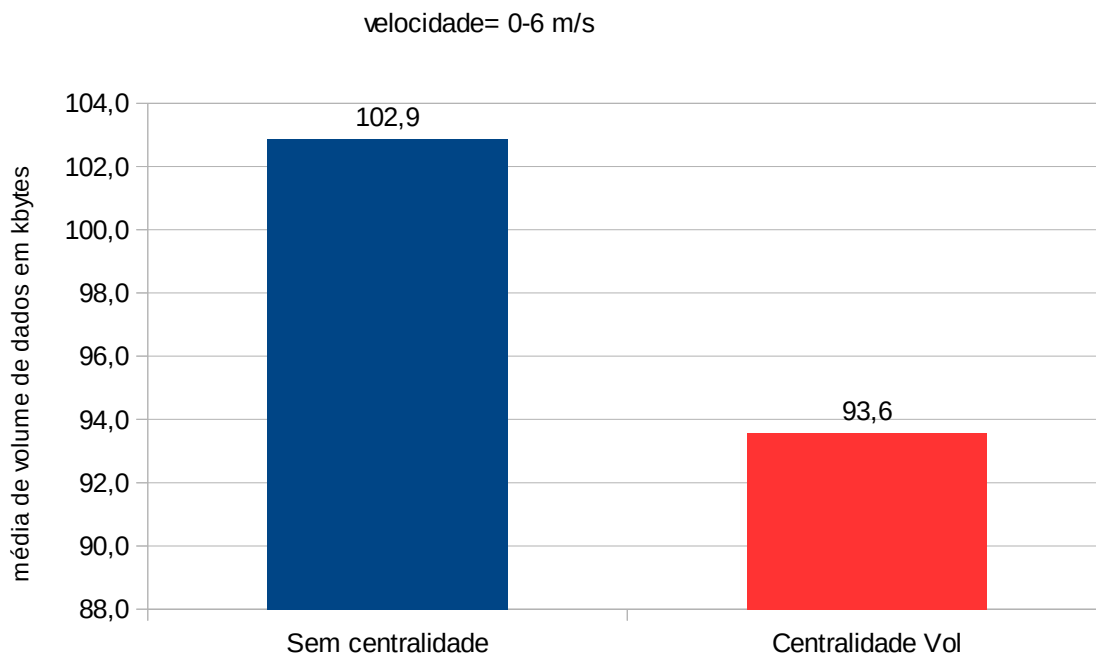
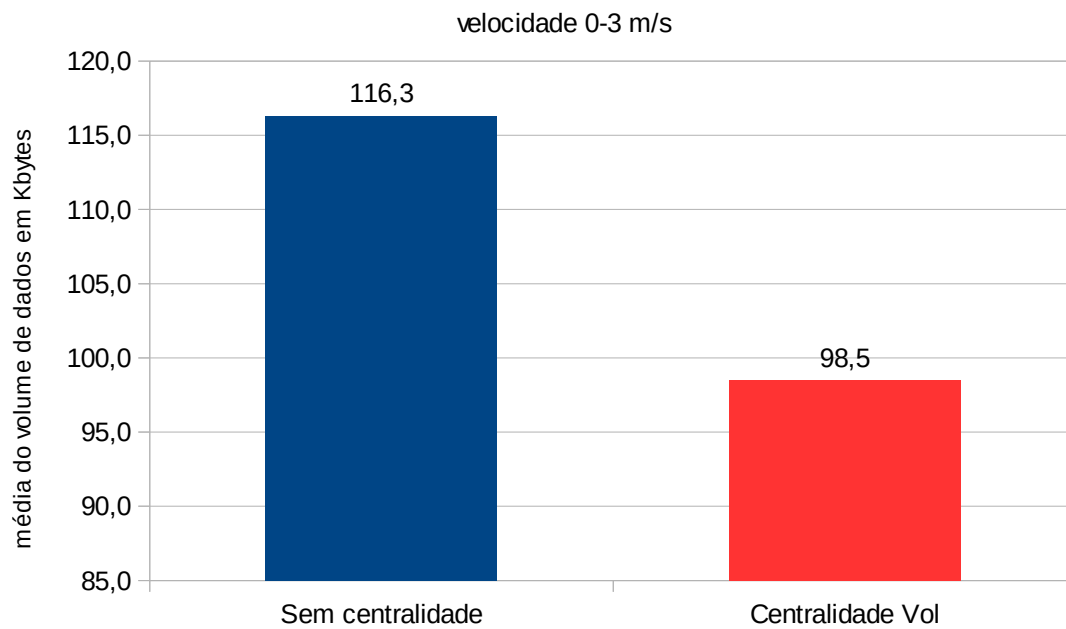


Figura 5.3: Volume de dados para acessar a réplica na fase de alocação dos dados.
 (a) com velocidade de 0-3m/s. (b) com velocidade de 0-6m/s.

Na Figura 5.3 (a), com a velocidade de 0-3m/s, o método sem centralidade utiliza 116,3 Kbytes para obter a réplica, enquanto o método da centralidade *Vol* (que localiza o nó mais central para armazenar a réplica) utiliza 98,5 Kbytes. Desta forma, ocorre uma economia de 17,8 Kbytes utilizando a centralidade *Vol*.

Na Figura 5.3 (b), com a velocidade de 0-6m/s, a centralidade *Vol* também necessita de um menor volume de dados para acessar a réplica, mesmo somando-se as mensagens que são gastas para o cálculo da centralidade. Enquanto o método sem centralidade utiliza 102,9 Kbytes para acessar a réplica, o método da centralidade *Vol* utiliza 93,6 Kbytes, ocorrendo uma economia de 9,3 Kbytes.

Em comparação com a Figura 5.3 (a), na Figura 5.3 (b) a economia no volume de dados é menor. Com velocidade de 0-3m/s, a economia é de 20% (Figura 5.3 (a)) utilizando a centralidade *Vol*) e com velocidade de 0-6m/s, a economia é de 9% (Figura 5.3 (b)) utilizando a centralidade *Vol*). Uma velocidade maior não foi testada pois seria uma velocidade muito superior ao cenário aqui proposto para as simulações, que é a velocidade de pedestre.

Afirmar que utilizar *Vol* na alocação é viável ou até mesmo fundamental para o método de replicação de dados como um todo, depende de muitos fatores. Um dos fatores é, que a viabilidade em utilizar centralidade *Vol*, depende da velocidade e do padrão de deslocamento dos nós. Quanto maior a velocidade dos nós, provavelmente a centralidade do grafo altera-se mais rapidamente, e o volume dos dados utilizados para obter a réplica tendem a aumentar. No entanto, dependendo do padrão de movimentação (pois os nós podem se aproximar do centro do grafo ou se afastar do centro) um maior ou menor volume de dados serão economizados. Também de acordo com a dinamicidade da rede, se esta é alta, como por exemplo, a velocidade de deslocamento dos nós, talvez não seja viável definir a centralidade na fase de alocação, pois na fase de realocação da réplica, o nó definido mais central, poderia não estar mais centralizado na rede, já que esta centralidade, calculada anteriormente, poderia estar desatualizada.

Outro fator é o valor de τ . Se o τ é pequeno, conseqüentemente, o tempo da fase de alocação em relação ao tempo total da simulação será menor, pois a tendência é que com τ menor, a réplica irá mover-se mais rapidamente e a fase de realocação iniciará antes. Assim, o volume de dados economizados é menor, pois deve-se considerar os dados que são gastos para o cálculo da centralidade *Vol*. Ainda outro fator, é que por exemplo, na fase de realocação, já será usada a centralidade para definir o potencial de um nó, $P(i)$. De qualquer forma se a réplica, mesmo estando inicialmente nas bordas da rede, tenderá a deslocar-se para a centralidade.

Em geral, em uma MANET típica, a relação custo-benefício no uso da centralidade *Vol* na fase de alocação é baixo, ou seja, o ganho em relação ao volume de dados transferidos para obter a réplica é mínima. Nestas simulações foi utilizada a centralidade *Vol* na fase de alocação pois essa centralidade faz parte da fase de realocação no cálculo do potencial e a dinamicidade da rede não é alta. Se a dinamicidade da rede é alta e a topologia da rede altera rapidamente, o cálculo poderia ser realizado no decorrer na fase de realocação, juntamente com o cálculo do potencial dos nós, economizando mensagens pois os mesmos poderiam ser enviados na mesma mensagem. Caso não fosse utilizado na fase seguinte, poderiam ser feitos outros estudos para avaliar o impacto de usar centralidade *Vol* somente na fase de alocação da réplica quando a rede é muito dinâmica em relação a mobilidade.

5.4.2. Efeito da variação no número de consultas (requisições)

Primeiramente foram analisados os efeitos causados pelo acréscimo no número de consultas iniciando de 1 e estendendo até 40 requisições. As simulações foram realizadas até 40 requisições porque com este número, o volume de dados transmitidos para obter a réplica permanece praticamente constante, e em alguns casos, pode chegar a zero (quando a réplica está armazenada no nó que faz a requisição). As Figuras 5.4 e 5.5 mostram os

resultados desta simulação utilizando o método sem centralidade, a centralidade *Vol* e a centralidade ponderada por potencial, em relação ao diâmetro do grafo. Esses métodos foram comparados com o diâmetro do grafo³ para determinar um limite superior da distância em saltos entre dois nós na rede.

Nos gráficos da Figura 5.4, somente dois nós requerem o acesso à réplica. Como o número de réplicas permitidas para esta simulação é dois, é possível verificar com maior clareza o potencial de atração que o nó que acessa a réplica possui. A Figura 5.4 (a) mostra a métrica 1, distância média da réplica. O eixo vertical mostra a média do número de saltos requeridos para obter a réplica por requisição. Já a Figura 5.4 (b) mostra a métrica 2, volume de dados transmitidos. O eixo vertical mostra a média do volume de dados (em Kbytes) transmitidos por requisição. Em ambas as Figuras, (a) e (b), o eixo horizontal indica o número de requisições efetuadas.

