



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

**Proposta para Alocação de Canais e para Comunicação
Cooperativa em Redes Ad Hoc**

Thiago Fernandes Neves

Brasília
2014



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

Proposta para Alocação de Canais e para Comunicação Cooperativa em Redes Ad Hoc

Thiago Fernandes Neves

Dissertação apresentada como requisito parcial
para conclusão do Mestrado em Informática

Orientador
Prof. Dr. Jacir Luiz Bordim

Brasília
2014

Universidade de Brasília — UnB
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Mestrado em Informática

Coordenador: Prof. Dr. Ricardo Pezzuol Jacobi

Banca examinadora composta por:

Prof. Dr. Jacir Luiz Bordim (Orientador) — CIC/UnB

Prof. Dr. Renato Mariz de Moraes — ENE/UnB

Prof.^a Dr. Alba Cristina Magalhães Alves de Melo — CIC/UnB

CIP — Catalogação Internacional na Publicação

Neves, Thiago Fernandes.

Proposta para Alocação de Canais e para Comunicação Cooperativa em
Redes Ad Hoc / Thiago Fernandes Neves. Brasília : UnB, 2014.

75 p. : il. ; 29,5 cm.

Dissertação (Mestrado) — Universidade de Brasília, Brasília, 2014.

1. Alocação de múltiplos canais; Comunicação cooperativa; Controle
de topologia; Economia de energia; Redes *ad hoc*; Redes sem fio.

CDU 004.7

Endereço: Universidade de Brasília
Campus Universitário Darcy Ribeiro — Asa Norte
CEP 70910-900
Brasília-DF — Brasil

Agradecimentos

Agradeço aos meus familiares e amigos pelo apoio, ao meu orientador pela dedicação e companherismo e aos membros da banca examinadora pela análise crítica e sugestões construtivas. Agradeço também à CAPES pela bolsa de estudos que foi fundamental para que este trabalho pudesse ter sido concluído com sucesso.

Resumo

A popularização de tecnologias sem fio, aliado com aplicações que exigem conexão contínua e altas taxas de transmissão, impulsionam o desenvolvimento de protocolos de Controle de Acesso ao Meio (do Inglês, *Medium Access Control – MAC*) eficientes em energia. Mecanismos que permitem melhorar o desempenho da rede utilizando a disponibilidade de múltiplos canais de comunicação têm sido explorados na literatura. No entanto, desenvolver protocolos eficientes em energia que permitam realizar a atribuição de canais e agendamento de comunicação, melhorando o desempenho da rede, tem sido uma tarefa desafiadora. Neste contexto, a primeira parte dessa dissertação propõe um protocolo de alocação de canais e de agendamento de comunicação para redes sem fio, chamado EEMC-MAC, que permite reduzir o consumo de energia e o tempo de comunicação. A segunda parte dessa dissertação possui seu foco em mecanismos para melhorar a conectividade em redes *ad hoc*. Nesse contexto, Comunicação Cooperativa (CC) é utilizada para explorar a diversidade espacial na camada física e permitir que múltiplos nós cooperem na transmissão de um sinal para um mesmo receptor. Uma vez que CC pode reduzir a potência de transmissão e estender o raio de transmissão, a técnica tem sido combinada com protocolos de controle de topologia em redes *ad hoc*. Os primeiros trabalhos de controle de topologia em redes *ad hoc* cooperativas buscam aumentar a conectividade da rede, enquanto o consumo de energia é minimizado em cada nó. Trabalhos posteriores focam na eficiência das rotas criadas na topologia final. No entanto, a nosso conhecimento, nenhum trabalho até então explorou CC para aumentar a conectividade com o sorvedouro em redes *ad hoc*. Na segunda parte dessa dissertação, é proposta uma nova técnica, chamada *CoopSink*, que utiliza CC e controle de topologia em redes *ad hoc* para aumentar a conectividade com um nó sorvedouro, além de garantir a eficiência das rotas para o sorvedouro.

Palavras-chave: Alocação de múltiplos canais; Comunicação cooperativa; Controle de topologia; Economia de energia; Redes *ad hoc*; Redes sem fio.

Abstract

The popularization of wireless technology allied with high throughput and continuous Internet access applications has boosted the development of energy efficient Medium Access Control (MAC) protocols. Mechanisms to improve network performance using the availability of multiple communication channels have been explored in the literature. However, the development of energy efficient protocols to perform channel allocation and data scheduling to improve the network performance is a challenging task. In this context, the first part of this dissertation proposes a protocol, named EEMC-MAC, for multi-channel allocation and data scheduling for wireless networks that allows the reduction of energy consumption and communication time. The second part of this dissertation focuses on techniques to improve connectivity in ad hoc networks. In this context, Cooperative Communication (CC) is employed to explore spatial diversity in the physical layer, allowing multiple nodes to cooperatively relay signals to the receiver so that it can combine the received signals to obtain the original message. Once CC can be used to reduce the power of the transmission node and extend the transmission range, the technique has been combined with topology control protocols in wireless ad hoc networks. Early works in topology control in cooperative ad hoc networks aimed to increase network connectivity while minimizing energy consumption in each node. Later works focused in route efficiency in the final topology. Nevertheless, to the best of our knowledge, no work so far explored CC to increase connectivity to a sink node in wireless networks. As a second contribution of this work, a new technique named CoopSink is proposed, that uses CC and topology control in ad hoc networks to increase connectivity to a sink node, while ensuring efficient routes.

Keywords: Ad hoc networks; Cooperative communication; Energy saving; Multi-channel allocation; Topology control; Wireless networks.

Sumário

1	Introdução	1
1.1	Justificativa	3
1.2	Objetivos e Contribuições	3
1.3	Metodologia	4
1.4	Estrutura do Documento	4
2	Revisão Teórica	5
2.1	Redes sem Fio e Protocolos de Acesso ao Meio	5
2.2	Classificação dos protocolos MAC em redes <i>Ad Hoc</i>	7
2.2.1	Protocolos Livres de Contenção	8
2.2.2	Protocolos Baseados em Contenção	9
2.3	Protocolos MAC para Múltiplos Canais	9
2.4	Introdução à Comunicação Cooperativa (CC)	12
2.5	Controle de Topologia e Comunicação Cooperativa em Redes Ad Hoc	14
2.6	Discussão	15
3	Alocação Eficiente em Energia em Redes Sem Fio com Múltiplos Canais	16
3.1	Contextualização e Motivação	16
3.2	Modelo de Comunicação	18
3.3	O Problema de Atribuição de Canais (CAP)	19
3.3.1	ECO: Uma Heurística para o Problema de Atribuição de Canais	19
3.3.2	ECO: Complexidades Envolvidas	21
3.4	Descrição da Proposta	21
3.4.1	Rotinas para Agrupamento de Conjuntos de Transmissão	22
3.4.2	Detalhes do Protocolo EEMC-MAC	23
3.4.3	EEMC-MAC: Procedimento Principal e Complexidades Envolvidas	25
3.4.4	EEMC-MAC: Exemplo de Funcionamento	26
3.5	Simulação	28
3.5.1	Parâmetros de Simulação e Métricas de Avaliação	29
3.5.2	Resultados de Simulação	29
3.6	EEMC-MAC para Múltiplos Saltos	31
3.7	Discussão	32
4	Controle de Topologia em Redes <i>Ad Hoc</i> Cooperativas	33
4.1	Contextualização e Motivação	33
4.2	Modelo e Definição do Problema	34
4.2.1	Modelo de Comunicação Cooperativa	35

4.2.2	Modelo de Rede Cooperativa	36
4.2.3	Exemplificação do Modelo	39
4.2.4	Formulação do Problema	42
4.3	Descrição da Proposta	42
4.3.1	Algoritmo: <i>Greedy Helper Set Selection</i>	42
4.3.2	<i>CoopSink</i> : Descrição da Técnica	46
4.4	Resultados de Simulação	48
4.4.1	M1: Conectividade com o sorvedouro	51
4.4.2	M2: Potência média de transmissão	51
4.4.3	M3: ESF para o sorvedouro	52
4.4.4	M4: Número de arestas CC	54
4.5	Discussão	55
5	Conclusão	57
	Referências	58

Lista de Figuras

1.1	Medidas de utilização do espectro [1].	2
2.1	Ilustração dos problemas de terminal exposto e escondido.	7
2.2	Diagrama de espaço-tempo com colisão na transmissão entre dois nós.	8
2.3	Exemplo de funcionamento do protocolo <i>MMAC</i>	11
3.1	Exemplo de grafo de comunicação com $n = 4$ nós.	19
3.2	Fluxograma do protocolo EEMC-MAC.	27
3.3	Exemplo de grafo de comunicação com $n = 8$ nós.	28
3.4	Possível representação gráfica para o uso de 4 canais de comunicação durante a execução do protocolo EEMC-MAC para o grafo de comunicação representado pela Figura 3.3.	28
3.5	Porcentagem de tempo do protocolo em estágio de transmissão para várias configurações (métrica α).	30
3.6	Comparação entre o tempo do estágio de transmissão pelo tempo ótimo (métrica γ).	31
4.1	(a) Cenário onde existem 3 nós próximos (v_1, v_2 e v_3) e um nó distante (v_4). (b) Aumento do raio de cobertura de v_1 com a utilização nós ajudantes (v_2 e v_3) para criar um enlace cooperativo entre v_1 e v_4	36
4.2	ESF - <i>Energy Stretch Factor</i> de um par de nós $v_i, v_j \in V$ em $G'_P = (V, E'_P)$ com respeito a $G_P = (V, E_P)$, $E'_P \subseteq E_P$, G'_P e G_P conexos. $\rho_{G'_P}^{G'_P}(v_i, v_j) = \frac{\min(\pi_{G'_P}(v_i, v_j))}{\min(\pi_{G_P}(v_i, v_j))}$ representa a diferença entre o custo do caminho mais eficiente em energia entre os nós v_i e v_j no grafo G'_P e o caminho mais eficiente entre esses mesmos nós em G_P	39
4.3	(a) Exemplo de topologia de rede, onde $ V = 50$ e $ E_P = 0$. (b) Grafo formado a partir de (a) quando os nós operam em sua potência máxima P_{MAX} . (c) Grafo criado a partir de (b) adicionando todas as arestas CC possíveis. (d) COE-t-S criada a partir de (c) inserindo as arestas CC e com $t = 1, 4$	49
4.4	Resultados de simulação para a métrica M1.	52
4.5	Resultados de simulação para a métrica M2.	53
4.6	Resultados de simulação para a métrica M3.	54
4.7	Resultados de simulação para a métrica M4.	55

Lista de Tabelas

1.1	Lista das contribuições feitas durante o período de mestrado.	4
2.1	Técnicas para alocação de múltiplos canais em redes <i>ad hoc</i>	10
2.2	Técnicas que utilizam comunicação cooperativa.	14
3.1	Tabela de notações.	20
4.1	Condições necessárias e análises que precisam ser feitas para se criar uma aresta CC $\widetilde{v_1v_4}$ para o grafo exemplo da Figura 4.1a	42

Lista de Algoritmos

3.1	: ECOH(G, k)	21
3.2	: CombineGroup(g_i, c_i)	22
3.3	: CombineTS(g_i, C)	23
3.4	: ManagementStage(V, C)	24
3.5	: TransmissionStage	26
4.1	: <i>GreedyHelperSetSelection</i> ($v_i, N(v_i), v_j$)	43
4.2	: <i>CoopSinkStep1</i> (V, P_{MAX})	46
4.3	: <i>CoopSinkStep2</i> ($\overline{G_P}$)	47
4.4	: <i>CoopSinkStep3</i> (G_P, t, v_o)	48
4.5	: <i>CoopSinkStep4</i> (G'_P)	50

Capítulo 1

Introdução

Nos últimos anos, a indústria eletrônica tem concentrado esforços no desenvolvimento de dispositivos sem fio cada vez menores e com maiores recursos de processamento, conectividade e multimídia. Esse investimento está muito ligado ao barateamento dos processos de produção e à popularização das redes sociais, como *facebook* e *twitter*. Dispositivos sem fio utilizam faixas do espectro de radiofrequências para comunicação e possuem o objetivo primordial de minimizar o consumo de energia, uma vez que, em geral, operam por meio de baterias. No entanto, o crescimento desenfreado no número de dispositivos e de usuários resultou em um sério problema: as faixas de radiofrequências destinadas para esses dispositivos estão, cada vez mais, congestionadas e esse congestionamento resulta em uma baixa qualidade de serviço para o usuário [2].

Embora as faixas de radiofrequências, como a banda *ISM* (do Inglês, *Industrial, Scientific and Medical*) que é utilizada no WiFi [3], estejam com a utilização saturada, outras faixas são pouco utilizadas, como aponta o estudo feito por *Cabric et. al* [1]. Nesse estudo, foi feito um sensoriamento de diversas faixas de radiofrequências e mediu-se o quão congestionadas essas faixas estavam. Uma das principais constatações foi que, como ilustrado pela Figura 1.1, determinadas faixas de radiofrequências, principalmente abaixo de 3GHz, são amplamente utilizadas para transmissão de dados, enquanto outras faixas, como acima de 3GHz, são pouco utilizadas. Isso ocorre pois as faixas de radiofrequências são divididas estaticamente para diversas entidades e aplicações, como para rádio, televisão, telefonia, satélites, uso militar, etc. Algumas das faixas mais saturadas são justamente as utilizadas para comunicação móvel. Estudos posteriores aos feitos por *Jia et. al* confirmaram essa tendência [4].

Devido ao uso desigual do espectro de radiofrequências, notou-se a necessidade de utilizá-lo de maneira dinâmica. Em um estudo pioneiro, *Mitola* [5] propôs rádios inteligentes, ou rádios cognitivos, que, utilizando um *hardware* especial, são capazes de reconfiguração autônoma, aprendendo e se adaptando ao ambiente de comunicação. Dessa maneira, identificou-se que o acesso oportunista ao espectro eletromagnético poderia ser uma das muitas aplicações para rádios cognitivos. Essa abordagem de acesso oportunista busca alocar “buracos”, ou oportunidades, nas faixas do espectro licenciado para um usuário oportunista, ou usuário secundário, de maneira que o usuário primário, licenciado, não seja prejudicado. Essa idéia resultou no conceito de redes de acesso oportunista (do Inglês, *Opportunistic Spectrum Access – OSA*) [6].

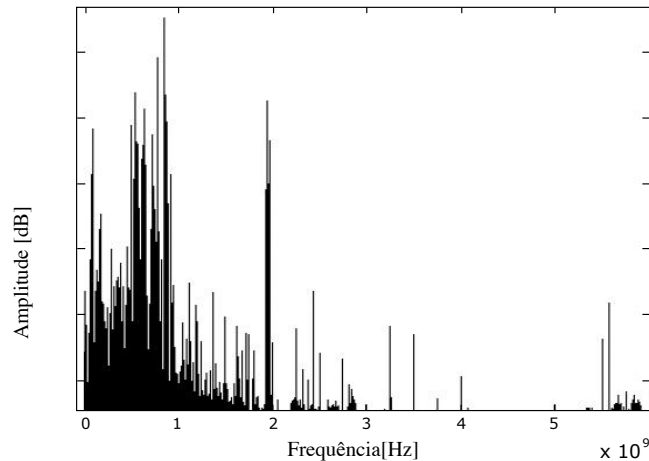


Figura 1.1: Medidas de utilização do espectro [1].

Existem muitas classificações e debates quanto aos componentes básicos de uma rede *OSA*. Segundo [6], redes *OSA* possuem os seguintes componentes básicos:

- **Identificação oportunista de espectro:** Um canal de comunicação pode ser considerado disponível para um usuário oportunista quando ele não está sendo utilizado por seus usuários primários. Um componente de identificação oportunista de espectro é responsável pela identificação desses canais, que são dinâmicos tanto espacial quanto temporalmente. Questões específicas incluem: *(i)* identificação de um nível máximo aceitável de interferência; *(ii)* sensoriamento inteligente dos canais para minimizar o consumo de energia; e *(iii)* processamento de sinais e técnicas para detecção de oportunidades;
- **Exploração oportunista de espectro:** Esse tipo de componente recebe como entrada as oportunidades detectadas e decide se uma transmissão deve ser feita. Questões específicas incluem: *(i)* como transmitir um dado; *(ii)* qual modulação (ou qual canal) e qual potência de transmissão (relacionado ao alcance do sinal) usar; e *(iii)* como compartilhar as oportunidades de maneira justa;
- **Política regulatória:** Define as regras de cooperação e uso conjunto entre usuários primários e secundários. Deve estar de acordo com as agências reguladoras de telecomunicações.

Cada um desses componentes possuem muitos desafios em aberto e diversas oportunidades de pesquisa. Por esse motivo, a primeira parte desse trabalho foca em técnicas para **exploração oportunista de espectro**. Mais especificamente, considera-se o cenário onde um conjunto de canais são identificados por um mecanismo de sensoriamento e a tarefa de realizar a alocação desses canais. A segunda parte desse trabalho consiste em tratar cenários muito comuns em redes sem fio, que é o caso da rede perder conectividade. Dessa forma, estuda-se a utilização de uma técnica chamada comunicação cooperativa, que pode ser utilizada para preservar ou mesmo aumentar a conectividade da rede. As seções que seguem apresentam a justificativa, os objetivos, a metodologia de trabalho.

1.1 Justificativa

Com o crescimento do número de dispositivos sem fio, a demanda por acesso contínuo e de alta velocidade à Internet e a dificuldade em se obter faixas de radiofrequências adicionais, resultou a necessidade de melhor utilizar os recursos de espectro existentes para minimizar a interferência na comunicação. Com a exploração oportunista do espectro, o problema de interferência seria atenuado, uma vez que permitiria uma melhor utilização de faixas até então sub-utilizadas e, conseqüentemente, aumentaria a vazão da rede. Com isso, o espectro seria melhor utilizado, permitindo qualidade de serviço às aplicações que exigem cada vez mais recursos de banda, como vídeo, fotos e jogos.

Fazer o acesso oportunista de maneira eficiente em energia também é importante pois dispositivos móveis, tipicamente, possuem uma quantidade limitada de energia. Os ganhos em economizar energia são inúmeros: maior tempo de utilização do dispositivo, maior vida útil da bateria, possível diminuição do tamanho dos dispositivos, entre outras vantagens. No entanto, mesmo utilizando técnicas para aumentar a vida útil das baterias, os dispositivos eventualmente irão ficar sem carga, podendo causar a desconexão da rede. Dessa forma, estudar mecanismos para preservar ou mesmo recuperar a conexão em redes sem fio também é uma área de pesquisa promissora. Com isso, existiria um ganho na confiabilidade da rede, além de reduzir os custos de manutenção, caso algum dispositivo tenha problema e precise ser substituído. Um cenário onde esse estudo seria desejável pode ser encontrado no projeto da torre observatório da Amazônia ou projeto ATTO (do Inglês, *Amazon Tall Tower Observatory*). Nesse projeto, pretende-se implantar uma grande torre no coração da floresta amazônica e, com a ajuda de sensores menores e estrategicamente posicionados, obter estimativas de fontes de gases estufa como CO_2 , CH_4 e N_2O [7]. Dessa forma, caso algum sensor menor apresente problema e torne uma parte da rede desconexa, a rede poderia se auto-organizar para que a conectividade seja mantida.

1.2 Objetivos e Contribuições

Essa dissertação é dividida em duas partes que se relacionam por estarem inseridas no contexto de redes sem fio, onde um dos principais objetivos é minimizar o consumo de energia nos dispositivos. A primeira parte consiste em levantar o estado da arte das técnicas que podem ser utilizadas para alocação de múltiplos canais de maneira eficiente em energia em redes sem fio. Após isso, propor uma nova técnica para esse cenário, que possua vantagens em relação ao que existe na literatura. Como objetivos secundários, pretende-se avaliar os seguintes tópicos: (i) como realizar a alocação dos canais de comunicação; (ii) como fazer a coordenação e agendamento de transmissões de dados; (iii) como medir o consumo de energia dos dispositivos e avaliar se esse consumo é adequado.

A segunda parte dessa dissertação consiste em estudar como a técnica de comunicação cooperativa pode ser utilizada para manter ou aumentar a conectividade de uma rede sem fio. Mais especificamente, pretende-se propor uma nova técnica para o cenário onde existe um nó sorvedouro e diversos outros nós espalhados pela rede que precisam reportar informações para o nó sorvedouro. Como objetivos secundários, pretende-se avaliar os seguintes tópicos: (i) qual a vantagem da técnica em relação ao que existe na literatura; e (ii) qual o custo para se realizar essa tarefa. A Tabela 1.1 sumariza as principais contribuições feitas pelo autor durante o período de mestrado.

Tabela 1.1: Lista das contribuições feitas durante o período de mestrado.

Autores	Título	Conferência/Revista	Situação	Ano/Ref
Neves, Thiago F.; Bordim, Jacir L.	<i>Topology Control in Cooperative Ad hoc Networks</i>	<i>Journal Electronic Notes in Theoretical Computer Science – ENTCS</i>	Aceito	2014 [8]
Neves, Thiago F.; Bordim, Jacir L.	<i>EEMC-MAC: An Energy Efficient Protocol for Multi-Channel Wireless Networks</i>	<i>The Eighth International Conference on Systems and Networks Communications – ICSNC2013</i>	Publicado (Best Paper)	2013 [9]
Neves, Thiago F.; Bordim, Jacir L.	<i>An energy efficient multi-channel protocol for wireless ad hoc networks</i>	<i>XXXVIII Conferencia Latinoamericana En Informatica – CLEI2012</i>	Publicado	2013 [10]
Jr, Juarez; Almeida, Lucas; Modesto, Felipe; Neves, Thiago; Weigang, Li	<i>An investigation on repost activity prediction for social media events</i>	<i>Web Information Systems Engineering – WISE 2012</i>	Publicado	2012 [11]
Neves, Thiago F.; Caetano, Marcos F.; Bordim, Jacir L.	<i>An energy optimal technique for multi-channel allocation and data scheduling in wireless networks</i>	<i>XXXVIII Conferencia Latinoamericana En Informatica – CLEI2012</i>	Publicado	2012 [12]
Neves, Thiago F.; Caetano, Marcos F.; Bordim, Jacir L.	<i>An Energy-Optimum and Communication-Time Efficient Protocol for Allocation, Scheduling and Routing in Wireless Networks</i>	<i>26th IEEE International Parallel and Distributed Processing Symposium Workshops – IPDPSW2012</i>	Publicado	2012 [13]

1.3 Metodologia

A metodologia desse trabalho consistiu inicialmente no estudo das principais técnicas de exploração oportunista de espectro, controle de topologia e comunicação cooperativa em redes sem fio. Em seguida, foram coletados dados de desempenho dessas técnicas em cenários similares aos utilizados na literatura por meio de simulação. Após a coleta e análise dos dados, foram identificados pontos que poderiam ser melhorados e novas abordagens foram propostas. Essas novas abordagens foram validadas por meio de simulações e análise da complexidade dos algoritmos. Relatórios técnicos foram produzidos em todas as etapas do processo para auxiliar a organização das ideias e na escrita da dissertação.

1.4 Estrutura do Documento

O restante dessa dissertação é organizada como segue. O Capítulo 2 faz uma breve revisão do estado da arte sobre redes sem fio, protocolos de acesso ao meio, protocolos para alocação de múltiplos canais e comunicação cooperativa. O Capítulo 3 é baseado no trabalho descrito em [9], discute sobre alocação eficiente em energia de múltiplos canais, apresenta uma nova proposta para esse cenário e demonstra as vantagens dessa proposta. O Capítulo 4 é baseado no trabalho descrito em [8], discute como comunicação cooperativa pode ser utilizada com controle de topologia para obter redes conexas e propõe uma nova técnica que possui vantagens em relação ao que existe na literatura em um determinado cenário. O Capítulo 5 conclui a dissertação.

Capítulo 2

Revisão Teórica

Este capítulo faz uma breve revisão teórica dos principais elementos e ideias básicas necessárias para compreensão dessa dissertação. A Seção 2.1 discorre sobre redes sem fio e seus protocolos de acesso ao meio. A Seção 2.2 classifica os protocolos de acesso ao meio. A Seção 2.3 apresenta alguns protocolos de acesso ao meio para múltiplos canais. A Seção 2.4 descreve uma técnica da camada física em redes sem fio chamada “comunicação cooperativa”. A Seção 2.5 trata do controle de topologia e de como essa técnica pode ser combinada com comunicação cooperativa.

2.1 Redes sem Fio e Protocolos de Acesso ao Meio

Dispositivos sem fio se comunicam através de ondas eletromagnéticas confinadas em uma faixa de frequência, ou canal, previamente alocado para este fim. O número de canais e a capacidade de cada dispositivo depende principalmente da tecnologia que é usada. Por exemplo, o padrão de redes sem fio *IEEE 802.11* [14], ou popular *WiFi*, opera na faixa entre 2.4GHZ e 5GHZ, também conhecida como banda *ISM* (do Inglês, *Industrial, Scientific and Medical applications*). Quando um dispositivo opera em uma região de frequência, ou espectro, ele está sujeito à interferência de outros dispositivos que operam na mesma frequência. Redes sem fio podem possuir ou não intra-estrutura, são classificadas em [15]:

- **Redes sem fio com infraestrutura:** Esse tipo de rede utiliza um ponto de controle central, como uma estação rádio base (ERB), que gerencia o envio e recebimento de dados de e para um hospedeiro sem fio associado à ela. Exemplos de redes com infraestrutura são redes de celular ou pontos de acesso que utilizam *WiFi*;
- **Redes sem fio sem infraestrutura:** Nesses tipos de redes, também conhecidas por redes *ad hoc*, os dispositivos sem fio não dispõem de uma infraestrutura para se conectar. Como em redes *ad hoc* não existe infraestrutura e o alcance do rádio é limitado, a comunicação é feita por meio de cooperação entre os nós dessa rede. Essa cooperação requer que os nós encaminhem pacotes de outros nós, agindo assim como roteadores.

