



TRABALHO DE GRADUAÇÃO

Comunicação Cooperativa em Redes Móveis Ad Hoc
Consciente em Energia e Atraso Fim-a-Fim

Camila Feitosa Rêgo

Brasília, Junho de 2014

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia

TRABALHO DE GRADUAÇÃO

**Comunicação Cooperativa em Redes Móveis Ad Hoc
Consciente em Energia e Atraso Fim-a-Fim**

Camila Feitosa Rêgo

*Relatório submetido ao Departamento de Engenharia
Elétrica como requisito parcial para obtenção
do grau de Engenheira de Redes de Comunicação*

Banca Examinadora

Prof. Jacir Luiz Bordim, CIC/UnB
Orientador

Prof. Marcelo Menezes de Carvalho, ENE/UnB
Examinador interno

Prof. Ugo Silva Dias, ENE/UnB
Examinador interno

Dedicatória

Dedico esse trabalho a Deus e ao meu grande exemplo: minha mãe.

Camila Feitosa Rêgo

Agradecimentos

Agradeço primeiramente a Deus, pois sem a presença dEle na minha vida essa conquista não seria possível. Agradeço a minha querida e amada mãe que me apoiou em todos os momentos da minha vida e é o meu exemplo de pessoa, profissional, mulher e mãe. Agradeço aos meus amigos e colegas de curso que me ajudaram e me deram forças em cada momento desta jornada. Agradeço a todos os professores que participaram da minha formação por cada lição (acadêmica, ética e moral) que pude aprender com cada um. Também ao meu noivo Rodrigo que me apoiou em todos os passos dessa conquista. Agradeço também ao colega Tiago Neves por todos os seus conselhos, disponibilidade e paciência. Em especial, agradeço ao Professor Jacir Luiz Bordim que me acompanhou durante todo o projeto de graduação e demonstrou imensa consideração, empenho e dedicação na evolução deste trabalho.

Camila Feitosa Rêgo

RESUMO

O presente trabalho considera a utilização de técnicas de comunicação cooperativa em ambientes sem fio *ad hoc* como alternativa para aumentar a conectividade e, ao mesmo tempo, para reduzir o atraso e aumentar a eficiência energética das rotas para um sorvedouro. Neste contexto, são desenvolvidas duas novas técnicas: *Cooperative Sink Communication (CSC)* e *Cooperative Sink Communication - Helper Set (CSC-HS)*. O *CSC* utiliza comunicação cooperativa para aumentar a conectividade entre os nós sensores e o nó sorvedouro e controle de topologia para diminuir o atraso fim-a-fim entre estes nós. A técnica *CSC-HS* utiliza comunicação cooperativa e controle de topologia para aumentar a conectividade e diminuir o atraso fim-a-fim mantendo a eficiência energética em redes *ad hoc* utilizando como métrica a quantidade de nós auxiliares que compõem a aresta cooperativa. Num primeiro momento, foi feita uma revisão teórica dos principais conceitos relativos ao entendimento deste trabalho: redes *ad hoc*, modelos de propagação em ambientes sem fio e vazão máxima teórica de redes *ad hoc*. Em um segundo momento, são apresentados os conceitos de comunicação cooperativa e é feita uma revisão bibliográfica sobre controle de topologia em redes *ad hoc*. Em seguida, as duas novas técnicas são propostas e são avaliadas por meio de comparação com outras técnicas da literatura e métricas definidas no escopo deste trabalho. O *CSC*, comparado com alternativas semelhantes, permite reduzir o atraso em até 50% nos cenários avaliados, enquanto o *CSC-HS* permite reduzir o atraso em até 45% mantendo uma eficiência energética por rota próxima de outras técnicas da literatura.

ABSTRACT

This work considers the use of cooperative communication technique in mobile *ad hoc* networks as an alternative to increase connectivity and, simultaneously, reduce the delay and increase the energy efficiency of routes to a sink. In this context, two new techniques are developed: *Cooperative Sink Communication (CSC)* and *Cooperative Sink Communication - Helper Set (CSC-HS)*. The *CSC* uses cooperative communication to increase connectivity between the sensor nodes and the sink node and topology control to reduce the end-to-end delay. *CSC-HS* technique uses cooperative communication and topology control to increase connectivity and reduce the delay by maintaining end-to-end energy efficiency in mobile *ad hoc* networks using as metric the number of helper nodes that compose the cooperative edge. At first, the main concepts of this work are briefly summarized: mobile *ad hoc* networks, propagation models and theoretical maximum throughput. In a second moment, the concepts of cooperative communications are presented and previous works in topology control in mobile *ad hoc* networks are reviewed. Then the two new techniques are proposed and evaluated by means of comparison with other techniques in the literature and metrics defined in this work. The *CSC*, compared to similar alternatives, can reduce delay by up to 50% in the evaluated scenarios, while the *CSC-HS* reduces delay by up to 45% maintaining the energy efficiency per route close to other techniques.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	OBJETIVOS	2
1.2	METODOLOGIA	2
1.3	RESULTADOS OBTIDOS	2
1.4	ESTRUTURA DO DOCUMENTO	3
2	REVISÃO TEÓRICA	4
2.1	REDES <i>Ad Hoc</i> SEM FIO	4
2.1.1	HISTÓRICO	6
2.1.2	VANTAGENS E DESVANTAGENS	6
2.1.3	REDES DE SENSORES	7
2.2	MODELOS DE PROPAGAÇÃO	8
2.2.1	MODELO DE PROPAGAÇÃO NO ESPAÇO LIVRE	8
2.2.2	MODELO DE PROPAGAÇÃO DE DOIS RAIOS	9
2.3	ROTEAMENTO EM REDES AD HOC	10
2.3.1	PROTOCOLOS DE ROTEAMENTO PRÓ-ATIVOS	10
2.3.2	PROTOCOLOS DE ROTEAMENTO REATIVOS	11
2.3.3	PROTOCOLOS HÍBRIDOS	12
2.4	VAZÃO MÁXIMA TEÓRICA EM REDES AD HOC SEM FIO	13
2.4.1	INTRODUÇÃO AO CSMA/CA COM RTS/CTS	13
2.4.2	CÁLCULO DA VAZÃO MÁXIMA TEÓRICA	14
2.5	DISCUSSÃO	17
3	CONTROLE DE TOPOLOGIA EM REDES AD HOC COOPERATIVAS	18
3.1	COMUNICAÇÃO COOPERATIVA	18
3.1.1	MODELO DE COMUNICAÇÃO COOPERATIVA	18
3.1.2	MODELO DE REDE COOPERATIVA	20
3.2	TÉCNICAS DE RETRANSMISSÃO COOPERATIVA	23
3.2.1	AMPLIFICAR E ENCAMINHAR	24
3.2.2	DECODIFICAR E ENCAMINHAR	24
3.2.3	COMPRESS AND FORWARD	24
3.3	CONTROLE DE TOPOLOGIA E REDES AD HOC COOPERATIVAS	25
3.3.1	MST	25

3.3.2	LMST	25
3.3.3	DIJKTRA	26
3.3.4	DTCC/ITCC	26
3.3.5	COOPBRIDGES.....	26
3.3.6	GREEDY(ADD/DEL)LINK.....	26
3.3.7	COOPSINK	27
3.4	DISCUSSÃO	27
4	PROPOSTA DO COOPERATIVE SINK COMMUNICATION	28
4.1	FORMULAÇÃO DO PROBLEMA.....	28
4.2	DESCRIÇÃO DA PROPOSTA	29
4.2.1	ALGORITMO: <i>Greedy Helper Set Selection</i>	30
4.2.2	<i>CSC</i> : DESCRIÇÃO DA TÉCNICA	32
4.2.3	<i>CSC - HS</i> : DESCRIÇÃO DA TÉCNICA	38
4.3	RESULTADOS DE SIMULAÇÃO.....	39
4.3.1	M1: MÉDIA DE SALTOS	42
4.3.2	M2: MÉDIA DE TRANSMISSÃO POR ENLACE.....	45
4.3.3	M3: POTÊNCIA MÉDIA DA ROTA.....	47
4.3.4	M4: CONECTIVIDADE COM O SORVEDOURO.....	48
4.3.5	M5: POTÊNCIA MÉDIA DE TRANSMISSÃO POR NÓ.....	50
4.3.6	M6: NÚMERO DE ARESTAS CC	52
4.4	DISCUSSÃO	54
5	CONSIDERAÇÕES FINAIS	55
	REFERÊNCIAS BIBLIOGRÁFICAS	56

LISTA DE FIGURAS

2.1	Exemplo de rede sem fio com infraestrutura.....	5
2.2	Exemplo de rede <i>ad hoc</i> [1].....	5
2.3	Exemplo de rede de sensores com um sorvedouro (Figura à esquerda) e múltiplos sorvedouros (Figura à direita).....	8
2.4	Diagrama de tempo para CSMA/CA com RTS/CTS [2].....	14
3.1	(a) Cenário onde existem 3 nós próximos v_1, v_2 e v_3 e um distante v_4 e (b) Primeiro momento da CC onde v_1 transmite o dado para v_2 e v_3	19
3.2	(a) Segundo momento da CC cooperativa onde v_1, v_2 e v_3 transmitem o dado para v_4 e (b) Aumento do raio de cobertura de v_1 com a utilização dos nós ajudantes v_2 e v_3 para criar um enlace cooperativo entre v_1 e v_4	20
3.3	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	23
4.1	Exemplo de topologia de rede, onde $ V = 70$ e $ E_P = \emptyset$	34
4.2	Grafo formado a partir da Figura 4.1 quando os nós operam em sua potência máxima P_{MAX}	34
4.3	Grafo criado a partir da Figura 4.2 adicionando todas as arestas CC possíveis.....	35
4.4	Grafo criado a partir da Figura 4.3 em que as arestas resultantes são as que promovem a menor quantidade de saltos até o sorvedouro.....	37
4.5	Grafo criado a partir da Figura 4.3 em que as arestas resultantes são as que promovem a menor quantidade de transmissões até o sorvedouro.....	39
4.6	Média de Saltos para $R_{MAX} = 50$ metros.....	43
4.7	Média de Saltos para $R_{MAX} = 70$ metros.....	44
4.8	Média de Transmissão por Enlace para $R_{MAX} = 50$ metros.....	46
4.9	Média de Transmissão por Enlace para $R_{MAX} = 70$ metros.....	46
4.10	Potência Média da Rota para $R_{MAX} = 50$ metros.....	47
4.11	Potência Média da Rota para $R_{MAX} = 70$ metros.....	48
4.12	Conectividade com o Sorvedouro para $R_{MAX} = 50$ metros.....	49
4.13	Conectividade com o Sorvedouro para $R_{MAX} = 70$ metros.....	49
4.14	Potência Média de Transmissão por Nó para $R_{MAX} = 50$ metros.....	51
4.15	Potência de Transmissão por Nó para $R_{MAX} = 70$ metros.....	51

4.16	Número de Arestas CC para $R_{MAX} = 50$ metros.	53
4.17	Número de Arestas CC para $R_{MAX} = 70$ metros.	53

LISTA DE TABELAS

2.1	Componentes de atraso para o CSMA/CA com a utilização dos protocolos de controle RTS/CTS para diferentes tipos de tecnologias da camada física [2].	15
2.2	Valores de a e b para o CSMA/CA com a utilização dos protocolos de controle RTS/CTS para diferentes tipos de tecnologias da camada física [2].	15
2.3	Atraso fim-a-fim para um salto utilizando CSMA/CA com RTS/CTS	16
2.4	Vazão Máxima Teórica utilizando CSMA/CA com RTS/CTS	16
4.1	Parâmetros de simulação dos cenários 1 e 2	41
4.2	Atraso de Fim-a-Fim utilizando o <i>CSC</i> e o <i>CSC-HS</i> com quantidade de saltos = 6, utilizando CSMA/CA com RTS/CTS, $R_{MAX} = 50m$ e $n = 120$	44
4.3	Atraso de fim-a-fim utilizando o <i>CoopSink</i> e com quantidade de saltos = 11, utilizando CSMA/CA com RTS/CTS, $R_{MAX} = 50m$ e $n = 120$	45

LISTA DE ALGORITMOS

4.1	: <i>GreedyHelperSetSelection</i> ($v_i, N(v_i), v_j, d_{ij}, \alpha$)	32
4.2	: <i>CSCStep1</i> (V, P_{MAX})	33
4.3	: <i>CSCStep2</i> ($\overline{G_P}$)	35
4.4	: <i>CSCStep3</i> ($G_P, E_P, v_o, \mathfrak{R}$)	36
4.5	: <i>CSCStep4</i> (G'_P)	38