Observa-se que em ambos os gráficos da Figura 5.4, a centralidade ponderada por potencial causa um decréscimo no número de saltos e no volume de dados em Kbytes, que são necessários para obter a réplica, quando o número de requisições aumenta. Já, no método sem centralidade o número de saltos e o volume de dados, que são necessários atravessar para obter a réplica, permanece praticamente constante, ou seja, não decrementa com o aumento do número de requisições. A variação que ocorre, é ocasionada pela mobilidade dos nós. Como esperado, o número de saltos e mensagens diminui com o aumento do número de requisições no método *CPP*. Isso ocorre porque a réplica é “atraída” para o nó com maior valor potencial ($P(i)$).

Pode-se observar também, que na Figura 5.4 (b), no início (primeira requisição), o gasto de mensagens é maior devido as mensagens que se referem ao cálculo do algoritmo, mas logo após o cálculo, começa diminuir consideravelmente as mensagens gastas para obter a réplica.

³ Diâmetro do grafo: maior distância entre dois vértices em um grafo [NAS12].

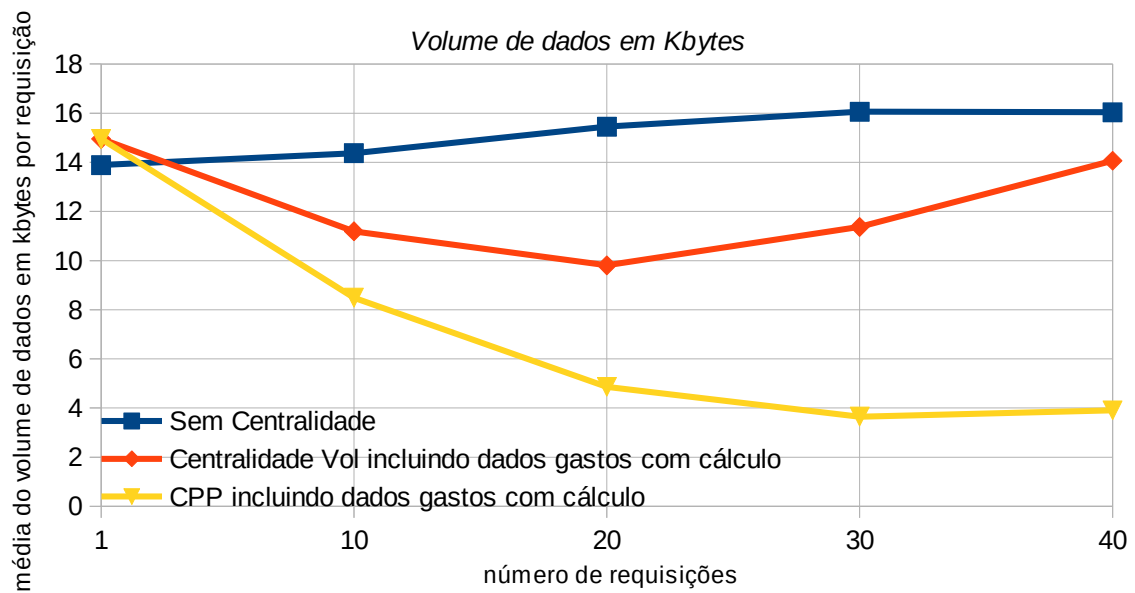
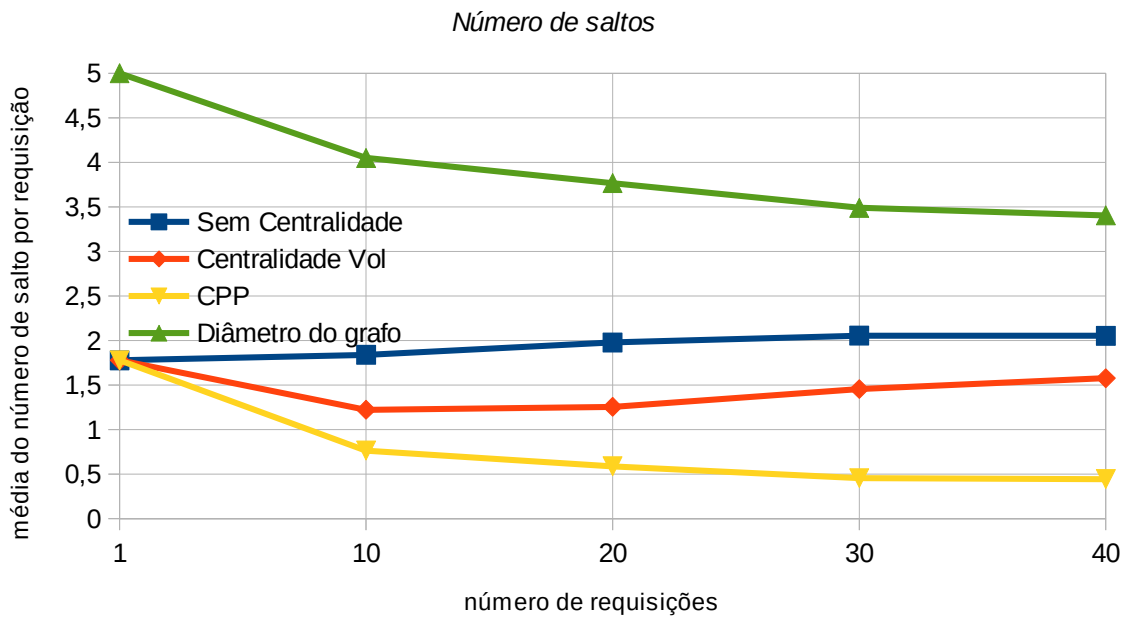


Figura 5.4: Efeito da variação no número de requisições com 2 nós acessando a réplica.
 (a) Distância média da réplica em saltos. (b) Volume de dados transmitidos.

Mesmo considerando que a quantidade de mensagens utilizadas para executar o algoritmo é grande (aproximadamente 260 em toda a simulação) o tamanho dessas mensagens são muito menores que o tamanho das mensagens utilizadas para enviar a réplica. Enquanto o algoritmo utiliza mensagens de aproximadamente 4 bytes para calcular o $P(i)$, a réplica neste cenário é de 7,81 Kbytes. De acordo com a Figura 5.4 (b), em 20 requisições tem-se uma economia de aproximadamente 70 % na sobrecarga do volume de dados necessários para obter a réplica. O método sem centralidade utiliza 15,45 Kbytes, o método CPP utiliza 4,86 Kbytes, e a medida que aumenta o número de requisições (até 40 requisições), esse número aumenta.

Nos gráficos da Figura 5.5, foi analisado o desempenho do algoritmo quando vários nós (5) obtêm a réplica e não somente dois nós, como no caso da Figura 5.4. A Figura 5.5 mostra os resultados obtidos.

Nas simulações da Figura 5.4, onde somente dois nós realizam as requisições, foi observada um decremento maior, em comparação com a Figura 5.5, com vários nós realizando as requisições. Isto acontece porque os nós sorteados para acessar a réplica estão distribuídos sobre a rede (o algoritmo de geração de números aleatórios utilizado possui distribuição uniforme) ficando o algoritmo de centralidade ponderada por potencial mais próximo dos resultados do método de *Vol*, onde somente a centralidade é considerada. Entretanto, se os nós estão concentrados em uma região do grafo, ou se o algoritmo utilizado para realizar o sorteio dos nós que acessam a réplica permite a repetição, o decremento continua considerável. Mesmo assim, o número de saltos necessários para acessar a réplica, utilizando a centralidade ponderada por potencial, continua sendo menor do que no método sem centralidade, Figura 5.5 (a). Isso também é válido para o volume de dados em Kbytes, como mostra a Figura 5.5 (b).

Outra razão para o gráfico da Figura 5.4 apresentar melhores resultados comparado com o gráfico da Figura 5.5, é o número de cópias da réplica. Em ambas as simulações é 2, e o número de nós acessando a réplica nas simulações da Figura 5.4 também é 2. Já, nas simulações da Figura 5.5, o número de nós acessando a réplica é 5.

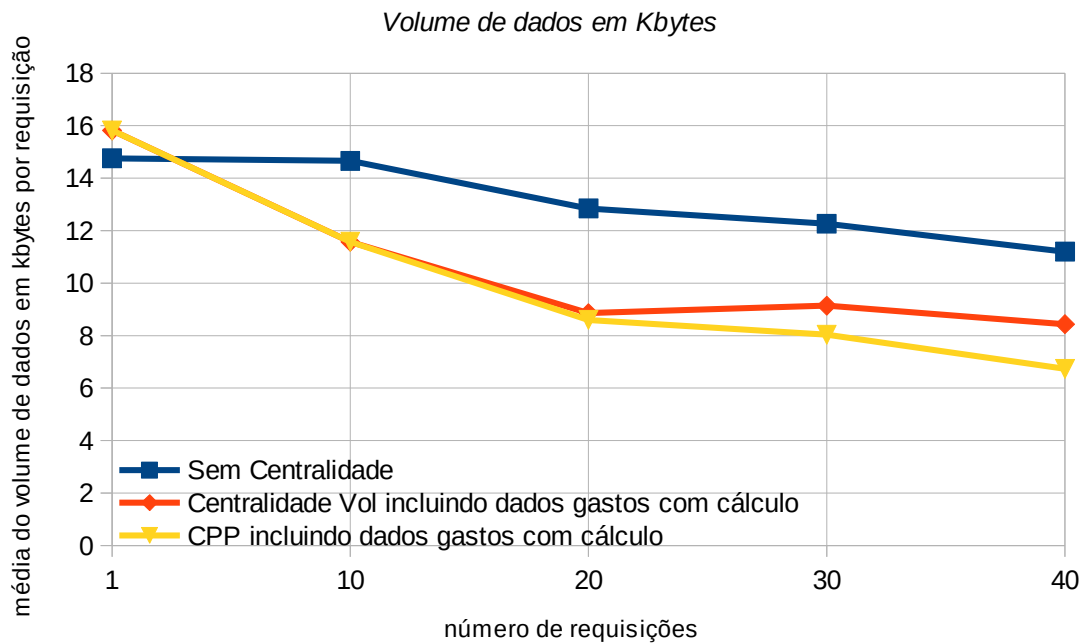
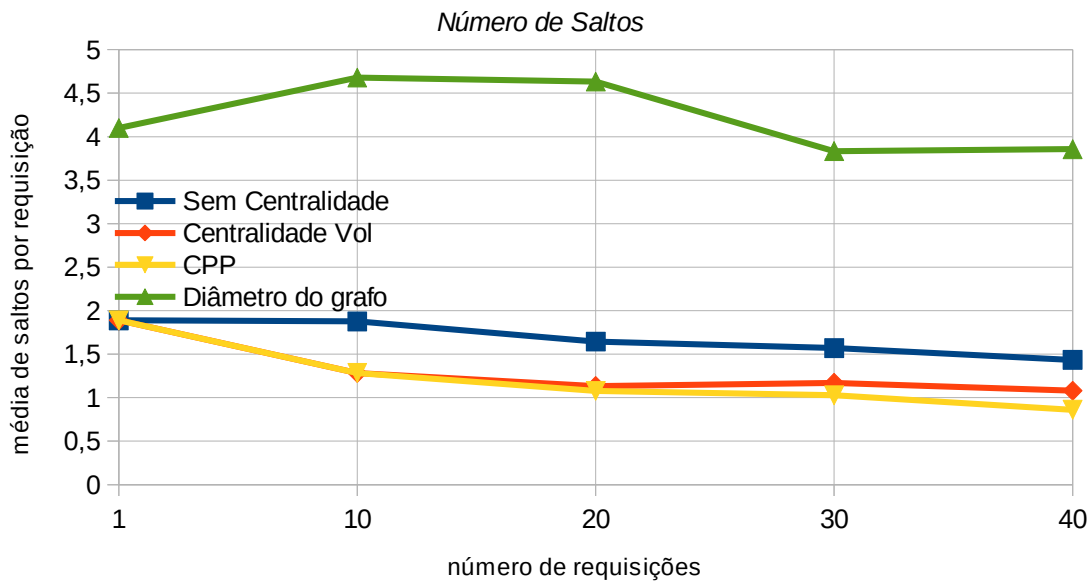