Redes sem fio utilizam enlaces de difusão para troca de dados. O uso desse tipo de enlace resulta, potencialmente, em vários dispositivos transmissores e vários receptores

conectados no mesmo canal de comunicação. Esse canal de comunicação, comumente, é chamado de canal Multi-acesso, onde vários dispositivos podem se conectar nele ao mesmo tempo. Protocolos de controle de acesso ao meio (do Inglês, *Medium Access Control - MAC*) são protocolos que definem regras que tornam possível o compartilhamento justo e eficiente dos canais multi-acesso, contornando os diversos problemas inerentes à natureza da comunicação sem fio. A seguir, as principais características que devem ser consideradas na elaboração de protocolos *MAC* são descritas [16]:

1. **Alcance da Transmissão de Rádio:** O alcance de rádio é determinado pela distância máxima que um transmissor e um receptor podem ficar um do outro, de maneira que a conexão tenha potência suficiente para trocar dados. A potência do sinal cai rapidamente com o aumento da distância entre o transmissor e o receptor. No entanto, mesmo se um nó receptor estiver dentro do alcance do rádio do transmissor, é possível que ele fique incapaz de receber pacotes se o nível de interferência em um canal estiver alto.
2. **Reuso de Frequência:** Dadas as características do alcance de um rádio, temos que um mesmo canal pode ser usado simultaneamente em duas localizações diferentes que estão suficientemente distantes uma da outra, desde que a interferência entre um e outro estejam dentro de um nível aceitável. Essa característica é chamada de reuso de frequência.
3. **Teste de Portadora:** O teste de portadora é um mecanismo usado em vários protocolos *MAC* para redes sem fio [17]. Consiste no processo de determinar se um canal de comunicação está ocupado ou ocioso. O teste de portadora é realizado comparando a potência do sinal recebido no canal com um limite. Esse limite é escolhido tal que o receptor possa detectar um sinal transmitido de uma distância maior do que o seu alcance de transmissão de rádio. Uma vez que a força do sinal em um meio sem fio decai em proporção ao quadrado da distância do transmissor, a presença de um sinal no nó receptor pode não ser claramente detectada em outros nós, como se eles estivessem fora de alcance.
4. **Problema do Terminal Exposto e Escondido:** Para ilustrar a ocorrência do problema do terminal exposto e escondido utilizaremos como exemplo o cenário da Figura 2.1. Nota-se que o nó B está dentro do raio de alcance dos nós A e C , mas os nós A e C estão fora do alcance um do outro. Considere o caso onde o nó A está transmitindo para B . O nó C , estando fora do raio de A , não pode testar a portadora e pode, eventualmente, enviar uma transmissão para B , causando uma colisão. Esse problema se chama **problema do terminal escondido**, uma vez que os terminais A e C estão escondidos um do outro. Agora considere o caso onde o nó B está transmitindo para A . Desde que C está dentro do raio de B , ele testa a portadora e decide adiar sua própria transmissão para um outro nó, digamos D que está fora do alcance de B e A . No entanto, isso é desnecessário, pois, não existe maneira da transmissão do nó C para D causar uma colisão no receptor A . Esse problema é chamado de **problema do terminal exposto** [17], uma vez que o fato de B ser exposto para C levou esse último a adiar desnecessariamente sua transmissão.

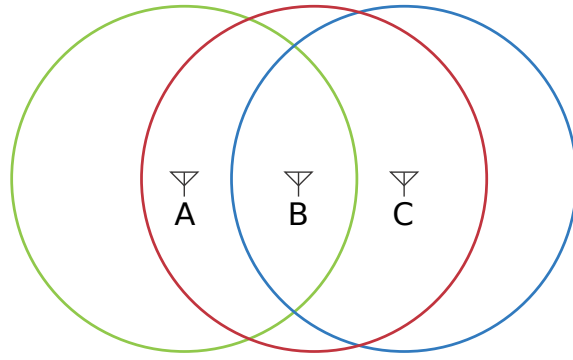


Figura 2.1: Ilustração dos problemas de terminal exposto e escondido.

5. **Colisão e Justiça:** Um dos principais problemas dos protocolos que utilizam canais de multi-acesso são as colisões. Uma colisão ocorre quando dois, ou mais, dispositivos transmitem dados ao mesmo tempo e no mesmo canal, causando erros nos dados que estão sendo transmitidos [18]. O grande desafio desses protocolos é reduzir a quantidade de colisões em transmissões sem fio. Uma resposta inicial para esse problema seria a simples utilização de um teste de portadora, um nó testaria o meio e só transmitiria se nenhum outro nó estiver usando o canal. Esse mecanismo, porém, não é suficiente para evitar colisões. A Figura 2.2 apresenta um diagrama espaço/tempo de quatro nós (A, B, C, D) acessando o mesmo canal de comunicação. O eixo horizontal mostra a posição de cada nó no espaço, o eixo vertical representa o tempo. Observando a Figura 2.2, observa-se que o nó B percebe que o canal está ocioso e faz uma transmissão no tempo t_0 . A partir desse momento t_0 , o nó B começa a transmitir sinais e seus *bits* se propagam em todas as direções ao longo do meio de transmissão. Em um tempo t_1 ($t_1 > t_0$) o nó D , porém, percebe que o canal está ocioso e também faz uma transmissão, os dados se propagam em todas as direções, assim como ocorreu com a transmissão de B . Porém, pouco tempo depois, a transmissão de D começa a interferir na transmissão de B e ambos os pacotes são perdidos. Isso mostra que o problema de colisões nas comunicações é um problema difícil e o protocolo *MAC* deve ser capaz de detectar colisões e realizar re-transmissões dos pacotes de maneira eficiente, embora o tempo de propagação seja bastante curto (cerca de $1\mu s$). Paralelamente, um protocolo *MAC* deve ser justo, isto é, garantir que todos os nós tenham oportunidades de usar o canal de comunicação e que os recursos sejam divididos igualmente. Obviamente, esse conceito pode ser adaptado para privilegiar um conjunto de nós em particular, mas em uma rede com nós com privilégios iguais a oportunidade acesso deve ser igual.

2.2 Classificação dos protocolos MAC em redes *Ad Hoc*

Diversos autores, classificam protocolos *MAC* em redes *ad hoc* de maneiras distintas. Segundo [17], esses protocolos podem ser classificados em protocolos livres de contenção e protocolos baseados em contenção. A seguir, descreve-se brevemente essa classificação.

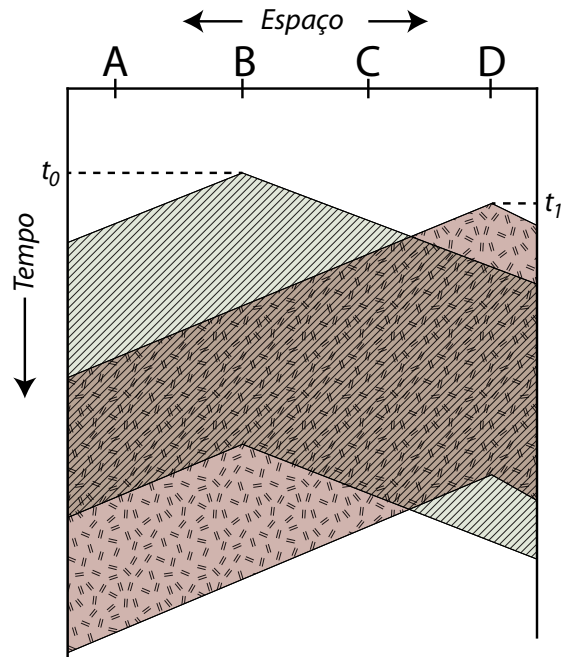


Figura 2.2: Diagrama de espaço-tempo com colisão na transmissão entre dois nós.

2.2.1 Protocolos Livres de Contenção

Protocolos livres de contenção usam técnicas que buscam eliminar as colisões, tais como:

- **Técnicas de divisão do canal:** O canal principal é dividido em várias faixas e essas faixas são atribuídas aos nós, de modo que essas faixas não se sobrepõem e não interferem entre si. Essas técnicas podem ser feitas no domínio do tempo (ex: *Time Division Multiple Access - TDMA* [19]), por códigos (ex: *Code Division Multiple Access - CDMA* [20]) ou no domínio da frequência (ex: *Frequency Division Multiple Access - FDMA* [21]).
- **Técnicas de revezamento:** Buscam garantir a propriedade de que: se existem n nós ativos em uma rede e o canal de acesso mútuo tem vazão V , então cada nó ativo têm vazão aproximadamente $\frac{V}{n}$. As técnicas principais são protocolos de seleção (ex: *Polling* [22]) e protocolos de passagem de permissão (ex: *Token Ring* [23]).
 - Protocolo *Polling*: Esse protocolo designa um dos nós da rede como nó líder. O nó líder seleciona cada um dos nós por alternância circular para cada nó da rede poder transmitir um certo número de pacotes por um certo período de tempo, eliminando colisões e intervalos. As principais desvantagens estão no atraso para eleição de um líder na rede e, se o líder falhar, o canal inteiro ficará inoperante.
 - Protocolo *Token Ring*: Nesse protocolo não há um nó líder, uma mensagem de permissão é passada entre os nós obedecendo uma determinada ordem fixa (como um identificador único de cada nó). Quando um nó recebe essa mensagem de permissão, ele retém a mensagem até enviar todos os seus pacotes

ou repassa a mensagem para o próximo nó da fila. A passagem de permissão é descentralizada e têm uma alta eficiência. As principais desvantagens estão na possibilidade da falha de um nó derrubar o canal inteiro ou um nó, por um erro, não liberar a permissão após concluir sua tarefa. Um exemplo de padrão que usa *token ring* é o *IEEE 802.5* [24].

2.2.2 Protocolos Baseados em Contenção

Protocolos baseados em contenção estão cientes que colisões podem ocorrer e buscam identificar, corrigir e prevenir problemas de colisão. Esses protocolos podem ser classificados em duas categorias:

- **Acesso Aleatório:** Em um protocolo de acesso aleatório, um nó transmissor transmite no canal sempre que estiver pronto. Quando ocorre uma colisão, cada nó envolvido nela retransmite repetidamente o pacote até que ele passe sem colisão. Essa retransmissão, porém, espera um tempo aleatório antes de ser realizada. Cada nó envolvido em uma colisão escolhe atrasos aleatórios independentes. Como após uma colisão os tempos de atraso são escolhidos arbitrariamente, é possível que um dos nós escolha um atraso suficientemente mais curto a ponto de evitar uma nova colisão. Como exemplos desses tipos de protocolos temos o *ALOHA* e o *Slotted ALOHA* [25].
- **Reserva de Canal:** Esse tipo de protocolo usa mecanismos de reserva de canal antes de uma transmissão (tipicamente utilizando pacotes de controle pergunte-antes-de-enviar e pronto-para-enviar, ou *RTS/CTS*) e para tratar a ocorrência de colisões, caso ocorram (utilizando mecanismos que detectam colisões e fazem a retransmissão dos pacotes). Existem protocolos que fazem reserva de canal, porém, sem teste de portadora, isto é, sem testar se o canal está ocupado ou ocioso (ex: *MACA*, *MACAW* [26]) e protocolos que fazem a reserva do canal e também teste de portadora (ex: *CSMA/CA* [27] e o *FAMA* [28]).

2.3 Protocolos MAC para Múltiplos Canais

Este trabalho trata principalmente dos protocolos MAC para múltiplos canais de comunicação. Esses protocolos, em geral, assumem que existem k canais de comunicação disponíveis e o objetivo é utilizar esses recursos da melhor maneira possível seguindo algumas premissas. Essas premissas variam desde obter economia de energia, proporcionar QoS até maximizar a vazão da rede [16]. A Tabela 2.1 lista alguns dos principais protocolos MAC para múltiplos canais da literatura, além de indicar se o protocolo assume sincronização entre os nós e quantos rádios são necessários para cada dispositivo. Esses protocolos são brevemente discutidos mais adiante.

MMAC (2003)

O MMAC (do Inglês, *Multichannel MAC Protocol*) [29] é um protocolo de acesso múltiplo que utiliza um único transceptor por nó. Esse protocolo usa a idéia de mensagens de

Tabela 2.1: Técnicas para alocação de múltiplos canais em redes *ad hoc*.

Proposta/Ano	Assume sincronização entre os nós	Rádios
MMAC/2003 [29]	sim	1
CMAC/2006 [30]	não	2
TMMAC/2007 [31]	sim	1
<i>Marina et al.</i> /2010 [32]	não	N
MC-LMAC/2011 [33]	sim	1

indicação de tráfego *ad hoc* *ATIM* (do Inglês, *Ad hoc Traffic Indication Messages*) do mecanismo de economia de potência do *IEEE 802.11*. Janelas *ATIM* são usadas pelo *IEEE 802.11* para colocar os nós em modo inativo, onde um nó consome muito menos energia não enviando ou recebendo pacotes. O *MMAC* adota esse conceito enviando, periodicamente, avisos para dividir o tempo em intervalos de *beacon*. Os nós são sincronizados, permitindo que seus intervalos de *beacon* comecem ao mesmo tempo. Cada intervalo de *beacon* começa com uma janela *ATIM* no qual é usada pelos nós comunicantes para trocar informações de controle. Um dos k canais de dados é escolhido como canal padrão e todos os nós escutam esse canal durante a janela *ATIM*. Dessa maneira, ao invés de se ter um único canal de controle, o *MMAC* usa um dos canais de dados por uma fração de tempo. Esta técnica é particularmente útil quando o número de canais disponíveis é baixo e alocar um único canal para controle seria custoso.

No *MMAC*, cada nó mantém uma estrutura de dados chamada lista de canais preferíveis (LCP) para registrar o uso de diferentes canais na sua vizinhança. Cada canal, nessa lista, é categorizado baseado nas suas preferências como:

- ALTO: Esse canal está sendo usado pelo próprio nó no intervalo de *beacon* atual. Apenas um canal pode estar nesse estado em um dado tempo.
- MÉDIO: Esse canal não está sendo usado por nenhum dos nós vizinhos.
- BAIXO: Esse canal já está sendo usado por um, ou mais, nós vizinhos.

O estado do canal, na LCP, é alterado da seguinte maneira: todos os canais da LCP estão no estado MÉDIO no começo de cada intervalo de *beacon*. Se dois nós escolherem um canal para se comunicarem, aquele canal muda para estado ALTO. Se um nó ouvir mensagens de controle que especificam que um canal em particular é usado por algum outro nó da sua vizinhança, então este nó muda o estado desse canal para BAIXO.

A Figura 2.3 ilustra o funcionamento do protocolo *MMAC*. Nessa figura, temos representado um intervalo de *beacon* onde 4 agentes, nós X , Y , Z e W , tentam se comunicar. As linhas horizontais indicam a ação realizada pelo agente correspondente em relação ao tempo. Nesse exemplo, temos dois canais, representados por setas contínuas e tracejadas. Quando um intervalo de *beacon* é iniciado, todos os nós da rede, que usam mecanismos de sincronização do protocolo *MMAC*, se voltam para o canal de controle, esse período de gerenciamento é chamado de janela *ATIM*. Quando um nó X tem pacotes para um nó Y , ele envia um pacote *ATIM* para Y , contendo a LCP de X . Ao receber esse pacote, o nó

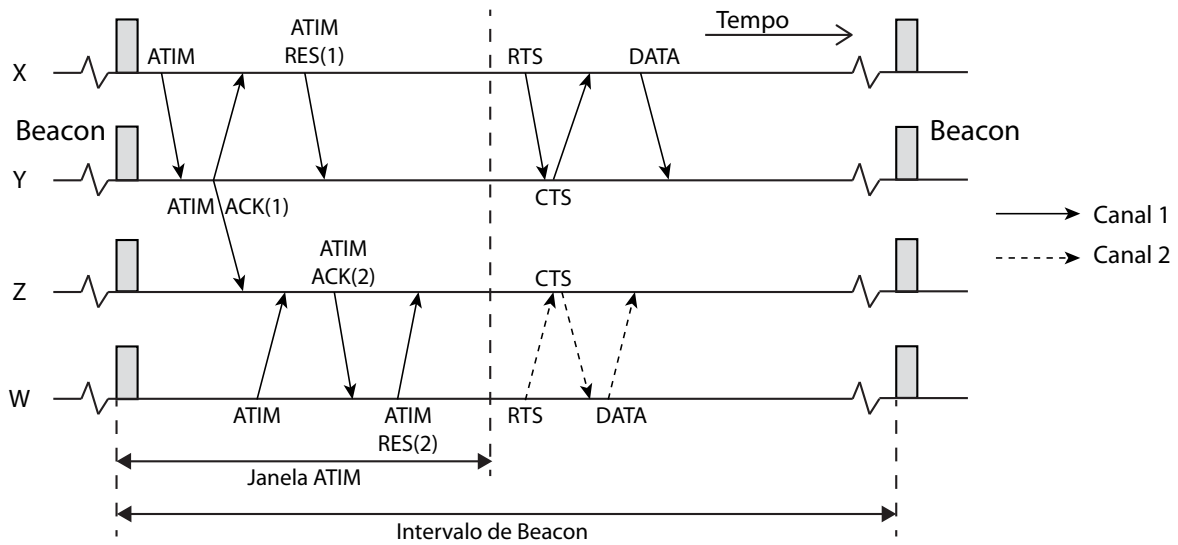


Figura 2.3: Exemplo de funcionamento do protocolo *MMAC*

Y escolhe um canal baseado na LCP de *X* e na sua própria LCP e envia um *ATIM-ACK* para *X*. Se o canal escolhido for aceito por *X* ele envia um pacote *ATIM-RES* para *Y* reservar o canal e também para outros nós na vizinhança saberem que o canal em particular foi reservado. Esta informação é usada pelos vizinhos para atualizar suas LCPs. Depois da janela *ATIM*, os nós *X* e *Y* mudam para o canal escolhido e começam a transmissão de dados. Por outro lado, se o canal escolhido não for aceito por *X*, ele terá que esperar pela próxima janela *ATIM* e renegociar. Desde que um nó *Z* também sintonize no canal padrão durante a janela de *ATIM*, ele ouve as mensagens de controle trocadas entre *X* e *Y* e escolhe outro canal para fazer suas transmissões. Um *backoff* randômico é usado para resolver colisões de pacotes *ATIM* quando múltiplos nós competem no canal padrão.

CMAC (2006)

Kim et al. [30] propõem o protocolo CMAC, um protocolo MAC multi-canal assíncrono e que utiliza 2 radios. Um dos radios é um radio de baixa potência LR (do Inglês, *Low power wakeup Radio*) que só envia ou recebe pulsos fracos e o outro é um rádio principal *half-duplex* MR (do Inglês, *Main half-duplex Radio*). O LR está sempre ligado e é usado para monitorar o canal de controle enquanto o radio MR fica inativo. O LR executa duas funções. Primeiramente, quando um nó precisa enviar, o receptor é “acordado” através de uma série de pulsos e, em seguida, a negociação de canais é feita antes do rádio MR ser mudado para modo ativo. O MR é capaz de mudar dinamicamente para qualquer um dos canais pré-definidos. Ele transmite em uma potência de transmissão constante na banda selecionada, embora, ele possa ser desligado também. O MR lida com todas as transmissões e recepções. Por necessitar de dois transceptores por nó, esse protocolo possui a desvantagem de aumentar a complexidade e o custo da rede como um todo. Além disso, o canal de controle pode se tornar um gargalo quando muitos nós iniciam a negociação de canais e solicitam transmissão de dados simultaneamente.

TMMAC (2007)

Zhang et al. [31] propõem o TMMAC, um protocolo MAC multi-canal para redes *ad hoc*. Em adição ao processo de negociação de frequências convencional (uso de mensagens de controle) o TMMAC introduz um processo de negociação de baixo custo. Nesse protocolo, o tempo é dividido em períodos fixos que consiste em uma janela ATIM (do Inglês, *Ad Hoc Traffic Indication Messages*) seguida por uma janela de comunicação. Na janela ATIM mensagens de controle são trocadas para determinar os canais e as fatias de tempo em que as transmissões vão ocorrer. Com isso, os nós da rede podem economizar energia entrando em modo ativo apenas quando necessitam enviar ou receber dados na janela de comunicação. O tamanho da janela ATIM é ajustado dinamicamente baseado em diferentes padrões de tráfego para atingir maior vazão e menor consumo de energia.

Marina et al. (2010)

Marina et al. [32] propõem uma formulação baseada em teoria dos grafos para o problema de atribuição de canais guiada por uma abordagem de controle de topologia. Os autores então propõem uma heurística gulosa, chamada CLICA, para obter topologias conexas, com baixa interferência e utilizando vários canais. Os autores mostram, através de simulação e prova matemática, que a proposta proporciona uma grande redução na interferência mesmo considerando um número pequeno de rádios por nó.

MC-MAC (2011)

Incel et al. [33] propõem o MC-LMAC, um protocolo baseado em agendamento que utiliza um único rádio para transmissão. Esse protocolo é projetado para redes de sensores com altas necessidades de vazão. O MC-LMAC toma vantagem tanto da comunicação agendada quanto a comunicação em múltiplos-canal. O agendamento das comunicações é feito com objetivo de minimizar as colisões. O principal problema desse protocolo está no custo da troca de mensagens, e se torna pior quando a densidade da rede aumenta.

2.4 Introdução à Comunicação Cooperativa (CC)

Canais de transmissão em redes sem fio sofrem de desvanecimento, isso significa que a atenuação do sinal pode variar significativamente ao longo de uma transmissão. Transmitir cópias independentes de um mesmo sinal gera diversidade e pode efetivamente combater os efeitos prejudiciais do desvanecimento. Em particular, diversidade espacial é gerada transmitindo sinais de diferentes localizações, permitindo assim que versões independentes do sinal sejam detectadas e decodificadas no dispositivo receptor [34].

MIMO (do Inglês, *Multiple-Input Multiple-Output*) é uma tecnologia que utiliza múltiplas antenas tanto em dispositivos transmissores quanto em dispositivos receptores. Foi demonstrado pela literatura que MIMO pode significativamente aumentar a vazão e a confiabilidade de um sistema usando técnicas que aumentam a diversidade espacial do sinal [35]. Devido à essas propriedades, a tecnologia MIMO foi incorporada em diversos padrões de comunicação sem fio, como *IEEE 802.11n*, *4G*, *WiMAX* e *HSPA+* [36]. Maiores detalhes sobre a tecnologia MIMO e técnicas que fazem uso dessa tecnologia podem ser encontradas em [37].

Comunicação Cooperativa (CC) é uma técnica que permite que dispositivos com uma única antena se beneficiem de algumas das vantagens dos sistemas MIMO como o uso da diversidade espacial. Em redes de múltiplos saltos tradicionais, nós intermediários cooperam com um nó fonte retransmitindo o dado para o nó destino, processo que é realizado pela camada de rede. Dessa maneira, o nó destino recebe apenas uma cópia do dado enviado pelo nó fonte. CC, por outro lado, é uma técnica da camada física. Nessa técnica, quando um nó fonte transmite um dado, nós ajudantes em volta do fonte sobre escutam o sinal e retransmitem várias cópias independentes do mesmo sinal para o nó destino, formando um sistema MIMO virtual [35]. Existem duas técnicas principais de retransmissão por parte dos nós ajudantes em CC [38]:

- **Amplifica e encaminha** (*Amplify and Forward*): Nessa técnica, quando um nó fonte quer enviar um dado para um nó destino ele envia o sinal normalmente. Os nós ajudantes do nó fonte, ao receber o sinal ruidoso, amplificam e retransmitem o sinal sem nenhum processamento adicional. Essa técnica possui a desvantagem de propagar erros, caso eles ocorram e a vantagem de ser uma solução com baixa complexidade;
- **Decodifica e encaminha** (*Decode and Forward*): De maneira similar à técnica anterior, o nó fonte transmite um sinal e os nós ajudantes deste nó fonte recebem um sinal ruidoso. Eles então decodificam o sinal e o retransmitem, diminuindo o ruído no receptor em relação à técnica anterior, porém, aumentando a complexidade e o custo operacional.

Em ambas alternativas, o nó destino pode combinar as várias cópias do sinal recebido tanto do nó fonte quanto dos nós ajudantes, decodificar e obter o sinal original. O conceito de combinar sinais parciais tem sido tradicionalmente chamado de MRC (*Maximal Ratio Combining*) [39].

Diversas técnicas de CC utilizam retransmissão do tipo “amplifica e encaminha”. *Lane-man et al.* [40] propõem diferentes protocolos que utilizam comunicação cooperativa com a técnica de “amplifica e encaminha” e avalia seu desempenho em termos de comportamento de falha. *Chatnagar et al.* [41] propõem uma dupla modulação diferencial em um sistema de CC com um protocolo de “amplifica e encaminha” para evitar o problema de compensação de portadora em *m-Nakagami* canais de desvanecimento. O esquema proposto apresenta melhores resultados do que os sistemas duplamente diferenciais tradicionais.

No campo das técnicas de CC que utilizam “decodifica e encaminha”, *Sendonaris et al.* [42] incorporam essa técnica ao CDMA, obtendo ganhos em vazão e em área de cobertura em relação a um sistema CDMA convencional. *Bletsas et al.* [43] apresentam uma estratégia oportunista de retransmissão que utiliza tanto as técnicas de “amplifica e encaminha” quanto de “decodifica e encaminha”. Essa técnica pode ser implementada de maneira distribuída, sem a necessidade do estado global do canal e é capaz de obter resultados bastante eficientes na seleção de nós retransmissores. Os trabalhos em [44, 45, 46, 47] focam em maneiras ótimas (ou semi-ótimas) para a escolha de nós ajudantes em CC.

Existe um grande esforço de pesquisa para aprimorar os mecanismos da camada física para permitir CC e estudos em camadas superiores estão começando a ganhar importância. Uma vez que CC pode ser utilizado para melhorar a conectividade, expandir a cobertura

Tabela 2.2: Técnicas que utilizam comunicação cooperativa.

Proposta/Ano	Foco
<i>DTCC/ITCC (2006) [55]</i>	Melhorar o consumo de energia
<i>CoopBridges (2010) [39]</i>	Melhorar a conectividade
<i>Greedy(Add/Del)Link (2012) [54]</i>	Manter rotas eficientes

e melhorar o consumo de energia, a técnica tem sido aplicada em diversas áreas como: difusão [48, 49, 50], roteamento [51, 52], rádios cognitivos [53] e controle de topologia [39, 54, 55]. A Seção 2.5 descreve os principais trabalhos da literatura que combinam controle de topologia com CC.

2.5 Controle de Topologia e Comunicação Cooperativa em Redes Ad Hoc

Controle de topologia é uma técnica que altera a topologia da rede baseado em algumas condições. Por exemplo, controle de topologia pode ser utilizada para otimizar o consumo de energia da rede, reduzir o custo de roteamento e o número de mensagens de controle, melhorar a vazão ou atender a requisitos de QoS [55, 56, 57].

De acordo com [55], protocolos de controle de topologia podem ser classificados como: (i) centralizados; ou (ii) distribuídos. Protocolos centralizados consideram que informações globais estão disponíveis, tal como informação de topologia, informação de roteamento, estado da memória, etc. No entanto, mesmo quando informação global está disponível, foi provado que encontrar topologias fortemente conexas com o mínimo de energia é considerado um problema NP-completo [58]. Entre os protocolos centralizados, *Ramanathan et al.* [59] propõem alternativas para otimizar a conectividade da rede enquanto o tempo de vida da rede é aprimorado. Protocolos distribuídos consideram informação de vizinhança de até k saltos, onde k é tipicamente um ou dois. *Li et al.* [60] propõem um algoritmo baseado em cone para controle de topologia cujo objetivo é otimizar o consumo de energia enquanto mantém a conectividade. Para alcançar esse objetivo, cada nó ajusta sua potência de transmissão para cobrir um certo número de nós vizinhos, sob a condição de que esses vizinhos fiquem dentro de um cone com até α graus. Os autores demonstram que um grau de $\alpha = \frac{5\pi}{6}$ é suficiente para preservar a conectividade da rede. Em um trabalho mais recente, *Li et al.* [61] propuseram um algoritmo chamado *Localised Minimum Spanning Tree (LMST)*. No LMST cada nó cria uma MST utilizando informação de até um salto. A topologia final é construída de tal forma que o grau máximo do grafo é igual a 6. Mais informações sobre controle de topologia podem ser encontrados em [62, 63].