LISTA DE SIGLAS

Siglas

AODV	<i>Ad Hoc On Demand Distance Vector Routing Protocol</i>
CC	Comunicação Cooperativa (do Inglês, <i>Cooperative Communication</i>)
CoopBridges	Pontes Cooperativas (do Inglês, <i>Cooperative Bridges</i>)
CoopSink	<i>Cooperative Sink</i>
CSC	<i>Cooperative Sink Communication</i>
CSC-HS	<i>Cooperative Sink Communication - Helper Set</i>
CSMA/CA	Protocolo de acesso múltiplo com detecção de portadora e prevenção de colisão (do Inglês, <i>Carrier Sense Multiple Access/Collision Avoidance</i>)
CTS	Liberado para envio (do Inglês, <i>Clear To Send</i>)
DFC	Função coordenada de distribuição (do Inglês, <i>Distributed Coordination Function</i>)
DIFS	Tempo entre quadro distribuído (do Inglês, <i>Distributed Interframe Space</i>)
DSDV	<i>Dynamic Destination Sequenced Distance-Vector Routing Protocol</i>
DSR	<i>Dynamic Source Routing Protocol</i>
ERB	Estação Rádio-Base
HSPA	Acesso de Pacotes Evoluído a Alta Velocidade (do Inglês, <i>Evolved High-Speed Packet Acces</i>)
IEEE	Instituto de Engenheiros Elétricos e Eletrônicos (do Inglês, <i>Institute of Electrical and Electronics Engineers</i>)
IETF	Força Tarefa de Engenharia da Internet (do Inglês, <i>Internet Engineering Task Force</i>)
LMST	Árvore Geradora Mínima Localizada (do Inglês, <i>Localized Minimum Spanning Tree</i>)
MAC	Controle de Acesso ao Meio (do Inglês, <i>Medium Access Control</i>)
MANET	Rede <i>Ad Hoc</i> Sem Fio Móvel (do Inglês, <i>Mobile Ad Hoc Networks</i>)
MIMO	Técnica de transmissão sem fio que utiliza múltiplas antenas para transmissão e recepção (do Inglês, <i>Multiple-Input, Multiple-Output</i>)
MPR	<i>Multipoint Relaying</i>
MSDU	Unidade de Dados de Serviço da Camada de Enlace (do Inglês, <i>MAC Service Data Unit</i>)
MST	Árvore geradora mínima (do Inglês, <i>Minimum Spanning Tree</i>)

NAV	Vetor de Alocação da Rede (do Inglês, <i>Network Allocation Vector</i>)
OLSR	<i>Optimized Link State Routing Protocol</i>
PLE	Expoente de perda de caminho (do Inglês, <i>Path Loss Exponent</i>)
PRNET	<i>Packet Radio Network</i>
QoS	Qualidade de Serviço (do Inglês, <i>Quality of Service</i>)
RTS	Requisição para Envio (do Inglês, <i>Request To Send</i>)
SDU	Unidade de Dados de Serviço (do Inglês, <i>Service Data Unit</i>)
SIFS	Curto Intervalo entre Quadros (do Inglês, <i>Short Interframe Space</i>)
SNR	Razão sinal ruído (do Inglês, <i>Signal Noise Ratio</i>)
TMT	Vazão Máxima Teórica (do Inglês, <i>Theoretical Maximum Throughput</i>)
U.S. DARPA	Agência de Projetos de Defesa Avançados do Estados Unidos (do Inglês, <i>United States Defense Advanced Research Project Agency</i>)
WiMAX	Interoperabilidade Mundial para Acesso de Micro-ondas (do Inglês, <i>Worldwide Interoperability for Microwave Access</i>)

Capítulo 1

Introdução

Na última década, diversos avanços tecnológicos nas áreas de eletrônica, rádios e comunicação sem fio estimularam o desenvolvimento e uso de sensores em áreas ligadas a processos físicos, químicos, biológicos, entre outros. Uma rede de sensores sem fio é composta por diversos módulos distribuídos, os nós sensores, e um ou mais nós sorvedouros. Os nós sensores captam informações do ambiente e enviam essa informação para o nó sorvedouro que possui maior capacidade de processamento e de comunicação com outras redes.

Os nós sensores apresentam, em geral, dimensões pequenas, alcance de rádio limitado e utilizam baterias como fonte de alimentação. Desta forma, os nós apresentam limitações quanto à capacidade de processamento, armazenamento, de energia e de alcance do seu rádio de comunicação. Os nós sensores podem ser agrupados para realizarem tarefas de forma cooperativa de maneira a suplantar as limitações de bateria e alcance.

Devido às limitações das redes de sensores sem fio, manter a comunicação com o sorvedouro e executar essa comunicação de forma rápida e energeticamente eficiente é essencial. Assim, estuda-se a utilização de comunicação cooperativa para manter ou aumentar a conectividade em uma rede de sensores.

Alguns trabalhos da literatura abordam a utilização de comunicação cooperativa para: aumentar a conectividade [3], garantir eficiência energética [4] ou ambos [5]. Em especial, a técnica *CoopSink* [5] utiliza comunicação cooperativa e controle de topologia para aumentar a conectividade mantendo a eficiência energética numa rede de sensores sem fio. Entretanto, esta técnica não avalia o impacto no atraso fim-a-fim.

Este trabalho faz um estudo do uso de comunicação cooperativa para aumentar a conectividade numa rede de sensores sem fio e do uso de controle de topologia para diminuir o atraso fim-a-fim da comunicação entre os nós sensores e o nó sorvedouro e, simultaneamente, manter as rotas para o sorvedouro as mais eficientes possíveis.

1.1 Objetivos

Numa rede de sensores, existe a necessidade de que os nós se comuniquem com o nó sorvedouro. Este nó sorvedouro concentra as informações recebidas pelos nós sensores e pode utilizar as informações localmente ou enviá-la para outras redes. Como o alcance de transmissão dos nós sensores é limitado, esta comunicação é feita por múltiplos saltos. Além de se comunicar com o sorvedouro, é interessante que a comunicação aconteça de forma rápida e energeticamente eficiente já que os nós sensores, normalmente, utilizam baterias como fonte de alimentação. Assim, a proposta deste trabalho consiste em estudar como a técnica de comunicação cooperativa pode ser utilizada para aumentar a conectividade de uma rede de sensores e para, em conjunto com controle de topologia, reduzir o atraso fim-a-fim sem comprometer a eficiência energética. Mais especificamente, duas novas técnicas são propostas para o cenário onde exista um nó sorvedouro e diversos nós espalhados pela rede que precisam reportar informações para o nó sorvedouro da forma mais rápida e com o menor custo energético possível. Como objetivos secundários, *(i)* pretende-se avaliar as duas propostas com outras técnicas da literatura, *(ii)* mensurar o atraso fim-a-fim e *(iii)* mensurar o custo energético para realizar esta tarefa.

1.2 Metodologia

A metodologia deste trabalho consistiu na revisão do estado da arte de redes *ad hoc* sem fio, modelos de propagação, comunicação cooperativa e controle de topologia. Em seguida, foram implementadas por meio do software Matlab [6] e analisadas técnicas da literatura que utilizam as técnicas de comunicação cooperativa e controle de topologia. Após a implementação, coleta e análise de dados foram propostas melhorias por meio do desenvolvimento de duas técnicas. Por fim, as duas novas técnicas propostas foram implementadas e validadas e tiveram seu desempenho comparado com outras técnicas da literatura por meio de simulação.

1.3 Resultados Obtidos

Neste trabalho é apresentado um estudo da utilização de comunicação cooperativa para aumentar a conectividade de uma rede de sensores e, em conjunto com controle de topologia, reduzir o atraso fim-a-fim mantendo a eficiência energética. A partir deste estudo, são propostas duas novas técnicas: *Cooperative Sink Communication (CSC)* e *Cooperative Sink Communication - Helper Set (CSC-HS)*. A primeira técnica proposta, *CSC*, tem por objetivos aumentar a conectividade da rede e reduzir o atraso fim-a-fim para o sorvedouro. A segunda técnica proposta, *CSC-HS*, promove um aumento da conectividade e faz um balanço entre diminuir o atraso fim-a-fim e manter o custo energético das rotas para o sorvedouro o mais reduzido possível. As duas técnicas propostas são validadas e avaliadas e os resultados comparados com outras técnicas da literatura: CoopSink [5] e MST [7]. Os resultados obtidos mostram que:

- Os resultados obtidos mostram que as técnicas *CSC* e *CSC-HS* apresentam uma conecti-

vidade de mais de 80% dos nós com o sorvedouro quando temos uma quantidade de nós maior que 90 e um alcance de 50 metros. Quando o alcance é de 70 metros, a conectividade aumenta para quase 100% com a mesma quantidade de nós;

- A técnica *CSC* apresenta o menor atraso fim-a-fim para o sorvedouro, obtendo um resultado até 50% melhor do que a outra técnica da literatura estudada, o *CoopSink* (abreviação para *Cooperative Sink*). A técnica *CSC-HS* apresenta um atraso fim-a-fim até 45% melhor do que o resultado do *CoopSink*[5] e, ainda, consegue obter uma eficiência energética por rota muito próxima do *CoopSink*, que é uma técnica da literatura voltada para a eficiência energética em redes de sensores.

Assim, os resultados obtidos validam o *CSC* como uma técnica para redes de sensores que aumenta a conectividade da rede e garante o menor atraso fim-a-fim dos nós para o sorvedouro e validam o *CSC-HS* como uma técnica que aumenta a conectividade da rede e garante o menor atraso fim-a-fim com a maior eficiência energética.

1.4 Estrutura do Documento

Este trabalho é estruturado como segue. O Capítulo 2 faz uma introdução às redes *ad hoc* sem fio, modelos de propagação, controle de potência e roteamento em redes *ad hoc* e apresenta o cálculo da vazão máxima teórica de redes *ad hoc*. O Capítulo 3 apresenta o modelo de comunicação cooperativa, as principais técnicas de retransmissão cooperativa e faz uma breve revisão do estado da arte sobre controle de topologia em redes *ad hoc* que utilizam, ou não, comunicação cooperativa. O Capítulo 4 apresenta as principais contribuições deste trabalho. Mais especificamente, este capítulo apresenta duas técnicas: *CSC*, que tem como objetivo aumentar a conectividade com o sorvedouro e reduzir o atraso fim-a-fim; e *CSC-HS*, que tem como objetivo aumentar a conectividade com o sorvedouro e reduzir o atraso fim-a-fim mantendo a eficiência energética. Além disso, este capítulo apresenta uma avaliação dessas propostas comparando-as com as outras técnicas da literatura. O Capítulo 5 faz uma breve revisão do documento, conclui o trabalho e apresenta uma proposta futura.

Capítulo 2

Revisão Teórica

Este capítulo faz uma breve revisão dos conceitos básicos necessários para o entendimento deste trabalho. A Seção 2.1 faz uma introdução às redes *ad hoc* sem fio. A Seção 2.2 apresenta dois modelos de propagação: espaço livre e dois raios. A Seção 2.3 apresenta os principais protocolos de roteamento em redes *ad hoc*. A Seção 2.4 trata do cálculo da vazão máxima teórica em redes *ad hoc* sem fio. Por fim, a Seção 2.5 faz uma discussão dos assuntos tratados neste capítulo.

2.1 Redes *Ad Hoc* Sem Fio

As redes sem fio tem se tornado crescentemente populares nas últimas décadas, principalmente com a adaptação de redes sem fio para suportar mobilidade. As redes sem fio podem ser classificadas em dois tipos: redes sem fio com infraestrutura e redes sem fio sem infraestrutura [8].

As redes sem fio infraestruturadas, são redes que utilizam um ponto de controle, como um ponto de acesso ou uma estação rádio base (ERB). Um dispositivo móvel se conecta e comunica com o ponto de acesso mais próximo dentro do seu raio de alcance. Quando o sinal se propaga para fora da área de abrangência de um ponto de acesso, um “hand off” ocorre do antigo ponto de acesso para a novo ponto de acesso. Exemplos de redes sem fio com infraestrutura são redes de celular ou pontos de acesso que utilizam *WiFi*. A Figura 2.1 apresenta um exemplo de rede sem fio com infraestrutura: rede *WiFi* com ponto de acesso.

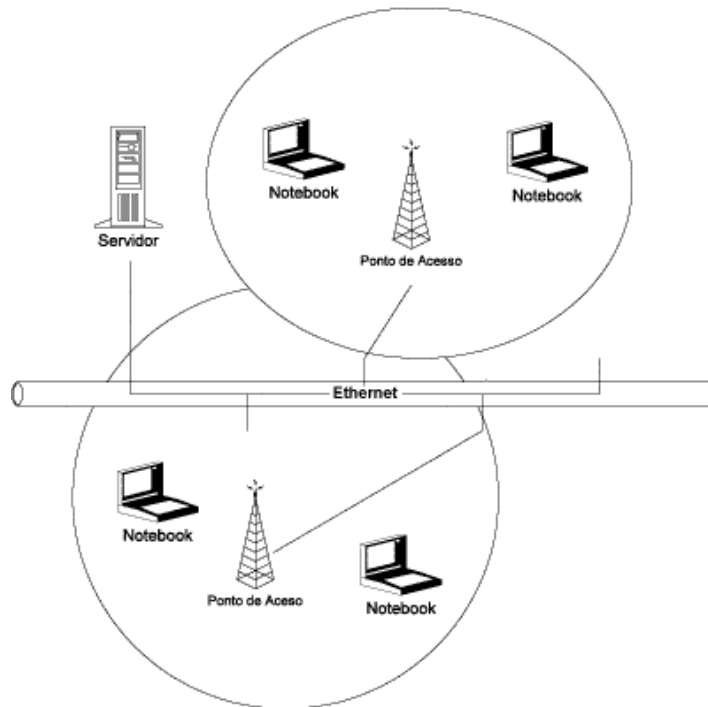


Figura 2.1: Exemplo de rede sem fio com infraestrutura.