Figura 5.5: Efeito da variação no número de requisições com vários nós acessando a réplica. (a) Distância média da réplica em saltos. (b) Volume de dados transmitidos.

Se fosse aumentado o número de cópias da réplica para aproximadamente 5, os resultados de ambas as Figuras 5.4 e 5.5 seriam muito semelhantes, pois as réplicas poderiam se aproximar ainda mais dos nós que estão fazendo requisições às réplicas, diminuindo o número de saltos necessários e também o volume de dados transmitidos sobre a rede.

O algoritmo de centralidade ponderada por potencial também mostrou, de acordo com os resultados das Figura 5.4 (b), que em aproximadamente 10 requisições seu uso se torna viável, visto que o volume de dados (em Kbytes) já é menor em relação ao método sem centralidade.

De acordo com os resultados obtidos pode-se concluir que quanto maior o número de requisições para acesso à réplica, um menor volume de dados são necessários para alcançar a réplica e devido a isso, um menor número de nós são atravessados. Desta forma, o volume dos dados necessários para obter a réplica é inversamente proporcional ao número de requisições efetuadas.

Como esperado, isso demonstra que o algoritmo de *CPP*, no início, tem um volume maior de dados enviados, pois o algoritmo precisa que sejam trocadas mensagens entre os nós para a sua execução. Porém, com o aumento do número de requisições, o volume de dados enviados diminui, pois a réplica é atraída para o nó que tem maior força potencial. Ficando a réplica mais próxima, um menor volume de dados são enviados, tornando o acesso mais rápido à réplica, reduzindo o tráfego de mensagens no meio de comunicação, economizando tempo e recursos, do nó e da rede.

Como a velocidade aqui utilizada é na faixa de 0-3m/s, não houve mudanças significativas na topologia da rede no tempo das simulações aqui utilizadas. Mas, como a rede é móvel, caso fosse utilizado em um cenário onde a mobilidade fosse mais alta, a ponto de causar uma mudança considerável na topologia da rede, alternativas poderiam ser utilizadas para que o desempenho do algoritmo de *CPP* não fosse afetado de maneira significativa. Uma solução seria diminuir a força que a centralidade *Vol* tem para atrair a réplica, diminuindo sua influência no potencial. Uma alternativa seria analisar a

viabilidade da atualização da centralidade Vol , pois o nó calculado como mais central no início, muito provavelmente mudaria com o tempo. Para que o cálculo do Vol não acrescente um custo nas mensagens transmitidas, as mensagens utilizadas para encontrar o nó mais central poderiam ser enviadas de “carona”, nas requisições por exemplo, do cálculo do $P(i)$. Assim o cálculo seria realizado somente nos nós do caminho por onde as mensagens de requisições passam, reduzindo assim o número total de mensagens necessárias para o cálculo do Vol . A questão de mobilidade será tratada com maiores detalhes na Seção 5.4.4.

No gráfico da Figura 5.7 foi analisado o efeito da variação no número de requisições com 2 nós acessando a réplica em uma área retangular 600m x 100m. Com esta área é possível verificar com clareza a força de atração que um nó acessando a réplica possui. Como nestas simulações somente 2 nós solicitam a réplica e os mesmos encontram-se em lados opostos, como no exemplo da Figura 5.6, a força de atração dos nós fica evidente. A réplica encontra-se na parte central da área simulada e o nó está destacado com um quadrado.

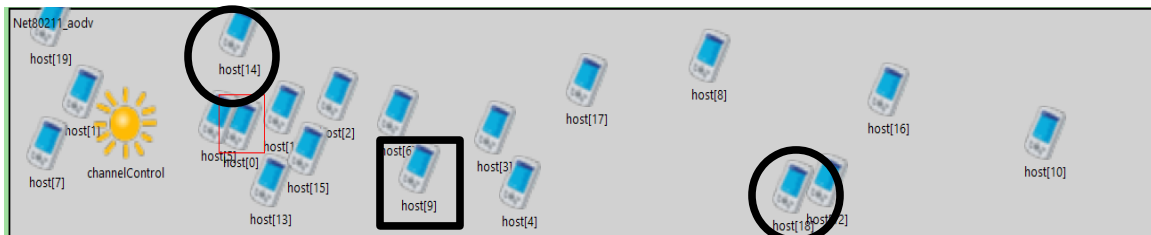


Figura 5.6: Área retangular 600m x 100m, com 2 nós em lados opostos acessando a réplica

E de acordo com o gráfico da Figura 5.7, pode-se perceber a força de atração que os nós que acessam a réplica, como por exemplo o nó 18 e 14 (Figura 5.6). Em 50 requisições o

número de saltos necessários para alcançar a réplica é zero porque as réplicas estão nos nós que acessam. Neste gráfico, a réplica 1 se deslocou 4 saltos, sendo um salto cada vez. Então, se deslocou 4 vezes até chegar ao nó que fazia as requisições. E a réplica 2 foi criada em aproximadamente 20 requisições. Apesar disso não poder ser observado no gráfico da Figura 5.7.

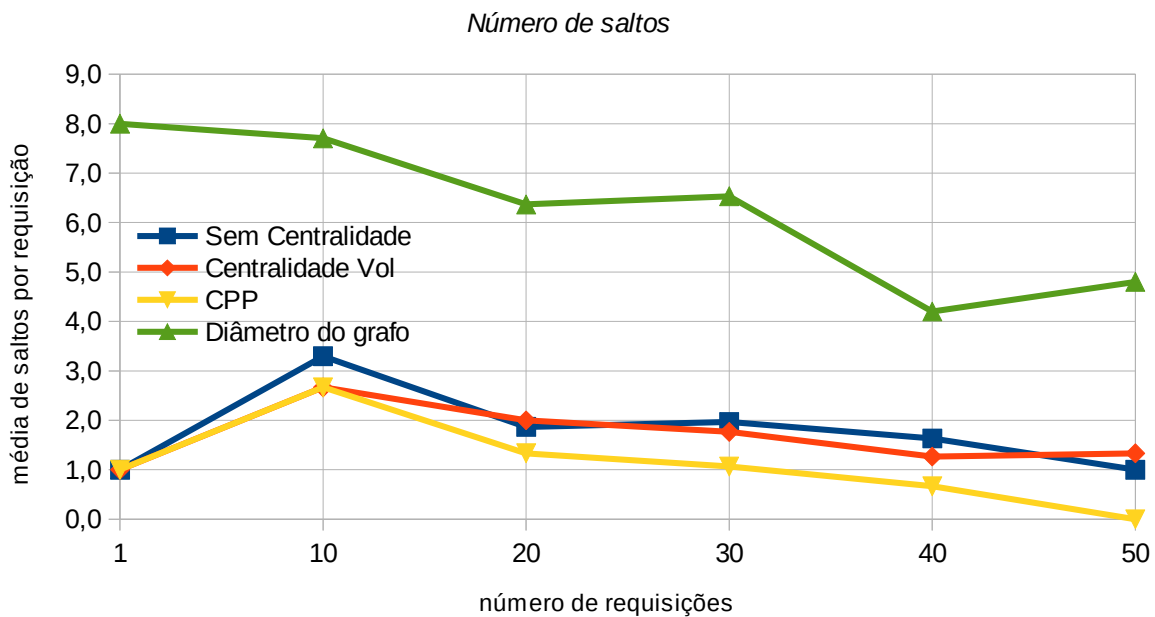


Figura 5.7: Efeito da variação no número de requisições com 2 nós acessando a réplica em uma área retangular 600m x 100m

5.4.3. Efeito da variação do alfa (α) - (Ajuste fino de α)

O parâmetro alfa permite ajustar o peso da centralidade e dos fatores dinâmicos (W) no cálculo do potencial. Quanto menor o valor de α , a centralidade *Vol* terá um peso

menor na fórmula do potencial de um nó. Já o $W(i)$, que representa as informações de contexto, terá uma influência maior. Isso acontece porque na Equação $P(i)$, o Vol é multiplicado por α e $W(i)$ é multiplicado por $1 - \alpha$. Com o objetivo de encontrar o valor ideal de α foram realizados experimentos variando o seu valor. A Figura 5.8 mostra o impacto da variação do parâmetro α no desempenho da CPP.