A Tabela 2.2 lista algumas das principais propostas da literatura que consideram o uso de controle de topologia em redes *ad hoc* cooperativas. As subseções que seguem descrevem brevemente essas propostas.

DTCC/ITCC (2006)

Cardei et al. [55] foram um dos primeiros a estudar o uso de controle de topologia com comunicação cooperativa com o objetivo de obter topologias de rede fortemente conexas e com o mínimo de consumo de energia. Os autores mostram que esse problema é NP-completo e propuseram dois algoritmos localizados e distribuídos. Ambos os algoritmos recebem como entrada o resultado de algoritmos tradicionais de controle de topologia (sem comunicação cooperativa). O primeiro algoritmo usa um processo de decisão distribuído onde cada nó utiliza informação de até dois saltos. O segundo algoritmo iterativamente atribui potências de transmissão para os nós, utilizando apenas informação de um salto.

CoopBridges (2010)

Yu et al. [39] propõem um esquema de controle de topologia centralizado com o objetivo de aumentar a conectividade da rede e de reduzir o consumo de energia. Para minimizar o número de enlaces cooperativos e reduzir o custo de transmissão, um algoritmo polinomial e outro exponencial (porém ótimo) de seleção de nós ajudantes são propostos.

Greedy(Add/Del)Link (2012)

Zhu et al. [54] consideram o problema de selecionar caminhos eficientes quando links cooperativos são utilizados. Os autores propõem dois algoritmos de controle de topologia que constroem *energy spanners* no qual a eficiência energética dos caminhos é garantida. Ambos os algoritmos podem ser executados de maneira distribuída ou localizada.

2.6 Discussão

Esse capítulo fez uma breve revisão de conceitos básicos que são necessários para a compreensão deste trabalho. Primeiramente, uma breve descrição sobre redes sem fio e protocolos de acesso ao meio é feita. Em seguida, classifica-se os protocolos de acesso ao meio em redes *ad hoc* e alguns exemplos de protocolos de acesso ao meio em múltiplos canais são apresentados. Por fim, introduz-se o conceito de comunicação cooperativa e alguns trabalhos que combinaram essa técnica com controle de topologia são citados.

Capítulo 3

Alocação Eficiente em Energia em Redes Sem Fio com Múltiplos Canais

Este capítulo fala sobre alocação eficiente em energia de redes sem fio com múltiplos canais de comunicação. Em particular, este capítulo apresenta um novo protocolo, chamado EEMC-MAC, que pode ser utilizado na alocação de múltiplos canais e agendamento de transmissões em redes sem fio. A Seção 3.1 faz uma breve contextualização sobre o cenário em que esta proposta está inserida, falando sobre a motivação da proposta e fazendo uma revisão específica de trabalhos relacionados. A Seção 3.2 descreve o modelo de comunicação considerado. A Seção 3.3 discorre sobre o problema de atribuição de canais e apresenta uma solução para esse problema. A Seção 3.4 apresenta a proposta do capítulo e a Seção 3.5 apresenta os resultados empíricos. A Seção 3.6 propõe uma possível extensão do EEMC-MAC para múltiplos saltos. A Seção 3.7 faz uma discussão dos resultados.

3.1 Contextualização e Motivação

A necessidade de acesso sem fio ininterrupto e de alto desempenho à Internet tem sido influenciada pela popularização dos dispositivos móveis e o advento das redes sociais. Essa tendência em aplicações móveis tem proporcionado um crescente esforço de pesquisa por protocolos de acesso ao meio (MAC) capazes de lidar com essa demanda crescente. Uma das maiores preocupações no desenvolvimento desses protocolos é manter o consumo de energia em um nível aceitável, uma vez que esses dispositivos tipicamente operam utilizando baterias. Controle de topologia (*topology control*) e ciclo de trabalho (*duty-cycle*) são duas das principais estratégias utilizadas para proporcionar economia de energia em redes sem fio [64]. Técnicas de controle de topologia permitem que dispositivos sem fio ajustem suas potências de transmissão com o objetivo de economizar energia sem afetar a conectividade da rede [54]. Esquemas de ciclo de trabalho permitem que dispositivos sem fio alternem entre modos ativo e inativo. Quando em modo ativo, o dispositivo está disponível para enviar ou receber dados, enquanto em modo inativo o dispositivo está em um modo de economia de energia onde não é capaz de receber ou transmitir dados. Essa última estratégia é bastante desafiadora uma vez que necessita de mecanismos para garantir que dispositivos que desejam se comunicar estejam em modo ativo no mesmo instante [30].

Grande parte dos protocolos MAC disponíveis são desenvolvidos para ambientes com um único canal de comunicação [65]. Tais protocolos, especialmente em cenários muito densos, possuem altas taxas de colisão de pacotes, causando retransmissão, atraso fim-a-fim e diminuindo a vazão efetiva do canal de comunicação. Múltiplos canais de comunicação têm sido utilizados para aumentar a vazão em redes sem fio [6]. Tais canais podem ser obtidos por técnicas de acesso oportunista, obtendo assim acesso temporário à frequências licenciadas não utilizadas [66]. Com a disponibilidade de múltiplos canais, técnicas baseadas no FDMA (do Inglês, *Frequency Division Multiple Access*), por exemplo, permitem selecionar diversos canais de comunicação com frequências que não se sobrepõem e não se interferem. Assim, pares de nós podem comunicar ao mesmo tempo e sem interferência desde que sejam atribuídos para canais distintos.

Diversos trabalhos da literatura consideram a utilização de múltiplos canais de comunicação em redes sem fio [33, 31, 67]. Alguns desses trabalhos combinam protocolos MAC de múltiplos canais com técnicas de ciclo de trabalho para aumentar a vazão da rede e diminuir o consumo de energia. *Tang et al.* propõem o EM-MAC [68], um protocolo multi-canal para minimizar o consumo de energia em redes de sensores sem fio. O EM-MAC permite que os nós transmissores estimem o tempo em que os nós receptores estarão em modo ativo e não necessita de um canal de controle. *Incel et al.* propõem o MC-LMAC [33], um protocolo MAC multi-canal para redes de sensores sem fio. O MC-LMAC é totalmente distribuído e faz o agendamento da comunicação baseado em algoritmos TDMA (do Inglês, *Time Division Multiple Access*). Essa abordagem minimiza a quantidade de colisões na comunicação e informa aos nós em quais períodos eles devem ficar em modo ativo. O MC-LMAC foca mais em maximizar a vazão do que em minimizar o consumo de energia.

Zhang et al. [31] propõem o TMMAC, um protocolo MAC multi-canal para redes *ad hoc*. O TMMAC divide sua operação em estágios de gerenciamento e transmissão. No início do estágio de gerenciamento, todos os nós que desejam se comunicar se voltam para um canal de controle e negociam um canal de comunicação. O estágio de gerenciamento, inicialmente, possui duração fixa, mas a duração é ajustada dinamicamente de acordo com o tráfego. Durante o estágio de transmissão, os nós comunicantes se voltam para os canais negociados no estágio anterior enquanto os nós que não desejam comunicar permanecem em modo inativo. Em trabalhos anteriores, propusemos o ECOA [13], uma técnica para alocação e agendamento de dados eficiente em energia. Assim como o TMMAC, essa técnica divide sua operação em estágios de gerenciamento e transmissão. A técnica utiliza um agendamento eficiente de transmissões e a estratégia de ciclo de trabalho para alternar os nós entre modos ativo e inativo de maneira eficiente, obtendo assim um consumo de energia bastante reduzido.

As propostas anteriores mostram que é possível diminuir o consumo de energia ao custo de um tempo maior para realizar a comunicação (como é o caso do EM-MAC). Reciprocamente, é possível minimizar o tempo de comunicação relaxando restrições de energia (como é o caso do MC-LMAC). Claramente, achar um compromisso entre esses dois parâmetros é um desafio. Tanto o TMMAC quanto o ECOA visam balancear esses dois parâmetros, no entanto, a necessidade de um estágio de gerenciamento que ocorre em um canal de controle único, independente do número de canais disponíveis, pode se tornar um gargalo pois aumenta o tempo de comunicação.

Este capítulo aborda os problemas de alocação de múltiplos canais, agendamento de

transmissões e economia de energia em redes sem fio. Assim como em trabalhos relacionados [31], assume-se que os dispositivos funcionam com bateria e possuem um único transceptor, capaz de sintonizar em um de vários canais de comunicação disponíveis e alternar entre modos ativo (consumo normal de energia) e inativo (consumo reduzido de energia). Assume-se que o tempo é dividido em fatias de tempo e cada fatia possui a duração necessária para garantir uma transmissão ou recepção de pacote. Neste contexto, este capítulo propõe o EEMC-MAC, um protocolo MAC eficiente em energia para redes sem fio com múltiplos canais. Essa técnica divide sua operação em estágios de gerenciamento e transmissão. O estágio de gerenciamento diverge de outras propostas da literatura, uma vez que considera a utilização de todos os canais de comunicação disponíveis, enquanto mantém um baixo consumo de energia e tempo de comunicação. Resultados empíricos mostram que o tempo gasto com gerenciamento no EEMC-MAC é, em média, cerca de 5% do tempo total de execução, enquanto o consumo de energia no estágio de transmissão é bastante eficiente.

3.2 Modelo de Comunicação

Esta seção detalha o modelo de comunicação considerado unicamente para este capítulo. Considera-se que cada estação da rede possui um único transceptor e um identificador único, que é conhecido pelas demais estações da rede. Cada nó da rede possui um enlace para todos os demais nós da rede. Ou seja, considera-se um ambiente de um único salto, representado por um grafo de topologia G_P . O cenário de comunicação, por outro lado, é representado por um grafo direcionado $G = (V, E)$ (grafo de comunicação), onde $V = \{v_1, v_2, \dots, v_n\}$ é o conjunto de estações (nós) e $E \subseteq V^2$ é o conjunto de dados (arestas) a serem transmitidos. Considere $E = \{e_1, e_2, \dots, e_p\}$, onde $e_h = \{(v_s, v_d) | \{v_s, v_d\} \subseteq V, s \neq d\}$, $1 \leq h \leq p$, como o conjunto de arestas que representam o grafo de comunicação da rede K_n . Cada aresta $e_h = (v_s, v_d) \in E$ representa uma comunicação entre um nó fonte v_s e um nó destino v_d . Não existem arestas paralelas no grafo de comunicação. Cada nó $v_i \in V$ possui um conjunto de transmissão s_{v_i} que identifica os destinatários dos dados que v_i tem para enviar e um conjunto de recepção d_{v_i} que indica quais nós possuem dados para enviar para v_i . Desta forma, cada nó v_i possui $\tau_{v_i} = |s_{v_i}| + |d_{v_i}|$ dados para enviar e receber.

Como exemplo, a Figura 3.1 ilustra um possível grafo de comunicação para a topologia de rede G_t . Nessa figura, $V = \{v_1, v_2, v_3, v_4\}$ e $E = \{e_1, e_2, e_3\}$, onde $e_1 = (v_1, v_2)$, $e_2 = (v_1, v_4)$ e $e_3 = (v_3, v_2)$. Nesse grafo, o nó v_1 possui dados para enviar para v_2 e v_4 , assim $s_{v_1} = \{v_2, v_4\}$, e nenhum dado para receber, ou seja, $d_{v_1} = \emptyset$. Similarmente, $s_{v_2} = \emptyset$, $d_{v_2} = \{v_1, v_3\}$, $s_{v_3} = \{v_2\}$, $d_{v_3} = \emptyset$, $s_{v_4} = \emptyset$ e $d_{v_4} = \{v_1\}$.

Assim como nos estudos apresentados por [69, 31], a proposta apresentada nesse capítulo assume que transmissões/recepções de dados ocorrem em fatias de tempo, com cada transmissão/recepção ocupando exatamente uma fatia de tempo. Em cada fatia de tempo t_j , $j \geq 0$, onde t_j equivale ao intervalo de tempo $[t_j, t_{j+1})$, um nó pode alternar entre os modos ativo e inativo. Quando ativo, um nó pode enviar ou receber dados. Caso contrário, o nó pode economizar energia ficando em modo inativo. Ou seja, o consumo de energia em cada nó é relacionado com o número de fatias de tempo que ele fica em modo ativo. Considere $C = \{c_1, c_2, \dots, c_k\}$ como o conjunto de canais disponíveis para comunicação. Quando um canal c_i , $1 \leq i \leq k$, é utilizado por um par de nós em uma fatia

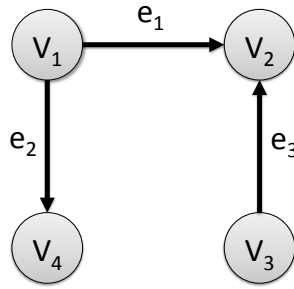


Figura 3.1: Exemplo de grafo de comunicação com $n = 4$ nós.

de tempo t_j , ele ficará indisponível para outros nós nessa mesma fatia de tempo. Quando duas, ou mais, estações transmitem em um mesmo canal e ao mesmo tempo, uma colisão ocorrerá e os dados são perdidos. A Tabela 3.1 sumariza todas as notações utilizadas nesse capítulo.

3.3 O Problema de Atribuição de Canais (CAP)

Em um ambiente de rede onde múltiplos canais de comunicação estão disponíveis, o problema de atribuição de canais que satisfaz as restrições de interferência e maximiza a vazão é conhecido como problema de atribuição de canais (ou CAP, do Inglês *Channel Assignment Problem*). Para prevenir interferência entre comunicações, um mesmo canal não pode ser atribuído para dois pares de nós comunicantes simultaneamente. Em sua forma geral, o CAP é equivalente ao problema de coloração de grafos (ou GCP, do Inglês *Graph-Coloring Problem*), o qual é conhecido por ser um problema NP-completo [70]. Dado um grafo de comunicação G e k canais de comunicação do modelo proposto anteriormente, o CAP consiste em realizar as comunicações representadas em G simultaneamente no menor tempo e utilizando o mínimo de canais possível. Note que se $k = 1$ esse problema é simplificado, uma vez que todas as comunicações necessariamente seriam serializadas. No entanto, no caso geral, soluções ótimas são complexas de se obter.

Por ser um problema NP-completo, muitos pesquisadores propuseram heurísticas e algoritmos aproximados para resolver o problema de atribuição de canais, estes, no entanto, não garantem soluções ótimas. As soluções propostas utilizam desde redes neurais [71] até teoria de jogos e de grafos [70]. A seguir, é apresentada uma heurística, proposta por esse trabalho, que usa a coloração de arestas e permite obter uma solução para o problema de atribuição de canais.

3.3.1 ECOH: Uma Heurística para o Problema de Atribuição de Canais

O Algoritmo 3.1 apresenta uma heurística baseada no algoritmo de coloração de arestas para resolver o problema de atribuição de canais, chamada *ECOH* (sigla para *Edge Coloring Heuristic*). A heurística recebe como entrada um grafo de comunicação $G = (V, E)$ e o número k de canais disponíveis. Como saída, o algoritmo retorna uma lista de conjuntos de comunicação definida por $CS = \{CS_1, CS_2, \dots, CS_r\}$, com $CS_i \subseteq E$, $1 \leq i \leq r$, onde os elementos de cada conjunto CS_i são disjuntos, $1 \leq i \leq r \leq |E|$. A

Tabela 3.1: Tabela de notações.

Símbolo	Definição
G_P	Grafo que representa a topologia da rede sem fio;
$G = (V, E)$	Grafo (ou cenário) de comunicação;
$V = \{v_1, v_2, \dots, v_n\}$	Conjunto de nós (ou de dispositivos);
$E = \{e_1, e_2, \dots, e_p\}$	Conjunto de arestas (ou de transmissões);
n	Números de nós (dispositivos) na rede;
p	Números de arestas (transmissões) na rede;
s_{v_i}	Conjunto de transmissão do nó v_i ;
d_{v_i}	Conjunto de recepção do nó v_i ;
τ_{v_i}	Equivale a $ s_{v_i} + d_{v_i} $;
t_j	Fatia de tempo equivalente ao intervalo $[t_j, t_{j+1})$;
$C = \{c_1, c_2, \dots, c_k\}$	Conjunto de canais;
k	Número de canais;
g_i	Subconjunto de V ($g_i \subseteq V$);
l	Equivale a $ g_i $;
T_m	Tempo do protocolo para o estágio de gerenciamento;
T_t	Tempo do protocolo para o estágio de transmissão;
T_t'	Tempo ótimo do estágio de transmissão;
α	Equivale a $\frac{T_t}{T_m + T_t}$;
γ	Equivale a $\frac{T_t'}{T_t}$;

ideia básica por trás da heurística proposta é a distribuição das arestas pertencentes a E em r conjuntos de comunicação, tal que as arestas pertencentes a cada conjunto não possuam dependências de comunicação entre si. O critério de seleção escolhe uma aresta do nó de maior grau em G . Esta aresta será parte do conjunto de transmissão inicial CS_i e servirá de base de comparação para as demais arestas em E . Apenas as arestas sem dependências com outros elementos de CS_i serão removidas de E e inseridas em CS_i . Uma aresta é considerada não dependente com um conjunto de outras arestas se ela não compartilha nenhum vértice em comum com as demais arestas do conjunto. O processo é repetido até que todas as arestas em E tenham sido tratadas.

Para melhor entender as operações do Algoritmo 3.1, considere como entrada o grafo de comunicação ilustrado pela Figura 3.1 e o número de canais disponíveis igual a $k = 2$. Dessa maneira, temos $E = \{e_1, e_2, e_3\}$, onde $e_1 = (v_1, v_2)$, $e_2 = (v_1, v_4)$ e $e_3 = (v_3, v_2)$. Suponha que a aresta e_2 seja inserida no primeiro conjunto de comunicação CS_1 , linha 5. Percorrendo todas as arestas de E , linha 6, o algoritmo checa que a aresta e_3 não possui dependência com os itens do conjunto CS_1 e decide inserir essa aresta em CS_1 , linha 8. Como não existem mais arestas em E sem dependências com CS_1 , o algoritmo termina o laço. Um novo laço é iniciado e a variável r é incrementada para 2, linha 3. No novo laço, o algoritmo insere a aresta e_1 no conjunto de comunicação CS_2 e finaliza o algoritmo, uma vez que a condição $E = \emptyset$ é alcançada, linha 2. Nesse exemplo, a saída do algoritmo seria $CS = \{CS_1, CS_2\}$, com $CS_1 = \{e_2, e_3\}$ e $CS_2 = \{e_1\}$. Observe que, segundo o algoritmo, $|CS_i| \leq k$, $1 \leq i \leq r$, ou seja, cada conjunto terá no máximo k

Algoritmo 3.1 : $\text{ECOH}(G,k)$

```
1:  $G = (V, E)$ ;  $r \leftarrow 0$ ;  
2: Enquanto ( $E \neq \emptyset$ ) faça  
3:    $r \leftarrow r + 1$ ;  
4:   Selecione a aresta “ $e$ ” do vértice de maior grau em  $E$ ;  
5:    $CS_r \leftarrow e$ ,  $E \leftarrow E \setminus e$ ;  
6:   Para (cada  $e' \in E$ ) faça  
7:     Se ( $e' \cap CS_r = \emptyset$ ) e ( $|CS_r| \leq k$ ) então  
8:        $CS_r \leftarrow CS_r \cup e'$ ;  
9:        $E \leftarrow E \setminus e'$ ;  
10:  Fim Se  
11:  Fim Para  
12: Fim Enquanto  
13:  $CS \leftarrow \{CS_1, CS_2, \dots, CS_r\}$ ;
```

elementos disjuntos. Essa construção permite que todas as transmissões representadas em um conjunto de comunicação CS_i , possam ser realizadas em paralelo nos k canais disponíveis e na mesma fatia de tempo.

3.3.2 ECOH: Complexidades Envolvidas

O Lema 1 descreve as complexidades envolvidas na heurística ECOH (Algoritmo 3.1).

Lema 1 *Dado um conjunto de canais k e um grafo de comunicação $G = (V, E)$, a heurística ECOH (Algoritmo 3.1) calcula uma lista de conjuntos de comunicação $CS = \{CS_1, CS_2, \dots, CS_r\}$ tal que as arestas em cada lista CS_i , $1 \leq i \leq r$, não possuem dependências entre si e $|CS_i| \leq k$. O ECOH calcula as r listas em $O(p^2)$ operações, onde $|E| = p$.*

Prova do Lema 1 *A heurística ECOH (Algoritmo 3.1) consiste em dois laços principais alinhados. O primeiro laço (linha 2), é executado enquanto $E \neq \emptyset$, ou seja, um máximo de $|E|$ iterações. O segundo laço (linha 6) é executado para cada $e' \in E$, ou seja, um máximo de $|E|$ operações novamente. É trivial ver que as operações de conjuntos podem ser executadas em $O(1)$ operações. Dessa forma, temos que, no pior caso, a heurística ECOH pode ser executada em $O(|E|^2)$ operações.*

3.4 Descrição da Proposta

Esta seção apresenta os detalhes do protocolo proposto, chamado EEMC-MAC (sigla para *Energy Efficient Multi-Channel MAC Protocol*). Este protocolo busca realizar a alocação de múltiplos canais e agendamento de transmissões para permitir a comunicação de dados em redes sem fio. Adicionalmente, o protocolo realiza estas tarefas de forma a reduzir o consumo de energia e o tempo necessário para transmitir os dados. Começamos apresentando as rotinas que serão mais tarde utilizadas pelo protocolo.

Algoritmo 3.2 : $\text{CombineGroup}(g_i, c_i)$

- 1: Seja $|g_i| = l$;
 - 2: Cada nó em g_i computa seu ID local dentro de $[1..l]$ tal que $g_i = \{v_1, v_2, \dots, v_l\}$;
 - 3: **Para** ($j \leftarrow 1$ até $|g_i| - 1$) **faça**
 - 4: Nós v_j e v_{j+1} entram em modo ativo;
 - 5: v_j envia seu conjunto de transmissão s_{v_j} para v_{j+1} usando o canal c_i ;
 - 6: Nó v_{j+1} faz $s_{v_{j+1}} = s_{v_{j+1}} \cup s_{v_j}$;
 - 7: Nó v_j entra em modo inativo;
 - 8: **Fim Para**
-

3.4.1 Rotinas para Agrupamento de Conjuntos de Transmissão

Da Seção 3.2, temos que cada nó $v_i \in V$ possui um conjunto de transmissão s_{v_i} que identifica o conjunto de nós que v_i possui dados para enviar, que existe sincronia entre os nós e que cada nó possui um ID único na rede e sabe o número de canais disponíveis. Nessa subseção, estamos interessados em combinar todos os conjuntos de transmissão em um único nó. A rotina CombineGroup , descrita pelo Algoritmo 3.2, busca alcançar essa meta utilizando um único canal de comunicação. A rotina possui como entrada: um conjunto de nós g_i , $g_i \subseteq V$, $|g_i| = l$, e um canal de comunicação c_i , $1 \leq i \leq k$. No primeiro passo do algoritmo, cada nó em g_i computa um ID local consecutivo, linha 2. Seja v_l o nó com o maior ID em g_i . O laço das linhas 3-8 combina os conjuntos de transmissão s_{v_j} , $1 \leq j \leq l$ de tal forma que o nó local v_l contenha os conjuntos de transmissões $s_{v_l} \cup s_{v_{l-1}} \cup \dots \cup s_{v_1}$.

Agora, suponha que $k > 1$ canais estão disponíveis. Neste caso, a rotina CombineGroup poderia ser aprimorada para tomar vantagem de se ter vários canais disponíveis. A rotina descrita pelo Algoritmo 3.3, chamada CombineTS , mostra como os conjuntos de transmissão podem ser combinados utilizando diversos canais simultaneamente. De maneira similar ao CombineGroup , a rotina CombineTS possui dois parâmetros: um grupo de nós g_i , $g_i \subseteq V$, $|g_i| = l$, e um conjunto de canais $C = \{c_1, c_2, \dots, c_k\}$. Essa rotina só é executada se $k \geq \lfloor \frac{l}{2} \rfloor$ para que todas as transmissões em g_i possam ser paralelizadas dentro dos k canais disponíveis. No começo do algoritmo, todos os nós computam um ID local, linha 4. O procedimento cresce em uma árvore binária, combinando os nós folhas e fazendo o seu caminho até a raiz através dos k canais disponíveis, linhas 5-13. Ao final do algoritmo, o nó v_1 terá todos os conjuntos de transmissão $s_{v_l} \cup s_{v_{l-1}} \cup \dots \cup s_{v_1}$. Para referência futura, considere o lema a seguir:

Lema 2 *A rotina CombineGroup (Algoritmo 3.2) combina os conjuntos de transmissão dos nós de g_i em $|g_i| - 1$ fatias de tempo, utilizando um único canal c_i e com cada nó ficando em modo ativo por 2 fatias de tempo. A rotina CombineTS (Algoritmo 3.3) combina os conjuntos de transmissão dos nós de g_i em até $\log k + 1$ fatias de tempo, usando $|C| = k$ canais e com cada nó ficando em modo ativo por não mais que $\log k + 1$ fatias de tempo, onde $k \geq \lfloor \frac{l}{2} \rfloor$, $|g_i| = l$.*

Prova do Lema 2 *A rotina CombineGroup (Algoritmo 3.2) possui um único que laço (linha 3) que possui $|g_i| - 1$ iterações. Dentro desse laço, uma única mensagem é enviada (linha 5), tomando uma fatia de tempo. Assume-se que as outras operações podem ser executadas em um tempo desprezível. Dessa forma, a rotina CombineGroup necessita de*

Algoritmo 3.3 : $\text{CombineTS}(g_i, C)$

- 1: Seja $|g_i| = l$, $|C| = k$, $b = l$;
 - 2: **Se** $(k \geq \lfloor \frac{l}{2} \rfloor)$ **então**
 - 3: Seja $C = \{c_1, c_2, \dots, c_k\}$;
 - 4: Cada nó em g_i computa seu ID local dentro de $[1..l]$ tal que $g_i = \{v_1, v_2, \dots, v_l\}$;
 - 5: **Enquanto** $(b > 1)$ **faça**
 - 6: **Para** $(i \leftarrow 0$ até $(\frac{l}{2} - 1))$ em paralelo **faça**
 - 7: Atribua o canal c_{i+1} ao par (v_{i+1}, v_{l-i}) ;
 - 8: v_{l-i} envia seu conjunto de transmissão $s_{v_{l-i}}$ para v_{i+1} ;
 - 9: v_{i+1} faz $s_{v_{i+1}} = s_{v_{i+1}} \cup s_{v_{l-i}}$;
 - 10: v_{l-i} entra em modo inativo;
 - 11: **Fim Para**
 - 12: $b \leftarrow b/2$;
 - 13: **Fim Enquanto**
 - 14: **Fim Se**
-

$|g_i| - 1$ fatias de tempo para ser executada. O consumo de energia máximo por nó é de duas fatias de tempo, sendo uma para receber uma mensagem na iteração j e outra para enviar uma transmissão na fatia de tempo $j+1$. A rotina CombineTS (Algoritmo 3.3) possui dois laços alinhados. O laço mais interno (linha 6) ocorre em paralelo, ou seja, cada uma das $\frac{l}{2}$ iterações, onde $|g_i| = l$, ocupam uma única fatia de tempo. Dessa forma, a complexidade de tempo está atrelada apenas ao tempo consumido pelo laço mais externo (linha 5). Assume-se que as outras operações podem ser executadas em um tempo desprezível. O laço mais externo é executado $\log l + 1$ vezes e, como $k \geq \lfloor \frac{l}{2} \rfloor$, pode-se assumir um tempo máximo de execução de $\log k + 1$ fatias de tempo. Assumindo o pior caso onde um nó fica ativo durante toda a execução da rotina CombineTS , temos que o consumo máximo de energia é da ordem de $\log k + 1$ fatias de tempo.