Redes sem fio sem infraestrutura: este tipo de rede, também chamada de rede *ad hoc*, não dispõe de roteadores fixos, os nós podem ser capazes de movimento e podem se conectar dinamicamente. A conectividade entre os nós pode mudar frequentemente requerendo uma permanente adaptação e reconfiguração das rotas. Em redes *ad hoc*, os nós agem como roteadores que descobrem e mantêm rotas para outros nós da rede e a comunicação é feita por meio de cooperação entre os nós. Exemplos de aplicações de redes *ad hoc* são: operações de busca-resgate de emergência, convenções onde existe a necessidade de difundir informações rapidamente, aquisições de dados em terrenos inóspitos (aplicações militares) e redes de sensores onde os nós fixos captam informações do ambiente e transmitem por difusão para um nó sorvedouro. A Figura 2.2 apresenta um exemplo de rede *ad hoc*: uma rede *WiFi* sem ponto de acesso.



Figura 2.2: Exemplo de rede *ad hoc* [1].

2.1.1 Histórico

As aplicações de redes *ad hoc* datam do início da década de 70 quando a U.S. DARPA (*United States Defense Advanced Research Projects Agency*) iniciou o projeto PRNET (*Packet Radio Network*) [9] para explorar o uso de redes de pacotes de rádio num ambiente tático para comunicação de dados. Mais tarde, em 1983, a DARPA lançou o programa SURAN (*Survivable Adaptive Network*) para expandir a tecnologia desenvolvida no projeto PRNET para suportar grandes redes, e para desenvolver protocolos de rede adaptativos os quais pudessem adaptar-se às rápidas mudanças de condições em um ambiente tático. O último da série dos programas iniciados pela DARPA para satisfazer os requisitos de defesa para sistemas de informações robustos e rapidamente expansíveis foi o GloMo (*Global Mobile Information Systems*), que teve início em 1994.

A pesquisa em redes *ad hoc* permaneceu por muito tempo apenas na área militar. A partir da década de 90, aplicações comerciais ganharam força e em 1995, numa sessão de conferência da IETF (*Internet Engineering Task Force*) inicia-se uma pesquisa ativa em redes *ad hoc*. Em 1996, o trabalho do IETF evoluiu para *Mobile Ad Hoc Network* (MANET), que tinha o foco em redes de satélite militares, redes de computadores portáteis e adaptação dos protocolos de roteamento existentes para redes IP dinâmicas. Atualmente, o foco do grupo de trabalho MANET é padronizar as funcionalidades do protocolo IP adequando-o para aplicação de roteamento sem fio dentro de topologias estáticas ou dinâmicas.

2.1.2 Vantagens e Desvantagens

Existem várias vantagens e desvantagens ao se comparar redes *ad hoc* com redes sem fio com infraestrutura e redes fixas (cabramento estruturado).

Vantagens

- Instalação rápida: redes *ad hoc* podem ser estabelecidas dinamicamente em locais onde não exista uma infraestrutura previamente instalada;
- Tolerância à falhas: a permanente adaptação e reconfiguração das rotas em redes *ad hoc* permitem que perdas de conectividade entre os nós possam ser facilmente resolvidas desde que uma nova rota possa ser estabelecida;
- Conectividade: dois nós móveis podem se comunicar diretamente desde que cada nó encontre-se dentro do raio de alcance do outro;

Desvantagens

- Roteamento: uma topologia de rede dinâmica dificulta a construção de algoritmos de roteamento já que há a necessidade de atualização das rotas com grande frequência;
- Localização: como o endereço da máquina não tem relação com a localização do nó e não há informações geográficas que auxiliem na determinação do posicionamento do nó, a dificuldade de determinar a localização de um nó é uma desvantagem das redes *ad hoc*;

- Taxa de erro: perdas de pacotes não são causadas apenas por erros de transmissão mas também por mudanças na topologia de rede;
- Potência de Transmissão: a potência de transmissão é limitada pelos dispositivos ou nós;
- Alcance da transmissão: o alcance da transmissão é limitado pela potência do rádio. Isso torna necessário o roteamento das mensagens através de múltiplos saltos;
- Meio de Transmissão: o meio de transmissão sem fio apresenta desvanecimento e multipercurso;
- Consumo de energia: como os dispositivos sem fio normalmente operam com baterias, existem restrições quanto ao consumo de energia;
- Restrição de banda: redes sem fio tem uma capacidade menor do que redes fixas. Um efeito disso é que, devido a baixa capacidade, existe mais congestionamento.

2.1.3 Redes de Sensores

Redes de sensores podem ser vistas como um tipo especial de rede *ad hoc* já que as redes de sensores não tem infraestrutura, ou seja, um ponto de acesso e os nós são fixos. Uma rede de sensores pode ser definida como um rede de dispositivos, chamados de nós, que podem perceber o meio e comunicar a informação reunida no campo monitorado por enlaces sem fio [10]. Os dados são enviados, possivelmente por múltiplos saltos, para um sorvedouro que pode usar a informação localmente ou enviá-la para outras redes por meio de um *gateway*.

Os nós sensores podem ser estacionários ou dinâmicos, podem armazenar a informação de sua localização ou não e podem ser homogêneos (todos os nós tem a mesma potência de transmissão) ou heterogêneos. Além disso, uma rede de sensores pode ter um ou múltiplos sorvedouros [10]. A Figura 2.3 apresenta dois exemplos de redes de sensores: uma com um sorvedouro apenas e outro com mais sorvedouros.

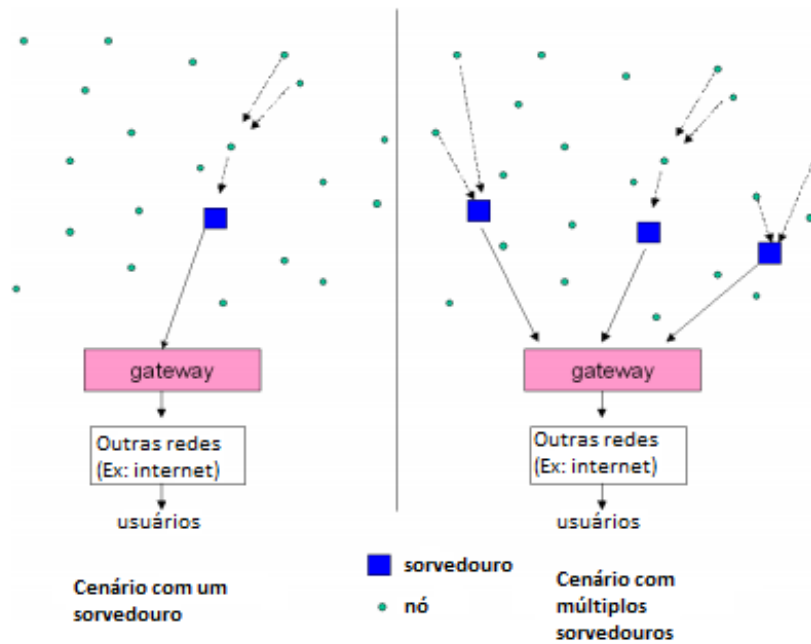


Figura 2.3: Exemplo de rede de sensores com um sorvedouro (Figura à esquerda) e múltiplos sorvedouros (Figura à direita).

2.2 Modelos de Propagação

O canal de comunicação rádio móvel impõem limitações fundamentais ao desempenho dos sistemas de comunicação sem fio. No percurso de transmissão entre o transmissor e o receptor pode haver tanto uma linha de visão desobstruída quanto múltiplos obstáculos. Diferente dos canais de sistemas com fios, que são estacionários e previsíveis, os canais de rádio são extremamente aleatórios e de difícil análise (modelagem). Modelo de propagação é uma descrição matemática baseado na teoria da propagação eletromagnética e/ou em medidas experimentais que pretende descrever a influência dos fatores que condicionam a propagação nas características do sinal recebido. Nas subseções a seguir, serão descritos dois modelos de propagação: Modelo de *Friis* para o Espaço Livre e Modelo de Dois Raios.

2.2.1 Modelo de Propagação no Espaço Livre

Em redes sem fio, o modelo de propagação no espaço livre é usado para prever a potência do sinal recebido quando o transmissor e o receptor possuem uma linha de visão desobstruída entre eles. Este modelo leva em consideração que a potência é atenuada na razão do quadrado da distância entre o transmissor e o receptor. De acordo com este modelo, a *Equação de Friis para o Espaço Livre* [11] para antenas não isotrópicas é dada a seguir:

$$P_r = P_t \left(\frac{\lambda}{4\pi d} \right)^n G_t G_r, \quad (2.1)$$

onde P_r é a potência recebida, P_t é a potência transmitida (em Watts ou milliwatts), λ é o comprimento de onda (em metros), d é a distância entre transmissor e receptor (em metros), n é o coeficiente de perda de percurso, G_t é o ganho da antena no transmissor e G_r é o ganho da antena no receptor (adimensional). Para antena isotrópica ideal, a equação de perda no espaço livre é dada por:

$$\frac{P_t}{P_r} = \frac{(4\pi d)^2}{\lambda^2} = \frac{(4\pi f d)^2}{c^2}, \quad (2.2)$$

onde c é a velocidade da luz ($3 * 10^8 m/s$) e f é a frequência (em Hertz).

2.2.2 Modelo de Propagação de Dois Raios

O modelo de propagação de dois raios [11] considera tanto o caminho direto quanto caminho refletido no chão entre o transmissor e o receptor. Este modelo prediz que a potência recebida é atenuada com a distância elevada a quarta potência. A equação para a potência recebida utilizando o modelo de dois raios é dada a seguir:

$$P_r = P_t \frac{h_t^2 h_r^2}{d^4} G_t G_r, \quad (2.3)$$

onde h_t e h_r são a altura das antenas transmissoras e receptoras, respectivamente, onde P_r é a potência recebida, P_t é a potência transmitida (em Watts ou miliwatts), d é a distância entre transmissor e receptor e G_t é o ganho da antena no transmissor e G_r é o ganho da antena no receptor (adimensional).

Para os dois modelos, é possível medir a atenuação da potência do sinal. O *decibel* é a medida da razão do nível de dois sinais, o ganho em decibéis é dado na equação a seguir:

$$G_{dB} = 10 \log_{10} \frac{P_{out}}{P_{in}}, \quad (2.4)$$

Algumas outras unidades relacionadas ao *decibel* são utilizadas para referenciar ganhos ou perdas de potência como o *dBW* (*decibel Watt*) e o *dBm* (*decibel milli-Watt*). As equações a seguir mostram as conversões de potência entre dBW e W e entre dBm e mW:

$$Potencia_{dBW} = 10 \log_{10} \left(\frac{Potencia_W}{1W} \right) \quad (2.5)$$

$$Potencia_W = 10^{\frac{Potencia_{dBW}}{10}} \quad (2.6)$$

$$Potencia_{dBm} = 10 \log_{10} \left(\frac{Potencia_{mW}}{1mW} \right) \quad (2.7)$$

$$Potencia_{mW} = 10^{\frac{Potencia_{dbmW}}{10}} \quad (2.8)$$

2.3 Roteamento em Redes Ad Hoc

De modo a facilitar a comunicação dentro de uma rede *ad hoc*, um protocolo de roteamento é utilizado para descobrir rotas entre nós antes da troca de dados. Os protocolos de roteamento em redes *ad hoc* podem ser divididos em três categorias: pró-ativos, reativos e híbridos.

2.3.1 Protocolos de Roteamento Pró-ativos

Nos protocolos de roteamento pró-ativos, ou dirigido por tabelas (do Inglês, *Table-Driven*), cada nó mantém uma ou mais tabelas para armazenar dados de roteamento e propagar as mudanças pela rede. Estes protocolos tentam manter rotas válidas para todas as comunicações entre nós durante todo o tempo, ou seja, antes que a rota seja necessária. Periodicamente, as mudanças nas tabelas de roteamento são trocadas de forma a manter sincronizadas todas as tabelas de roteamento. O principal problema dos protocolos pró-ativos é a propagação e a manutenção de informação de roteamento, sendo esta necessária ou não.

Alguns exemplos de protocolos pró-ativos são: *Dynamic Destination Sequenced Distance-Vector Routing Protocol* (DSDV) [12] e *Optimized Link State Routing Protocol* (OLSR) [13].

2.3.1.1 Dynamic Destination Sequenced Distance-Vector Routing Protocol (DSDV)

O protocolo DSDV [12] é baseado no algoritmo de *Bellman-Ford* [14], com a eliminação de ciclos. Cada nó armazena todos os possíveis destinos, e uma distância, que pode ser a quantidade de saltos, na tabela de roteamento. Cada entrada na tabela é marcada com um número de sequência determinado pelo nó de destino. Este número tem a função de distinguir rotas velhas de rotas novas, evitando a formação de loops.

A manutenção da tabela é feita através do envio de mensagens periódicas para cada nó, informando as alterações ocorridas nas suas tabelas devido a mudanças de topologia na rede. Para diminuir a quantidade de tráfego na rede para atualização de tabelas de roteamento, existem dois tipos de pacotes: *full dump* e *incremental*. Pacotes *full dump* contêm toda a informação de roteamento disponível e os pacotes *incremental* são utilizados apenas para transmitir as informações que foram modificadas desde o último pacote *full dump*.