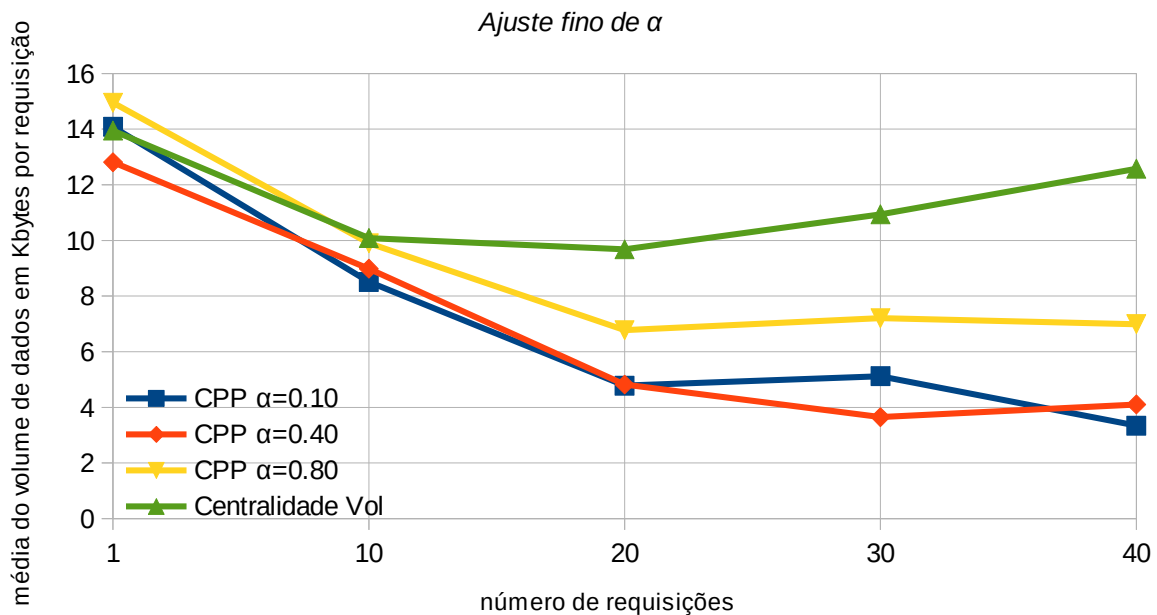


Figura 5.8: Efeito da variação do α na centralidade ponderada por potencial

Por padrão nas simulações anteriores foi utilizado $\alpha=0,40$. Se for definido um valor de α muito grande como, por exemplo, da Figura 5.8, $\alpha=0,80$, a centralidade ponderada por potencial terá resultados mais próximos da centralidade Vol . Já quando o valor de α diminui, como é o caso de $\alpha=0,40$, os resultados são melhores, e o volume de dados gastos por requisição tende a diminuir. Como é criada uma cópia da réplica mesmo que o valor de α diminua muito, como é o caso de $\alpha=0,10$, o volume de dados gastos por requisição permanece melhor do que $\alpha=0,80$. Assim, mesmo afastando a réplica do “centro do grafo”

e aproximando de um nó, existe outra réplica, e os nós restantes continuam acessando esta réplica próxima a eles. Também porque ao longo do tempo, com a mobilidade dos nós, a centralidade Vol (que é calculada no início das simulações) pode mudar e talvez o nó que foi calculado como mais central, ao longo do tempo das simulações, não seja o mais central.

5.4.4. Efeito do aumento da mobilidade dos nós

Nesta seção, foi analisado o desempenho da CPP quando aumentado a velocidade de deslocamento dos nós da rede na faixa de 0-3m/s para 0-6m/s. O objetivo é investigar o impacto da mudança da topologia da rede no desempenho do algoritmo. A Figura 5.9 (a) mostra os resultados desta simulação. Se comparada a velocidade de deslocamento dos nós na faixa de 0-3m/s, com a velocidade de deslocamento dos nós na faixa de 0-6m/s, com velocidade menor os resultados são melhores. Uma solução para o desempenho do algoritmo de CPP melhorar os resultados, quando a velocidade de deslocamento aumenta, é diminuir ou até mesmo zerar a influência que a centralidade Vol tem na fórmula $P(i)$. Pois esta provavelmente estaria desatualizada com o aumento da mobilidade dos nós, durante o decorrer da simulação.

A Figura 5.9 (b) mostra os resultados quando o $\alpha = 0,20$ para a mobilidade de 0-6m/s, ou seja, a influência que a centralidade Vol tem na fórmula $P(i)$ é diminuída quando a velocidade é maior. Assim, as informações de contexto, tais como a AF , que é dinâmica, tem um peso maior no cálculo do potencial de um nó. O problema de desconsiderar a centralidade Vol no cálculo de $P(i)$, é que se um nó iniciar fazendo requisições à réplica, e se este nó estiver na borda da rede e a réplica na outra borda da rede, ou seja, os nós que requisitam e o nó que tem a réplica estão distantes um do outro, o custo inicial para o nó acessar a réplica será grande, até a réplica deslocar-se mais próxima a este nó.

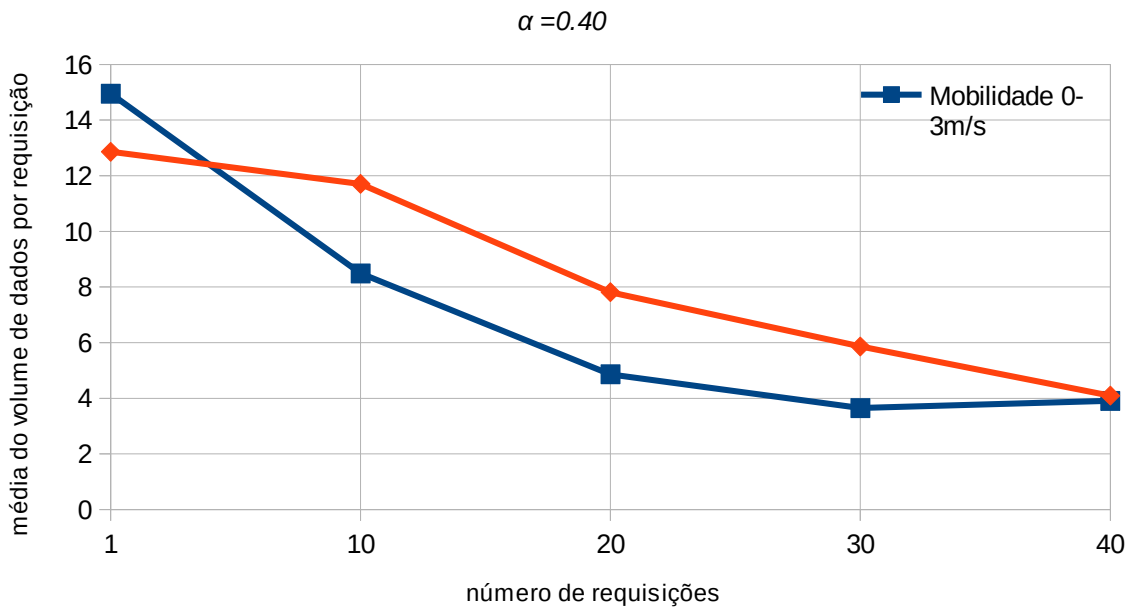
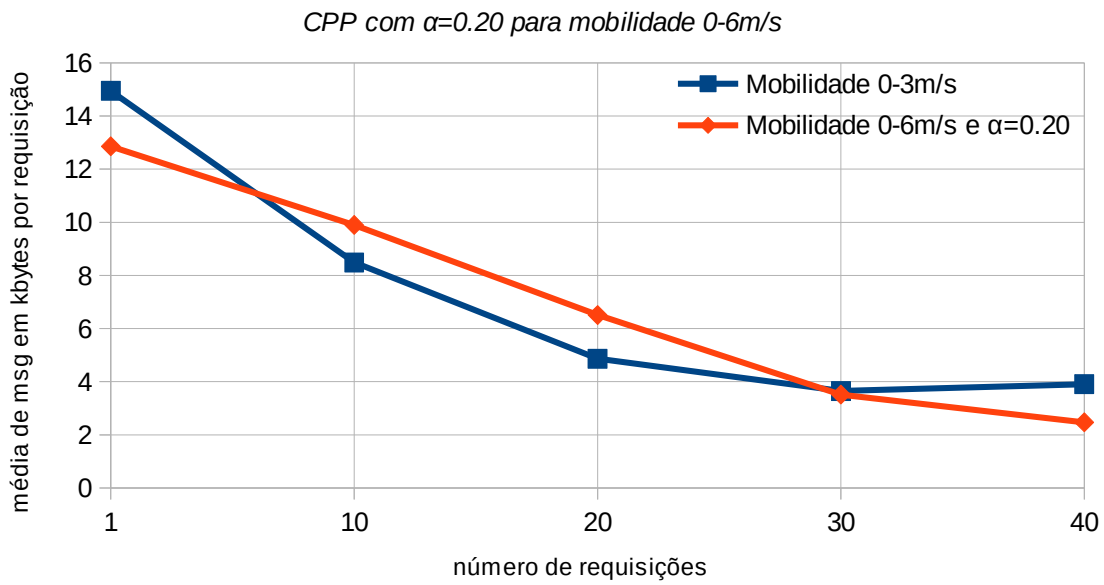


Figura 5.9: Aumento da mobilidade dos nós. (a) Com $\alpha=0,40$ (b) Com $\alpha=0,20$ para mobilidade 0-6m/s, diminuindo a influência da centralidade *Vol* no cálculo do potencial.