3.4.2 Detalhes do Protocolo EEMC-MAC

A seguir, apresentamos os detalhes do protocolo EEMC-MAC. A operação do protocolo EEMC-MAC é dividida em duas fases: **gerenciamento** e **transmissão**. Ambas as fases serão melhor discutidos a seguir.

EEMC-MAC: Fase de Gerenciamento

A ideia principal da fase de gerenciamento é garantir que um nó líder obtenha todos os conjuntos de transmissão s_{v_i} , $\forall v_i \in V$. Este processo deve ocorrer de uma maneira eficiente em energia e ao mesmo tempo utilizar o maior número de canais possível para minimizar o tempo de execução. Então, o nó líder pode unir todos os conjuntos de transmissão e criar o grafo de comunicação $G = (V, E)$. O Algoritmo 3.4 descreve os passos da fase de gerenciamento. Se a relação $k < \frac{n}{2}$ for verdadeira, os n nós no conjunto $V = \{v_1, v_2, \dots, v_n\}$ são divididos em k grupos de nós g_1, g_2, \dots, g_k , linhas 2-3. Tendo em vista que cada nó conhece k , n e seu identificador local, ele tem condições de identificar o grupo a qual pertence. A meta é reduzir o número de nós ativos até k . No próximo

Algoritmo 3.4 : ManagementStage(V, C)

- 1: Todos os nós em $V = \{v_1, v_2, \dots, v_n\}$ entram em modo inativo;
 - 2: **Se** ($k < \lfloor \frac{n}{2} \rfloor$) **então**
 - 3: Divida os nós em V em k grupos: g_1, g_2, \dots, g_k ;
 - 4: **Para** ($i \leftarrow 1$ **até** k) em paralelo **faça**
 - 5: Execute $CombineGroup(g_i, c_i)$;
 - 6: **Fim Para**
 - 7: **Fim Se**
 - 8: Seja g_l o conjunto de arestas que estão em modo ativo;
 - 9: Os nós ativos executam $CombineTS(g_l, C)$;
 - 10: Seja v_m a última estação ativa após o passo anterior;
 - 11: O nó v_m utiliza os conjuntos de transmissão $s_{v_1} \cup s_{v_2} \cup \dots \cup s_{v_n}$ para construir G ;
-

passo, k chamadas da rotina $CombineGroup$ são realizadas, linhas 4-5. Como descrito anteriormente, a rotina $CombineGroup$ irá combinar os conjuntos de transmissão em cada grupo g_i em apenas um nó por grupo com os demais nós em modo inativo. A rotina $CombineTS$ é chamada para todos os nós que ainda estão ativos. A rotina irá garantir que todos os conjuntos de transmissão serão combinados e enviados para um único nó em V , linha 9-10. Este único nó, chamado v_m , possuirá todos os conjuntos de transmissão da rede. Por último, o nó v_m utiliza todos os conjuntos de transmissão para construir o grafo de comunicação da rede $G = (V, E)$, linha 11. Note que ao final, o nó v_m possui condições de verificar, para cada nó $v_i \in V$, o conjunto d_{v_i} . Para referência futura, considere o Lema 3.

Lema 3 *Dado um conjunto V de nós e um conjunto C de canais, onde $|V| = n$ e $|C| = k$, o Algoritmo ManagementStage (Algoritmo 3.4) combina todos os conjuntos de transmissão em $O(\lceil \frac{n}{k} \rceil + \log k)$ fatias de tempo com o consumo de energia individual por nó de até $\log k + 3$ fatias de tempo em modo ativo.*

Prova do Lema 3 *O Algoritmo ManagementStage (Algoritmo 3.4), no pior caso onde a condicional $k < \lfloor \frac{n}{2} \rfloor$ é verdadeira (linha 2), divide os n nós em k grupos. Cada grupo possui até $\lceil \frac{n}{k} \rceil$ elementos e executa em paralelo a rotina $CombineGroup$ (linha 5) que, seguindo o Lema 2, possui tempo de execução de $\lceil \frac{n}{k} \rceil - 1$ fatias de tempo e consumo de energia de 2 fatias de tempo. Em seguida, o algoritmo executa a rotina $CombineTS$ (linha 9) que, segundo o Lema 2, possui um tempo de execução da ordem de $\log k + 1$ fatias de tempo e consumo de energia de $\log k + 1$ fatias de tempo. Dessa forma, o tempo total de execução e consumo de energia da fase de gerenciamento do protocolo EEMC-MAC consistem, respectivamente, na soma do tempo de execução e consumo de energia das rotinas $CombineGroup$ e $CombineTS$. Dessa forma, temos que o tempo de comunicação é da ordem de $O(\lceil \frac{n}{k} \rceil + \log k)$ fatias de tempo enquanto o consumo de energia é da ordem de $O(\log k)$ fatias de tempo.*

EEMC-MAC: Fase de Transmissão

O estágio de transmissão do protocolo EEMC-MAC começa imediatamente após o estágio de gerenciamento. Nesse estágio, o nó líder v_m já computou o grafo de comunicação

G. O Algoritmo 3.5 descreve o estágio de transmissão do protocolo EEMC-MAC. No começo do algoritmo, o nó líder v_m possui condições de construir o grafo de comunicação G . Para resolver as dependências nas comunicações, o nó líder executa a heurística *ECOH* e obtém os conjuntos de comunicação $CS = \{CS_1, CS_2, \dots, CS_r\}$, linha 2. A heurística *ECOH* garante que $k \leq |CS_i|$, isto é, cada conjunto de comunicação CS_i não possui mais elementos que o número de canais disponíveis e seus elementos são disjuntos. Isto significa que todas as transmissões representadas em cada conjunto não possuem dependências entre si. Em um próximo passo, todos os nós entram em modo ativo e sintonizam no canal c_1 para receber a difusão de CS do nó líder v_m e então retornam para o modo inativo, linha 3. O primeiro laço vai de 1 até r (o número de conjuntos de comunicação), linhas 4-12, e o segundo vai de 1 até o número de elementos no conjunto de comunicação indicado no primeiro laço, linhas 5-11. Os nós em cada conjunto entram em modo ativo, linha 8, e sintonizam em um canal apenas quando possuem dados para enviar ou receber, linha 9, retornando ao modo inativo após a transmissão ou recepção.

Lema 4 *O Algoritmo TransmissionStage (Algoritmo 3.4) necessita entre $\Omega(\lceil \frac{p}{k} \rceil)$ e $O(p)$ fatias de tempo para execução e possui um gasto energético individual em cada nó $v_i \in V$ de $\tau_{v_i} + 1$ fatias de tempo em modo ativo.*

Prova do Lema 4 *O tempo de execução do estágio de transmissão depende de como a heurística *ECOH* cria a lista de conjuntos de comunicação $CS = \{CS_1, CS_2, \dots, CS_r\}$. Analisando o Algoritmo 3.5, temos que r fatias de tempo são necessárias para transmitir todos os elementos em CS , uma vez que cada conjunto CS_i não possui dependências e podem ser transmitidos em paralelo, mais 1 fatia de tempo para a difusão de CS . Temos que, no pior caso onde nenhuma transmissão pode ser paralelizada, r será igual ao número de arestas do grafo de comunicação, ou seja, $r = p$. No entanto, no melhor caso, todas as transmissões em E podem ser paralelizadas nos k canais disponíveis, ou seja, $r = \lceil \frac{p}{k} \rceil$. Dessa forma, o tempo total do estágio de transmissão fica entre $\Omega(\lceil \frac{p}{k} \rceil)$ e $O(p)$ fatias de tempo. Considera-se que a execução da heurística *ECOH* e a difusão de CS podem ser realizadas numa mesma fatia de tempo.*

3.4.3 EEMC-MAC: Procedimento Principal e Complexidades Envolvidas

O procedimento principal do protocolo EEMC-MAC consiste em executar as fases de gerenciamento e de transmissão em sequência. O fluxograma apresentado pela Figura 3.2 ilustra a sequência de eventos necessários para a execução do protocolo. Após a inicialização do protocolo, os nós utilizam mecanismos para permitir a sincronização com o restante da rede. Essa sincronização é uma hipótese do modelo de comunicação, no entanto, um mecanismo de sincronização como o descrito por [72] poderia ser utilizado nesse momento. Após a sincronização, assume-se que os nós possuem um ID único na rede e sabem o número de canais que estão disponíveis. A fase de gerenciamento é iniciada após a sincronização ter sido finalizada. Se existirem canais suficientes, os nós executam a rotina *CombineTS* diretamente. Caso contrário, eles executam a rotina *CombineGroup* antes. A fase de transmissão é iniciada após os nós finalizarem a execução da rotina *CombineTS*. Nesse momento, o nó líder executa o Algoritmo *ECOH* para realizar o

Algoritmo 3.5 : TransmissionStage

- 1: Seja v_m o nó líder da rede (do estágio anterior) com o grafo de comunicação G ;
 - 2: O nó v_m executa $ECOH(G, k)$ e obtém os conjuntos de comunicação $CS = \{CS_1, CS_2, \dots, CS_r\}$;
 - 3: Todos os nós em V entram em modo ativo e sintonizam no canal c_1 . O nó v_m difunde CS no canal c_1 . Após todos os nós em V receberem a difusão de CS eles entram em modo inativo;
 - 4: **Para** ($i \leftarrow 1$ até r) **faça**
 - 5: **Para** ($j \leftarrow 1$ até $|CS_i|$) em paralelo **faça**
 - 6: Seja $e_h = \{v_s, v_d\}$, $e_h \in CS_i$;
 - 7: $CS_i \leftarrow CS_i \setminus e_h$;
 - 8: Os nós v_s e v_d entram em modo ativo;
 - 9: O nó v_s envia um pacote para v_d usando o canal c_j ;
 - 10: Os nós v_s e v_d entram em modo inativo;
 - 11: **Fim Para**
 - 12: **Fim Para**
-

agendamento das transmissões e posteriormente faz a difusão desse agendamento. Todos os nós, sabendo o agendamento, ficam ativos apenas quando tiverem dados para enviar ou receber, finalizando uma rodada de execução do protocolo EEMC-MAC após isso. O Teorema 1 sintetiza as complexidades do protocolo EEMC-MAC, concatenando os Lemas 3 e 4.

Teorema 1 *O protocolo EEMC-MAC resolve o acesso ao meio em múltiplos canais e agendamento de transmissões em uma rede sem fio síncrona, de único salto, com o tempo dividido em fatias de tempo, representada por um grafo de comunicação $G = (V, E)$ em $O(\lceil \frac{n}{k} \rceil + \log k + p)$ fatias de tempo, com cada nó $v_i \in V$ ativo em $O(\log k + \tau_{v_i})$ fatias de tempo, onde $|V| = n$, $|E| = p$, $|C| = k$ e τ_{v_i} é a quantidade de dados que o nó v_i possui para enviar e receber.*

Prova do Teorema 1 *Considerando que o protocolo EEMC-MAC consiste na execução em seqüência das fases de gerenciamento e transmissão, o tempo de execução e consumo de energia consistem na soma dos custos das respectivas fases. Esses custos podem ser obtidos dos Lemas 3 e 4.*

3.4.4 EEMC-MAC: Exemplo de Funcionamento

Para exemplificar o funcionamento do protocolo, considere o grafo de comunicação representado pela Figura 3.3. Nesse grafo, temos 8 vértices, $V = \{v_1, v_2, \dots, v_8\}$, e 12 arestas, $E = \{e_1, e_2, \dots, e_{12}\}$. Considere a presença de $k = 4$ canais de comunicação.

A Figura 3.4 representa como seriam as transmissões de dados utilizando 4 canais, o grafo de comunicação anterior e o protocolo EEMC-MAC. O protocolo inicia na fase de gerenciamento (Algoritmo 3.4). Como o número de canais é suficientemente grande (a relação $k < \frac{n}{2}$ é falsa), a rotina *CombineTS* (Algoritmo 3.3) é chamada. Essa rotina irá agrupar os conjuntos de transmissão s_{v_i} dos nós em V utilizando múltiplos canais até o nó

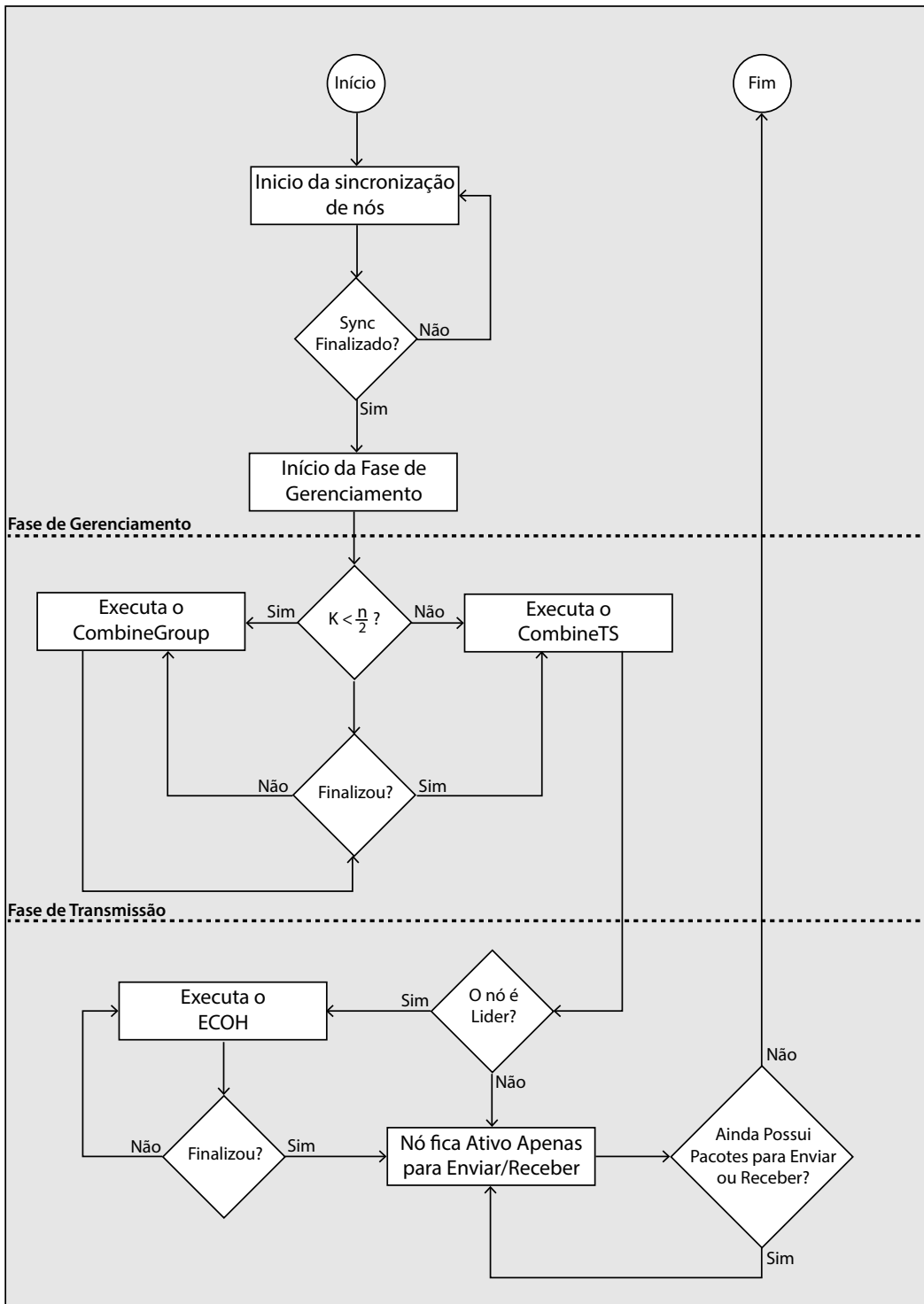


Figura 3.2: Fluxograma do protocolo EEMC-MAC.

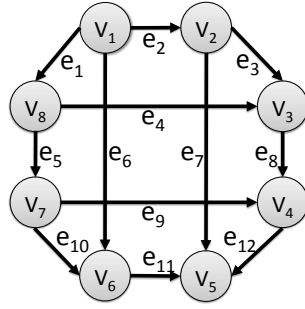


Figura 3.3: Exemplo de grafo de comunicação com $n = 8$ nós.

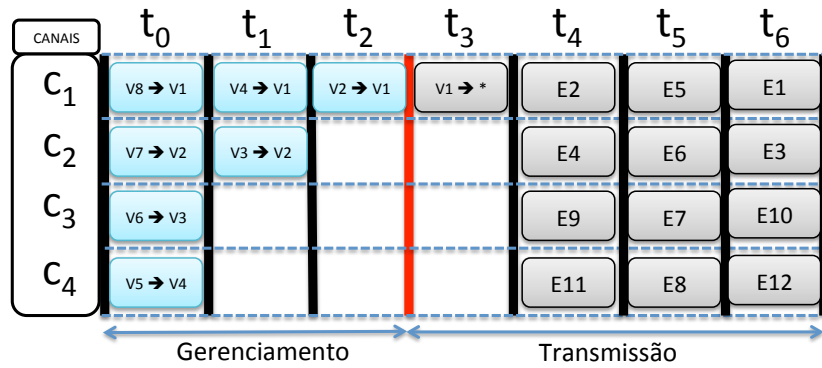


Figura 3.4: Possível representação gráfica para o uso de 4 canais de comunicação durante a execução do protocolo EEMC-MAC para o grafo de comunicação representado pela Figura 3.3.

líder v_1 obtenha todos os conjuntos de transmissão, fatias de tempo t_0 a t_2 . Esse procedimento de agrupar os conjuntos de transmissão finaliza a etapa de gerenciamento. A etapa de transmissão (Algoritmo 3.5) inicia logo após a etapa de gerenciamento. Nessa etapa, o nó líder v_1 utiliza o algoritmo de resolução de dependências ECOH (Algoritmo 3.1) para obter a lista de conjuntos de comunicação CS . Essa lista permite realizar o agendamento de transmissões, contendo o canal e fatia de tempo cada nó deve sintonizar para enviar e/ou receber dados. Observe que o algoritmo ECOH garante que transmissões que ocorrem na mesma fatia de tempo não devem ter vértices em comum. O nó líder então difunde essas listas na fatia de tempo t_3 . As fatias que seguem são as transmissões dos pacotes agendados, t_4 a t_6 .

3.5 Simulação

A avaliação do protocolo proposto foi realizada através de simulação. Para esse propósito, um simulador foi desenvolvido incorporando as características descritas na Seção 3.4. O tempo de execução do protocolo EEMC-MAC é comparado com os valores da solução ótima. Investiga-se também o impacto do estágio de gerenciamento no tempo total de execução do protocolo. Essa seção descreve os parâmetros de simulação, as métricas de avaliação e expõe os resultados obtidos.

3.5.1 Parâmetros de Simulação e Métricas de Avaliação

A simulação foi conduzida variando o número de nós, número de pacotes por nó (0 ou 1 pacote para cada destino) e número de canais de comunicação. O número de nós (n) utilizados nas simulações assumem os seguintes valores $n = 8, 16, 32, 64, 128, 256$. O número de transmissões de dados em cada nó é definido com base no percentual máximo de transmissões possíveis. Mais precisamente, utilizou-se 5 configurações de transmissão, variando o número de transmissões entre: 0% a 20%, 21% a 40%, 41% a 60%, 61% a 80% e 81% a 100% do número máximo de transmissões possíveis. Por exemplo, em uma configuração com 8 nós utilizando a primeira configuração (0% até 20%), existirão de 0 até, no máximo, 11 arestas ($20\% * (8 * 7) = 11.2$) no grafo de comunicação. Uma vez que um grafo direcionado de n nós possui até $n * (n - 1)$ arestas. O número de canais k assume valores entre 2^0 até $2^{\lfloor \frac{n}{2} \rfloor}$.

Os parâmetros propostos permitem avaliar grafos de comunicação tanto esparsos (0% a 20% de transmissões) quanto densos (81% a 100% de transmissões), além de permitir que o máximo de canais sejam usados no sistema sem superar o número máximo de paralelizações possíveis. Cada configuração de simulação é realizada 100 vezes e uma média aritmética é tomada para os resultados. Lembre que uma aresta no grafo de comunicação equivale a uma transmissão de dados entre nós adjacentes.

O desempenho do protocolo em relação ao tempo de execução é avaliado considerando as seguintes métricas: (i) a porcentagem de tempo que o protocolo necessitou para o estágio de transmissão (α); e (ii) a razão entre o tempo do estágio de transmissão pelo que seria o tempo ótimo (γ). A métrica α é definida como:

$$\alpha = \frac{T_t}{T_m + T_t}, \quad (3.1)$$

onde T_m é o número de fatias de tempo necessárias para concluir o estágio de gerenciamento e T_t é o número de fatias de tempo para concluir o estágio de transmissão. Note que valores altos para α (próximos de 1) indicam que o custo de gerenciamento foi baixo comparado com o tempo total de execução do protocolo. A métrica γ é definida de maneira similar:

$$\gamma = \frac{T_t}{T'_t}, \quad (3.2)$$

onde T'_t é o tempo ótimo do estágio de transmissão (em fatias de tempo). Os valores de γ indicam a diferença entre o tempo de execução do estágio de transmissão do protocolo e o tempo ótimo. Note que, quanto $\gamma = 1$, o protocolo EEMC-MAC alcançou o tempo mínimo possível para completar o estágio de transmissão.

3.5.2 Resultados de Simulação

Nesta seção, apresentamos os resultados de simulação do protocolo EEMC-MAC para os parâmetros de simulação descritos anteriormente. A Figura 3.5 de (a) até (f) apresenta os valores para a métrica “porcentagem de tempo do protocolo em estágio de transmissão” (α). No eixo x temos uma variação no número de canais enquanto no eixo y temos os valores para a métrica α (em porcentagem). Note que, a medida que o número de nós e transmissões aumenta, a porcentagem de tempo requerida para a fase de transmissão

diminui, se tornando quase desprezível. Tomando a média das simulações, temos que o tempo gasto com gerenciamento é inferior à 5% do tempo total de execução.

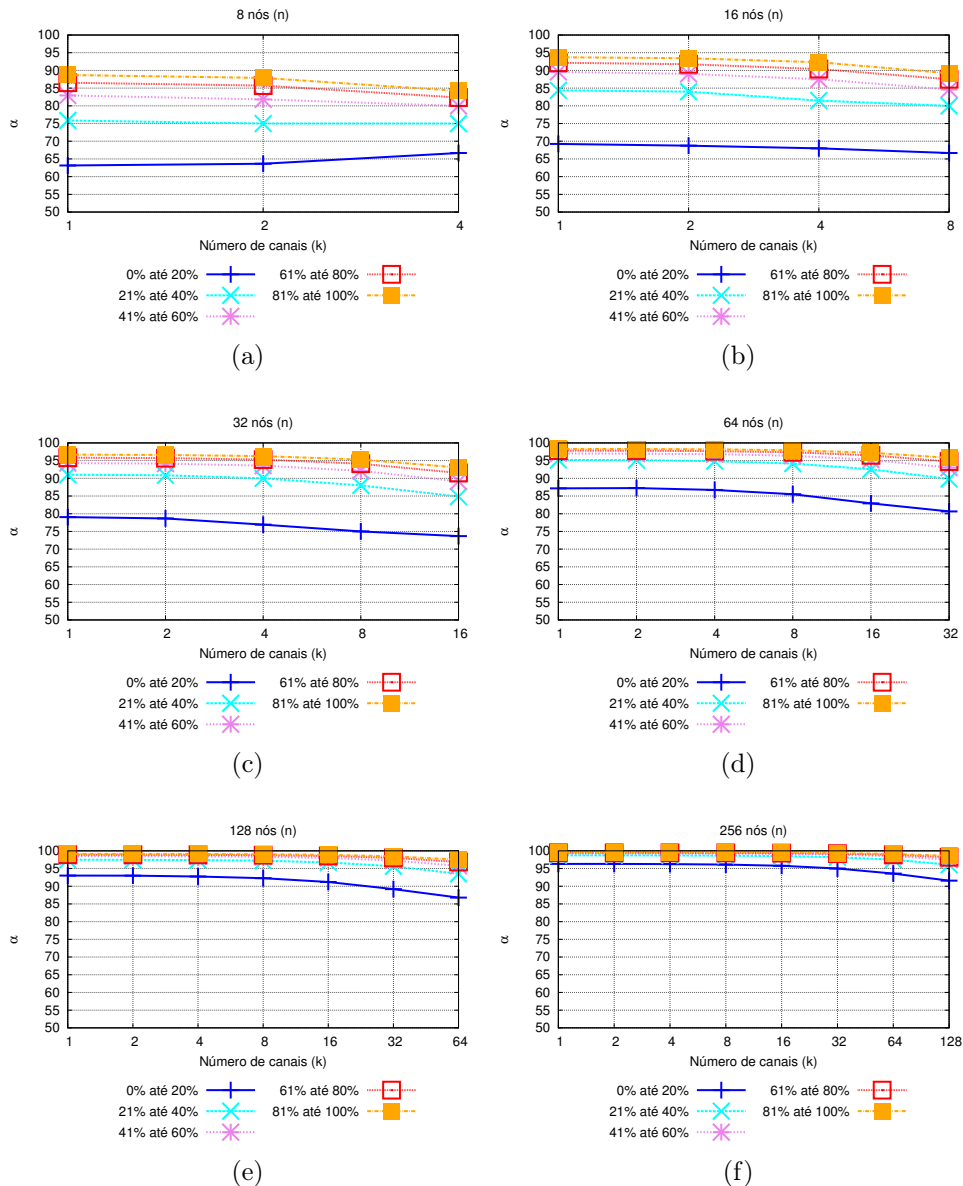


Figura 3.5: Porcentagem de tempo do protocolo em estágio de transmissão para várias configurações (métrica α).