A difusão de uma nova rota contém o endereço de destino, o número de saltos para alcançar o destino, o número de sequência da informação original sobre o destino e o número de sequência único da mensagem de difusão. Utiliza-se sempre a rota com o número de sequência mais recente e, no caso de duas atualizações terem o mesmo número de sequência, utiliza-se a rota com a menor métrica.

2.3.1.2 Optimized Link State Routing Protocol (OLSR)

O OLSR [13] é um protocolo ponto-a-ponto baseado no algoritmo estado de enlace (do Inglês, *link state*). Este algoritmo utiliza trocas periódicas de mensagens para manter a informação de

topologia da rede em cada nó. O OLSR é uma otimização do algoritmo de estado de enlace puro já que compacta o tamanho da informação enviada na mensagem e reduz o número de nós transmitindo durante cada atualização de rota através da utilização da estratégia de *multipoint relaying* (MPR). Para isso, durante cada atualização de topologia, cada nó seleciona um conjunto de nós vizinhos para retransmitir seus pacotes. Este conjunto de nós é chamado de *multipoint relays* daquele nó. Qualquer nó que está fora do conjunto lê o pacote, mas não o retransmite. O conjunto de nós MPRs é definido como o subconjunto de vizinhos a um salto de distância e que cobre todo o subconjunto de vizinhos a dois saltos de distância. A utilização da estratégia de MPRs garante rotas ótimas em termos de número de saltos, e estas rotas estão disponíveis imediatamente quando necessários. O OLSR é mais adequado para redes *ad hoc* grandes e densas [13].

2.3.2 Protocolos de Roteamento Reativos

Nos protocolos de roteamento reativos, as rotas só são criadas quando há uma necessidade do nó. Desta forma, é necessário um procedimento de descoberta de rotas dentro da rede. Assim que uma rota é estabelecida, esta é mantida por um procedimento de manutenção de rota ou até a rota não ser mais necessária. A principal vantagem deste tipo de protocolo de roteamento é que não são necessárias atualizações periódicas da tabela de roteamento já que para estes protocolos a informação de roteamento só está disponível quando necessária.

Alguns exemplos de protocolos pró-ativos são: *Dynamic Source Routing Protocol* (DSR) [15] e *Ad Hoc On Demand Distance Vector Routing Protocol* (AODV)[16].

2.3.2.1 Dynamic Source Routing Protocol (DSR)

O DSR [15] é um algoritmo de roteamento onde o nó de origem determina qual rota que o pacote deve seguir pela rede. Não são enviadas mensagens periódicas para a troca de informações de roteamento, garantindo economia de banda e de bateria. Cada nó mantém um *cache* com as rotas que aprende e cada rota armazenada possui um tempo de vida que quando é expirado resulta na eliminação da rota do *cache*.

O protocolo consiste em duas fases: descoberta de rota e manutenção da rota. No protocolo de descobrimento de rotas, o nó de origem envia por difusão um pacote de requisição de rota contendo o endereço de origem, destino da comunicação e o registro de rotas para o seu vizinho. Ao receber esse pacote, cada nó verifica em seu *cache* se existe uma rota para o nó de destino. Se a rota não existir, o nó insere seu endereço no registro de rota e propaga a mensagem por difusão para os seus vizinhos. Quando o pacote chega ao destino ou a algum nó que tenha uma rota para ele, é enviado um pacote de resposta utilizando um caminho que está armazenado no registro de rota. A manutenção das rotas acontece por meio de pacotes de erro de rota e pacotes de confirmação. Os pacotes de erro de rota são gerados quando um enlace encontra um problema de transmissão. Quando um pacote de erro de rota é recebido, o nó com o erro é removido do cache e todas as rotas contendo nó com erro são truncadas. Já os pacotes de confirmação são utilizados para verificar a correta operação das rotas.

Uma das vantagens do DSR é a utilização de roteamento pelo nó de origem, o que permite uma economia de banda já que as informações de rotas não precisam ser atualizadas constantemente. Algumas desvantagens são: a utilização de inundação na descoberta das rotas e problemas de escalabilidade.

2.3.2.2 Ad Hoc On Demand Distance Vector Routing Protocol (AODV)

O protocolo AODV [16] usa o conceito de vetor de distância e é uma melhoria do protocolo DSDV descrito anteriormente já que minimiza o número de mensagens de difusão necessárias para criar rotas sob demanda.

Quando um nó de origem deseja mandar uma mensagem para um nó de destino e não tem uma rota válida para este nó, o nó de origem inicia uma descoberta de caminho para o outro nó. O nó de origem faz uma difusão do pacote de requisição de rota para os vizinhos que propagam este pacote até que o nó de destino seja alcançado ou que seja localizada uma rota para o destino. O AODV utiliza número de sequência de destino para garantir que todas as rotas estão livres de ciclos e contém a informação de rota mais recente. Cada nó mantém seu próprio número de sequência bem como o difusão ID. O difusão ID é incrementado para cada requisição de rota que o nó inicia e, junto com o endereço IP, identifica unicamente uma requisição de rota. O nó de origem inclui na requisição de rota seu próprio número de sequência, o difusão ID e o número de sequência mais recente do nó de destino.

Durante o processo de propagação da requisição de rota, os nós intermediários guardam, em suas tabelas de roteamento, o endereço do nó vizinho de onde a primeira cópia da mensagem de difusão é recebida, estabelecendo o caminho reverso. Quando a mensagem de difusão chega ao nó de destino ou a um nó intermediário que tenha uma rota para o nó de destino, um pacote de resposta de rota é criado. Este pacote é roteado de volta ao nó de origem pelo caminho reverso. Enquanto o pacote é roteado de volta ao nó de origem, os nós intermediários adicionam a rota às suas tabelas de roteamento.

No protocolo DSR, cada nó da rede possui em sua tabela de roteamento todas as rotas “salto a salto”, da origem até o nó de destino, ou seja, armazena em seu cache todo o caminho, que pode ser mais de um, com todos nós pelo qual o pacote deverá percorrer. Por outro lado, o AODV utiliza, em sua tabela, no máximo uma entrada para cada destino, ou seja, não contém a rota inteira até um determinado nó, mas apenas o próximo salto para qual o nó deverá passar. Segundo [17], o DSR apresenta um desempenho melhor em redes com baixa mobilidade e baixo número de nós enquanto o AODV apresenta um melhor desempenho em redes com uma quantidade maior de nós e maior mobilidade.

2.3.3 Protocolos Híbridos

Para redes com uma grande quantidade de nós, existe uma dificuldade em implementar algoritmos do tipo pró-ativos e reativos. Uma solução possível é a organização da rede em grupos e a utilização de algoritmos de roteamento diferentes dentro e entre os grupos [1]. Se os grupos forem

bem divididos, apenas poucos dados sairão do grupo, e as alterações de topologias dentro do grupo podem ser passadas apenas para os membros do próprio grupo, sendo transparente para os nós que não pertencem ao grupo. Os nós não pertencentes ao grupo, necessitam saber apenas como chegar no grupo, geralmente através de um nó por onde passam os dados que saem ou entram no grupo.

2.4 Vazão Máxima Teórica em Redes Ad Hoc sem Fio

Nesta seção será feito o cálculo da Vazão Máxima Teórica (do Inglês, *Theoretical Maximum Throughput*(TMT)) para redes *ad hoc* sem fio para uma variedade de tecnologia e taxa de dados. As informações para o cálculo do TMT estão disponíveis nos padrões IEEE [18] e em [2].

2.4.1 Introdução ao CSMA/CA com RTS/CTS

O CSMA/CA é um protocolo de Controle de Acesso ao Meio (MAC, do Inglês, *Medium Access Control*) de redes *ad hoc* baseado em contenção que utiliza mecanismos de reserva de canal antes de uma transmissão e para tratar a ocorrência de colisões [19].

No protocolo CSMA/CA, o mecanismo fundamental de acesso ao meio é chamado de (DCF, do Inglês, *Distributed Coordination Function*). Este mecanismo é baseado em um esquema de acesso aleatório para evitar colisões. Sempre que existe um dado a ser transmitido por uma estação, esta monitora a atividade no canal. Se o canal estiver ocioso por um período maior que o tempo entre quadro distribuído (DIFS, do Inglês, *Distributed Interframe Space*), a estação envia um quadro de reserva (RTS, do Inglês, *Request to Send*) contendo a duração do pacote de dados endereçado a estação de destino. Se a estação recebe o RTS corretamente, esta espera um curto intervalo de tempo (SIFS, do Inglês, *Short Interframe Space*) e envia um quadro liberação para enviar (CTS, do Inglês, *Clear To Send* (CTS) indicando que a estação de origem pode iniciar a sua transmissão. A estação transmissora, após receber o CTS, espera um tempo igual ao SIFS para iniciar a sua transmissão.

Durante a transmissão do dado, a estação não consegue escutar o canal para detectar colisões. Assim, quando o dado chega corretamente ao destino, um ACK é transmitido pela estação de destino logo após um curto período de tempo (SIFS). Caso ocorre um *timeout* e o ACK não seja recebido, a estação transmissora identifica uma colisão ou perda e reagenda a transmissão de acordo com o algoritmo de recuo binário exponencial.

Nos quadros RTS/CTS existe a especificação do tamanho do *payload* que a estação deseja transmitir. Como em redes em fio o canal é de difusão, as estações que receberem o RTS/CTS podem usar essa informação para atualizar seu vetor de alocação da rede (NAV, do Inglês, *network allocation vector*). Desta forma, a estação só precisa escutar o canal quando houver um estouro do contador do tamanho de NAV. A Figura 2.4 mostra o diagrama de tempo para CSMA/CA com RTS/CTS.

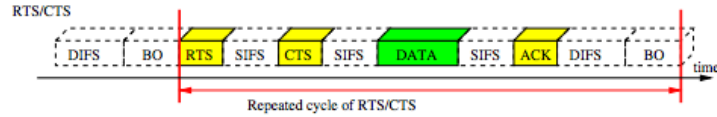


Figura 2.4: Diagrama de tempo para CSMA/CA com RTS/CTS [2].

Na subseção a seguir, será apresentado o cálculo da vazão máxima teórica de uma rede *ad hoc* sem fio utilizando o protocolo CSMA/CA com a utilização dos pacotes de RTS/CTS.

2.4.2 Cálculo da Vazão Máxima Teórica

O limite superior de vazão que uma rede *ad hoc* sem fio pode assumir é definido como a vazão máxima teórica. O TMT é definido assumindo os seguintes pontos:

- Taxa de erro de bit (do Inglês, *Bit Error Rate*) igual a zero;
- Não há perda devido a colisões;
- Não há perda de pacotes devido a estouro de *buffer* no nó receptor;
- O nó fonte sempre tem pacotes para enviar;
- A camada de enlace não usa fragmentação;

O cálculo do TMT é dado por:

$$TMT = \frac{\text{Tamanho do MSDU}}{\text{Atraso por MSDU}} \quad (2.9)$$

Service Data Unit (SDU) é definido como o tamanho da carga útil que uma camada entrega para a camada acima. O MSDU é definido como o *MAC Service Data Unit*, ou seja, a carga útil que a camada de enlace recebe da camada física. O atraso por MSDU é definido na equação abaixo:

$$\text{Atraso por MSDU} = (T_{DIFS} + T_{SIFS} + T_{BO} + T_{RTS} + T_{CTS} + T_{ACK} + T_{DATA}) \times 10^{-6} s. \quad (2.10)$$

Sendo T_{DIFS} , o tempo entre quadro distribuído, T_{SIFS} o intervalo entre o envio de um RTS e um CTS, T_{BO} o tempo de recuo, ou seja, espera antes da transmissão, T_{RTS} o tempo de transmissão de pacotes RTS, T_{CTS} o tempo de transmissão de pacotes CTS, T_{ACK} o tempo de transmissão de um pacote ACK e T_{DATA} o tempo de transmissão de um pacote de dados. A Tabela 2.1 foi obtida a partir de [18] e mostra os componentes de atraso para o CSMA/CA com a utilização dos protocolos de controle RTS/CTS para diferentes tipos de tecnologias da camada física.

Tabela 2.1: Componentes de atraso para o CSMA/CA com a utilização dos protocolos de controle RTS/CTS para diferentes tipos de tecnologias da camada física [2].