Então, uma solução para quando a mobilidade dos nós aumenta, seria recalcular a centralidade *Vol*. Recalcular a centralidade *Vol* periodicamente poderia levar a um aumento momentâneo no volume de dados transmitidos para realizar o cálculo, como ocorre no início da fase de alocação dos dados. Desta forma, surgem picos no volume de dados necessários para obter a réplica. Assim, pode-se adaptar o cálculo do *Vol* de maneira a torná-lo viável, uma vez que o valor pode alterar rapidamente quando a mobilidade dos nós é alta. Para tanto, uma alternativa poderia ser considerada: enviar as mensagens para cálculo do *Vol* com as mensagens de requisições das réplicas. Assim, a atualização da centralidade *Vol* seria calculada somente nos nós por onde a réplica passa até chegar no nó que solicitou a réplica (no caminho percorrido pela réplica) e não na rede inteira. Neste caso, as mensagens iriam de “carona”, aumentando em alguns bytes o tamanho das mensagens necessárias, porém não o número delas. Desta forma não surgiriam picos no volume de dados necessários para obter a réplica. Isso pode ser feito para tentar gastar menos bytes com o cálculo do *Vol*, mas também, para manter a centralidade *Vol* atualizada. Quando um novo nó acessa a réplica então o *Vol* deverá ser atualizado e este nó tem a chance de “atrair a réplica” próximo a ele.

Nestas abordagens é importante definir o intervalo de tempo necessário para recalcular a centralidade *Vol*. Este tempo dependerá da velocidade de deslocamento dos nós. Por exemplo, quando a velocidade é de 0-6m/s, a centralidade *Vol* poderia ser recalculada a aproximadamente a cada 20s - 30s, dado que neste tempo a topologia da rede, muito provavelmente teria mudanças consideráveis alterando significativamente o centro do grafo. Outra possibilidade é monitorar a vizinhança, quando o nó que tem a réplica percebe mudanças nos seus vizinhos, dispara o cálculo do *Vol*. Entretanto esta possibilidade teria uma desvantagem: pode ser que todos os vizinhos se desloquem na mesma direção ou então, os vizinhos troquem entre si a localização. Ou seja, nem sempre alterações na topologia implicam alterações no seu centro.

Estratégias como as citadas acima poderiam amenizar o problema de mudança considerável na topologia, típica em MANETs de alta mobilidade.

Outro problema que poderia ocorrer devido à alta mobilidade dos nós, é a desconexão de um ou mais nós na rede. Mesmo a predição da instabilidade sendo executada, em algum momento a desconexão dos nós pode inevitavelmente ocorrer. Assim, a centralidade da rede mudaria tendo a necessidade de recalcular a centralidade *Vol*. No caso de particionamento da rede, se a réplica estivesse armazenada em cada uma das partes onde ocorreu o particionamento, a partir deste momento, o algoritmo seria executado em cada componente conexo do grafo.

Conclui-se que, aumentando a mobilidade dos nós na rede, o algoritmo de *CPP* pode continuar com o desempenho semelhante, no que diz respeito a média de volume de dados que são gastos para acessar a réplica, alterando-se parâmetros na execução do algoritmo para adaptar com a dinamicidade da rede.

5.4.5. Efeito da acessibilidade dos dados

Nesta seção foi analisado o desempenho do algoritmo em relação a acessibilidade dos dados. Protocolos de replicação de dados têm como objetivo aumentar a acessibilidade dos dados [ATS09]. A acessibilidade dos dados é medida pelo número de requisições efetuadas com sucesso dividindo-se pelo número total de requisições [ATS09],[ATS07].

Os resultados da Figura 5.10 mostram que utilizando a centralidade ponderada por potencial são encontrados os nós que são os mais adequados para mover uma réplica ou criar uma nova. No grafo, o eixo horizontal indica os métodos utilizados para comparação (sem Centralidade e *CPP*) e o eixo vertical indica a taxa de sucesso nas requisições dos dados. A taxa de acessibilidade usando centralidade ponderada por potencial fica em aproximadamente 97% e melhora significativamente a acessibilidade em relação ao método sem centralidade, que é de 89%.

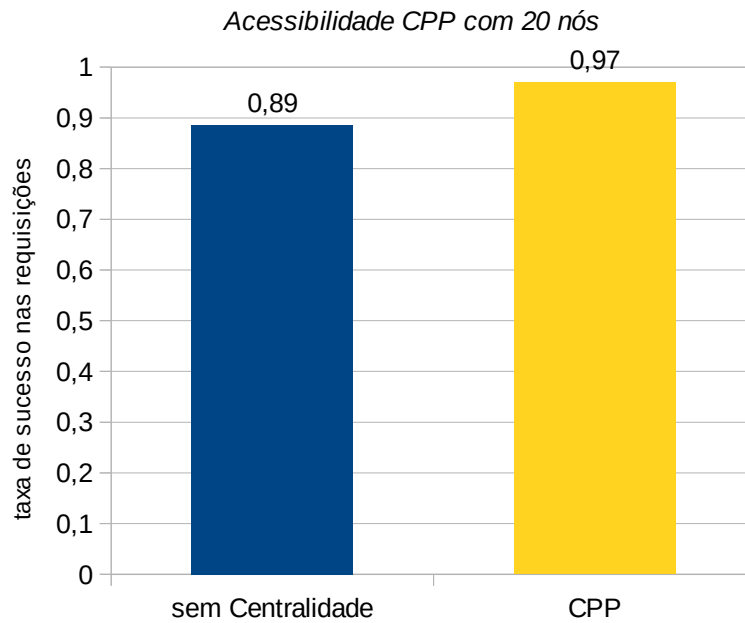


Figura 5.10: Acessibilidade dos dados com 20 nós

A centralidade *Vol* não está incluída nestes grafos, pois como neste método não tem predição da instabilidade, assim como no método sem Centralidade, os resultados seriam muito semelhantes.

Essas altas taxas de acessibilidade do método aqui proposto é devido ao fato dele suportar a predição de instabilidade de um nó, de tal forma que, se este nó se torna incapaz de manter a réplica, esta possivelmente será movida para outro nó mais estável. Foi considerado incapaz de manter e receber a réplica o nó que tivesse uma $IN = 5$, e a IN poderia variar de 1-5, onde 1 seria um nó estável e 5 um nó instável. A instabilidade de um nó depende do seu nível de bateria, da quantidade de memória e armazenamento de que dispõe, da sua sobrecarga de processamento e da sua velocidade de deslocamento. Esse valor foi definido aleatoriamente, sendo a faixa (por exemplo, 1-5) de conhecimento global dos nós da rede.

5.4.6. Efeito do aumento do número de nós– análise da escalabilidade

Outra avaliação de desempenho que foi analisada é em relação à escalabilidade do sistema [ATS08]. Essa métrica é o resultado do comportamento do sistema quando é aumentado o número de nós na rede. Escalabilidade é um critério importante, mas muitos protocolos de replicação em MANETs não a consideram. Muitos protocolos não tratam a escalabilidade do sistema visto que, eles propõem soluções que geram uma sobrecarga de mensagens muito alta na rede e à medida que aumenta o tamanho da rede, o número de mensagens aumenta demasiadamente, tornando o algoritmo inviável [ATS13].

Nas simulações aqui executadas, foram utilizadas redes com 20, 35 e 50 nós. Para manter a densidade da rede com o aumento do número de nós foi aumentada a área de simulação da rede para 350m x 350m (35 nós) e 400m x 400m (50 nós). A Figura 5.11 mostra a acessibilidade dos dados com 35 nós e a Figura 5.12 para 50 nós.

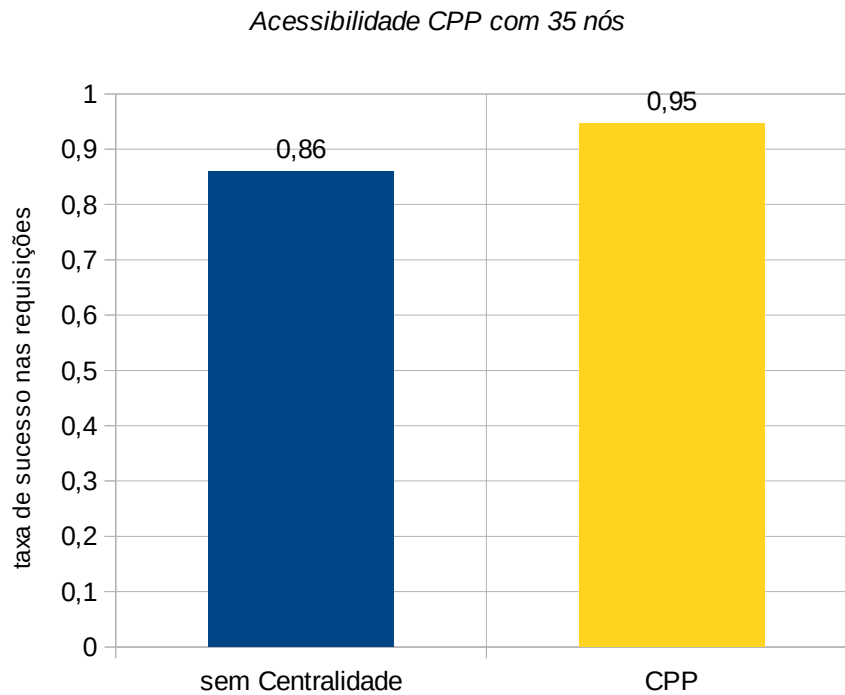


Figura 5.11: Acessibilidade dos dados com 35 nós.

Acessibilidade CPP com 50 nós

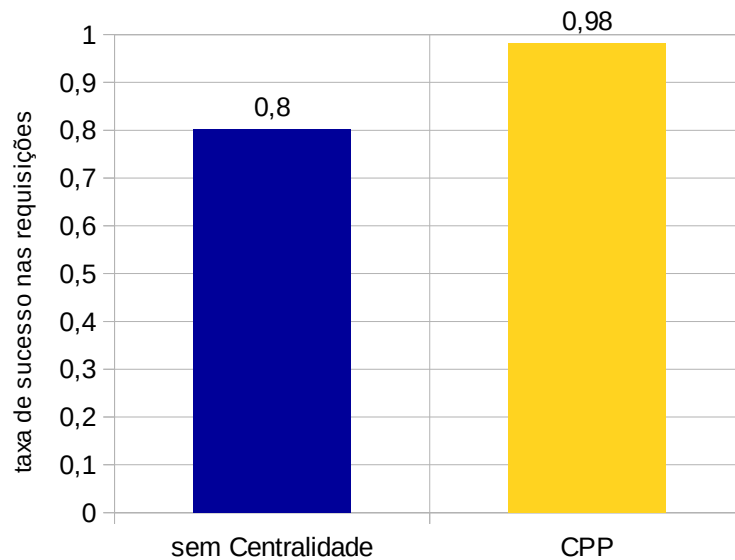


Figura 5.12: Acessibilidade dos dados com 50 nós.