Os resultados apresentados na Figura 3.5 mostraram que o tempo necessário para o estágio de gerenciamento do protocolo é muito pequeno. Com o objetivo de se ter uma indicação justa do tempo do estágio de transmissão, o protocolo é comparado com a atribuição ótima de canais. Do teorema de Vizing [73] temos que o número mínimo de cores necessárias para colorir um grafo está entre $\Delta(G)$ and $\Delta(G)+1$, onde $\Delta(G)$ é o maior grau do grafo. Posto de outra forma, verifica-se que todas as transmissões representadas com a mesma cor no grafo de comunicação são disjuntas e podem ser paralelizadas na mesma fatia de tempo. A tarefa de associar as arestas de G a cada cor e garantir o

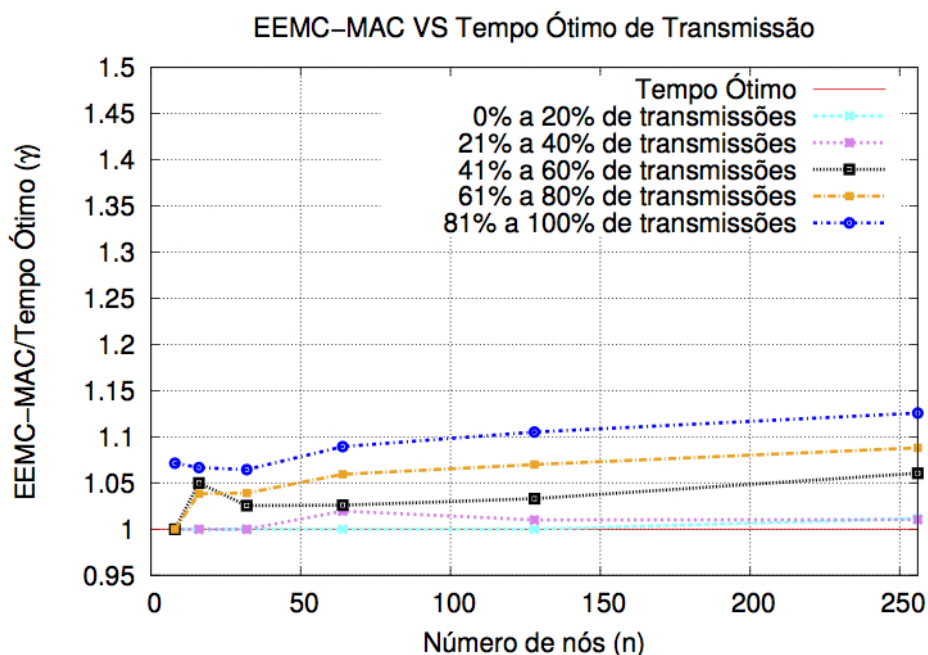


Figura 3.6: Comparação entre o tempo do estágio de transmissão pelo tempo ótimo (métrica γ).

arranjo ótimo é uma tarefa complexa. No entanto, este resultado pode ser uma cota inferior de comparação. Para permitir tal comparação, consideramos que $k = \lfloor \frac{n}{2} \rfloor$ canais estão disponíveis. Assim, é garantido que cada par de nós na rede possuirá um canal de comunicação disponível para comunicação. Por conveniência, definimos que $T_{\nu} = \Delta(G) + 1$, onde 1 é o tempo para difusão das informações de agendamento ótimo de canais.

A Figura 3.6 apresenta os resultados de simulação para a métrica que compara o tempo de execução do estágio de transmissão do protocolo com o tempo ótimo de execução (γ). No eixo x varia-se o número de vértices enquanto no eixo y é apresentado os valores para a métrica γ . Temos também cinco variações na quantidade de transmissões por grafo de comunicação. Observe que, em diversos casos, o tempo ótimo foi obtido ($\gamma = 1$) e também que todos os cenários de simulação ficaram abaixo do valor $\gamma < 1,15$, ou seja, em todos os cenários de simulação, o tempo do EEMC-MAC esteve sempre a menos de 15% do tempo ótimo. Quando tomamos a média dos valores obtidos, temos que o protocolo EEMC-MAC ficou a menos de 4% do tempo ótimo no estágio de transmissão. Note que o protocolo EEMC-MAC possui a característica de ser escalável, uma vez que, para diversas variações de nós, não houve uma variação significativa no tempo de execução quando comparado com o tempo ótimo.

3.6 EEMC-MAC para Múltiplos Saltos

Nesta seção, é discutido como o EEMC-MAC poderia ser estendido para múltiplos saltos utilizando um esquema de *clusters*. *Younis et al.* [74] propõem o HEED, uma abordagem de *cluster* eficiente em energia para redes *ad hoc* distribuídas. Esta abordagem se

encaixa na natureza determinística do EEMC-MAC uma vez que o processo de construção de *clusters* é concluído após um número constante de iterações (independente do diâmetro da rede) e o custo de controle é linear no tamanho de nós da rede. Cada nó utiliza apenas informações locais no processo de construção dos *clusters*.

Assume-se que, no início do protocolo EEMC-MAC para múltiplos saltos, todos os nós da rede estão organizados em *clusters*, seguindo o protocolo HEED [74]. Cada *cluster* possui as seguintes características: tempo síncrono; comunicação de único salto; uma lista dos canais disponíveis; e um nó líder do *cluster* CH (*Cluster Head*). O CH possui as seguintes deveres: agir como o nó líder do protocolo EEMC-MAC, sendo responsável por agrupar todos os conjuntos de transmissão, criar e entregar o escalonamento dos dados para os demais nós do *cluster*; criar e manter uma tabela de roteamento inter-*cluster*; e realizar a comunicação inter-*cluster*. Nós em *clusters* diferentes não são necessariamente sincronizados. Assume-se um canal comum para comunicação inter-*cluster*.

O EEMC-MAC pode ser executado dentro de cada *cluster* com pequenas modificações no algoritmo original. Se um nó v_s de um *cluster* A quiser enviar um pacote para o nó v_d de um *cluster* B , ele envia o pacote para CH do *cluster* A , que armazena esse pacote na sua memória local. Observe que, do protocolo EEMC-MC, o CH sabe exatamente, de ambos os estágios (gerenciamento e transmissão), quais fatias de tempo estão livres para transmitir ou receber dados. Dessa maneira, o CH utiliza essas fatias de tempo livres para realizar a comunicação inter-*cluster* no canal de comunicação comum, utilizando algoritmos para descoberta de rotas, se necessário. Quando o CH do *cluster* B receber o pacote retransmitido pelo CH do *cluster* A , ele irá adicionar esse pacote ao seu conjunto de transmissão, entregando o pacote ao nó v_d no próximo ciclo do EEMC-MAC.

3.7 Discussão

Neste capítulo, foi apresentada uma das propostas desta dissertação. A proposta desse capítulo é um protocolo, chamado EEMC-MAC, que pode ser utilizado no agendamento de transmissões eficiente em energia em redes sem fio com múltiplos canais. O EEMC-MAC combina um esquema que faz os nós da rede alternarem entre modos ativo e inativo com uma heurística que, ao mesmo tempo, resolve as dependências nas comunicações e faz o agendamento das transmissões. Para tal, O EEMC-MAC divide tais tarefas em fases de gerenciamento e transmissão. O gasto energético na fase de gerenciamento é da ordem de $O(\log k)$ fatias de tempo em modo ativo por nó na rede, onde k é o número de canais do sistema. Resultados empíricos mostram que a fase de gerenciamento representa menos de 5% do tempo total de operação do protocolo. O estágio de transmissão é ótimo em energia e, quando comparado com o tempo ótimo de transmissão, este é superior em apenas 4% nos casos simulados. O protocolo também possui a característica de ser escalável com o crescimento do número de nós da rede. Por fim, foi discutido como o EEMC-MAC poderia ser estendido para funcionar com múltiplos saltos. Em trabalhos futuros, pretende-se comparar o protocolo com outros da literatura, estender a proposta para múltiplos saltos além de relaxar algumas limitações do cenário de comunicação como: presunção de que os nós estão sincronizados e ausência de interferência externa.

Capítulo 4

Controle de Topologia em Redes *Ad Hoc* Cooperativas

Comunicação cooperativa é uma técnica da camada física que permite combinar versões diferentes de um sinal para obter a mensagem original. Essa técnica tem sido combinada com sucesso com técnicas de controle de topologia em redes *ad hoc*, uma vez que CC permite aumentar o alcance de transmissão dos nós da rede, podendo ser utilizada para aumentar a conectividade em uma rede sem fio [39, 54]. Este capítulo propõe uma nova técnica, chamada *CoopSink*, que faz o controle de topologia em redes *ad hoc* cooperativas. Mais especificamente, o *CoopSink* utiliza comunicação cooperativa para aumentar a conectividade da topologia em relação a um nó fixo, que pode ser um nó sorvedouro em uma rede *ad hoc*. Além disso, o *CoopSink* busca manter as rotas para o sorvedouro o mais eficientes possíveis. A Seção 4.1 faz uma breve introdução sobre o contexto em que essa técnica se insere e fala da motivação para este capítulo. A Seção 4.2 descreve os modelos de comunicação e rede considerados e formaliza o problema principal tratado nesse trabalho. A Seção 4.3 descreve a técnica. A Seção 4.4 apresenta os resultados da simulação que compara o *CoopSink* com outras técnicas da literatura. A Seção 4.5 faz uma discussão do que foi apresentado no capítulo.

4.1 Contextualização e Motivação

Redes *ad hoc* são redes sem fio que não utilizam uma infraestrutura para se conectar. Como não existe infraestrutura, o alcance do rádio é limitado e a comunicação é feita por cooperação entre os nós. Essas redes possuem uma série de aplicações civis e militares e têm atraído bastante atenção nos últimos anos. Um dos maiores desafios nesse tipo de rede é reduzir o consumo de energia, uma vez que os dispositivos sem fio tipicamente operam por meio de baterias. Controle de topologia [75, 62, 76] é uma estratégia amplamente utilizada em redes *ad hoc* para obter economia de energia. Essa estratégia consiste em permitir que os nós sem fio selecionem um subconjunto de nós vizinhos ou ajustem sua potência de transmissão com o objetivo de conservar energia [77, 78] e/ou manter a conectividade da rede [58, 79, 80].

Comunicação Cooperativa (CC) é uma técnica que permite que dispositivos com uma única antena se beneficiem de algumas das vantagens dos sistemas MIMO como o uso da diversidade espacial [35]. Em redes de múltiplos saltos tradicionais, nós intermediários

cooperam com um nó fonte retransmitindo o dado para o nó destino, processo que é realizado pela camada de rede. Dessa maneira, o nó destino recebe apenas uma cópia do dado enviado pelo nó fonte. CC, por outro lado, é uma técnica da camada física. Nessa técnica, quando um nó fonte transmite um dado, nós ajudantes em volta do fonte sobre-escutam o sinal e retransmitem simultaneamente várias cópias independentes do mesmo sinal para o nó destino. O nó destino é então capaz de combinar os sinais recebidos e obter o dado original.

Existe um grande esforço de pesquisa para aprimorar os mecanismos da camada física para permitir CC e estudos em camadas superiores estão começando a ganhar importância. Uma vez que CC pode ser utilizado para melhorar a conectividade, expandir a cobertura e melhorar o consumo de energia, a técnica tem sido aplicada em diversas áreas como: difusão [48, 49, 50], roteamento [51, 52], rádios cognitivos [53] e controle de topologia [55, 39, 54]. Este capítulo foca no uso de CC com controle de topologia.

Cardei et al. [55] estudam o problema de controle de topologia sobre o modelo de CC, com o objetivo de obter topologias fortemente conexas e com mínimo consumo de energia. Em seguida, mostram que esse problema é NP-completo e propõem dois algoritmos distribuídos e localizados que começam a partir da topologia resultante de um algoritmo tradicional de controle de topologia (sem utilizar CC). O primeiro algoritmo utiliza um processo de decisão distribuído em cada nó e que faz uso de informações de vizinhos de até 2 saltos. O segundo algoritmo atribui potências de transmissão aos nós iterativamente utilizando informações de vizinhos de um salto.

Yu et al. [39] utilizam controle de topologia e CC com o objetivo de aumentar a conectividade da rede e diminuir a potência de transmissão dos nós. O algoritmo proposto primeiramente constrói todos os candidatos a arestas bidirecionais utilizando o modelo de CC, conecta componentes desconexos e então gera uma estrutura de árvore para reduzir o consumo de energia.

Zhu et al. [54] consideram a eficiência energética dos caminhos entre os nós no seu processo de controle de topologia com CC. Propõem dois algoritmos que constroem uma topologia garantindo uma eficiência mínima para as rotas criadas.

Enquanto os trabalhos da literatura possuem seu foco em aumentar a conectividade e/ou diminuir o consumo de energia de maneira igualitária dos nós e das rotas, salvo melhor juízo, nenhum trabalho até então explorou comunicação cooperativa para aumentar a conectividade com o sorvedouro em uma rede de sensores. Essa aplicação é particularmente interessante pois permite que sensores distantes, antes sem a possibilidade de comunicação, se tornem capazes de se comunicar com o sorvedouro. Nesse contexto, a conectividade seria aumentada em direção ao *sorvedouro*, e esse poderia ser equipado com um rádio de maior potência para enviar as consultas, assim como é proposto em [81]. Nesse capítulo, é proposta uma nova técnica, chamada *CoopSink*, que utiliza CC e controle de topologia em redes de sensores para aumentar a conectividade com um nó sorvedouro, além de garantir a eficiência das rotas.

4.2 Modelo e Definição do Problema

Nesta seção, primeiramente descreve-se o modelo de comunicação cooperativa e o correspondente modelo de rede utilizado nas formulações e algoritmos deste capítulo.

Em seguida, exemplifica-se o modelo e o problema principal tratado neste capítulo é formalizado. O modelo definido nessa seção foi baseado no utilizado por *Zhu et al.* [54]

4.2.1 Modelo de Comunicação Cooperativa

Cada nó v_i pode ajustar sua potência de transmissão P_i com valores dentro do intervalo $[0, P_{MAX}]$, sendo que, quando $P_i = 0$, o rádio está desligado e, quando $P_i = P_{MAX}$, o rádio opera em sua potência máxima. Em modelos tradicionais de comunicação cooperativa, um nó transmissor v_i pode se comunicar com um nó receptor v_j diretamente, apenas quando a potência de transmissão de v_i satisfaz a Equação (4.1).

$$P_i(d_{i,j})^{-\alpha} \geq \tau \quad (0 \leq P_i \leq P_{MAX}), \quad (4.1)$$

onde: α é expoente de perda de caminho (do Inglês, *Path Loss Exponent – PLE*), que usualmente fica entre 2 e 4 e simboliza a taxa desvanecimento do sinal com a distância; $d_{i,j}$ é a distância Euclidiana entre os nós v_i e v_j ; e τ é a média da razão sinal-ruído (do Inglês, *Signal-to-Noise Ratio – SNR*) recebido pelo receptor, de modo que o sinal recebido possa ser decodificado com sucesso. Assim como em [39, 54], este trabalho foca em um regime limitado de interferência, onde o ruído é pequeno comparado com o sinal.

Comunicação Cooperativa (CC) [35] tira vantagem do desenho da camada física para combinar sinais parciais e obter a informação completa. Dessa maneira, uma comunicação completa entre o nó v_i e o nó v_j pode ser alcançada com CC se v_i transmite o sinal com um conjunto de nós ajudantes $H_{i,j}$ e a soma de suas potências de transmissão satisfazem a Equação (4.2).

$$\sum_{v_k \in v_i \cup H_{i,j}} P_k(d_{k,j})^{-\alpha} \geq \tau \quad (0 \leq P_i \leq P_{MAX}). \quad (4.2)$$

As Figuras 4.1a e 4.1b exemplificam um cenário onde CC poderia ser utilizado para aumentar a conectividade em uma rede *ad hoc*. Na Figura 4.1a, temos três nós próximos (v_1, v_2 e v_3) e um nó distante (v_4). O raio de transmissão, isto é, a potência de transmissão de v_1 permite que os nós v_2 e v_3 sejam alcançados diretamente, v_2 e v_3 não conseguem alcançar outro nó além de v_1 e o nó v_4 não consegue alcançar nenhum nó diretamente.

Utilizando CC, o nó v_1 poderia selecionar os nós v_2 e v_3 como ajudantes para transmitir para v_4 , isto é, $H_{1,4} = \{v_2, v_3\}$. Após selecionar esses nós como ajudantes, o nó v_1 transmite seu dado para v_2 e v_3 em um primeiro momento, então, os três juntos transmitem o mesmo dado para v_4 em um segundo momento, ampliando o raio de transmissão de v_1 . Se o SNR combinado for maior do que τ em v_4 , esse nó seria capaz de decodificar o sinal recebido e recuperar o dado oriundo de v_1 , assim como ilustrado pela Figura 4.1b. Quando o nó v_1 transmite seu dado para os nós ajudantes v_2 e v_3 no primeiro momento, v_4 também recebe dados parciais de v_1 . Em alguns modelos de CC tal dado parcial é utilizado por v_4 para ajudar na decodificação do dado durante o segundo momento da comunicação cooperativa. Neste trabalho, este dado parcial (vindo de v_1 no primeiro momento) é ignorado por razões de simplicidade (assim como [39] e [54] fizeram). Técnicas da camada física para implementação de CC podem ser encontradas em [51].

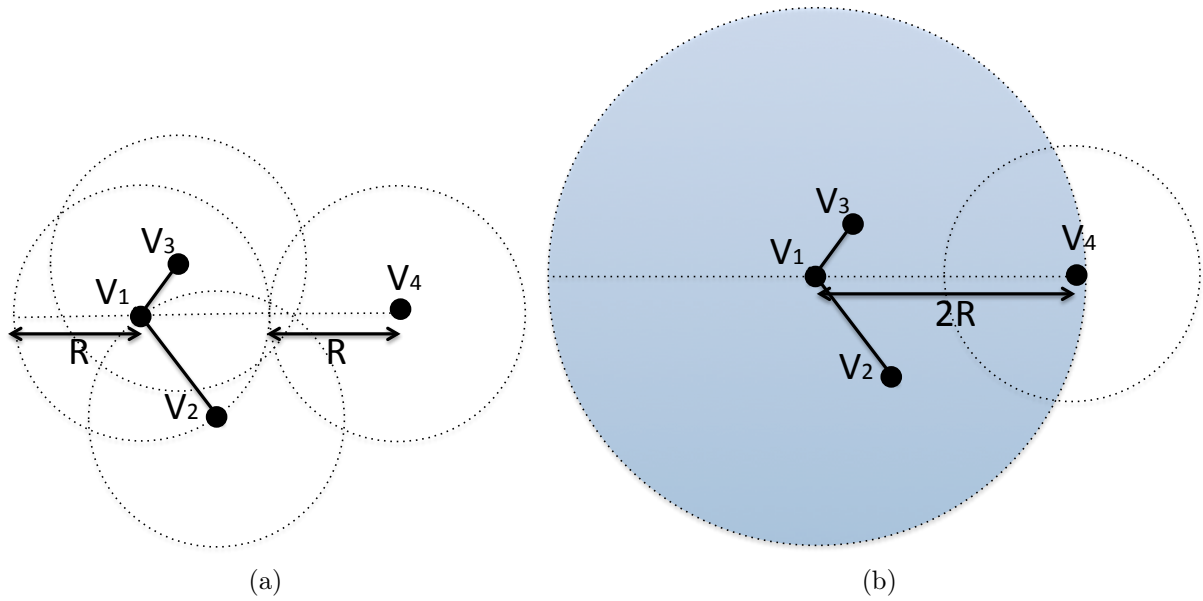


Figura 4.1: (a) Cenário onde existem 3 nós próximos (v_1 , v_2 e v_3) e um nó distante (v_4). (b) Aumento do raio de cobertura de v_1 com a utilização nós ajudantes (v_2 e v_3) para criar um enlace cooperativo entre v_1 e v_4 .

4.2.2 Modelo de Rede Cooperativa

Considere uma rede sem fio *ad hoc* com n nós que são capazes de receber e combinar sinais parciais recebidos em acordo com o modelo de comunicação cooperativa. A topologia de rede é modelada com grafo direcionado plano: $G_P = (V, E_P)$, onde $V = \{v_1, v_2, \dots, v_n\}$ é um conjunto que representa os nós de uma rede *ad hoc* e E_P é um conjunto de arestas que ligam dois nós distintos de V . Uma aresta $v_i v_j \in E_P$ simboliza que o nó v_i pode transmitir um dado para o nó v_j tanto diretamente quanto usando CC. $N(v_i)$ é o conjunto de vizinhos diretos de v_i dentro do seu alcance máximo de transmissão R_{max} , isto é, para todos $v_k \in N(v_i)$, existe $P_i \leq P_{MAX}$ tal que $P_i(d_{i,k})^{-\alpha} \geq \tau$, seguindo a Equação (4.1). Assume-se que cada nó $v_i \in V$ possui um ID único e também uma visão global da topologia da rede. Cada nó em $v_i \in V$ possui uma única antena e opera por baterias. Dado as representações anteriores, define-se diversos conceitos e métricas de avaliação.

Definição 1 (Aresta direta): Uma aresta direta $\overline{v_i v_j}$ é uma aresta pertencente a E_P , representando que um nó v_i pode transmitir informação para o nó v_j diretamente, isto é, a potência P_i de v_i é capaz de alcançar v_j com $P_i \leq P_{MAX}$. Utiliza-se uma traço horizontal reto para identificar uma aresta direta.

Definição 2 (Conjunto de nós ajudantes): $H_{i,j}$ representa o conjunto de nós ajudantes de v_i na comunicação cooperativa para v_j . Assume-se que todos os nós ajudantes precisam ser vizinhos diretos de v_i , isto é, $H_{i,j} \subseteq N(v_i)$, onde $N(v_i)$ é o conjunto com todos os vizinhos de v_i . Em outras palavras, todos os elementos em $N(v_i)$ são candidatos a nós ajudantes de v_i .

Definição 3 (Aresta CC): Uma aresta CC $\widetilde{v_i v_j}$ é uma aresta de E_P que representa que um nó v_i pode transmitir dados para o nó v_j cooperativamente, utilizando um conjunto de

nós ajudantes $H_{i,j}$. Utiliza-se uma traço horizontal ondulado para identificar uma aresta CC.

Definição 4 (Aresta ajudante): Uma aresta ajudante é uma aresta entre v_i e um dos seus nós ajudantes em $H_{i,j}$. Por exemplo, na Figura 4.1b, o nó v_1 utiliza os nós v_2 e v_3 como nós ajudantes para criar uma aresta CC entre v_1 e v_4 , isto é $H_{1,4} = \{v_2, v_3\}$, dessa forma, as arestas $\overline{v_1v_2}$ e $\overline{v_1v_3}$ são consideradas arestas ajudantes.

Definição 5 (Topologia de rede): A união de todas as arestas diretas e arestas CC são representadas por $\overline{E_P}$ e $\widetilde{E_P}$, respectivamente. Similarmente, os grafos de comunicação direto e CC são definidos por $\overline{G_P} = (V, \overline{E_P})$ e $\widetilde{G_P} = (V, \widetilde{E_P})$, respectivamente. Note que $E_P = \overline{E_P} \cup \widetilde{E_P}$. Seguindo a notação, observe que se $v_i v_j \in E_P$, então $v_i v_j = \overline{v_i v_j}$ se $v_i v_j$ for uma aresta direta e que $v_i v_j = \widetilde{v_i v_j}$ se $v_i v_j$ for uma aresta CC.

Definição 6 (Peso de uma aresta direta): O peso de uma aresta $\overline{v_i v_j}$ é definido como:

$$w(\overline{v_i v_j}) = \tau d_{i,j}^\alpha. \quad (4.3)$$

Definição 7 (Peso de uma aresta CC): O peso de cada aresta $\widetilde{v_i v_j}$ é definido como:

$$w(\widetilde{v_i v_j}) = w_d(H_{i,j}) + (|H_{i,j}| + 1)w_{CC}(H_{i,j}), \quad (4.4)$$

onde:

- $|H_{i,j}|$: é o número de elementos do conjunto de nós ajudantes $H_{i,j}$;
- $w_d(H_{i,j}) = \left(\frac{\tau}{\max_{v_k \in H_{i,j}} (d_{i,k})^{-\alpha}} \right)$: é a potência mínima necessária para o nó v_i se comunicar diretamente com seu nó ajudante mais distante em $H_{i,j}$;
- $w_{CC}(H_{i,j}) = \left(\frac{\tau}{\sum_{v_k \in v_i \cup H_{i,j}} (d_{k,j})^{-\alpha}} \right)$: é a potência mínima necessária para o nó v_i se comunicar com v_j , transmitindo em conjunto com seus ajudantes em $H_{i,j}$.

Observe que, de acordo com as Equações (4.1) e (4.2), a seguinte relação deve ser verdadeira para existir uma aresta CC:

$$\max(w_d(H_{i,j}), w_{CC}(H_{i,j})) \leq P_{MAX}. \quad (4.5)$$

Para uma comunicação cooperativa ocorrer de v_i para v_j , o nó fonte v_i deve, em um primeiro momento, enviar seu dado para seus nós ajudantes em $H_{i,j}$ e, em um segundo momento, o nó v_i e seus ajudantes transmitem simultaneamente o mesmo dado para v_j . Dessa forma, o peso de uma aresta CC consiste na soma dos custos de comunicação nesses dois momentos. $w_d(H_{i,j})$ é o custo do primeiro momento da comunicação cooperativa enquanto $w_{CC}(H_{i,j})$ é o custo para um único nó transmitir cooperativamente, devendo ser multiplicado por $(|H_{i,j}| + 1)$, que é o número de nós que participam da comunicação cooperativa.

Neste trabalho, o modelo de CC é simplificado assumindo que as potências de transmissão de v_i e de seus nós ajudantes em $H_{i,j}$ são as mesmas. Além disso, só é considerado a potência de transmissão em cada nó transmissor.

Definição 8 (*Custo de um caminho direcional*): Dado um nó fonte v_i e um nó destino v_j em um grafo $G_P = (V, E_P)$, existe um caminho direcional entre v_i e v_j se, e somente se, existe uma sequência de vértices:

$$v_i, v_{i_1}, v_{i_2}, \dots, v_{i_{k-1}}, v_{i_k}, v_j \in V,$$

tal que:

$$(v_i v_{i_1}), (v_{i_1} v_{i_2}), \dots, (v_{i_{k-1}} v_{i_k}), (v_{i_k} v_j) \in E_P.$$

O custo de um caminho direcional entre v_i e v_j em um grafo G_P é definido por:

$$\pi_{G_P}(v_i, v_j) = w(v_i v_{i_1}) + w(v_{i_1} v_{i_2}) + \dots + w(v_{i_{k-1}} v_{i_k}) + w(v_{i_k} v_j). \quad (4.6)$$

Ou seja, a soma do peso de todas as arestas, tanto arestas diretas quanto arestas CC, que fazem parte de um caminho possível entre v_i e v_j . O custo do menor caminho em G_P entre os nós v_i e v_j é definido como:

$$\min(\pi_{G_P}(v_i, v_j)). \quad (4.7)$$

Definição 9 (*EFS - Energy Stretch Factor*): Seja $G'_P = (V, E'_P)$ um subgrafo de $G_P = (V, E_P)$, $E'_P \subseteq E_P$, G_P e G'_P são grafos conexos. O EFS de um par de nós v_i, v_j em G'_P com respeito aos mesmos nós em G_P é definido como:

$$\rho_{G'_P}^{G_P}(v_i, v_j) = \frac{\min(\pi_{G'_P}(v_i, v_j))}{\min(\pi_{G_P}(v_i, v_j))}. \quad (4.8)$$

Define-se o EFS de G'_P com respeito a G_P como:

$$\rho_{G'_P}^{G_P} = \max_{v_i, v_j \in V} \rho_{G'_P}^{G_P}(v_i, v_j). \quad (4.9)$$

Ou seja, o maior ESF que existe entre quaisquer dois vértices $v_i, v_j \in V$ de G'_P em relação a G_P . A Figura 4.2 ilustra esse conceito.