Esquema	Taxa	T_{DIFS}	T_{SIFS}	T_{BO}	T_{RTS}	T_{CTS}	T_{ACK}	T_{DATA} (MSDU em bytes)
FHSS	1Mbps	128ms	84ms	375ms	288ms	240ms	240ms	$128 + 33/328(34 + MSDU)/1$
FHSS	2Mbps	128ms	84ms	375ms	288ms	240ms	240ms	$128 + 33/328(34 + MSDU)/2$
DSSS	1Mbps	50ms	30ms	310ms	352ms	304ms	304ms	$192 + 8(34 + MSDU)/1$
DSSS	2Mbps	50ms	30ms	310ms	352ms	304ms	304ms	$192 + 8(34 + MSDU)/2$
HR - DSSS	5,5Mbps	50ms	30ms	310ms	352ms	304ms	304ms	$192 + 8(34 + MSDU)/5,5$
HR - DSSS	11Mbps	50ms	30ms	310ms	352ms	304ms	304ms	$192 + 8(34 + MSDU)/11$
OFDM	6Mbps	34ms	27ms	67,5ms	52ms	44ms	44ms	$20 + 4 \times (16 + 6 + 8 \times (34 + MSDU))/24$
OFDM	12Mbps	34ms	27ms	67,5ms	36ms	32ms	32ms	$20 + 4 \times (16 + 6 + 8 \times (34 + MSDU))/38$
OFDM	24Mbps	34ms	27ms	67,5ms	28ms	28ms	28ms	$20 + 4 \times (16 + 6 + 8 \times (34 + MSDU))/96$
OFDM	54Mbps	34ms	27ms	67,5ms	24ms	24ms	24ms	$20 + 4 \times (16 + 6 + 8 \times (34 + MSDU))/216$

O tempo de atraso por MSDU pode ser simplificado para uma função do tamanho MSDU em bytes:

$$\text{Atraso por MSDU}(x) = (ax + b) \times 10^{-6} s. \quad (2.11)$$

Sendo a todos os componentes de atraso que dependem do tamanho do MSDU e b todos os componentes de atraso que não dependem do tamanho do MSDU. Utilizando os dados da Tabela 2.1 podemos calcular os valores de a e b para o CSMA/CA com a utilização dos protocolos de controle RTS/CTS para diferentes tipos de tecnologias da camada física. Estes valores de a e b são apresentados na Tabela 2.2.

Tabela 2.2: Valores de a e b para o CSMA/CA com a utilização dos protocolos de controle RTS/CTS para diferentes tipos de tecnologias da camada física [2].

Esquema	Taxa	a	b
FHSS	1Mbps	8,250	1763,50
FHSS	2Mbps	4,125	1623,25
DSSS	1Mbps	8,000	1814,00
DSSS	2Mbps	4,000	1678,00
HR - DSSS	5,5Mbps	1,454	1591,45
HR - DSSS	11Mbps	0,727	1566,73
OFDM	6Mbps	1,333	337,50
OFDM	12Mbps	0,666	273,00
OFDM	24Mbps	0,333	244,75
OFDM	54Mbps	0,148	225,94

Utilizando os valores mais comuns de MSDU: 512 bytes, 1024 bytes e 1480 bytes, podemos encontrar o atraso fim-a-fim de um nó fonte para um nó de destino que estão dentro do alcance um do outro. Este atraso fim-a-fim entre o nós corresponde ao atraso de um salto já que a conexão

entre o nó fonte e o nó de destino é direta. A Tabela 2.3 apresenta o atraso fim-a-fim quando se transmite um MSDU utilizando CSMA/CA com RTS/CTS. O cálculo da Tabela 2.3 é executado utilizando os dados da Tabela 2.2 e a Equação 2.11.

Tabela 2.3: Atraso fim-a-fim para um salto utilizando CSMA/CA com RTS/CTS

Esquema	Taxa	TMT		
		MSDU=512 bytes	MSDU=1024 bytes	MSDU=1480 bytes
FHSS	1Mbps	5,99ms	10,21ms	13,97ms
FHSS	2Mbps	3,74ms	5,85ms	7,73ms
DSSS	1Mbps	5,91ms	10,01ms	13,65ms
DSSS	2Mbps	3,73ms	5,77ms	7,60ms
HR - DSSS	5,5Mbps	2,34ms	3,08ms	3,74ms
HR - DSSS	11Mbps	1,94ms	2,31ms	2,64ms
OFDM	6Mbps	1,02ms	1,70ms	2,31ms
OFDM	12Mbps	0,61ms	0,96ms	1,26ms
OFDM	24Mbps	0,42ms	0,59ms	0,74ms
OFDM	54Mbps	0,30ms	0,38ms	0,45ms

O TMT pode ser obtido dividindo o número de *bytes* no MSDU pelo atraso total (Equação 2.11). Assim, o TMT é dado por:

$$TMT(x) = \frac{8x}{ax + b} \times 10^6 \text{bps}. \quad (2.12)$$

Utilizando a Equação 2.12 e Tabela 2.2 podemos calcular a vazão máxima teórica de redes *ad hoc* sem fio que utilizam o mecanismo CSMA/CA com RTS/CTS para diversas tecnologias da camada física. Os resultados da TMT são apresentados na Tabela 2.4.

Tabela 2.4: Vazão Máxima Teórica utilizando CSMA/CA com RTS/CTS

Esquema	Taxa	TMT		
		MSDU=512 bytes	MSDU=1024 bytes	MSDU=1480 bytes
FHSS	1Mbps	684,41kbps	802,22kbps	847,31kbps
FHSS	2Mbps	1096,58kbps	1401,00kbps	1532,04kbps
DSSS	1Mbps	693,06kbps	818,71kbps	867,14kbps
DSSS	2Mbps	1099,30kbps	1418,87kbps	1558,30kbps
HR - DSSS	5,5Mbps	1753,29kbps	2658,95kbps	3162,23kbps
HR - DSSS	11Mbps	2112,33kbps	3544,09kbps	4479,60kbps
OFDM	6Mbps	4015,70kbps	4811,77kbps	5124,79kbps
OFDM	12Mbps	6667,37kbps	8572,00kbps	9399,28kbps
OFDM	24Mbps	9860,02kbps	13977,62kbps	16041,66kbps
OFDM	54Mbps	13572,27kbps	21692,23kbps	26594,67kbps

Pela Tabela 2.4, podemos observar uma diferença razoável entre a taxa e a vazão máxima teórica. Isso ocorre devido os componentes de atraso *a* e *b*, em especial o componente *a* que

depende do tamanho do MSDU. Com o aumento do MSDU, a TMT se aproxima da taxa já que o componente de atraso b exerce, proporcionalmente, menos influência no resultado do TMT.

2.5 Discussão

Este capítulo fez uma revisão de conceitos básicos que são necessários para a compreensão deste trabalho. Primeiramente, uma introdução à redes *ad hoc* sem fio é feita. Em seguida, apresenta-se dois modelos de propagação: espaço livre e modelo de dois raios e faz-se uma breve revisão do estado da arte de protocolos de roteamento em redes *ad hoc*. Por fim, o cálculo da vazão máxima teórica em redes *ad hoc* com a utilização de CSMA/CA com RTS/CTS é feito. Com base nos conceitos discutidos neste capítulos, será apresentado o modelo de comunicação cooperativa, as técnicas de comunicação cooperativa e algoritmos de controle de topologia em redes *ad hoc* no capítulo a seguir.

Capítulo 3

Controle de Topologia em Redes Ad Hoc Cooperativas

Comunicação Cooperativa é uma técnica da camada física que permite combinar sinais parciais para obter a mensagem original. Este capítulo apresenta os conceitos de comunicação cooperativa e controle de topologia necessários para o entendimento deste trabalho. A Seção 3.1 apresenta o modelo de comunicação cooperativa e o modelo de rede cooperativa. A Seção 3.2 apresenta o conceito de MIMO virtual e descreve as principais técnicas de retransmissão cooperativa. A Seção 3.3 apresenta uma revisão bibliográfica do controle de topologia em redes *ad hoc*. A Seção 3.4 faz uma discussão dos pontos importantes deste capítulo.

3.1 Comunicação Cooperativa

Nesta seção são descritos os algoritmos e definições do modelo de comunicação cooperativa e do modelo de rede cooperativa. Os modelos definidos nesta seção são baseados em [3], [4], [5].

3.1.1 Modelo de Comunicação Cooperativa

Sabendo que cada nó v_i pode ajustar sua potência de transmissão P_i dentro do intervalo $[0, P_{MAX}]$ e, quando $P_i = 0$, o rádio está desligado e, para $P_i = P_{MAX}$, o rádio opera em sua potência máxima, podemos determinar a potência de transmissão necessária para a origem, em conjunto com os nós ajudantes, alcançar um nó de destino de forma semelhante à comunicação direta. Assim, no modelo tradicional de comunicação cooperativa, um nó transmissor v_i pode se comunicar com um nó receptor v_j diretamente apenas quando a potência de transmissão do nó de origem satisfaz a Equação 3.1.

$$P_i(d_{i,j})^{-\alpha} \geq \tau \quad (0 \leq P_i \leq P_{MAX}), \quad (3.1)$$

onde: α é o expoente de perda de caminho (PLE, do Inglês, *Path Loss Exponent*), que normal-

mente fica entre 2 e 4 e simboliza a taxa de desvanecimento do sinal com a distância d_{ij} ; d_{ij} é a distância Euclidiana entre os nós v_i e v_j ; e τ é a sensibilidade do receptor para iniciar a detecção de um pacote, isto é, o limiar de detecção do receptor para que este possa corretamente decodificar o sinal e obter a mensagem original. Este trabalho foca num regime limitado de interferência, onde o ruído é pequeno se comparado com o sinal. Além disso, só é considerado o expoente de perda de caminho, o efeito do desvanecimento do sinal em pequena escala não é considerado neste modelo.

A Comunicação Cooperativa (CC) [20] se beneficia de algumas vantagens dos sistemas MIMO, como o uso da diversidade espacial. Assim, a CC tira vantagem do projeto da camada física para combinar sinais e obter a informação completa. Desta forma, ocorrerá uma comunicação completa entre o nó v_i e o nó v_j utilizando CC se v_i transmite o sinal simultaneamente com um conjunto de nós auxiliares H_{ij} e a soma de suas potências de transmissão satisfazem a Equação 3.2.

$$\sum_{v_k \in v_i \cup H_{i,j}} P_k (d_{k,j})^{-\alpha} \geq \tau \quad (0 \leq P_k \leq P_{MAX}). \quad (3.2)$$

As Figuras 3.1a, 3.1b, 3.2a e 3.2b exemplificam um cenário de comunicação cooperativa. A Figura 3.1a mostra o cenário onde o nó v_1 tem dois nós dentro do raio de transmissão e um fora. O nó v_1 pode enviar dados para o nó v_4 , que está fora do seu raio de transmissão, utilizando CC. O nó v_1 pode selecionar os nós v_2 e v_3 como auxiliares para transmitir para v_4 , isto é H_{ij} . Após selecionar esses nós como ajudantes, o nó v_1 transmite seu dado para v_2 e v_3 num primeiro momento, como mostrado na Figura 3.1b e num segundo momento, os três nós juntos transmitem seu dado para v_4 , o que é mostrado na Figura 3.2a. Se a potência do sinal recebido for maior que o τ em v_4 , este nó seria capaz de decodificar o sinal recebido e recuperar o dado oriundo de v_1 , assim como ilustrado pela Figura 3.2b.

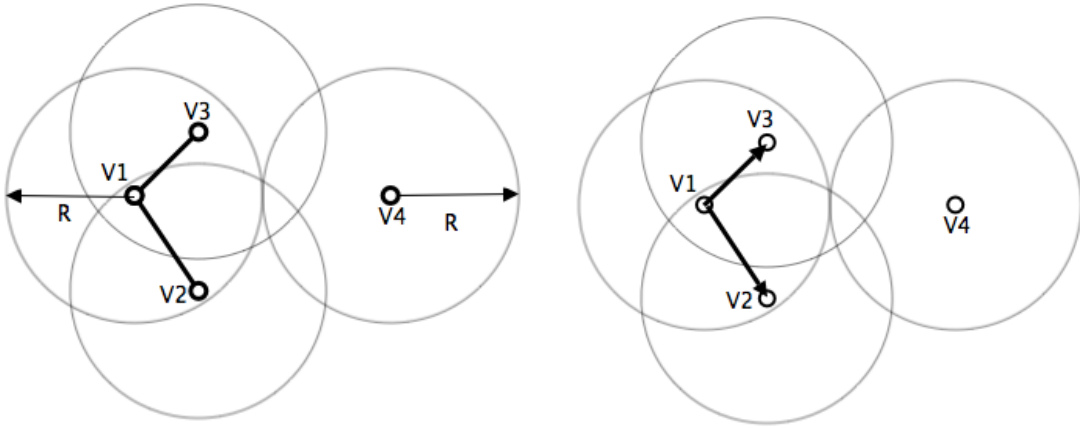


Figura 3.1: (a) Cenário onde existem 3 nós próximos v_1, v_2 e v_3 e um distante v_4 e (b) Primeiro momento da CC onde v_1 transmite o dado para v_2 e v_3 .