De acordo com os resultados da Figura 5.13, o algoritmo de *CPP* mostra ter uma escalabilidade boa quando ocorre um aumento no número de nós na rede. Desta maneira, aumentando o número de nós não degrada o algoritmo de *CPP*, pois o desempenho deste, continua semelhante nos testes realizados. Desta forma, podemos dizer que o sistema é escalável em relação a acessibilidade dos dados.

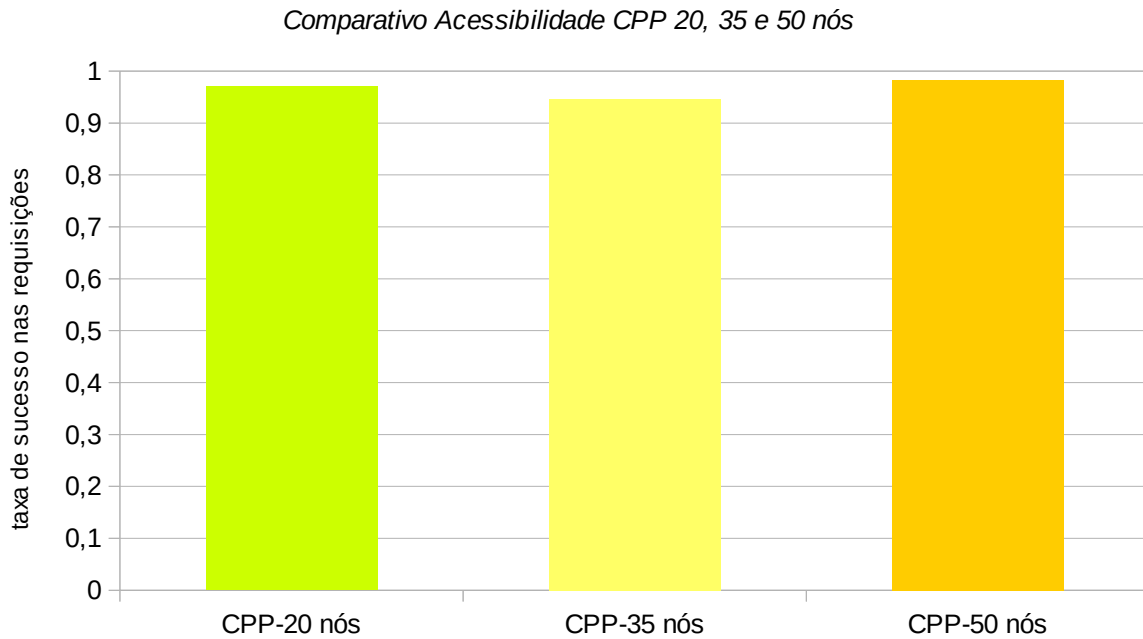


Figura 5.13: Comparativo da acessibilidade dos dados com 20, 35 e 50 nós na rede.

5.4.7. Efeito da variação do período para recalculer o Potencial (Ajuste fino do Potencial)

Finalmente, o efeito da variação do período para recalculer o potencial dos nós $P(i)$ é analisado. A Figura 5.14 mostra os resultados quando o potencial foi recalculado a cada 2s, 6s e 20s. Podemos observar que recalculando o potencial a cada 6s e 2s os resultados são melhores do que se calculado a cada 20s. Para os experimentos aqui realizados foi escolhido calcular o $P(i)$ a cada 6s, porque os resultados são semelhantes se comparado a realizar o cálculo cada 2s, porém consome mais energia e processamento do nó se o cálculo for realizado com uma frequência maior. Desta maneira, calculando o $P(i)$ a cada 6s, economiza-se energia do nó e tem-se bons resultados.

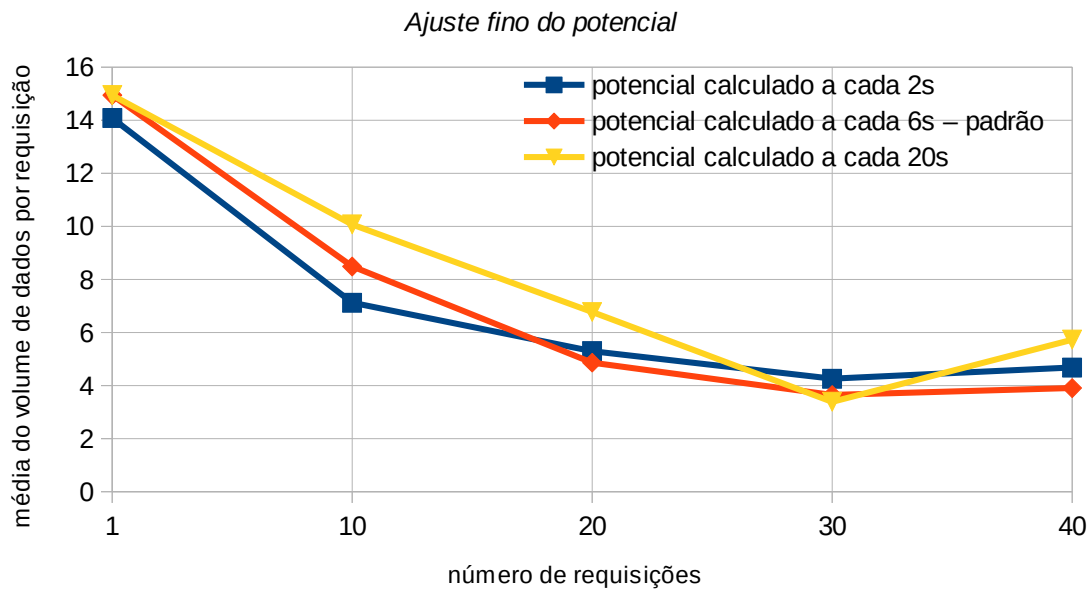


Figura 5.14: Ajuste fino do Potencial

5.5. Discussão

O método de replicação de dados proposto investigou a viabilidade da solução encontrada para os problemas existentes em MANETs e seus fatores dinâmicos. Os efeitos da mobilidade dos nós pode ser atenuada no método de replicação de dados se levado em conta a predição da instabilidade de um nó. Assim, se o nó torna-se impossibilitado de armazenar a réplica de um dado, esta réplica é transferida para outro nó que esteja apto a mantê-la, antes da réplica ficar inacessível. Com isso, a acessibilidade da réplica é mantida. Também, considerando a frequência de acesso de um nó à réplica e movimentando essa réplica mais próxima aos nós que acessam mais, diminui o número de saltos para acessar e, como resultado, um menor volume de dados são necessários para obter a réplica, economizando recursos do nó e da rede.

Os resultados das simulações mostram que, usando o método de replicação de dados, ocorre, na média, uma economia de mais de 70% na sobrecarga das mensagens que são geradas na rede para obter uma réplica, depois que o algoritmo estabiliza (aproximadamente em 10 requisições o volume de dados transmitidos diminui consideravelmente, devido também aos dados gastos para o cálculo da centralidade *Vol*) de acordo com a Figura 5.4 (b).

Diferenciando as duas fases de replicação de dados, como mostrado em um dos experimentos de alocação da réplica, houve uma economia de mensagens mesmo somando-se as mensagens que foram gastas para o cálculo do algoritmo da centralidade *Vol*. É evidente que a economia em volume de dados é menor pois são gastas uma quantidade considerável de mensagens, para realizar o cálculo dessa centralidade, se comparada com a fase de realocação. Mas, essa economia menor, em parte, é justificada, devido ao tempo de simulação da fase de alocação (aproximadamente 10s) em relação a fase de realocação (aproximadamente 30s). Mesmo que a réplica não tenha sido inicialmente alocada no nó central, se o algoritmo de centralidade ponderada por potencial for executado por um tempo suficientemente longo, o efeito seria muito semelhante, pois a réplica tenderia a movimentar-se em um nó mais central, visto que a centralidade *Vol* faz parte do cálculo da centralidade ponderada por potencial. Você não sabe disso.

Concluindo, a fase de alocação da réplica é viável se o tempo de simulação nesta fase for suficiente para economizar o volume de dados que foram gastos para o cálculo da centralidade *Vol*, e além disso, a mobilidade da rede não for alta, a ponto deste cálculo exigir ser atualizado periodicamente.

Nas simulações que avaliaram o aumento e o decréscimo de parâmetros tais como o α , é importante ressaltar que a definição de um valor que seja ideal para esses parâmetros, depende do tipo de aplicação. Por exemplo, se os nós tiverem uma velocidade de deslocamento superior a média de pedestres caminhando (foi utilizada velocidade de pedestre nas simulações aqui realizadas) o algoritmo de centralidade ponderada por

potencial deve considerar mais os fatores dinâmicos da rede, como frequência de acesso à réplica e instabilidade dos nós. Nas simulações aqui descritas, foi apresentado um exemplo onde poderiam ter variações destes parâmetros.

Conclusão

MANETs trazem consigo muitas oportunidades e inúmeros desafios. Se, por um lado, as possibilidades de comunicação abrem caminho para uma grande variedade de aplicações, por outro lado, a mobilidade dos nós em si e a escassez de seus recursos dificultam a disponibilidade de dados. Assim, muitos são os desafios para os pesquisadores da área. Este trabalho apresentou um novo modelo de posicionamento e ciclo de vida de réplicas de dados em MANETs baseado no conceito de campos magnéticos virtuais. Estes últimos são utilizados para quantificar fatores dinâmicos da rede (como por exemplo, frequência de acesso e instabilidade) sem ignorar, por outro lado, a tendência de manter réplicas próximas ao centro da topologia da rede. Para tanto, o domínio do problema foi apresentado juntamente com os conceitos que serviram de embasamento teórico e a análise de trabalhos relacionados. O modelo proposto foi implementado e avaliado em ambiente de simulação.