Definição 10 (*CE-t-S - Cooperative Energy t-Spanner*): Seja $G'_P = (V, E'_P)$ um subgrafo de $G_P = (V, E_P)$, $E'_P \subseteq E_P$, G_P e G'_P são grafos conexos. G'_P é uma CE-t-S em relação a G_P , se seu ESF não é maior que uma constante t , ou seja:

$$\rho_{G'_P}^{G_P} \leq t. \quad (4.10)$$

Definição 11 (*COE-t-S - Cooperative Oriented Energy t-Spanner*): A COE-t-S é um caso específico da CE-t-S. Seja $G'_P = (V, E'_P)$ um subgrafo de $G_P = (V, E_P)$, $E'_P \subseteq E_P$, G_P e G'_P não necessariamente são conexos (ao contrário do que ocorre na CE-t-S). Considere um nó fixo $v_o \in V$ e uma constante t . Dessa forma, o grafo G'_P é uma COE-t-S com relação a G_P se:

$$\max_{v_i \in V} \mu(v_i, v_o) \leq t, \quad (4.11)$$

onde:

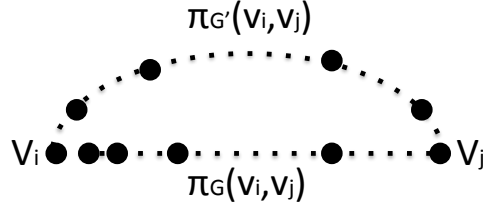


Figura 4.2: ESF - *Energy Stretch Factor* de um par de nós $v_i, v_j \in V$ em $G'_P = (V, E'_P)$ com respeito a $G_P = (V, E_P)$, $E'_P \subseteq E_P$, G'_P e G_P conexos. $\rho_{G'_P}^{G'_P}(v_i, v_j) = \frac{\min(\pi_{G'_P}(v_i, v_j))}{\min(\pi_{G_P}(v_i, v_j))}$ representa a diferença entre o custo do caminho mais eficiente em energia entre os nós v_i e v_j no grafo G'_P e o caminho mais eficiente entre esses mesmos nós em G_P .

$$\mu(v_i, v_o) = \begin{cases} \max_{v_i \in V} \rho_{G'_P}^{G'_P}(v_i, v_o), & \text{se existe caminho de } v_i \text{ até } v_o \text{ em } G_P; \\ 0, & \text{caso contrário.} \end{cases}$$

4.2.3 Exemplificação do Modelo

Para exemplificar o modelo definido anteriormente, considere o grafo representado pela Figura 4.1a. Este é um grafo direto $\overline{G_P} = (V, \overline{E_P})$, onde $V = \{v_1, v_2, v_3, v_4\}$ e $E_P = \{\overline{v_1v_2}, \overline{v_1v_3}, \overline{v_2v_1}, \overline{v_3v_1}\}$. Considere, por simplicidade, que $\alpha = 1$, $\tau = 1$ e que $R_{max} = R$. Observe que $\alpha = 1$ é irrealista e será utilizado apenas como exemplo. Dessa forma, podemos calcular a potência máxima de transmissão de cada nó $v_i \in V$ usando como base a Equação (4.1):

$$\begin{aligned} P_{MAX}(d_{MAX})^{-\alpha} &= \tau, \\ P_{MAX}(R)^{-1} &= 1, \\ P_{MAX} &= R. \end{aligned} \tag{4.12}$$

Observe que cada nó $v_i \in V$ possui uma potência de transmissão máxima de R com as configurações fornecidas. Analisando G_P , note que os nós v_2 e v_3 são nós vizinhos de v_1 , ou seja, $N(v_1) = \{v_2, v_3\}$, dessa maneira, o nó v_1 poderia selecionar qualquer subconjunto não vazio de $N(v_1)$ como seus nós ajudantes para criar uma aresta CC entre v_1 e v_4 , ou seja, $H_{1,4} \subseteq N_{v_1}$, segundo a Definição 2. O novo grafo com a aresta CC que sai de v_1 e vai até v_4 é ilustrado pela Figura 4.1b.

Considere que $d_{1,2} = R$, $d_{1,3} = \frac{R}{2}$, $d_{1,4} = 2R$, $d_{2,4} = 1,73R$ e $d_{3,4} = 1,80R$. Nas subseções que seguem, utiliza-se as definições feitas anteriormente para calcular o peso da aresta CC entre v_1 e v_4 utilizando diversos subconjuntos não-vazios de $N(v_i)$ como nós ajudantes e, por último, utiliza-se o exemplo dessa seção para introduzir o problema da seleção de nós ajudantes em CC.

Exemplo: Criação da aresta $\widetilde{v_1v_4}$ com $H_{1,4} = \{v_2, v_3\}$

Utilizando a Equação (4.2), podemos calcular a potência que o nó v_1 precisa para alcançar o nó v_4 utilizando os nós v_2 e v_3 como nós ajudantes:

$$\begin{aligned} \sum_{v_k \in v_1 \cup \{v_2, v_3\}} P_k(d_{k,4})^{-1} &\geq 1, \\ P_1(d_{1,4})^{-1} + P_2(d_{2,4})^{-1} + P_3(d_{3,4})^{-1} &\geq 1, \\ P_1 &\geq \frac{1}{\frac{1}{2R} + \frac{1}{1,73R} + \frac{1}{1,80R}}, \\ P_1 &\geq 0,612R. \end{aligned} \tag{4.13}$$

Da Equação (4.13), observa-se que, na aresta CC, todos os nós (v_1 , v_2 e v_3) operam com a mesma potência, isto é, $P_1 = P_2 = P_3$. Além disso, a potência mínima necessária para que v_1 alcance v_4 com comunicação cooperativa é igual a $P_1 = 0,612R$. Essa potência está abaixo da potência máxima de P_1 que é $P_{MAX} = R$, como visto anteriormente. Para calcular o peso da aresta CC criada ($\widetilde{v_1v_4}$) utiliza-se a Equação (4.4):

$$\begin{aligned} w(\widetilde{v_1v_4}) &= \frac{1}{R^{-1}} + \frac{(|H_{1,4}| + 1)}{\sum_{v_k \in v_1 \cup \{v_2, v_3\}} (d_{k,4})^{-1}}, \\ w(\widetilde{v_1v_4}) &= R + (|H_{1,4}| + 1) * 0,612R, \quad |H_{1,4}| = 2 \\ w(\widetilde{v_1v_4}) &= 2,836R. \end{aligned} \tag{4.14}$$

Da Equação (4.14), temos que o peso da aresta $\widetilde{v_1v_4}$ utilizando os nós v_2 e v_3 como ajudantes é igual a $w(\widetilde{v_1v_4}) = 2,836R$.

Exemplo: Criação da aresta $\widetilde{v_1v_4}$ com $H_{1,4} = \{v_2\}$ e com $H_{1,4} = \{v_3\}$

De maneira análoga ao exemplo anterior, utiliza-se a Equação (4.2) para calcular a potência que o nó v_1 precisa para alcançar o nó v_4 utilizando apenas o nó v_2 como ajudante:

$$\begin{aligned} \sum_{v_k \in v_1 \cup \{v_2\}} P_k(d_{k,4})^{-1} &\geq 1, \\ P_1(d_{1,4})^{-1} + P_2(d_{2,4})^{-1} &\geq 1, \\ P_1 &\geq \frac{1}{\frac{1}{2R} + \frac{1}{1,73R}}, \\ P_1 &\geq 0,928R \end{aligned} \tag{4.15}$$

Da Equação (4.15), temos que a potência mínima necessária para v_1 alcançar v_4 é igual a $P_1 = 0,928R$, o que está abaixo da potência máxima ($P_{MAX} = R$). O peso da aresta CC criada ($\widetilde{v_1v_4}$) pode ser calculada utilizando a Equação (4.4):

$$\begin{aligned}
w(\widetilde{v_1v_4}) &= \frac{1}{R^{-1}} + \frac{(|H_{1,4}| + 1)}{\sum_{v_k \in v_1 \cup \{v_2\}} (d_{k,4})^{-1}}, & (4.16) \\
w(\widetilde{v_1v_4}) &= R + (|H_{1,4}| + 1) * 0,928R, & |H_{1,4}| = 1 \\
w(\widetilde{v_1v_4}) &= 2,855R
\end{aligned}$$

Da Equação (4.16), temos que o peso da aresta $\widetilde{v_1v_4}$ utilizando somente o nó v_2 como ajudante é igual a $w(\widetilde{v_1v_4}) = 2,855R$. Seguindo o mesmo raciocínio, temos que a potência necessária no nó v_1 para alcançar v_4 utilizando apenas o nó v_3 como ajudante está representado pela Equação (4.17) e o peso da aresta $\widetilde{v_1v_4}$ é representado pela Equação (4.18).

$$P_1 \geq 0,947R \quad (4.17)$$

$$w(\widetilde{v_1v_4}) = 2,395R \quad (4.18)$$

A seguir, utiliza-se o exemplo dessa seção para introduzir o problema da seleção de nós ajudantes em CC.

Problema da seleção dos nós ajudantes em CC

O exemplo dessa seção mostra que é possível estender o raio de transmissão do nó v_1 utilizando o modelo de comunicação cooperativa. Isso proporcionou um aumento de conectividade, uma vez que o nó v_4 , que está a $2R$ de distância de v_1 , passou a poder ser acessado por v_1 , sendo que o limite de v_1 originalmente é igual a R . Note que a aresta CC criada entre v_1 e v_4 é direcional, ou seja, v_1 consegue alcançar v_4 transmitindo o sinal em parceria com seus ajudantes, mas v_4 continua sem poder acessar qualquer nó.

A Tabela 4.1 sumariza todos os cálculos feitos para esse exemplo. Quando tenta-se criar a aresta $\widetilde{v_1v_4}$ utilizando os nós v_2 e v_3 como nós ajudantes, a potência que v_1 precisa para transmitir o dado cooperativamente para v_4 é igual a $P_1 = 0,612R$ (Equação 4.13), enquanto o peso da aresta CC é igual a $w(\widetilde{v_1v_4}) = 2,836R$ (Equação 4.14). Analogamente, a potência de v_1 para criar a aresta CC até v_4 utilizando apenas v_2 como ajudante é igual a $0,928R$ (Equação 4.15) com o peso da aresta igual a $2,855R$ (Equação 4.16). Utilizando v_3 como ajudante os valores são iguais a $0,947R$ (Equação 4.17) e $2,395R$ (Equação 4.18).

Analisando os valores, observa-se que, quanto mais nós ajudantes, menor a potência mínima que v_1 e seus ajudantes precisam para cooperativamente transmitirem para v_4 . A maior diferença aparece quando analisa-se o peso da aresta CC. O peso da aresta $\widetilde{v_1v_4}$ é menor quando utiliza-se apenas o nó v_3 como ajudante e, mesmo utilizando apenas o nó v_2 , esse peso é bastante próximo do que utilizando ambos v_2 e v_3 como ajudantes. Essa diferença está fortemente relacionada com o modelo de comunicação cooperativa, onde, como pode ser observado na Equação (4.4), o peso de uma aresta CC é a soma de dois valores: o custo que o nó fonte precisa para enviar seu dado para seus nós ajudantes, em um primeiro momento; e o custo para os nós cooperativamente transmitirem o dado, em um segundo momento. Como, no exemplo, o nó v_2 está mais distante de v_1 do que o nó v_3 , respectivamente distâncias de R e $\frac{R}{2}$, a potência que o nó v_1 precisa para enviar seus dados para o nó v_2 , em um primeiro momento, acaba sendo custoso, não compensando utilizar

Tabela 4.1: Condições necessárias e análises que precisam ser feitas para se criar uma aresta CC $\widetilde{v_1v_4}$ para o grafo exemplo da Figura 4.1a

$H_{1,4}$	P_1	$w(\widetilde{v_1v_4})$
$\{v_2, v_3\}$	0,612R (4.13)	2,836 (4.14)
$\{v_2\}$	0,928R (4.15)	2,855R (4.16)
$\{v_3\}$	0,947R (4.17)	2,395R (4.18)

esse nó como ajudante. O problema da escolha eficiente de nós ajudantes em CC tem se mostrado uma tarefa desafiadora, tendo sido abordado por diversos autores [44, 45, 46, 47].

4.2.4 Formulação do Problema

Considere uma rede de sensores *ad hoc*, onde os sensores estão espalhados em uma área fixa e existe um único nó sorvedouro na borda da rede. Um nó sorvedouro é um nó responsável por coletar e solicitar informações dos sensores. Esses sensores são fixos, ou seja, não admitem mobilidade e a topologia de rede criada a partir desses sensores não necessariamente é conexa. A proposta desse capítulo consiste em utilizar comunicação cooperativa na topologia formada a partir dos nós dessa rede para aumentar a conectividade dos nós com o sorvedouro, de forma que a energia de acesso esteja dentro de limites bem definidos, conforme descrito pela Definição 11.

Utilizando a notação definida, dado uma topologia de rede representada pelo grafo $\overline{G_P} = (V, \overline{E_P})$, sendo $v_o \in V$ o nó sorvedouro, propor uma técnica que utilize CC para criar um grafo G'_P de tal forma que G'_P seja uma COE-t-S em relação a G_P e ao nó v_o , seguindo a Definição 11. Na seção que segue, descreve-se uma heurística gulosa para a escolha de nós ajudantes e propõe-se uma técnica para resolver o problema descrito anteriormente.

4.3 Descrição da Proposta

Nesta seção, descreve-se o *CoopSink* (abreviação para *Cooperative Sink*), uma técnica de controle de topologia centralizada para redes de sensores *ad hoc* cooperativas que busca aumentar a conectividade dos nós de uma rede em relação a um sorvedouro e diminuir o consumo de energia das rotas que levam até o sorvedouro. Em um primeiro momento, descreve-se um algoritmo guloso da literatura que pode ser utilizado para seleção de nós ajudantes na criação de arestas cooperativas e, em seguida, a técnica *CoopSink* é descrita. As formulações e algoritmos descritos nessa seção seguem as definições feitas na Seção 4.2.

4.3.1 Algoritmo: *Greedy Helper Set Selection*

O Algoritmo 4.1 descreve uma heurística gulosa, proposta por [39], e com algumas otimizações no código para diminuir o tempo de execução. Essa heurística é utilizada para selecionar os nós ajudantes mais eficiente em energia em uma comunicação cooperativa e serve para grafos tanto direcionais quanto bi-direcionais. O algoritmo recebe

Algoritmo 4.1 : *GreedyHelperSetSelection*($v_i, N(v_i), v_j$)

Entrada: $v_i, N(v_i), v_j$

Saída: $H_{i,j}$;

- 1: # (Passo 1)
- 2: conjunto: $A \leftarrow N(v_i)$;
- 3: vetor: $B \leftarrow []$, $C \leftarrow []$;
- 4: **Enquanto** ($v_k \leftarrow head(A) \neq \emptyset$) **faça**
- 5: $b_k \leftarrow \frac{\tau}{(d_{i,j})^{-\alpha}} - \frac{\tau}{\sum_{v_l \in \{v_i, v_k\}} (d_{l,j})^{-\alpha}}$;
- 6: $c_k = \frac{\tau}{(d_{i,k})^{-\alpha}}$;
- 7: $B \leftarrow [B, \frac{b_k}{c_k}]$;
- 8: $remove(A, v_k)$;
- 9: **Fim Enquanto**
- 10: $B \leftarrow ordenarOrdemDecrescente(B)$;
- 11: $C \leftarrow indexarTermos(B)$
- 12: # (Passo 2)
- 13: $k \leftarrow 0$, $\Omega \leftarrow v_i$;
- 14: **Enquanto** ($\sum_{v_k \in \Omega} P_{MAX}(d_{k,j})^{-\alpha} < \tau$) **faça**
- 15: $k \leftarrow k + 1$;
- 16: **Se** $k > |C|$ **então**
- 17: retorne erro;
- 18: **Fim Se**
- 19: $H_{i,j} \leftarrow H_{i,j} \cup A[k]$;
- 20: $\Omega \leftarrow \Omega \cup H_{i,j}$
- 21: **Fim Enquanto**
- 22: # (Passo 3)
- 23: **Enquanto** ($k \leq |N(v_i)|$) **faça**
- 24: **Se** ($k = |N(v_i)|$) **ou**
 ($PesoCC(\widetilde{v_i v_j}, \Omega) < PesoCC(\widetilde{v_i v_j}, \Omega \cup C[k + 1])$) **então**
- 25: retorne $H_{i,j}$;
- 26: **else**
- 27: $k \leftarrow k + 1$;
- 28: $H_{i,j} \leftarrow H_{i,j} \cup C[k]$;
- 29: $\Omega \leftarrow \Omega \cup H_{i,j}$;
- 30: **Fim Se**
- 31: **Fim Enquanto**

como entrada: o nó transmissor v_i ; o conjunto $N(v_i)$ de nós vizinhos de v_i ; e nó destino v_j . Retorna um conjunto de nós ajudantes $H_{i,j}$ de tal forma que o peso da aresta CC $\widetilde{v_i v_j}$ seja reduzido (embora não garanta soluções ótimas). Considere o protótipo de função $GreedyHelperSetSelection(v_i, N(v_i), v_j)$ para se referir ao Algoritmo 4.1. A seguir, descreve-se, de maneira geral, os passos do algoritmo, a função de ganho utilizada e, por último, um exemplo é apresentado.

Descrição geral do algoritmo

O Algoritmo 4.1 possui 3 passos principais:

- **Passo 1 (linhas 1-11):** Esse passo consiste em ordenar os nós vizinhos de v_i ($N(v_i)$) de acordo com uma heurística de ganho. As seguintes funções são utilizadas nesse passo: a função $head(X)$ retorna o primeiro elemento de um vetor ou conjunto X ; a função $remove(X, Y)$ retira o elemento Y do conjunto X ; a função $ordenarOrdemDecrescente(X)$ recebe um vetor X e retorna o mesmo vetor ordenado em ordem decrescente; por fim, a função $indexarTermos(X)$, onde $X = [\frac{b_{k_1}}{c_{k_1}}, \frac{b_{k_2}}{c_{k_2}}, \dots, \frac{b_{k_{|N(v_i)|}}}{c_{k_{|N(v_i)|}}]$, retorna um vetor $Y = [v_{k_1}, v_{k_2}, \dots, v_{k_{|N(v_i)|}}]$. A função de ganho será melhor detalhada adiante;
- **Passo 2 (linhas 12-21):** Esse passo consiste em adicionar elementos do vetor C ao conjunto de nós ajudantes $H_{i,j}$ até que v_i e seus vizinhos possuam a potência mínima necessária para criar a aresta $\widetilde{v_i v_j}$, de acordo com a Equação 4.2. Se, após adicionar todos os elementos de C , a potência somada desses nós ainda não seja suficiente para criar a aresta $\widetilde{v_i v_j}$, uma mensagem de erro é retornada (linhas 15-17);
- **Passo 3 (linhas 22-31):** Esse passo consiste em adicionar o máximo de nós ajudantes ao conjunto $H_{i,j}$ de modo que a inclusão de um novo nó não aumente o peso da aresta CC. Caso a inclusão dessa aresta aumente o peso da aresta CC ou todos os elementos em C já tiverem sido inseridos em $H_{i,j}$, a função retorna o conjunto $H_{i,j}$ (linhas 23-24). A função $PesoCC(v_i v_j, \Omega)$, onde $v_i v_j \in E_t$ e $\Omega \subseteq V$, retorna o peso da aresta CC $v_i v_j$ utilizando os nós presentes no conjunto Ω como ajudantes.

Função de ganho

A função de ganho, presente no Algoritmo 4.1 (Passo 1), é utilizada para ordenar os nós de $N(v_i)$ que trarão mais ganho em termos de custos para criar a aresta CC $\widetilde{v_i v_j}$. Considere, para cada nó $v_k \in N(v_i)$, os seguintes valores:

- $b_k \leftarrow \frac{\tau}{(d_{i,j})^{-\alpha}} - \frac{\tau}{\sum_{v_l \in \{v_i, v_k\}} (d_{l,j})^{-\alpha}}$: quantidade de potência que o nó v_i pode economizar se adicionar o nó v_k como ajudante. Observe que o primeiro elemento da subtração é o custo para v_i se comunicar com v_j diretamente e o segundo elemento é a potência para v_i se comunicar com v_j cooperativamente utilizando v_k como ajudante;
- $c_k = \frac{\tau}{(d_{i,k})^{-\alpha}}$: Custo para o nó v_i se comunicar com o nó ajudante v_k diretamente.

Como o objetivo é diminuir o peso da aresta CC é claro que os valores de b_k , que simbolizam o ganho, devem ser maximizados enquanto os valores de c_k , que representam

o custo de se comunicar com o nó ajudante, devem ser minimizados. Dessa maneira, a função de ganho considera a razão $\frac{b_k}{c_k}$ como métrica que indica qual nó trará mais ganho, ou seja, se $\frac{b_k}{c_k}$ é máximo, então a inclusão do nó v_k trará mais ganho que os outros.

Exemplo de funcionamento

Considere o exemplo apresentado na Seção 4.2.3. Nesse exemplo, temos um nó v_1 querendo criar uma aresta CC para v_4 , sendo que $N(v_1) = \{v_2, v_3\}$. Após cálculos exaustivos, chegou-se à conclusão de que apenas o nó v_3 deveria ser utilizado como ajudante para criar uma aresta CC mais eficiente, ou seja, $H_{1,4} = \{v_3\}$. O Algoritmo 4.1 pode ser utilizado para chegar à essa conclusão de maneira menos custosa, principalmente quando existe um número grande de nós vizinhos.

Nesse exemplo, a função *GreedyHelperSetSelection()* seria chamada com os parâmetros : $(v_1, \{v_2, v_3\}, v_4)$. Observando o Passo 1, o vetor B seria criado (linhas 4-9) com os seguintes valores:

$$B = \left[\frac{b_2}{c_2}, \frac{b_3}{c_3} \right] = \left[\frac{1,072R}{R}, \frac{1,053R}{0,5R} \right] = [1,072, 2,106]. \quad (4.19)$$

Esse cálculo já mostra que o ganho em adicionar o nó v_3 é maior do que adicionar o nó v_2 , ganho de 2,106 contra 1,072. O vetor C é criado de maneira que os nós com maior ganho aparecem na frente (linhas 10-11):

$$C = [v_3, v_2]. \quad (4.20)$$

No Passo 2, verifica-se qual é o conjunto mínimo de nós de tal forma que que aresta CC possa ser criada de v_1 para v_4 . Primeiramente, faz-se $\Omega = \{v_1\}$ (linha 13) e verifica-se que apenas o nó v_1 não possui a potência mínima necessária para acessar o nó v_4 (linha 14):

$$\begin{aligned} P_{MAX}(d_{1,4})^{-\alpha} &\leq \tau & (4.21) \\ R * (2R)^{-1} &= \frac{1}{2} \leq 1 & \text{(verdade),} \end{aligned}$$

Ω é então incrementado (linhas 19-20) com o primeiro elemento do vetor C , no caso, v_3 . Assim:

$$\begin{aligned} P_{MAX}(d_{1,4})^{-\alpha} + P_{MAX}(d_{3,4})^{-\alpha} &\leq \tau & (4.22) \\ R * (2R)^{-1} + R * (1,80R)^{-1} &= 1.056 < 1 & \text{(falso).} \end{aligned}$$

Como a condição de teste é falsa, o laço é encerrado. Nesse momento, temos que $\Omega = \{v_1, v_3\}$ e que $H_{i,j} = \{v_3\}$. No Passo 3, verifica-se se o peso da aresta CC $\widetilde{v_1 v_4}$ utilizando $\Omega = \{v_1, v_3\}$ como ajudante é menor do que o peso da aresta CC $\widetilde{v_1 v_4}$ utilizando $\Omega \cup v_2 = \{v_1, v_2, v_3\}$ como ajudantes (linha 24), finalizando o algoritmo e retornando $H_{1,4} = \{v_3\}$ (linha 25). Observe que $H_{1,4} = \{v_3\}$ é realmente a melhor configuração de nós ajudantes para esse exemplo, assim como foi verificado na Seção 4.2.3. Isso mostra

Algoritmo 4.2 : *CoopSinkStep1*(V, P_{MAX})

Entrada: V e P_{MAX} ;

Saída: $\overline{G_P}$;

- 1: $\overline{G_P} = (V, \emptyset)$;
 - 2: **Para** $v_i, v_j \in V$ **faça**
 - 3: **Se** $P_{MAX}(d_{ij})^{-\alpha} \geq \tau$ **então**
 - 4: Adicione a aresta $\overline{v_i v_j}$ a $\overline{G_P}$;
 - 5: **Fim Se**
 - 6: **Fim Para**
-

que o Algoritmo 4.1 é capaz de obter soluções eficientes para o problema de escolha de nós ajudantes em CC.

4.3.2 *CoopSink*: Descrição da Técnica

Esta seção descreve os passos do *CoopSink*, a proposta desse trabalho para os problemas descritos pela Seção 4.2.4. Assume-se uma unidade central de computação nessa seção. O *CoopSink* consiste na execução em sequência de 4 passos em sequência:

- **Passo 1:** Criar um grafo de topologia onde cada nó estabelece o máximo de arestas que sua potência máxima de transmissão permite;
- **Passo 2:** Utilizar a tecnologia de CC para criar o máximo de arestas CC possíveis entre os nós da rede;
- **Passo 3** Partindo o grafo de topologia criado no passo anterior, fazer controle de topologia para cortar arestas de maneira que rotas eficientes em relação ao nó sorvedouro sejam mantidas;
- **Passo 4:** Ajustar a potência dos nós para que o consumo de energia seja mínimo.

As subseções que seguem descrevem e exemplificam melhor os 4 passos da técnica.

Passo 1: Construção do grafo $\overline{G_P}$

O Algoritmo 4.2 descreve o primeiro passo da técnica *CoopSink*. Esse passo consiste em criar todas as arestas diretas possíveis em um grafo de topologia, desde que cada nó opere utilizando sua potência máxima P_{MAX} . Como entrada, temos: o conjunto de nós $V = \{v_1, v_2, \dots, v_n\}$ e suas localizações no mapa; e a potência máxima de transmissão P_{MAX} . Como saída, temos o grafo direto $\overline{G_P}$. Considere o protótipo de função *CoopSinkStep1*(V, P_{MAX}) para se referir ao Algoritmo 4.2.

As Figuras 4.3a e 4.3b exemplificam o que acontece nesse passo. Na Figura 4.3a temos uma área de 500x500m com $n = 50$ nós e $\overline{E_P} = \emptyset$. A Figura 4.3b ilustra o grafo resultante quando os nós criam aresta de acordo com sua potência máxima de transmissão, no exemplo, $P_{MAX} = 4900$ e $R_{MAX} = 70$. Note que o grafo resultante não necessariamente é conexo.