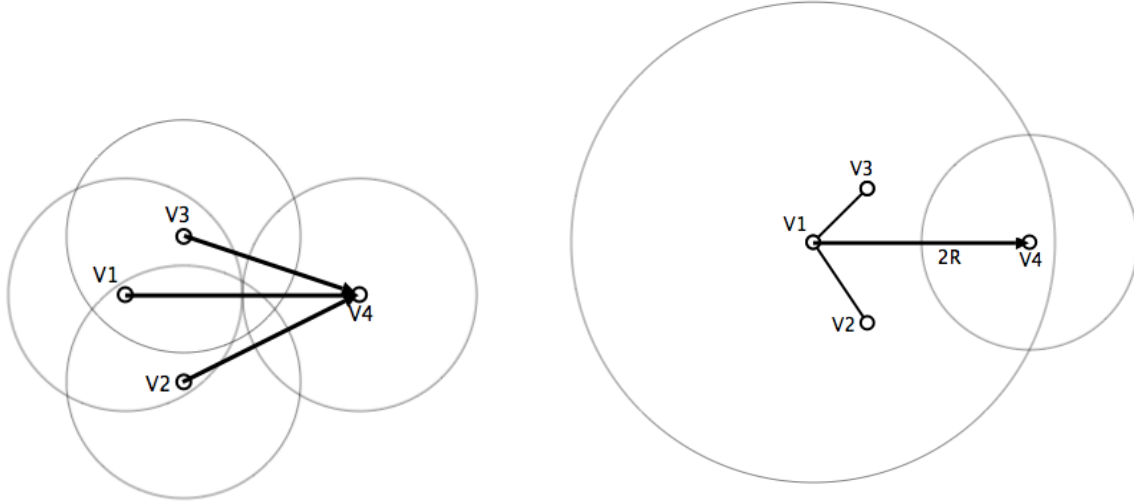


Figura 3.2: (a) Segundo momento da CC cooperativa onde v_1 , v_2 e v_3 transmitem o dado para v_4 e (b) Aumento do raio de cobertura de v_1 com a utilização dos nós ajudantes v_2 e v_3 para criar um enlace cooperativo entre v_1 e v_4 .

3.1.2 Modelo de Rede Cooperativa

Numa rede sem fio *ad hoc* com n nós que utiliza 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 comunicação cooperativa. O conjunto $N(v_i)$ 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 3.1.

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 auxiliares): $H_{i,j}$ representa o conjunto de nós auxiliares de v_i na comunicação cooperativa para v_j . Assume-se que todos os nós auxiliares 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 auxiliares de v_i .

Definição 3 (Aresta CC): Uma aresta CC $\widetilde{v_i v_j}$ é uma aresta pertencente a E_P se um nó v_i pode transmitir dados para o nó v_j cooperativamente, utilizando um conjunto de nós auxiliares $H_{i,j}$. Utiliza-se um traço horizontal ondulado para identificar uma aresta CC.

Definição 4 (Aresta auxiliares): Uma aresta auxiliar é uma aresta entre v_i e um dos seus nós ajudantes em $H_{i,j}$. Por exemplo, na Figura 3.2b, o nó v_1 utiliza os nós v_2 e v_3 como nós auxiliares

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 auxiliares.

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 direta 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 (Custo de uma aresta direta): O custo de uma aresta $\overline{v_i v_j}$ é definido como:

$$w(\overline{v_i v_j}) = \tau d_{i,j}^\alpha. \quad (3.3)$$

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

$$w(\widetilde{v_i v_j}) = w_d(H_{i,j}) + (|H_{i,j}| + 1)w_{CC}(H_{i,j}), \quad (3.4)$$

onde:

- $|H_{i,j}|$: é o número de elementos do conjunto de nós auxiliares $H_{i,j}$;
- $w_d(H_{i,j}) = \left(\frac{\tau}{\max_{v_k \in H_{i,j}} (d_{i,k})^{-\alpha}} \right)$: é a potência mínima necessária para o nó v_i se comunicar diretamente com seu nó auxiliar 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 3.1 e 3.2, a seguinte relação deve ser verdadeira para existir uma aresta CC :

$$\max(w_d(H_{i,j}), w_{CC}(H_{i,j})) \leq P_{MAX}. \quad (3.5)$$

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 auxiliares 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 considerada 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:

$$\pi_{G_P}(v_i, v_j) = w(v_i v_{i_1}) + w(v_{i_1} v_{i_2}) + \dots + w(v_{i_{k-1}} v_{i_k}) + w(v_{i_k} v_j). \quad (3.6)$$

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

$$\min(\pi_{G_P}(v_i, v_j)). \quad (3.7)$$

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:

$$\rho_{G'_P}^{G'_P}(v_i, v_j) = \frac{\min(\pi_{G'_P}(v_i, v_j))}{\min(\pi_{G_P}(v_i, v_j))}. \quad (3.8)$$

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

$$\rho_{G_P}^{G'_P} = \max_{v_i, v_j \in V} \rho_{G'_P}^{G'_P}(v_i, v_j). \quad (3.9)$$

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 3.3 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:

$$\rho_{G_P}^{G'_P} \leq t. \quad (3.10)$$

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:

$$\max_{v_i \in V} \mu(v_i, v_o) \leq t, \quad (3.11)$$

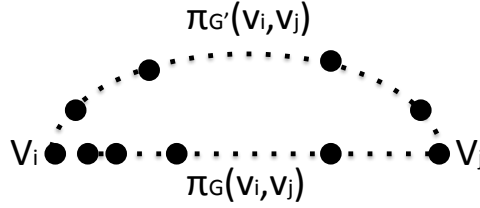


Figura 3.3: ESF - *Energy Stretch Factor* de um par de nós $v_i, v_j \in V$ em $G'_P = (V, E'_P)$ com respeito a $G_P = (V, E_P)$, $E'_P \subseteq E_P$, G'_P e G_P conexos. $\rho_{G'_P}^{G'_P}(v_i, v_j) = \frac{\min(\pi_{G'_P}(v_i, v_j))}{\min(\pi_{G_P}(v_i, v_j))}$ representa a diferença entre o custo 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 .

onde:

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

3.2 Técnicas de Retransmissão Cooperativa

A Comunicação Cooperativa permite que, dispositivos com uma única antena, tenham os benefícios dos sistemas MIMO (do Inglês, *Multiple-Input, Multiple-Output*) ao compartilhar antenas [20]. Em um sistema MIMO, os dispositivos utilizam, simultaneamente, tanto na origem quanto no destino, um conjunto de antenas para transmitir e receber dados. Segundo *Sadek et al.* [20], a utilização de sistemas MIMO combate o desvanecimento do sinal, aumenta o desempenho e a confiabilidade do sistema. Devido a essas propriedades, a tecnologia MIMO foi incorporada em diversos padrões de comunicação sem fio, como IEEE 802.11n, 4G, WiMAX e HSPA+ [21]. Uma das propriedades de um sistema MIMO é o ganho de diversidade que pode ser obtido quando existem diferentes canais entre o receptor e o transmissor. Isto significa que quando um canal está bloqueado ou tendo um alto desvanecimento, outros canais que estão bons o suficiente para a transmissão do sinal podem ser utilizados. No caso da comunicação cooperativa, a diversidade espacial pode explorar melhor as características de um sistema MIMO do que a diversidade no tempo ou frequência [22].

Em redes de múltiplos saltos tradicionais, nós intermediários cooperam com um nó fonte retransmitindo o dado para o nó de destino, processo que é realizado pela camada de rede. Dessa maneira, o nó de destino recebe apenas uma cópia do dado enviado. Entretanto, CC é uma técnica da camada física. Em CC, um conjunto de dispositivos com uma única antena enviam a mesma informação em um processo de duas fases: transmissão do dado original para um conjunto de nós auxiliares e transmissão dos dados pela origem e pelos auxiliares de forma sincronizada para o destino. O nó de destino pode combinar as várias cópias do sinal recebido tanto do nó fonte quanto dos nós auxiliares, decodificar e obter o sinal original. O resultado é um sistema MIMO virtual que é gerado pela transmissão de cópias independentes de um mesmo dado através de vários nós

em uma rede sem fio. O sistema MIMO virtual gerado acaba por combater o desvanecimento do sinal. Os três métodos principais de retransmissão por parte dos nós auxiliares em CC segundo [23], [24], [25] são descritos a seguir.

3.2.1 Amplificar e Encaminhar

No método Amplificar e Encaminhar (do Inglês, *Amplify and Forward*), o nó fonte envia o dado para os nós auxiliares que, ao receber o sinal ruidoso, amplificam e retransmitem o sinal sem nenhum processamento adicional. Ao chegar ao destino final, este combina as informações recebidas. Este método possui a desvantagem de propagar erros e a vantagem de ser uma solução com baixa complexidade.

Um exemplo de aplicação da técnica de retransmissão do tipo “amplifica e encaminha” é *Laneman et al.* [26] que propões diferentes protocolos CC e avalia sua performance em termos de comportamento mediante a falha. *Chatnagar et al.* [27] propõem uma dupla modulação diferencial em um sistema CC com um protocolo “amplifica e encaminha” para evitar o problema de compensação de portadora em canais de desvanecimento *m-Nakagami* .

3.2.2 Decodificar e Encaminhar

No método Decodificar e Encaminhar (do Inglês, *Decode and Forward*), o nó fonte envia o sinal para os nós auxiliares que recebem um sinal ruidoso. Estes decodificam o sinal e o retransmitem, diminuindo o ruído no receptor em relação ao método “amplificar e encaminhar”, entretanto, aumenta-se a complexidade e o custo operacional.

Como exemplo de utilização da técnica de “decodificar e encaminhar” temos *Bletsas et al.* [28] que apresenta uma estratégia oportunista de retransmissão que utiliza tanto as técnicas de “decodificar e encaminhar” quanto “amplificar e encaminhar”.

3.2.3 Compress and Forward

No método Comprimir e Encaminhar (do Inglês, *Compress and Forward*) , o nó fonte envia o sinal para os nós auxiliares que recebem um sinal ruidoso. Estes comprimem o sinal e o retransmitem para o nó de destino. O método de “comprimir e encaminhar” tende a ter uma performance melhor que o o método “decodificar e encaminhar” quando os nós auxiliares estão próximos do nó de destino [29].

Um exemplo da utilização do método “comprimir e encaminhar” é mostrado em *Chia-Hung Yeh et al.* [30] que propõe uma nova estratégia para este método que utiliza amostragem compressiva para diminuir o tamanho da mensagem retransmitida quando o nó estima que a decodificação ocorrerá com sucesso.

3.3 Controle de Topologia e Redes Ad Hoc Cooperativas

O controle de topologia permite a manutenção de diversas características de rede, enquanto reduz a interferência e o consumo de energia através do ajuste de potência de transmissão dos nós. *Santi et. al* [31] define controle de topologia como:

Definição 12 (*Controle de Topologia*): *O controle de topologia consiste em controlar os enlaces de comunicação, através do raio de comunicação, entre os nós com o objetivo de manter alguma propriedade global enquanto reduz o consumo de energia.*

Assim, o controle de topologia pode ser utilizado para otimizar o consumo de energia da rede, reduzir o custo de roteamento e o número de mensagens de controle, melhorar a vazão, aumentar a conectividade ou atender a requisitos de QoS. De acordo com [23], os protocolos de controle de topologia podem ser classificados em: (i) centralizados; ou (ii) distribuídos. Os protocolos centralizados assumem que informações globais, como informação de topologia, informação de estado de memória, etc, estão disponíveis. Entretanto, mesmo quando a informação global está disponível, encontrar topologias fortemente conexas com o mínimo de energia é considerado um problema NP-completo [32]. Já os protocolos distribuídos, consideram a informação de vizinhos de até k saltos, onde k é tipicamente um ou dois. Nas subseções a seguir, temos algumas descrições das propostas da literatura que consideram o controle de topologia em redes *ad hoc* com ou sem CC.

3.3.1 MST

O algoritmos de Prim [33] e Kruskal [34] podem ser utilizados para obter a árvore geradora mínima (do Inglês, *Minimum Spanning Tree*). O algoritmo de Kruskal considera, inicialmente, que cada nó está em um componente separadamente e, sucessivamente, adiciona a aresta de menor peso que junta dois componentes. O algoritmo é finalizado quando existir um único componente e as arestas adicionadas correspondem a MST. No algoritmo de Prim, dado um nó inicial qualquer, são adicionadas as arestas de menor peso que ligam um nó inicial a um nó qualquer não visitado. O algoritmo encerra quando todos os nós forem visitados e as arestas adicionadas correspondem a MST. O MST não é um algoritmo de controle de topologia mas pode ser usado para tal finalidade.

3.3.2 LMST

O *Localized Minimum Spanning Tree* (LMST) proposto em [7] consiste na união de árvores mínimas calculadas em cada nó de forma independente usando o algoritmo de Prim [33]. No LMST cada nó cria sua própria árvore utilizando informações de até um salto. A topologia final é construída de tal forma que o grau máximo do grafo é igual a 6.

3.3.3 Dijkstra

O algoritmo Dijkstra [35] é utilizado para resolver o problema do caminho de menor custo quando todas as arestas tem peso não negativo. Este algoritmo é iterativo, guloso e é utilizado para obter o menor caminho de um nó particular para todos os outros nós do grafo. O algoritmo original apresentado em [35] pode ser facilmente modificado para funcionar em grafos direcionais, para se obter a distância mínima entre todos os nós do grafo (do Inglês, *all pair shortest path*) e para se obter o menor caminho, não só a distância, entre todos os nós. No contexto de controle de topologia, o algoritmo Dijkstra pode ser utilizado para encontrar o menor custo entre dois nós num grafo e as arestas que não pertencem ao menor caminho são removidas. Como MST, o algoritmo Dijkstra não é um algoritmo de controle de topologia mas pode ser usado para tal.