Este trabalho atingiu os objetivos aqui propostos: conceber e constatar a viabilidade de utilizar um algoritmo da métrica de centralidade para a alocação dos dados. O algoritmo para realocação dos dados utilizando campos magnéticos virtuais foi desenvolvido para determinar a distribuição das réplicas em função de características como frequência de uso e localização. Por último, foi cumprido o objetivo de avaliar os

resultados obtidos com o método de replicação de dados, o qual compreende os algoritmos distribuídos para a alocação e a realocação de dados através de métricas de desempenho predefinidas.

Apesar de existirem inúmeros trabalhos relacionados a esta proposta, não foi encontrado na literatura um método que utilize o conceito de campos magnéticos virtuais a fim de distribuir as réplicas de acordo com mudanças no contexto do ambiente. Também se observou que, nenhum deles diferencia a etapa de alocação e realocação de dados, utilizando a mesma técnica da primeira etapa, na segunda etapa, com modificações que se ajustem às características individuais em cada uma delas. Para o melhor de nosso conhecimento, não foi encontrado trabalhos na literatura da aplicação de centralidade ponderada por potencial em técnicas de replicação de dados para MANETs. O modelo de centralidade ponderada por potencial foi definido a partir da exploração do problema, em um estudo bibliográfico aprofundado que permitiu identificar a possibilidade de melhorar o desempenho de um método de replicação. Também, foram pesquisados e estudados algoritmos que poderiam distribuir as réplicas na rede de acordo com parâmetros iniciais. Diversos algoritmos de grafos foram encontrados e que poderiam ser utilizados para esta finalidade. Este estudo também foi importante para o aprendizado de grafos e como os seus algoritmos poderiam ser aplicados em redes de computadores de forma a aumentar o desempenho do sistema. Também foi validado campos magnéticos em redes móveis, visto que esse conceito ainda não tinha sido testado com mobilidade.

Outra contribuição alcançada, o formalismo matemático do modelo proposto, foi capaz de representar a solução em um contexto formal e que possibilite a organização das informações e uma melhor descrição do problema. O algoritmo definido e simulado é a principal contribuição deste trabalho. O código será disponibilizado podendo ser melhor explorado futuramente e até mesmo realizado comparações com outros métodos. A definição de coeficientes como o k e o α permitem flexibilidade ao modelo e a utilização com pequenas adaptações em cenários diferentes ao proposto. O foco principal do estudo

foi a definição de um modelo formal, conciso e flexível. Embora os resultados alcançados já sejam promissores, ajustes finos para a utilização do algoritmo em diferentes cenários ainda poderiam ser feitos, abrindo novos caminhos para pesquisas.

Ainda, como contribuição pode-se citar o estudo do custo de comunicação do algoritmo desenvolvido. Cabe aqui ressaltar a importância dessa análise. Estimar o custo é importante para verificar se a solução é viável. Como em MANETs, o número de conexões simultâneas de cada nó é tipicamente baixo e limitado, mostramos que o algoritmo proposto tem complexidade linear com pequenas constantes multiplicativas, o que é desejável [CUN12]. Em MANETs, isso se torna ainda mais importante, visto a limitação de recursos dessa rede.

Por fim, apesar de menor amplitude, outra contribuição obtida foi a revisão bibliográfica, que poderá ser utilizada para novas pesquisas e trabalhos relacionados nas áreas abordadas neste trabalho: MANETs, replicação de dados, medidas de centralidade e campos magnéticos virtuais.

Este projeto resultou em dois trabalhos apresentados⁴ em eventos, os quais tiveram como objetivos obter um *feedback* sobre o modelo proposto, encaminhamentos para a continuidade do trabalho, comprovação da sua eficiência e também para a escrita de outros artigos.

Nos trabalhos futuros, pretende-se realizar outras simulações e avaliar os resultados obtidos. Muitos são os parâmetros que poderão ser modificados e que novos resultados surgirão. Pode-se aplicar o método em problemas e situações que estão além do escopo desta pesquisa e também outras métricas poderão ser avaliadas. Como se observou, a solução de realocação dos dados através de campos magnéticos virtuais é

⁴ Michelon, Gisane A.; Lima, Luiz A. P.; Oliveira Jr., J. Aélío; Calsavara, Alcides; de Andrade Gil. *A Strategy for Data Replication in Mobile Ad Hoc Networks*. Proceedings of IEEE 22nd International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, Paris, 2014.

Michelon, Gisane A.; Lima, Luiz A. P. *Replicação distribuída de dados em redes ad hoc móveis (MANETs)*. Trabalho apresentado no 3º Congresso Sul Brasileiro de Iniciação Científica e Pós-Graduação, Curitiba, 2014.

uma solução de baixo custo, em termos de sobrecarga de mensagens. Então, poderia ser testada em redes de larga escala, em cenários distintos da proposta desta tese, a fim de verificar seu desempenho, também neste tipo de rede. Pretende-se expandir o alcance dos resultados aqui obtidos. Por exemplo, como ajustar os parâmetros para adequarem-se a cenários específicos, ou como a mobilidade afeta o desempenho do sistema em redes maiores e mais dinâmicas. Algumas simulações poderão ser ajustadas para calcular o algoritmo e melhorar os custos quando ocorre mudança na topologia da rede. Assim, poderá ser necessário recalculer o Vol (ou calcular ele em momentos diferentes) e também calcular o $P(i)$ com maior frequência. Com estas modificações poderá ser avaliado se, com uma maior movimentação dos nós, o algoritmo continua viável. Como simulações futuras também pretende-se testar o algoritmo em funcionamento por um tempo maior, a fim de comprovar se os resultados se mantêm. Porém a tendência é que mesmo com um tempo de simulação maior os resultados permaneçam semelhantes aos apresentados nas simulações.

A comparação deste trabalho foi realizada utilizando uma abordagem comparativa semelhante de [ATS07], [JIN04]. Estes métodos de replicação de dados comparam seu desempenho escolhendo um nó aleatório para armazenar a réplica e esta alocação ocorre de forma estática. Desta forma, um estudo comparativo detalhado foi considerado como trabalho futuro.

Logo na sequência, como continuidade da parceria entre orientador e aluno, e em expansão dos objetivos aqui propostos, será estudada a viabilidade de implementação no mesmo ambiente de simulação da centralidade ponderada por potencial, a comparação com um ou mais métodos de replicação de dados citados nos trabalhos relacionados, por exemplo [ATS07], [PUS14].

Referências Bibliográficas

- [ATS07] ATSAN, E.; OZKASAP, O. *Applicability of eigenvector centrality principle to data replication in MANETs*. 22nd International Symposium on Computer and Information Sciences (ISCIS 2007), p.1-6, 2007.
- [ATS09] ATSAN, E.; ALTINBUKEN, D.; OZKASAP, O. *SCALAR data replication performance in mobile ad hoc applications*. 24th International Symposium on Computer and Information Sciences (ISCIS 2009), p. 369-374, 2009.
- [ATS13] ATSAN, E.; OZKASAP, O. *SCALAR: Scalable data lookup and replication protocol for mobile ad hoc networks*. *Computer Networks, Vol. 57, No. 17*, p. 3654–3672, 2013.
- [BAI04] BAI, F.; HELMY, A. *A Survey of Mobility Models*. Chapter 1 in *Wireless Adhoc Networks*, Kluwer Academic Publishers, *Vol. 2, No. 5*, p. 1-30, 2004.
- [BAL07] BALDAUF, M.; DUSTDAR, S.; ROSENBERG, F. *A Survey on Context-aware Systems*. *International Journal of Ad hoc and Ubiquitous Computing, Vol. 2, No. 4*, p. 263-277, 2007.

- [BAR11] BARBOSA, C. *Blog LinkPBNet - Tecnologia e Nada Mais*. Disponível em <http://linkpb.net/?m=20110519>. Postado em 19.05.2011. Acessado em 10.07.2012.
- [BLE10] BLECHA, R; MICHELON, G. A. *Replicação de dados para redes sem fio*. XIX Encontro Anual de Iniciação Científica - EAIC, 2010.
- [BON87] BONACICH, P. *Power and Centrality: A Family of Measures*. American Journal of Sociology, Vol. 92, No. 5, p. 1170-1182, 1987.
- [BRA01] BRANDES, U. *A faster algorithm for betweenness centrality*. Journal of Mathematical Sociology, Vol. 25, No. 2, p. 163–177, 2001.
- [BRI15] BRITO, J. M. *Redes IP*. Disponível em http://www.inforede.net/Technical/Layer_3_and_4/Network/Paper20Rede20IP.pdf. Acessado em 21.01.2015.
- [BRI98] BRIN, S.; PAGE, L. *The Anatomy of a Large-Scale Hypertextual Web Search Engine*. Seventh International World-Wide Web Conference (WWW 1998), Brisbane, Australia, 1998.
- [CAM02] CAMP, T.; BOLENG, J.; DAVIES V. *A Survey of Mobility Models for Ad hoc Network Research*. Wireless Communications & Mobile Computing, Vol. 2, p. 483-502, 2002.
- [CHE00] CHEN, G.; KOTZ, D. *A Survey of Context-Aware Mobile Computing Research*. Technical Report. Dartmouth College, Hanover, NH, USA. Dept. of Computer Science, 2000.