Algoritmo 4.3 : *CoopSinkStep2*($\overline{G_P}$)

Entrada: $\overline{G_P}$ **Saída:** G_P ;

- 1: $\widetilde{G_P} = (V, \emptyset)$
 - 2: **Para** $v_i, v_j \in V$ **faça**
 - 3: $H_{ij} \leftarrow GreedyHelperSetSelection(v_i, N(v_i), v_j)$
 - 4: **Se** $\frac{\tau}{\sum_{v_k \in v_i \cup H_{i,j}} (d_{k,j})^{-\alpha}} \leq P_{MAX}$ **então**
 - 5: Adicione a aresta $\widetilde{v_i v_j}$ a $\widetilde{G_P}$;
 - 6: **Se** $\overline{v_i v_j} \in \overline{G_P}$ e $w(\overline{v_i v_j}) > w(\widetilde{v_i v_j})$ **então**
 - 7: Retire a aresta $\overline{v_i v_j}$ de $\overline{G_P}$;
 - 8: **Fim Se**
 - 9: **Fim Para**
 - 10: **Fim Para**
 - 11: $G_P = \overline{G_P} + \widetilde{G_P}$
-

Passo 2: Construção dos grafos $\widetilde{G_P}$ e G_P

O Algoritmo 4.3 descreve o segundo passo da técnica *CoopSink*. Esse passo consiste em criar todas as arestas CC possíveis tendo como entrada o grafo $\overline{G_P} = (V, \overline{E_P})$ do passo anterior. Para isso, utiliza-se a função *GreedyHelperSetSelection*($v_i, N(v_i), v_j$), definida anteriormente, para a escolha dos nós ajudantes de uma aresta CC. Como entrada, temos o grafo direto $\overline{G_P}$ e, como saída, temos o grafo G_P . Considere o protótipo de função *CoopSinkStep2*($\overline{G_P}$) para se referir ao Algoritmo 4.3.

Como exemplo, considere os grafos representados pelas Figuras 4.3b e 4.3c. O grafo da Figura 4.3c é o grafo resultante do Algoritmo 4.3 quando recebe como entrada o grafo da Figura 4.3b. Nesse grafo, as arestas $\overline{v_i v_j} \in E_P$ estão em “azul”, as arestas $\widetilde{v_i v_j} \in E_P$ estão em “vermelho”. Note que, nesse exemplo, uma grande quantidade de arestas CC foram criadas, no entanto, essas arestas são direcionais, ou seja, a existência de $\widetilde{v_i v_j} \in E_P$ não implica que $\widetilde{v_j v_i} \in E_P$.

Passo 3: Cortando arestas de G_P e criando uma COE-t-S

O Algoritmo 4.4 descreve o terceiro passo da técnica *CoopSink*. Esse passo consiste em criar um grafo G'_P que é uma COE-t-S em relação ao grafo G_P e a um nó $v_o \in V$. Recebe como entrada: um grafo G_P ; uma constante t ; e um nó $v_o \in V$. Retorna um grafo G'_P que é uma COE-t-S em relação ao grafo G_P e ao nó $v_o \in V$. As seguintes funções são usadas nesse passo: a função *arestaMaiorPeso*(X) recebe um conjunto de arestas X e retorna a aresta de maior peso, ou seja, a aresta $v_i v_j$ cujo valor de $w(v_i v_j)$ seja maior; a função *remove*(X, Y) recebe um conjunto X e remove de X o elemento Y . Considere o protótipo de função *CoopSinkStep3*(G_P, t, v_o) para se referir ao Algoritmo 4.4.

Como exemplo, considere os grafos representados pelas Figuras 4.3c e 4.3d. Nesse exemplo, o grafo representado pela Figuras 4.3c é a entrada do Algoritmo 4.4, junto com $t = 1,4$, que é um parâmetro utilizado na literatura, e v_o na posição (250,250) – pintado de “verde” – do eixo cartesiano, sendo o grafo representado pela Figura 4.3d a saída do algoritmo. Note que várias arestas foram retiradas do grafo e que uma estrutura de

Algoritmo 4.4 : *CoopSinkStep3*(G_P, t, v_o)

Entrada: $G_P = (V, E_P)$;

Saída: $G'_P = (V, E'_P)$;

```
1: # Ordena todas as arestas pelo peso
2:  $A \leftarrow E_P$ ;
3:  $B \leftarrow []$ ;
4: Para  $A \neq \emptyset$  faça
5:    $v_i v_j \leftarrow \text{arestaMaiorPeso}(A)$ ;
6:    $B \leftarrow [B, v_i v_j]$ ;
7:    $\text{remove}(A, v_i v_j)$ ;
8: Fim Para
9: # Remove as arestas e cria uma COE-t-S
10:  $G'_P \leftarrow G_P$ ;
11:  $k \leftarrow 1$ ;
12: Para  $v_i v_j \leftarrow B[k]$  e  $k \leq |B|$  faça
13:    $G'_P \leftarrow G'_P - v_i v_j$ ;
14:   Se  $\max_{v_i \in V} \mu(v_i, v_o) \leq t$  então
15:     # A aresta  $v_i v_j$  não deve ser deletada;
16:      $G'_P \leftarrow G'_P + v_i v_j$ ;
17:     Se  $v_i v_j = \widetilde{v_i v_j}$  então
18:       # Note que, se  $v_i v_j$  for uma aresta CC, os nós ajudantes de  $v_i$  para criar essa
19:       # aresta devem ser protegidos contra deleção;
20:       Retire as arestas entre  $v_i$  e seus vizinhos em  $H_{i,j}$  de  $B$ ;
21:     Fim Se
22:   Fim Para
23:  $k \leftarrow k + 1$ ;
24: Fim Para
```

árvore foi criada em direção ao nó v_o que, nesse exemplo, fica na posição (250, 250) do eixo cartesiano.

Passo 4: Ajuste da potência P_i dos nós

O Algoritmo 4.5 descreve o último passo da técnica *CoopSink*. Esse passo consiste em permitir que os nós ajustem suas potências de transmissão de maneira a COE-t-S criada seja mantida. Recebe como entrada G'_P , que é uma COE-t-S em relação a G_P e não possui retorno. Considere o protótipo de função *CoopSinkStep4*(G'_P) para se referir ao Algoritmo 4.5. Na próxima seção, descreve-se os resultados da comparação do *CoopSink* com outras técnicas da literatura.

4.4 Resultados de Simulação

Na seção anterior, foi proposta uma técnica de controle de topologia em redes *ad hoc* cooperativas, na qual garante um aumento de conectividade e mantém a eficiência das rotas para o sorvedouro. Nesta seção, diversas simulações são realizadas para compa-

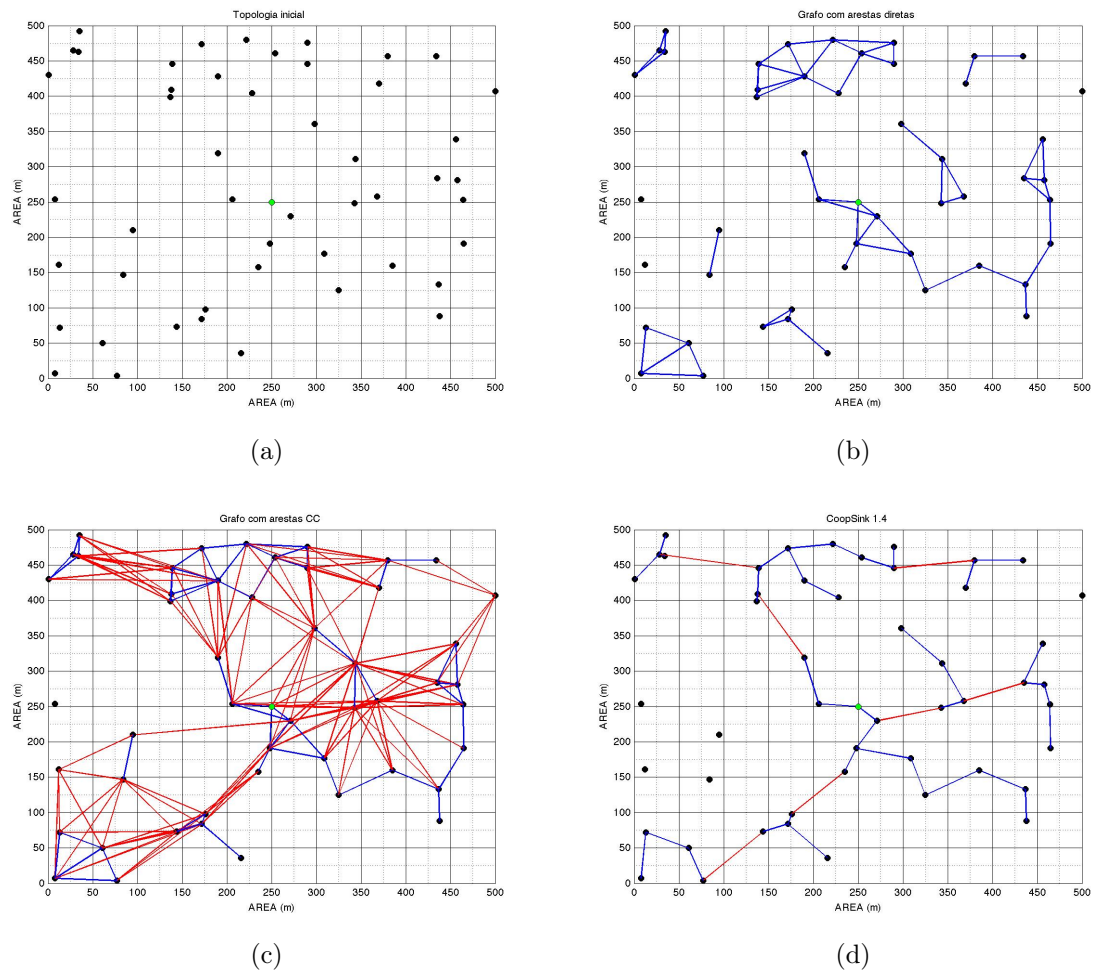


Figura 4.3: (a) Exemplo de topologia de rede, onde $|V| = 50$ e $|E_P| = 0$. (b) Grafo formado a partir de (a) quando os nós operam em sua potência máxima P_{MAX} . (c) Grafo criado a partir de (b) adicionando todas as arestas CC possíveis. (d) COE-t-S criada a partir de (c) inserindo as arestas CC e com $t = 1, 4$.

Algoritmo 4.5 : *CoopSinkStep4*(G'_P)

Entrada: $G'_P = (V, E'_P)$

- 1: **Para** $v_i \in V$ **faça**
 - 2: $a \leftarrow \max_{\overline{v_i v_j} \in E'_P} w_d(H_{i,j});$
 - 3: $b \leftarrow \max_{\overline{v_i v_j} \in E'_P} w_{CC}(H_{i,j});$
 - 4: $P_i = \max\{a, b\}$
 - 5: **Fim Para**
-

rar o desempenho do *CoopSink* com outras da literatura. As técnicas escolhidas para comparação são:

- *MST - Minimum Spanning Tree* [82]: Parte do grafo direto $\overline{G_P}$ (saída do passo 1 do *CoopSink*) e aplica o algoritmo *MST-Kruskal* para retirar as arestas menos eficientes;
- *CoopBridges* [39]: Parte do grafo direto $\overline{G_P}$ (saída do passo 1 do *CoopSink*) e divide todos os elementos conexos em clusters. Então cria o máximo de arestas CC não-direcionais possíveis entre os clusters, por fim, utiliza um algoritmo baseado na MST para cortar o máximo de arestas possíveis tanto intra como inter cluster.

Inicialmente, pretendia-se comparar o *CoopSink* também com o *Greedy-(Add/Del)-Link* [54], porém, essa técnica assume que a topologia gerada, após a inclusão das arestas CC, será necessariamente conexa, uma hipótese que não é assumida pelo *CoopSink* e pelo *CoopBridges*. Para avaliar o desempenho do *CoopSink* em relação às outras técnicas, utiliza-se as seguintes métricas:

- **M1 - Conectividade com o sorvedouro:** Consiste na porcentagem de nós que possuem uma rota direcional para o nó sorvedouro;
- **M2 - Potência média de transmissão:** Consiste na média das potências atribuídas a cada nó no grafo de topologia;
- **M3 - ESF para o sorvedouro:** Consiste em ter um grafo inicial $G_P = (V, E_P)$, criar um subgrafo $G'_P = (V, E'_P)$, $E'_P \subseteq E_P$, e verificar o valor de t quando considera-se que G'_P é uma COE-t-S em relação a G_P ;
- **M4 - Número de arestas CC:** Consiste em verificar quantas arestas CC foram criadas no grafo final de cada técnica.

A métrica M1 é justificada uma vez que um dos objetivos do *CoopSink* é aumentar a conectividade com o nó sorvedouro. A métrica M2 é utilizada uma vez que minimizar o consumo de energia entre os nós é um dos objetivos básicos de técnicas de controle de topologia. A métrica M3 é utilizada para verificar o quão eficiente são as rotas do *CoopSink* para o sorvedouro quando comparadas com outras técnicas. Por fim, a métrica M4 faz uma comparação entre o número de arestas CC resultantes na topologia final.

Para simulação, implementou-se os algoritmos descritos na Seção 4.3 no ambiente do Matlab [83]. A implementação do *CoopBridges* foi validada com os valores apresentados no artigo original, obtendo uma margem de erro de menos de 5%. Acredita-se que essa pequena variação se deve à maneira que as sementes randômicas de simulação são geradas. O ambiente de simulação é configurado de maneira similar às configurações feitas em [39, 54]:

- $n = 10, 20, \dots, 100$ nós são posicionados aleatoriamente em uma área de 500×500 m;
- O nó sorvedouro será sempre o nó v_1 e será posicionado no centro da área delimitada, ou seja, na posição $(250, 250)$ do eixo cartesiano;
- O *Path Loss Exponent* (PLE) é igual a 2;
- P_{MAX} é igual a 4900;
- O *Signal-to-Noise Ratio* (SNR) é igual a 1;
- Os valores para a constante t são variados entre 1, 1,4 e 1,8.

Variando os parâmetros, os resultados fixados nos gráficos consistem na média de 200 simulações para cada configuração de simulação. Essas 200 simulações são divididas em 5 grupos com 40 simulações cada e um intervalo de confiança de 95% é calculado observando a variação entre os valores médios de cada um dos 5 grupos de simulação. Os resultados apresentados dizem respeito a utilização das técnicas *CoopBridges*, *MST*, *CoopSink-1.0*, *CoopSink-1.4* e *CoopSink-1.8*, sendo as três últimas uma execução do *CoopSink* com, respectivamente, $t = 1,0$, $t = 1,4$ e $t = 1,8$ como parâmetros. As subseções que seguem expõem e analisam os resultados de simulação para cada uma das métricas propostas.

4.4.1 M1: Conectividade com o sorvedouro

A Figura 4.4 apresenta os resultados de simulação para a métrica M1. No eixo x , varia-se o número de nós na topologia de rede, enquanto, no eixo y , apresenta-se a porcentagem de nós na rede que possuem uma rota para o nó sorvedouro.

Observa-se que:

- O *CoopSink* apresentou conectividades com o sorvedouro iguais para cada uma das suas três variações: *CoopSink-1.0*, *CoopSink-1.4* e *CoopSink-1.8*. De fato, a conectividade resultante do *CoopSink* não depende do parâmetro t , apenas o custo das rotas para o sorvedouro são afetadas com a variação desse parâmetro;
- A conectividade do *CoopSink* com o sorvedouro superou o *CoopBridges* e o *MST* em todos os pontos observados e, a partir de 60 nós, essa conectividade foi de mais de 90%;
- No melhor caso, o *CoopSink* obteve uma conectividade com o sorvedouro 94,3% maior que o *MST* e 54,7% maior que o *CoopBridges*.

4.4.2 M2: Potência média de transmissão

A Figura 4.5 apresenta os resultados de simulação para a métrica M2. No eixo x , varia-se o número de nós na topologia de rede, enquanto, no eixo y , apresenta-se a potência média de transmissão dos nós na rede.

Observa-se que:

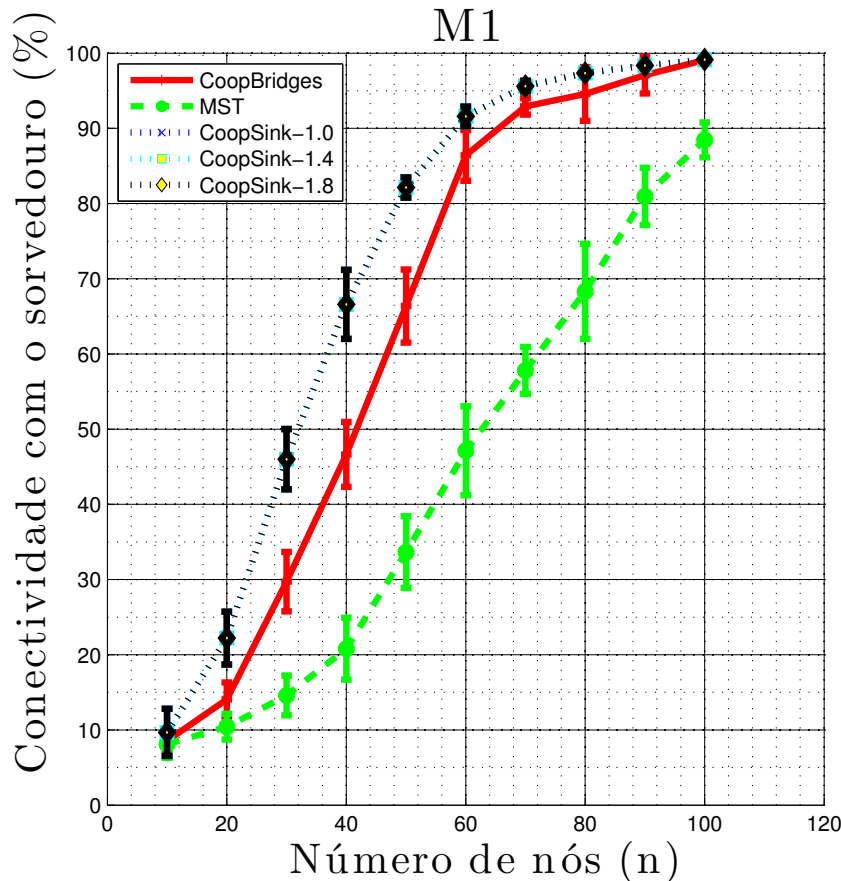


Figura 4.4: Resultados de simulação para a métrica M1.

- Como o *MST* não cria arestas CC, a potência média de transmissão utilizando essa técnica é, para a maioria dos pontos, a menor das técnicas observadas. Isso ocorre porque as arestas CC possuem a desvantagem de serem mais custosas energeticamente. No entanto, a partir de 90 nós, tanto o *CoopSink-1.4* quanto o *CoopSink-1.8* superam o *MST* no consumo de energia. Isso ocorre por causa do ajuste de potência que diminui o peso das arestas do grafo para favorecer apenas conexões direcionais com o nó sorvedouro;
- A potência média dos nós no *CoopSink* tende a ser maior quando a constante t é menor. Isso ocorre porque, quanto menor a constante t , menos arestas CC são retiradas de G_P no passo 3 do algoritmo e, quanto mais arestas CC, maior o consumo de energia;
- No caso do *CoopSink-1.4* e *CoopSink-1.8*, o consumo de energia foi maior que no *CoopBridges* nas simulações com até 60 nós. Com mais de 60 nós o consumo de energia do *CoopSink-1.4* e *CoopSink-1.8* foi até 28,7% menor que o *CoopBridges*.

4.4.3 M3: ESF para o sorvedouro

A Figura 4.6 apresenta os resultados de simulação para a métrica M3. No eixo x , varia-se o número de nós na topologia de rede, enquanto, no eixo y , apresenta-se o ESF

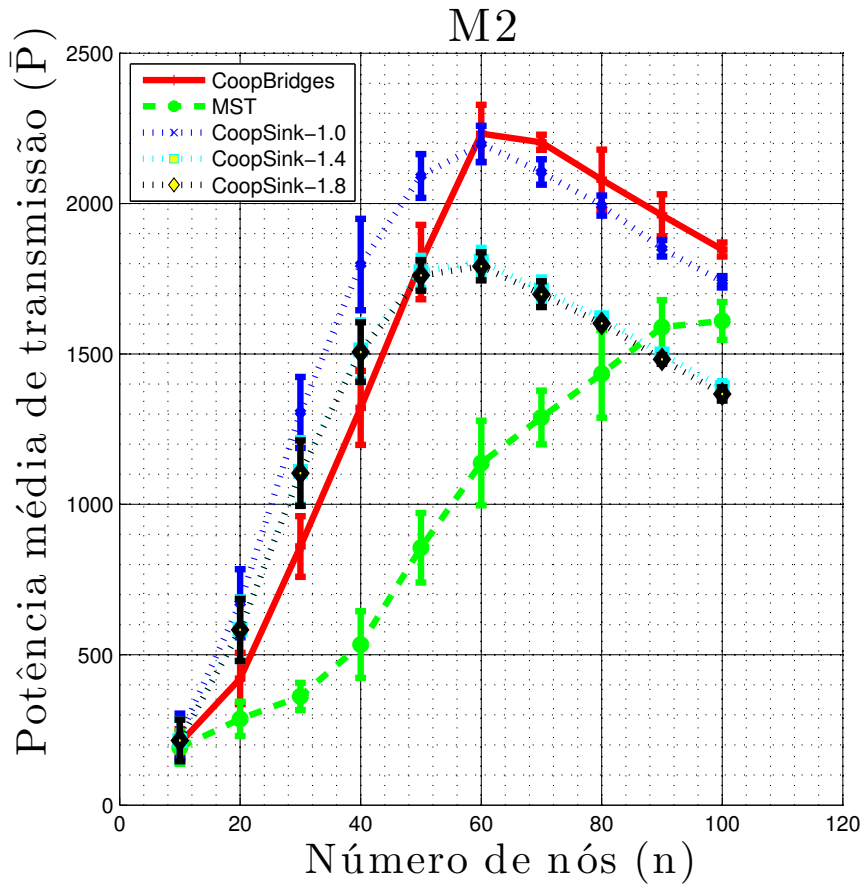


Figura 4.5: Resultados de simulação para a métrica M2.

em relação ao sorvedouro medido para cada uma das técnicas analisadas. Como o ESF é medido comparando dois grafos, mede-se o ESF em relação ao sorvedouro comparando os seguintes grafos para cada técnica:

- *CoopSink-(1.0/1.4/1.8)*: Grafo G'_P (obtido ao final do passo 3 do *CoopSink*) com o grafo G_P (obtido ao final do passo 2 do *CoopSink*);
- *CoopBridges*: Grafo obtido com a aplicação do *CoopBridges* com o grafo G_P (obtido ao final do passo 2 do *CoopSink*);
- *MST*: Grafo obtido após a aplicação do MST com o grafo $\overline{G_P}$ (obtido ao final do passo 1 do *CoopSink*).

Os resultados mostram que:

- As variações do *CoopSink* obedecem o ESF para o sorvedouro estabelecido durante a configuração, respectivamente, 1,0, 1,4 e 1,8. Isso significa que os valores observados para o ESF em relação ao sorvedouro devem ser menores que 1,0 (no caso do *CoopSink-1.0*), 1,4 (no caso do *CoopSink-1.4*) e 1,8 (no caso do *CoopSink-1.8*);
- O *CoopBridges* e o *MST* não possuem qualquer compromisso com o ESF em relação ao sorvedouro. Isso significa que algumas das rotas criadas podem ser bastante ineficientes energeticamente;

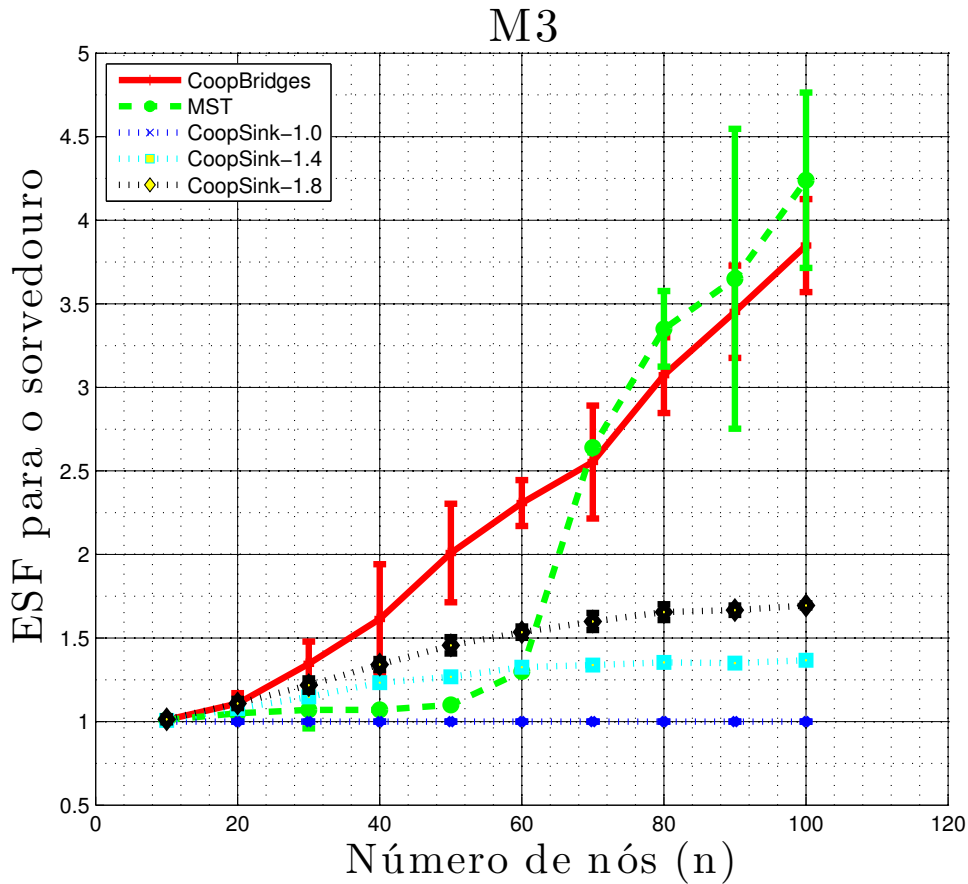


Figura 4.6: Resultados de simulação para a métrica M3.

- O *CoopSink-1.0* obteve ESF para o sorvedouro até 284% maior que o *CoopBridges* e até 324% maior que o *MST*. Como ESF maiores significam rotas menos eficientes, observa-se que as rotas para o sorvedouro de todas as variações do *CoopSink* são consideravelmente mais eficientes do que em outras propostas;
- Observa-se que, para valores menores de t , temos rotas bastante eficientes ao custo de um consumo maior de energia, assim como pode ser observado nos resultados de simulação para a métrica M2. Claramente, balancear esses parâmetros é um desafio e depende da aplicação. Em ambientes onde deseja-se aumentar o fluxo de dados, seria interessante manter valores baixos para o ESF, enquanto em ambientes onde o consumo de energia é mais importante os valores para o ESF devem ser mantidos mais altos.