3.3.4 DTCC/ITCC

Cardei et al. [23] fizeram um estudo do uso de controle de topologia com comunicação cooperativa com o objetivo de obter topologias fortemente conexas com o mínimo consumo de energia. Os autores demonstram que o problema é NP-completo e propõem dois algoritmos localizados e distribuídos. Ambos os algoritmos recebem como entrada o resultado de algoritmos tradicionais de controle de topologia, ou seja, sem a utilização de comunicação cooperativa. O primeiro algoritmo usa um processo de informação distribuída onde cada nó utiliza a informação até dois saltos. O segundo algoritmo atribui potência de transmissão aos nós iterativamente, usando informação de um salto.

3.3.5 CoopBridges

Yu et al. [3] propõem um esquema de controle de topologia centralizado que usa comunicação cooperativa para reduzir a potência de transmissão dos nós bem como aumentar a conectividade da rede. Os autores propõem dois algoritmos para selecionar os nós auxiliares de maior eficiência energética para auxiliar o nó fonte a se comunicar com o nó de destino: um método ótimo e uma heurística gulosa. Adicionalmente, os autores ainda propõem uma versão distribuída do esquema de controle de topologia proposto.

3.3.6 Greedy(Add/Del)Link

Zhou et al. [4] consideram o problema de selecionar caminhos eficientes quando enlaces cooperativos são utilizados. Os autores propõem dois algoritmos de controle de topologia que constroem *energy spanners* cooperativos em que a eficiência energética de cada caminho individual é garantida mesmo com apenas a informação local.

3.3.7 CoopSink

Neves et al. [5] propõem uma técnica de controle de topologia centralizada para redes de sensores 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. A técnica *CoopSink* apresentada obtém uma conectividade 6,8 vezes maior se comparada com outras técnicas da literatura e com rotas 2,34 vezes mais eficientes quando se considera o *energy stretch factor* para o sorvedouro.

3.4 Discussão

Neste capítulo foram apresentados tópicos importantes para o entendimento deste trabalho. Inicialmente, foi apresentado o modelo de comunicação e de rede cooperativa. Num segundo momento, foi discutido as técnicas de retransmissão cooperativa e apresentados alguns exemplos presentes na literatura. Finalmente, foram introduzidos algoritmos de controle de topologia em redes *ad hoc*. No Capítulo 4 serão apresentadas duas novas técnicas de controle de topologia em redes *ad hoc* cooperativas e estas técnicas foram baseadas nos algoritmos *CoopSink* e *CoopBridges* apresentados neste capítulo.

Capítulo 4

Proposta do Cooperative Sink Communication

A técnica de Comunicação Cooperativa (CC) tem sido combinada com sucesso com diversas 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 redes sem fio [3], [4]. Este capítulo propõem duas novas técnicas, chamadas *CSC* (abreviação para *Cooperative Sink Communication*) e *CSC-HS* (abreviação para *Cooperative Sink Communication - Helper Set*) que fazem o controle de topologia em redes *ad hoc* cooperativas. O *CSC* utiliza a 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* e o *CSC* busca minimizar a quantidade de saltos até o sorvedouro. O *CSC-HS* também busca aumentar a conectividade da topologia em relação ao nó sorvedouro e, além disso, diminui a quantidade de transmissões de dados necessárias para alcançar este nó de forma a manter as rotas para o sorvedouro o mais eficiente possíveis. A Seção 4.1 formaliza o problema principal tratado neste trabalho. A Seção 4.2 descreve as duas técnicas propostas. A Seção 4.3 apresenta os resultados de simulação que comparam o *CSC* e o *CSC-HS* com outras técnicas da literatura. A Seção 4.4 apresenta uma discussão do que foi tratado neste capítulo.

4.1 Formulação do Problema

Numa rede de sensores, onde exista um nó sorvedouro, há a necessidade de manter a conectividade com este nó. A técnica de comunicação cooperativa pode ser utilizada para garantir que nós distantes, antes sem conectividade com o sorvedouro, tornem-se capazes de se comunicar. As técnicas propostas neste capítulo não promovem apenas o aumento da conectividade com o sorvedouro, mas, adicionalmente, acham o menor caminho em termos de saltos ou de transmissões.

Enquanto o *CoopSink* [5] possui o seu foco em aumentar a conectividade e, ao mesmo tempo, obter rotas mais eficientes energeticamente para o sorvedouro, este não analisa o impacto no atraso de fim-a-fim, ou seja, o *CoopSink* não minimiza a quantidade de saltos para alcançar o sorvedouro.

Utilizar o controle de topologia em redes *ad hoc* cooperativas para diminuir o atraso fim-a-fim e aumentar a conectividade com o sorvedouro é uma tarefa importante pois permite que sensores distantes possam se comunicar com o sorvedouro e enviar dados críticos mais rapidamente. Assim, este capítulo descreve duas novas técnicas: *CSC* que utiliza controle de topologia em redes de sensores para aumentar a conectividade com um nó sorvedouro, além de diminuir a quantidade de saltos. A técnica *CSC - HS* também promove o aumento da conectividade com um nó sorvedouro e, além disso, diminuir a quantidade de transmissões necessárias para alcançar o sorvedouro tornando as rotas mais eficientes. A principal diferença entre o *CSC* e o *CSC-HS* é que o *CSC* diminui a quantidade de saltos para o sorvedouro enquanto o *CSC-HS* diminui a quantidade de transmissões para alcançar o nó sorvedouro.

Considere uma rede de sensores *ad hoc*, onde os sensores estão espalhados em uma área fixa e existe um único nó sorvedouro no centro 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 [10]. As duas propostas desse capítulo consistem em:

- 1 Utilizar comunicação cooperativa na topologia formada a partir dos nós dessa rede para aumentar a conectividade dos nós com o sorvedouro e diminuir a quantidade de saltos necessários para alcançar o sorvedouro;
- 2 Utilizar comunicação cooperativa na topologia formada a partir dos nós dessa rede para aumentar a conectividade dos nós com o sorvedouro e diminuir a quantidade de transmissões necessárias para alcançar o sorvedouro de forma a manter as rotas o mais eficientes possíveis.

Utilizando a notação definida, dada 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 tenha a menor quantidade de saltos necessários para todos os nós alcançarem o sorvedouro. E propor uma outra técnica que utilize CC para criar um grafo G'_P de tal forma que G'_P tenha a menor quantidade de transmissões necessárias para todos os nós alcançarem o sorvedouro. Na seção que segue, descreve-se uma heurística gulosa para a escolha de nós auxiliares e propõem-se duas técnicas para resolver os problemas descritos anteriormente.

4.2 Descrição da Proposta

Na Seção 4.2.2, descreve-se o *CSC*, uma técnica de controle de topologia centralizada, ou seja, o nós tem informações de toda a topologia, para redes de sensores *ad hoc* cooperativas que busca aumentar a conectividade dos nós de uma rede em relação ao sorvedouro. E, adicionalmente, diminuir a quantidade de média de saltos necessários para um nó alcançar o sorvedouro diminuindo, assim, o atraso fim-a-fim.

Na Seção 4.2.3 descreve-se o *CSC-HS* que é uma melhoria do *CSC*. O *CSC-HS*, além de aumentar a conectividade dos nós de uma rede em relação ao sorvedouro, promove a redução da

quantidade média de transmissões necessárias para um nó alcançar o sorvedouro tornando as rotas mais eficientes energeticamente.

Em um primeiro momento, descreve-se um algoritmo guloso que pode ser utilizado para seleção de nós auxiliares na criação de arestas cooperativas e, em seguida, as técnicas *CSC* e *CSC-HS* são descritas.

4.2.1 Algoritmo: *Greedy Helper Set Selection*

O Algoritmo 4.1 descreve uma heurística gulosa, proposta por [3], para selecionar os nós auxiliares mais eficientes em energia em uma comunicação cooperativa e serve para grafos tanto direcionais quanto bi-direcionais. O algoritmo recebe como entrada: o nó transmissor v_i ; o conjunto $N(v_i)$ de nós vizinhos de v_i ; e nó destino v_j , a distância d_{ij} entre os nós v_i e v_j e o coeficiente de desvanecimento α . Retorna um conjunto de nós auxiliares $H_{i,j}$ 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 e a função de ganho utilizada.

4.2.1.1 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 auxiliares $H_{i,j}$ 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 3.2. 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 auxiliares.

4.2.1.2 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 auxiliar. 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 auxiliar;
- $c_k = \frac{\tau}{(d_{i,k})^{-\alpha}}$: Custo para o nó v_i se comunicar com o nó auxiliar 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 auxiliar, 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.

Algoritmo 4.1 : *GreedyHelperSetSelection*($v_i, N(v_i), v_j, d_{ij}, \alpha$)

Entrada : $v_i, N(v_i), v_j, d_{ij}, \alpha$ **Saída** : $H_{i,j}$;

```
1: # (Passo 1)
2: conjunto:  $A \leftarrow N(v_i)$ ;
3: vetor:  $B \leftarrow [], C \leftarrow []$ ;
4: Enquanto ( $v_k \leftarrow \text{head}(A) \neq \emptyset$ ) 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:    $\text{remove}(A, v_k)$ ;
9: Fim do Enquanto
10:  $B \leftarrow \text{ordenarOrdemDecrescente}(B)$ ;
11:  $C \leftarrow \text{indexarTermos}(B)$ 
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 do Se
19:    $H_{i,j} \leftarrow H_{i,j} \cup A[k]$ ;
20:    $\Omega \leftarrow \Omega \cup H_{i,j}$ 
21: Fim do Enquanto
22: # (Passo 3)
23: Enquanto ( $k \leq |N(v_i)|$ ) Faça
24:   Se ( $k = |N(v_i)|$ ) or
     ( $\text{PesoCC}(\widetilde{v_i v_j}, \Omega) < \text{PesoCC}(\widetilde{v_i v_j}, \Omega \cup C[k + 1])$ ) 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 do Se
31: Fim do Enquanto
```

4.2.2 CSC: Descrição da Técnica

Esta seção descreve os passos do *CSC*, a proposta desse trabalho para aumentar a conectividade com o nó sorvedouro e diminuir o atraso de fim-a-fim. Assume-se uma unidade central de computação que tem informações globalizadas da rede. O *CSC* consiste na execução em sequência de 4 passos:

- **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 do grafo de topologia criado no passo anterior, fazer controle de topologia para cortar arestas de maneira que se diminua a quantidade de saltos necessários para alcançar o sorvedouro;

- **Passo 4:** Ajustar a potência dos nós para que o consumo de energia seja reduzido.

Observa-se que os passos 1, 2 e 4 são idênticos aos passos do *CoopSink* [5].

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

O Algoritmo 4.2 descreve o primeiro passo da técnica *CSC*. 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 $CSCStep1(V, P_{MAX})$ para se referir ao Algoritmo 4.2.

As Figuras 4.1 e 4.2 exemplificam o que acontece nesse passo. Na Figura 4.1 temos uma área de 500×500 m com $n = 70$ nós e $\overline{E_P} = \emptyset$. A Figura 4.2 ilustra o grafo resultante quando os nós criam arestas de acordo com sua potência máxima de transmissão, no exemplo, $P_{MAX} = -9,57$ dBm e $R_{MAX} = 70$ m. Note que o grafo resultante não é necessariamente conexo e que as arestas diretas geradas neste passo são bidirecionais.

Algoritmo 4.2 : $CSCStep1(V, P_{MAX})$

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 do Se**
 - 6: **Fim do Para**
-

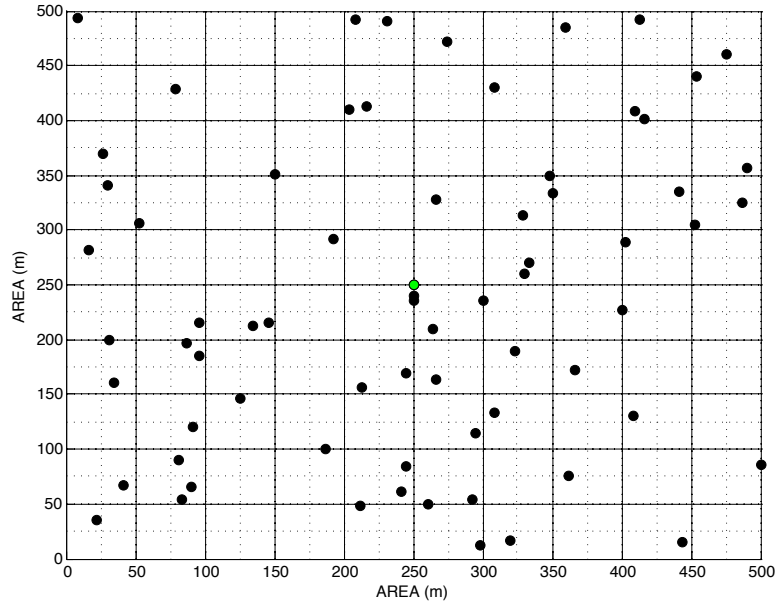


Figura 4.1: Exemplo de topologia de rede, onde $|V| = 70$ e $|E_P| = \emptyset$.