- [CHE14] CHEN, H.; ZHANG, Y.; CHENG G.; LIU B; LEI X. *Social-Based Hybrid Dissemination Strategy in Heterogeneous Opportunistic Networks*. Pervasive Computing and the Networked World. Vol. 8351, p. 15-26, 2014.
- [CLA03] CLAUSEN, T.; JACQUET, P. *Optimized link state routing protocol (OLSR)*. IETF Request for Comments - RFC 3626, outubro 2003.
- [COO02] COOK, S. A.; PACHL, J.K.; PRESSMAN, I. S. *The Optimal Location of Replicas in a Network Using a Read-One-Write-All Policy*. Distribute Computing, Vol.15, No.1, p. 57-66, 2002.
- [CUN12] CUNHA I. Apostila de Análise de Complexidade. Universidade Federal de Minas Gerais, 2012. Disponível em <http://homepages.dcc.ufmg.br/~cunha/teaching/20121/aeds2/complexity.pdf>. Acessado em 12.2014.
- [DEL10] DELICATO, F. C. *Middleware de Nova Geração*. Disponível em <http://www.dimap.ufrn.br/~flavia.delicato/MiddlewareNovaGeracao2008.pdf>. Acessado em 11.01.2010.
- [DER09] DERHAB, A.; BADACHE N. *Data Replication Protocols for Mobile Ad-Hoc Networks: A Survey and Taxonomy*. IEEE Communications Surveys & Tutorials, Vol. 11, No. 2, p. 33-51, 2009.
- [DRE12] DREIBHOLZ, T.; ZOLTAN BOJTHE Z.; MAUREIRA, J.; QUINTANA, A. A.; JONSSON, K.; BORBÉLY, T.; MÉSZÁROS, L.; YOUSAF, F. Z.; JANOTA, V.; SOMMER, C. *INET Framework for OMNEST/OMNeT++*. Disponível em <https://github.com/inet-framework/inet>. Acessado em 08/09/2012.

- [FRE10] FREITAS, L. Q. *Medidas de centralidade em grafos*. Dissertação do Programa de Pós-graduação em Engenharia de Produção, COPPE, da UFRJ, Disponível em http://objdig.ufrj.br/60/teses/coppe_m/LeandroQuintanilhaDeFreitas.pdf. Acesso em 14.08.2013.
- [FRE79] FREEMAN, L. C. *Centrality in social networks: Conceptual clarification*. *Social Networks*, Vol. 1, No. 3, p. 215-239, 1979.
- [GAN12] GANZERT, C. C. *Desenvolvimento Sistêmico, Equidade e Interdependência: A Busca por um Modelo Conceitual de Gestão do Equilíbrio das Relações entre agentes econômicos regionais*. Tese do Programa Pós-Graduação em Administração de Organizações - Faculdade de Economia, Administração e Contabilidade de Ribeirão Preto da Universidade de São Paulo, 2012.
- [GUE10] GUELLATI, N.; KHEDDOUCI, H. *A survey on self-stabilizing algorithms for independence, domination, coloring, and matching in graphs*. *Journal of Parallel Distributed Computing*, Vol. 70, No. 4, p. 406-415, 2010.
- [HAD06] HADIM, S.; AL-JAROODI, J.; MOHAMED, N. *Middleware Issues and Approaches for Mobile Ad hoc Networks*. *Proceedings 3rd IEEE Consumer Communications Networking Conference (CCNC)*, Vol. 1, p. 431-436, 2006.
- [HAN14] HANNEMAN, R. A.; RIDDLE, M. *Introduction to social network methods*. University of California, Disponível em http://www.faculty.ucr.edu/~hanneman/nettext/C10_Centrality.html#Bonacich. Acessado em 02.12.2014.

- [HAR05] HARA, T. *Data Replication Issues in Mobile Ad hoc Networks*. Proceedings Sixteenth International Workshop on Database and Expert Systems Applications, p. 753-757, 2005.
- [HAR06] HARA, T; MADRIA, S. K. *Data Replication for Improving Data Accessibility in Ad hoc Networks*. IEEE Transactions on Mobile Computing, Vol. 5, No.11, p. 1515-1532, 2006.
- [HAR08] HARRAS, K. A. *Challenged Networks - Protocol and Architectural Challenges in Delay and Disruption Tolerant Networks*, VDM Verlag, 2008.
- [HAR09] HARA, T; MADRIA, S. K. *Consistency Management Strategies for Data Replication in Mobile Ad hoc Networks*. IEEE Transactions on Mobile Computing, Vol. 8, No. 7, p. 950-967, 2009.
- [HER11] HEREK, T. A. *Análise Comparativa de Desempenho dos Protocolos de Roteamento em Redes Móveis Ad Hoc*. Monografia de Especialização em Teleinformática e Redes de Computadores, Universidade Tecnológica Federal do Paraná, 2011.
- [HUN13] HUNG-CHIN, J.; LEE P., *Social Aware Assisted Transmission in MANET*. Proceedings of the Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing. p. 342-347, 2013.
- [JIN04] JING, Z.; JINSHU, S.; KAN, Y.; YIJIE, W. *Stable Neighbor Based Adaptive Replica Allocation in Mobile Ad hoc Networks*, Computational Science (ICCS), Vol. 3036, p. 373-380, 2004.

- [KAW06] KAWASAKI, Y.; MATSUMOTO, N.; YOSHIDA, N. *Popularity- Based Content Replication in Peer-to-Peer Networks*, Computational Science (ICCS), Vol. 3994, p. 436-443, 2006.
- [KAT12] KATSAROS, K. *Data Management in Mobile Environments*. Athens University of Economics and Business. Disponível em <http://pages.cs.aueb.gr/~katsaros/>. Acesso em 27.04.2012.
- [KON14] KONAK, A., *Improving Network Connectivity in Emergency Ad Hoc Wireless Networks*. The 11th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2014). The Pennsylvania State University, University Park, 2014.
- [LIM09] LIMA, M. N.; SANTOS, A. L.; PUJOLLE, G. *A Survey of Survivability in Mobile Ad hoc Networks*. IEEE Communications Surveys and Tutorials, Vol. 11, No.1, p. 66-77, 2009.
- [LIM10] LIMA, L. A.; CALSAVARA, A. *Autonomic Application-Level Message Delivery Using Virtual Magnetic Fields*. Journal of Network and Systems Management, Vol. 18, No.1, p. 97-116, 2010.
- [LOU09] LOUREIRO, A.; OLIVEIRA, R.; SILVA, T.; PIRES, W. R. J.; OLIVEIRA, L.; MOREIRA, R.; SIQUEIRA, R.; ROCHA, S.; RUIZ, L. B. *Computação Ubíqua Ciente de Contexto: Desafios e Tendências*. 27º Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos. Livro texto dos minicursos, Cap. 3, p. 99-149, 2009.

- [LUI15] LUÍS, J. C. M. *IP - Internet Protocol*. Disponível em <http://www.jonny.eng.br/trabip/ip.html>. Acessado em 20.01.2015.
- [MAG09] MAGALHÃES, P. A. S. *Compressão de Cabeçalhos de routing em Redes Ad Hoc*. Disponível em [\http://paginas.fe.up.pt/~ee08059/pdf/PDI_relatorio_progresso.pdf. Acessado em 27.11.2014.
- [MIC14] MICHELON, G. A.; LIMA, L. A. P.; OLIVEIRA, J. A.; CALSAVARA, A.; DE ANDRADE G. *A Strategy for Data Replication in Mobile Ad Hoc Networks*. Proceedings of IEEE 22nd International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, Paris, 2014.
- [NAS12] NASCIMENTO, J. P.; MURTA, C. D. *Um Algoritmo Paralelo em Hadoop para Cálculo de Centralidade em Grafos Grandes*. XXX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, Ouro Preto, MG, 2012.
- [OLI10] OLIVERIA, G., LOUREIRO, A.; CAMPOS, M. *Replicação de dados em redes Ad Hoc*. Disponível em http://homepages.dcc.ufmg.br/~gabriel/publications/Conference/ERRC_2010.pdf. Acessado em 11.11.2014.
- [OKA08] OKAMOTO, K.; CHEN, W; LI, X. *Ranking of Closeness Centrality for Large-Scale Social Networks*. Proceedings of the 2nd annual international workshop on Frontiers in Algorithmics, p. 186-195, 2008.

- [PAR97] PARK, V. D.; SCOTT CORSON, M. *A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks*. IEEE INFOCOM'97, Vol. 3, p. 1405-1413, 1997.
- [PAU15] PAULA, F. B. Datagramas IP (Protocolo Internet). Disponível em <http://www.vivaolinux.com.br/artigo/Datagramas>. Acessado em 21.10.2015.
- [PER03] PERKINS, C.; BELDING-ROYER, E., DAS, S. *Ad hoc On-Demand Distance Vector (AODV) Routing*, IETF Request for Comments - RFC 3561, julho 2003.
- [PLA11] PLASS, W. *Greater sustained throughput and contention elimination in IEEE 802.11 with DS-CDMA*. Dissertação de mestrado em ciências, Iowa State University, 2011. Disponível em <http://lib.dr.iastate.edu/cgi/viewcontent.cgi?article=2994&context=etd>. Acessado em 10.01.2015.
- [PUS14] PUSHPALATHA, M; RAMARAO, T.; VENKATARAMAN, R. *Applicability of sub graph centrality to improve data accessibility among peers in MANETs*. Peer-to-Peer Networking and Applications, Vol. 7, No. 2, p. 129-146, 2014.
- [SAB66] SABIDUSSI, G. *The centrality index of a graph*. Psychometrika, Vol. 31, No. 4, p. 581-603, 1966.
- [SAI05] SAITO, Y.; SHAPIR, M. *Optimistic Replication*. ACM Computing Surveys, Vol. 37, No. 1, p. 42-81, 2005.
- [SIL10] SILES, R. *Wireless Forensics: Tapping the Air - Part One*. Disponível em <http://www.symantec.com/connect/articles/wireless-forensics-tapping-air-part-one>. Acessado em 13.01.2015.

- [VAR08] VARGA, A.; HORNIG, R. *An overview of the OMNeT++ simulation environment*, Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops (Simutools '08), p. 1-10, 2008.
- [VAR15] VARGA, A. *OMNeT++ Discrete Event Simulator*. Disponível em <http://omnetpp.org/>. Acessado em 10.02.2015.
- [WEH12] WEHMUTH, K.; ZIVIANI, A. *Distributed assessment of the closeness centrality ranking in complex networks*. Proceedings of the Fourth Annual Workshop on Simplifying Complex Networks for Practitioners, p. 43-48, 2012.
- [WEH13] WEHMUTH, K.; ZIVIANI, A. *DACCER: Distributed Assessment of the Closeness Centrality Ranking in complex networks*. Computer Networks, Vol. 57, No. 13, p. 2536–2548, 2013.