4.4.4 M4: Número de arestas CC

A Figura 4.7 apresenta os resultados de simulação para a métrica M4. No eixo x , varia-se o número de nós na topologia de rede, enquanto, no eixo y , apresenta-se o número de arestas CC presentes na topologia final de cada técnica comparada. Observe que:

- O *MST* não cria qualquer aresta CC, sendo o número de arestas CC igual a zero em todas as configurações;

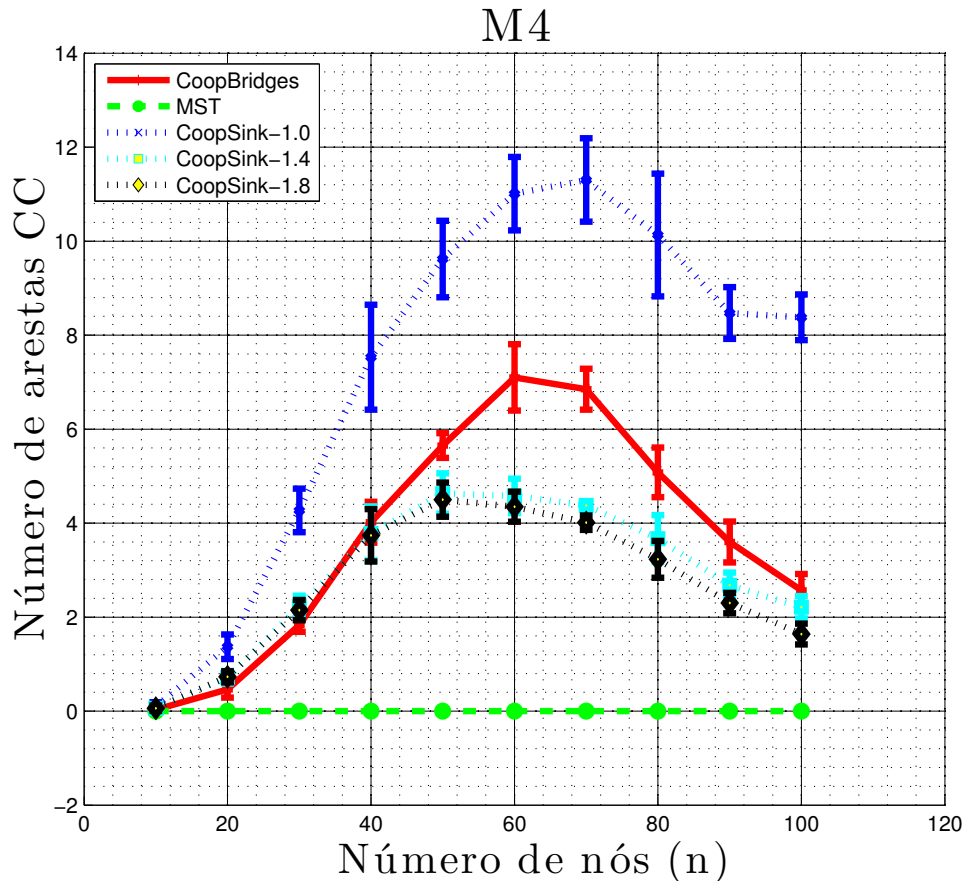


Figura 4.7: Resultados de simulação para a métrica M4.

- O *CoopSink* tende a ter mais arestas CC quando os valores de t são menores. Isso ocorre porque, quanto menor o valor de t , maior é a restrição em retirar uma aresta, ou seja, menos arestas são cortadas na topologia final. Esse resultado está condizente com o obtido nos resultados para a métrica M2, onde o consumo de energia do *CoopSink-1.0* foi maior do que nas outras variações, justamente porque existem mais arestas CC no seu grafo resultante;
- Até 40 nós, o *CoopSink-1.4*, o *CoopSink-1.8* e o *CoopBridges* possuem uma quantidade similar de arestas CC. No entanto, a partir de 40 nós, o *CoopSink-1.4* e o *CoopSink-1.8* chegam a ter até 56.4% menos arestas CC do que o *CoopBridges*. Isso mostra que o *CoopBridges* obtém uma quantidade maior de conectividade com o sorvedouro em todos os casos, enquanto mantém um consumo de energia, em alguns casos, menor e ainda necessita de uma quantidade igual ou menor de arestas CC na topologia final.

4.5 Discussão

Nesse capítulo, foi apresentado uma das propostas dessa dissertação. A proposta desse capítulo é uma técnica, chamada *CoopSink*, que pode ser utilizada para realizar controle de topologia em redes *ad hoc* cooperativas. Mais especificamente, o *CoopSink* é uma téc-

nica desenvolvida para redes *ad hoc* onde existe um nó sorvedouro e a necessidade de criar rotas eficientes para esse sorvedouro. Para isso, comunicação cooperativa é utilizada para aumentar a conectividade da rede e o controle de topologia diminui o consumo de energia dos nós enquanto mantém as rotas para o sorvedouro eficientes. Primeiramente, o modelo de comunicação e de rede foi definido, para depois descrever a técnica e expor os resultados obtidos por simulação. Como principais resultados temos que o *CoopSink* obteve uma conectividade com o sorvedouro até 94,3% mais eficiente do que outras técnicas da literatura enquanto as rotas foram até 324% mais eficientes quando se considera o EFS para o sorvedouro. Em trabalhos futuros, pretende-se implementar a técnica utilizando apenas informações locais.

Capítulo 5

Conclusão

Este trabalho abordou como temas principais redes sem fio *ad hoc*, protocolos de acesso ao meio, comunicação cooperativa e técnicas para economia de energia em redes sem fio. A motivação para estudar esses temas está na crescente popularização dos dispositivos sem fio, além do barateamento dos custos de produção e a necessidade de técnicas e protocolos que permitam uma melhor utilização desses dispositivos.

A primeira parte deste trabalho propôs um protocolo, chamado EEMC-MAC, que pode ser utilizado no agendamento de transmissões eficiente em energia em redes sem fio com múltiplos canais. O EEMC-MAC combina um esquema que faz os nós alternarem entre modos ativo e inativo com uma heurística que, ao mesmo tempo, resolve as dependências nas comunicações e faz o agendamento das transmissões. Para tal, o EEMC-MAC divide tais tarefas em fases de gerenciamento e transmissão. O gasto energético por nó na fase de gerenciamento é da ordem de $O(\log k)$ fatias de tempo, onde k é o número de canais do sistema, sendo o consumo de energia bastante eficiente na fase de transmissão. Resultados empíricos mostram que a fase de gerenciamento representa, em média, menos de 5% do tempo total de operação do protocolo enquanto o tempo de execução da fase de transmissão está, em média, a apenas 4% do tempo ótimo de transmissão. Por fim, foi sugerido como o EEMC-MAC poderia ser estendido para um ambiente com múltiplos saltos. Em trabalhos futuros pretende-se comparar o protocolo com outros da literatura, além de relaxar algumas limitações do cenário de comunicação como: presunção de que os nós estão sincronizados e ausência de interferência externa.

A segunda parte deste trabalho abordou controle de topologia em redes *ad hoc* cooperativas e propôs uma nova técnica, chamada *CoopSink*. O *CoopSink* é uma técnica desenvolvida para redes sem fio onde existe um nó sorvedouro e a necessidade de criar rotas eficientes para esse sorvedouro. Comunicação cooperativa é utilizada para aumentar a conectividade da rede e o controle de topologia diminui o consumo de energia dos nós enquanto mantém as rotas para o sorvedouro eficientes. Primeiramente, o modelo de comunicação e de rede foram definidos, para depois descrever a técnica e expor os resultados obtidos por simulação. Como principais resultados temos que o *CoopSink* obteve uma conectividade com o sorvedouro até 94,3% mais eficiente do que outras técnicas da literatura enquanto as rotas foram até 324% mais eficientes quando se considera o EFS para o sorvedouro. Em trabalhos futuros pretende-se explorar comunicação cooperativa para recuperar a conectividade em redes *ad hoc* de maneira pro-ativa e utilizando apenas informações localizadas.

Referências

- [1] Danijela Čabrić, Shridhar M Mishra, Daniel Willkomm, Robert Brodersen, and Adam Wolisz. A cognitive radio approach for usage of virtual unlicensed spectrum. In *14th IST Mobile and Wireless Communications Summit*, 2005. (Páginas ix, 1, e 2)
- [2] Felipe M Modesto, Marcos F Caetano, and Jacir L Bordim. An opportunistic mac protocol based on statistical spectrum analysis. In *Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), 2012 IEEE 26th International*, pages 855–862. IEEE, 2012. (Página 1)
- [3] Jin-A Park, Seung-Keun Park, Dong-Ho Kim, Pyung-Dong Cho, and Kyoung-Rok Cho. Experiments on radio interference between wireless lan and other radio devices on a 2.4 ghz ism band. In *Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual*, volume 3, pages 1798–1801. IEEE, 2003. (Página 1)
- [4] Juncheng Jia, Qian Zhang, and Xuemin Shen. HC-MAC: a hardware-constrained cognitive mac for efficient spectrum management. *Selected Areas in Communications, IEEE Journal on*, 26(1):106–117, 2008. (Página 1)
- [5] Joseph Mitola III and Gerald Q Maguire Jr. Cognitive radio: making software radios more personal. *Personal Communications, IEEE*, 6(4):13–18, 1999. (Página 1)
- [6] Qing Zhao and Brian M Sadler. A survey of dynamic spectrum access. *Signal Processing Magazine, IEEE*, 24(3):79–89, 2007. (Páginas 1, 2, e 17)
- [7] Moysés M Lima, Horácio A B F de Oliveira, Eduardo F Nakamura, and Antônio A Loureiro. Roteamento e agregação de dados baseado no rssi em redes de sensores sem fio. In *Simposio Brasileiro de Redes de Computadores, 2013. SBRC*. SBC, 2013. (Página 3)
- [8] Thiago F Neves and Jacirs L Bordim. Topology Control in Cooperative Ad Hoc Networks. *Journal Eletronic Notes in Theoretical Computer Science – ENTCS.*, A ser publicado em 2014. (Página 4)
- [9] Thiago F Neves and Jacir L Bordim. EEMC-MAC: An energy efficient protocol for multi-channel wireless networks. In *ICSNC 2013, The Eighth International Conference on Systems and Networks Communications*, pages 6–12, 2013. (Página 4)
- [10] Thiago F Neves and Jacir L Bordim. An energy efficient multi-channel protocol for wireless ad hoc networks. In *Computing Conference (CLEI), 2013 XXXIX Latin American*, pages 1–9. IEEE, 2013. (Página 4)

- [11] Juarez P S Junior, Lucas A Almeida, Felipe M Modesto, Thiago F Neves, and Li Weigang. An investigation on repost activity prediction for social media events. In *Web Information Systems Engineering - WISE 2012*, pages 719–725. Springer, 2012. (Página 4)
- [12] Thiago F Neves, Marcos F Caetano, and Jacir L Bordim. An energy optimal technique for multi-channel allocation and data scheduling in wireless networks. In *Informatica (CLEI), 2012 XXXVIII Conferencia Latinoamericana En*, pages 1–8. IEEE, 2012. (Página 4)
- [13] Thiago F Neves, Marcos F Caetano, and Jacir L Bordim. An energy-optimum and communication-time efficient protocol for allocation, scheduling and routing in wireless networks. In *Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), 2012 IEEE 26th International*, pages 848–854. IEEE, 2012. (Páginas 4 e 17)
- [14] Giuseppe Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *Selected Areas in Communications, IEEE Journal on*, 18(3):535–547, 2000. (Página 5)
- [15] Stefano Basagni, Marco Conti, Silvia Giordano, and Ivan Stojmenovic. *Mobile Ad Hoc Networking*. Wiley-IEEE press, 2004. (Página 5)
- [16] Azzedine Boukerche. *Handbook of Algorithms for Wireless Networking and Mobile Computing*. Chapman E Hall/CRC Computer and Information Science Series, 2006. (Páginas 6 e 9)
- [17] Sunil Kumar, Vineet S Raghavan, and Jing Deng. Medium access control protocols for ad hoc wireless networks: A survey. *Ad Hoc Networks*, 4(3):326–358, 2006. (Páginas 6 e 7)
- [18] Andrew S Tanenbaum. *Redes de Computadores*, volume 5. Campus, chapter 4, 2011. (Página 7)
- [19] Randolph Nelson and Leonard Kleinrock. Spatial TDMA: A collision-free multihop channel access protocol. *Communications, IEEE Transactions on*, 33(9):934–944, 1985. (Página 8)
- [20] Andrew J Viterbi. *CDMA: Principles of spread spectrum communication*, volume 3. Addison-Wesley Norwell, MA, 1995. (Página 8)
- [21] Hyung G Myung, Junsung Lim, and David J Goodman. Single carrier FDMA for uplink wireless transmission. *Vehicular Technology Magazine, IEEE*, 1(3):30–38, 2006. (Página 8)
- [22] Xuanming Dong, Pravin Varaiya, and Anuj Puri. An adaptive polling algorithm for pcf mode of 802.11 wireless lans. *EECS course 228a-COMMUNICATION NETWORKS, Department of Electrical Engineering and Computer Sciences, University of California, Berkeley*, 2003. (Página 8)

- [23] K C Sevick and Marjory J Johnson. Cycle time properties of the fddi token ring protocol. *Software Engineering, IEEE Transactions on*, 1(3):376–385, 1987. (Página 8)
- [24] Jay K Strosnider, Tom Marchok, and John Lehoczky. Advanced real-time scheduling using the IEEE 802.5 token ring. In *Real-Time Systems Symposium, 1988., Proceedings.*, pages 42–52. IEEE, 1988. (Página 9)
- [25] Christian Namislo. Analysis of mobile radio slotted aloha networks. *Vehicular Technology, IEEE Transactions on*, 33(3):199–204, 1984. (Página 9)
- [26] Phil Karn. MACA-A new channel access method for packet radio. In *ARRL/CRRL Amateur radio 9th computer networking conference*, volume 140, 1990. (Página 9)
- [27] Giuseppe Bianchi, Luigi Fratta, and Matteo Oliveri. Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LANs. In *Personal, Indoor and Mobile Radio Communications, 1996. PIMRC'96., Seventh IEEE International Symposium on*, volume 2, pages 392–396. IEEE, 1996. (Página 9)
- [28] Chane L Fullmer and J J Garcia-Luna-Aceves. FAMA-PJ: a channel access protocol for wireless lans. In *Proceedings of the 1st annual international conference on Mobile computing and networking*, pages 76–85. ACM, 1995. (Página 9)
- [29] Jungmin So and Nitin H Vaidya. A multi-channel mac protocol for ad hoc wireless networks. *TechnCSL/UIUC Technical Report*, 2003. (Páginas 9 e 10)
- [30] Kaushik R Chowdhury, Nagesh Nandiraju, Dave Cavalcanti, and Dharma P Agrawal. CMAC - A multi-channel energy efficient MAC for wireless sensor networks. In *Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE*, volume 2, pages 1172–1177. IEEE, 2006. (Páginas 10, 11, e 16)
- [31] Jingbin Zhang, Gang Zhou, Chengdu Huang, Sang Hyuk Son, and John A Stankovic. TMMAC: An energy efficient multi-channel mac protocol for ad hoc networks. In *Communications, 2007. ICC'07. IEEE International Conference on*, pages 3554–3561. IEEE, 2007. (Páginas 10, 12, 17, e 18)
- [32] M K Marina, S R Das, and A P Subramanian. A topology control approach for utilizing multiple channels in multi-radio wireless mesh networks. *Computer networks*, 54(2):241–256, 2010. (Páginas 10 e 12)
- [33] Ozlem Durmaz Incel, Lodewijk van Hoesel, Pierre Jansen, and Paul Havinga. MC-LMAC: A multi-channel mac protocol for wireless sensor networks. *Ad Hoc Networks*, 9(1):73–94, 2011. (Páginas 10, 12, e 17)
- [34] Aria Nosratinia, Todd E Hunter, and Ahmadreza Hedayat. Cooperative communication in wireless networks. *Communications Magazine, IEEE*, 42(10):74–80, 2004. (Página 12)
- [35] Ahmed K Sadek, Weifeng Su, and K J Ray Liu. Multinode cooperative communications in wireless networks. *Signal Processing, IEEE Transactions on*, 55(1):341–355, 2007. (Páginas 12, 13, 33, e 35)

- [36] Yang Xiao. IEEE 802.11n: enhancements for higher throughput in wireless lans. *Wireless Communications, IEEE*, 12(6):82–91, 2005. (Página 12)
- [37] Moray Rumney, Ryan Pirkel, Markus H Landmann, and David A Sanchez-Hernandez. MIMO over-the-air research, development, and testing. *International Journal of Antennas and Propagation*, 2012, 2012. (Página 12)
- [38] Ioannis Krikidis and John S Thompson. Overview of amplify-and-forward relaying. *Cooperative Communications for Improved Wireless Network Transmission: Framework for Virtual Antenna Array Applications*, page 29, 2010. (Página 13)
- [39] Jieun Yu, Heejun Roh, Wonjun Lee, Sangheon Pack, and Ding-Zhu Du. Cooperative bridges: topology control in cooperative wireless ad hoc networks. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE, 2010. (Páginas 13, 14, 15, 33, 34, 35, 42, e 50)
- [40] Nicholas Laneman, Gregory W Wornell, and David N C Tse. An efficient protocol for realizing cooperative diversity in wireless networks. In *Information Theory, 2001. Proceedings. 2001 IEEE International Symposium on*, page 294. IEEE, 2001. (Página 13)
- [41] Manav R Bhatnagar, A Hjørungnes, and Lingyang Song. Amplify-and-forward cooperative communications using double-differential modulation over nakagami-m channels. In *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, pages 350–355. IEEE, 2008. (Página 13)
- [42] Andrew Sendonaris, Elza Erkip, and Behnaam Aazhang. User cooperation diversity. part ii. implementation aspects and performance analysis. *Communications, IEEE Transactions on*, 51(11):1939–1948, 2003. (Página 13)
- [43] Aggelos Bletsas, Hyundong Shin, and Moe Z Win. Cooperative communications with outage-optimal opportunistic relaying. *Wireless Communications, IEEE Transactions on*, 6(9):3450–3460, 2007. (Página 13)
- [44] Ahmed S Ibrahim, Ahmed K Sadek, Weifeng Su, and K J Ray Liu. Cooperative communications with relay-selection: when to cooperate and whom to cooperate with? *Wireless Communications, IEEE Transactions on*, 7(7):2814–2827, 2008. (Páginas 13 e 42)
- [45] Yi Shi, Sushant Sharma, Y Thomas Hou, and Sastry Kompella. Optimal relay assignment for cooperative communications. In *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, pages 3–12. ACM, 2008. (Páginas 13 e 42)
- [46] Beibei Wang, Zhu Han, and K J Ray Liu. Distributed relay selection and power control for multiuser cooperative communication networks using buyer/seller game. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 544–552. IEEE, 2007. (Páginas 13 e 42)

- [47] Zhong Zhou, Shengli Zhou, Jun-Hong Cui, and Shuguang Cui. Energy-efficient cooperative communication based on power control and selective single-relay in wireless sensor networks. *Wireless Communications, IEEE Transactions on*, 7(8):3066–3078, 2008. (Páginas 13 e 42)
- [48] Manish Agarwal, Lixin Gao, Joon Ho Cho, and Jie Wu. Energy efficient broadcast in wireless ad hoc networks with hitch-hiking. *Mobile Networks and Applications*, 10(6):897–910, 2005. (Páginas 14 e 34)
- [49] Jie Wu, Mihaela Cardei, Fei Dai, and Shuhui Yang. Extended dominating set and its applications in ad hoc networks using cooperative communication. *Parallel and Distributed Systems, IEEE Transactions on*, 17(8):851–864, 2006. (Páginas 14 e 34)
- [50] Gentian Jakllari, Srikanth V Krishnamurthy, Michalis Faloutsos, and Prashant V Krishnamurthy. On broadcasting with cooperative diversity in multi-hop wireless networks. *Selected Areas in Communications, IEEE Journal on*, 25(2):484–496, 2007. (Páginas 14 e 34)
- [51] Gentian Jakllari, Srikanth V Krishnamurthy, Michalis Faloutsos, Prashant V Krishnamurthy, and Ozgur Ercetin. A framework for distributed spatio-temporal communications in mobile ad hoc networks. In *Proc. IEEE Infocom*, pages 1–13, 2006. (Páginas 14, 34, e 35)
- [52] Mathias Kurth, Anatolij Zubow, and J-P Redlich. Cooperative opportunistic routing using transmit diversity in wireless mesh networks. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 1310–1318. IEEE, 2008. (Páginas 14 e 34)
- [53] Khaled Ben Letaief and Wei Zhang. Cooperative communications for cognitive radio networks. *Proceedings of the IEEE*, 97(5):878–893, 2009. (Páginas 14 e 34)
- [54] Ying Zhu, Minsu Huang, Siyuan Chen, and Yu Wang. Energy-efficient topology control in cooperative ad hoc networks. *Parallel and Distributed Systems, IEEE Transactions on*, 23(8):1480–1491, 2012. (Páginas 14, 15, 16, 33, 34, 35, e 50)
- [55] Mihaela Cardei, Jie Wu, and Shuhui Yang. Topology control in ad hoc wireless networks using cooperative communication. *Mobile Computing, IEEE Transactions on*, 5(6):711–724, 2006. (Páginas 14, 15, e 34)
- [56] Xiaohua Jia, Deying Li, and Dingzhu Du. QoS topology control in ad hoc wireless networks. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 2 of 1, pages 1264–1272. IEEE, 2004. (Página 14)
- [57] Ting-Chao Hou and Victor Li. Transmission range control in multihop packet radio networks. *Communications, IEEE Transactions on*, 34(1):38–44, 1986. (Página 14)
- [58] W-T Chen and N-F Huang. The strongly connecting problem on multihop packet radio networks. *Communications, IEEE Transactions on*, 37(3):293–295, 1989. (Páginas 14 e 33)

- [59] Ram Ramanathan and Regina Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2 of 1, pages 404–413. IEEE, 2000. (Página 14)
- [60] Li Li, Joseph Y Halpern, Paramvir Bahl, Yi-Min Wang, and Rogert Wattenhofer. Analysis of a cone-based distributed topology control algorithm for wireless multihop networks. In *Proceedings of the twentieth annual ACM symposium on Principles of distributed computing*, pages 264–273. ACM, 2001. (Página 14)
- [61] Ning Li, Jennifer C Hou, and Lui Sha. Design and analysis of an mst-based topology control algorithm. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 3 of 1, pages 1702–1712. IEEE, 2003. (Página 14)
- [62] Xiang-Yang Li. Topology control in wireless ad hoc networks. *Mobile Ad Hoc Networking*, pages 175–204, 2003. (Páginas 14 e 33)
- [63] Nagesh Javali. *Topology Control for Wireless Ad-hoc Networks*. ProQuest, 2008. (Página 14)
- [64] Prasant Mohapatra and Srikanth Krishnamurthy. *Ad Hoc Networks - Technologies and Protocols*. Springer Science + Business Media, Inc. chapters 1, 3, 6, 2005. (Página 16)
- [65] Shiao-Li Tsao and Chung-Huei Huang. A survey of energy efficient mac protocols for IEEE 802.11 wlan. *Computer Communications*, 34(1):54–67, 2011. (Página 17)
- [66] C Stevenson, Gerald Chouinard, Zhongding Lei, Wendong Hu, S Shellhammer, and Winston Caldwell. IEEE 802.22: The first cognitive radio wireless regional area network standard. *IEEE Stands in Communications and Networking*, pages 130 – 138, January 2009. (Página 17)
- [67] Ashish Raniwala, Kartik Gopalan, and Tzi-cker Chiueh. Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 8(2):50–65, 2004. (Página 17)
- [68] Lei Tang, Yanjun Sun, Omer Gurewitz, and David B Johnson. EM-MAC: a dynamic multichannel energy-efficient mac protocol for wireless sensor networks. In *Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, page 23. ACM, 2011. (Página 17)
- [69] Jianfeng Wang, Yuguang Fang, and Dapeng Oliver Wu. A power-saving multi-radio multi-channel mac protocol for wireless local area networks. In *Proceedings of IEEE Infocom*. Citeseer, 2006. (Página 18)
- [70] Goutam K Audhya, Koushik Sinha, Sasthi C Ghosh, and Bhabani P Sinha. A survey on the channel assignment problem in wireless networks. *Wireless Communications and Mobile Computing*, 11(5):583–609, 2011. (Página 19)

- [71] Nobuo Funabiki and Yoshiyasu Takefuji. A neural network parallel algorithm for channel assignment problems in cellular radio networks. *Vehicular Technology, IEEE Transactions on*, 41(4):430–437, 1992. (Página 19)
- [72] Jeremy Eric Elson and Deborah Estrin. *Time synchronization in wireless sensor networks*. PhD thesis, University of California, Los Angeles, 2003. (Página 25)
- [73] V G Vizing. On an estimate of the chromatic class of a p-graph. *Diskret. Analiz*, 3(7):25–30, 1964. (Página 30)
- [74] Ossama Younis and Sonia Fahmy. HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *Mobile Computing, IEEE Transactions on*, 3(4):366–379, 2004. (Páginas 31 e 32)
- [75] Rajmohan Rajaraman. Topology control and routing in ad hoc networks: a survey. *ACM SIGACT News*, 33(2):60–73, 2002. (Página 33)
- [76] Carlos M Cordeiro, Hrishikesh Gossain, and Dbarma P Agrawal. Multicast over wireless mobile ad hoc networks: present and future directions. *Network, IEEE*, 17(1):52–59, 2003. (Página 33)
- [77] Xiang-Yang Li, Peng-Jun Wan, and Yu Wang. Power efficient and sparse spanner for wireless ad hoc networks. In *Computer Communications and Networks, 2001. Proceedings. Tenth International Conference on*, pages 564–567. IEEE, 2001. (Página 33)
- [78] Roger Wattenhofer, Li Li, Paramvir Bahl, and Y-M Wang. Distributed topology control for power efficient operation in multihop wireless ad hoc networks. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1388–1397. IEEE, 2001. (Página 33)
- [79] Andrea E F Clementi, Paolo Penna, and Riccardo Silvestri. On the power assignment problem in radio networks. *Mobile Networks and Applications*, 9(2):125–140, 2004. (Página 33)
- [80] Douglas M Blough, Mauro Leoncini, Giovanni Resta, and Paolo Santi. On the symmetric range assignment problem in wireless ad hoc networks. In *Proceedings of the IFIP 17th World Computer Congress-TC1 Stream/2nd IFIP International Conference on Theoretical Computer Science: Foundations of Information Technology in the Era of Networking and Mobile Computing*, pages 71–82, 2002. (Página 33)
- [81] Azzedine Boukerche, H A B Oliveira, Eduardo F Nakamura, and Alfredo F Loureiro. A novel location-free greedy forward algorithm for wireless sensor networks. In *Communications, 2008. ICC'08. IEEE International Conference on*, pages 2096–2101. IEEE, 2008. (Página 34)
- [82] Joseph B Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical society*, 7(1):48–50, 1956. (Página 50)

- [83] Michael Grant, Stephen Boyd, and Yinyu Ye. Cvx: Matlab software for disciplined convex programming, 2008. (Página 50)