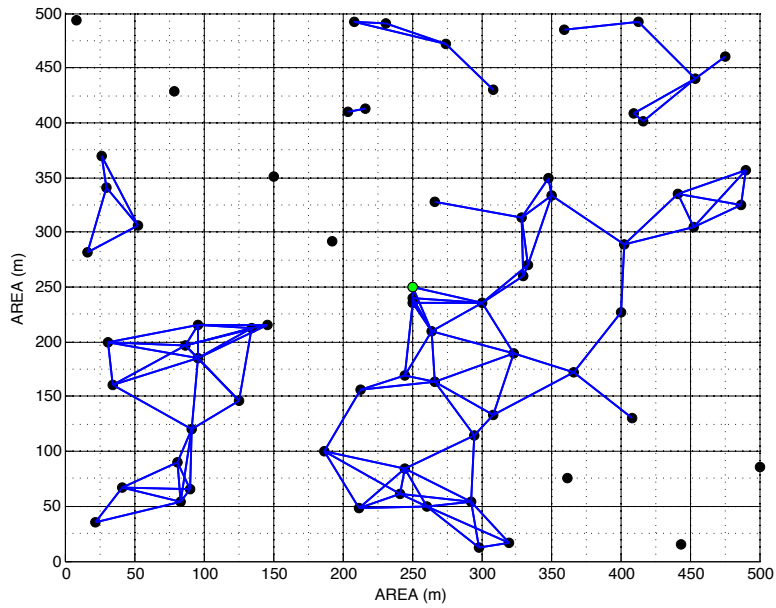


Figura 4.2: Grafo formado a partir da Figura 4.1 quando os nós operam em sua potência máxima P_{MAX} .

4.2.2.2 Passo 2: Construção dos grafos \widetilde{G}_P e G_P

O Algoritmo 4.3 descreve o segundo passo da técnica *CSC*. 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 auxiliares 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 $CSCStep2(\overline{G_P})$ para se referir ao Algoritmo 4.3.

Como exemplo, considere os grafos representados pelas Figuras 4.2 e 4.3. O grafo da Figura 4.3 é o grafo resultante do Algoritmo 4.3 quando recebe como entrada o grafo da Figura 4.2. 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 $\widetilde{v_j v_i} \in E_P$.

Algoritmo 4.3 : $CSCStep2(\overline{G_P})$

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 do Se**
 - 9: **Fim do Se**
 - 10: **Fim do Para**
 - 11: $G_P = \overline{G_P} + \widetilde{G_P}$
-

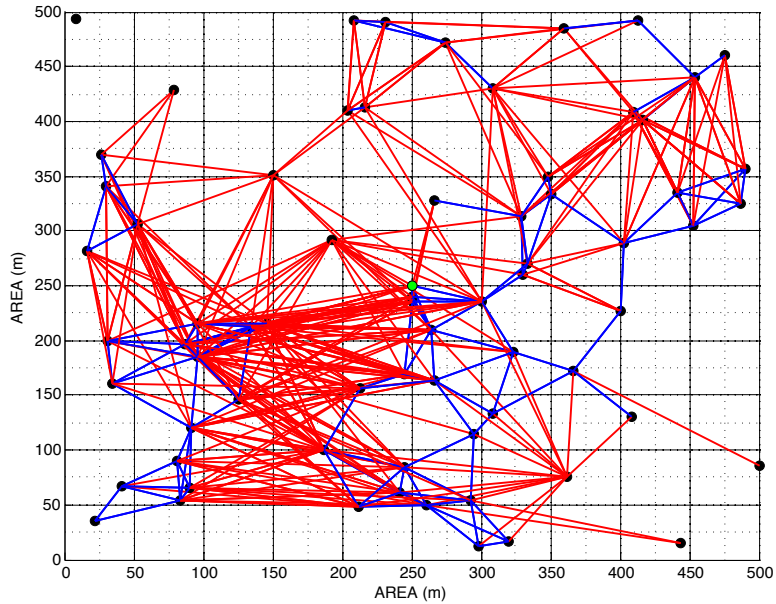


Figura 4.3: Grafo criado a partir da Figura 4.2 adicionando todas as arestas CC possíveis.

4.2.2.3 Passo 3: Cortando as arestas de G_P para criar o grafo G'_P

O Algoritmo 4.4 descreve o terceiro passo da técnica *CSC*. e recebe como entrada: um grafo G_P ; um nó v_o que, neste caso é o sorvedouro; a quantidade n de nós presente no grafo e uma variável booleana \mathfrak{R} . Esse passo consiste em criar um grafo G'_P que mantém as arestas baseado no valor do parâmetro \mathfrak{R} tal que:

- $\mathfrak{R} = 0$: Conjunto de arestas que obtém a menor quantidade de saltos para alcançar o nó sorvedouro;
- $\mathfrak{R} = 1$: Conjunto de arestas que obtém a menor quantidade de transmissões para alcançar no nó sorvedouro.

A seguinte função é utilizada nesse passo: $NextHopToSink(G_P, E_P, v_o, \mathfrak{R})$. Como o objetivo do *CSC* é diminuir a quantidade de saltos para o nó sorvedouro, o valor de \mathfrak{R} utilizado nesse passo é igual a 0. Esta função recebe o grafo G_P , o conjunto de arestas E_P , o nó sorvedouro v_o , o parâmetro \mathfrak{R} e retorna todas as arestas de menor caminho de algum nó v_i para o nó v_o . Na execução dessa função é utilizado o Algoritmo Dijkstra [35] para grafos direcionais e o resultado do menor caminho é baseado no número de saltos para transpor um enlace cuja definição é apresentada abaixo.

Definição 13 (*Número de saltos para transpor um enlace*): O número de saltos necessários para transpor um enlace direto $\overline{v_i v_j}$ é definido como:

$$hop(\overline{v_i v_j}) = 1, \quad (4.1)$$

representando a distância de um salto entre os nós v_i e v_j .

Enquanto o número de saltos necessários para transpor um enlace $CC \widetilde{v_i v_j}$ é definido como:

$$hop(\widetilde{v_i v_j}) = 2, \quad (4.2)$$

representando a distância virtual de dois saltos entre os nós v_i e v_j . O primeiro salto representa o primeiro momento da *CC* que é a transmissão do dado de v_i para os nós auxiliares e o segundo salto representa a transmissão, em sincronismo, do dado de v_i e dos nós auxiliares para v_j .

Considere o protótipo de função $CSCStep3(G_P, E_P, v_o, \mathfrak{R})$ para se referir ao Algoritmo 4.4.

Algoritmo 4.4 : $CSCStep3(G_P, E_P, v_o, \mathfrak{R})$

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

Saída : $G'_P = (V, E'_P)$;

- 1: $E'_P \leftarrow []$;
 - 2: **Para** $\forall v_i \in V$ **Faça**
 - 3: $e(v_i v_j) \leftarrow NextHopToSink(G_P, E_P, v_o, \mathfrak{R})$;
 - 4: v_i usa a aresta $e(v_i v_j)$ para alcançar o próximo salto em direção ao nó sorvedouro;
 - 5: $E'_P \leftarrow E'_P \cup e(v_i v_j)$;
 - 6: **Fim do Para**
-

Como exemplo, considere os grafos representados pelas Figuras 4.3 e 4.4. Nesse exemplo, o grafo representado pela Figura 4.3 é a entrada do Algoritmo 4.4, junto com v_o na posição (250,250), E_P que é o conjunto de arestas e o parâmetro \mathfrak{R} que é igual a zero. A saída do algoritmo é apresentada na Figura 4.4. Note que várias arestas foram retiradas do grafo e que uma estrutura de árvore foi criada em direção ao nó v_o que, nesse exemplo, fica na posição (250, 250) do eixo cartesiano. Se compararmos as Figuras 4.3 e 4.4, existe alguns nós que parecem desconectados e isso ocorre devido ao fato que as arestas cooperativas são direcionais. Assim, na Figura 4.3 alguns nós são alcançáveis a partir de outros nós mas estes não conseguem alcançar o nó sorvedouro como observado na Figura 4.4

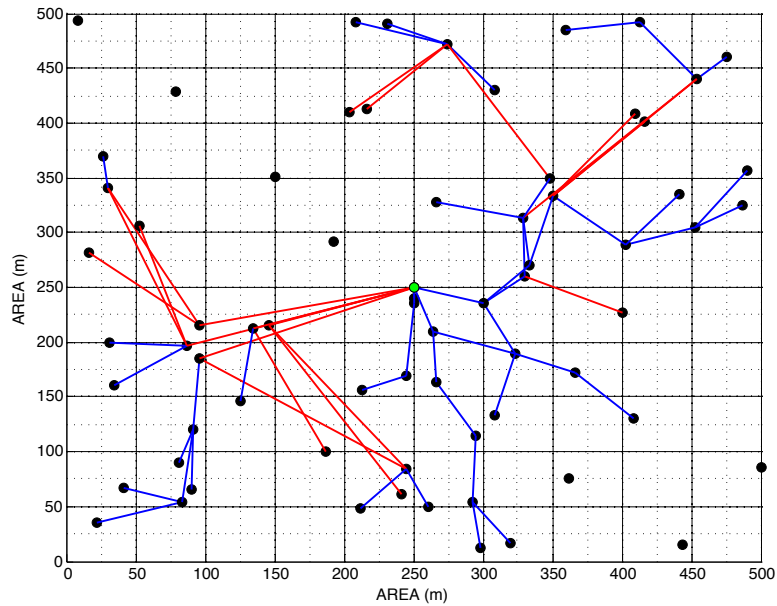


Figura 4.4: Grafo criado a partir da Figura 4.3 em que as arestas resultantes são as que promovem a menor quantidade de saltos até o sorvedouro.

4.2.2.4 Passo 4: Ajuste da potência P_i dos nós

O Algoritmo 4.5 descreve o último passo da técnica *CSC*. Esse passo consiste em permitir que os nós ajustem suas potências de transmissão de maneira que apenas a potência necessária para alcançar o nó mais distante dentro do alcance de transmissão mais a potência necessária para gerar as arestas cooperativas seja utilizada. Recebe como entrada G'_P e não possui retorno. Considere o protótipo de função $CSCStep4(G'_P)$ para se referir ao Algoritmo 4.5.

Algoritmo 4.5 : $CSCStep4(G'_P)$

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 do Para**
-

4.2.3 *CSC - HS*: Descrição da Técnica

Esta seção descreve os passos do *CSC-HS*, a proposta desse trabalho para aumentar a conectividade com o nó sorvedouro e diminuir a média de transmissões necessárias para um nó v_i alcançar o nó sorvedouro gerando rotas mais eficientes. Assume-se uma unidade central de computação que tem informações globalizadas da rede. O *CSC* consiste na execução em sequência de 4 passos:

- **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 do grafo de topologia criado no passo anterior, fazer controle de topologia para cortar arestas de maneira que se diminua a média de transmissões necessárias para um nó v_i alcançar o nó sorvedouro;
- **Passo 4**: Ajustar a potência dos nós para que o consumo de energia seja mínimo.

Como podemos observar, os passos 1, 2 e 4 são idênticos aos passos do *CSC*, por isso apresentaremos apenas o passo 3 a seguir.

4.2.3.1 **Passo 3**: Cortando as arestas de G_P para criar o grafo G'_P

O passo 3 da técnica *CSC-HS* também utiliza o Algoritmo 4.4. A principal diferença entre utilização deste algoritmo nas duas técnicas é o parâmetro de entrada \mathfrak{R} . No caso do *CSC*, o parâmetro \mathfrak{R} é definido como zero e no caso do *CSC-HS* o parâmetro \mathfrak{R} é definido como igual a um. Assim, utilizando o Algoritmo Dijkstra [35] para grafos direcionais é encontrado as arestas que constituem o caminho de menor custo baseado na métrica de quantidade de transmissões necessárias para transpor um enlace, cuja definição é apresentada abaixo.

Definição 14 (*Número de transmissões necessárias para transpor um enlace direto $\overline{v_i v_j}$ é definido como:*

$$trans(\overline{v_i v_j}) = 1, \tag{4.3}$$

representando a transmissão de um dado diretamente entre v_i e v_j .

Enquanto o número de transmissões necessárias para transpor um enlace $CC \widetilde{v_i v_j}$ é definido como:

$$\text{trans}(\widetilde{v_i v_j}) = |H_{ij}| + 2, \quad (4.4)$$

esta definição está relacionado com a quantidade de transmissões necessárias para o nó v_i alcançar o nó v_j . Inicialmente ocorre a transmissão do nó v_i para os nós auxiliares, num segundo momento ocorre a transmissão do nó v_i para o nó v_j em sincronismo com as transmissões dos nós auxiliares para v_j . Assim sendo, a quantidade de transmissões necessárias para v_i alcançar v_j depende diretamente da quantidade de nós auxiliares, ou seja, de $|H_{ij}|$.

Como exemplo, considere os grafos representados pelas Figuras 4.3 e 4.5. Nesse exemplo, o grafo representado pela Figuras 4.3 é a entrada do Algoritmo 4.4, junto com v_o na posição (250,250), o conjunto de arestas E_P e \mathfrak{R} que é igual a um. A saída do algoritmo é apresentada na Figura 4.5. Note que várias arestas foram retiradas do grafo e que uma estrutura de árvore foi criada em direção ao nó v_o que, nesse exemplo, fica na posição (250, 250) do eixo cartesiano.

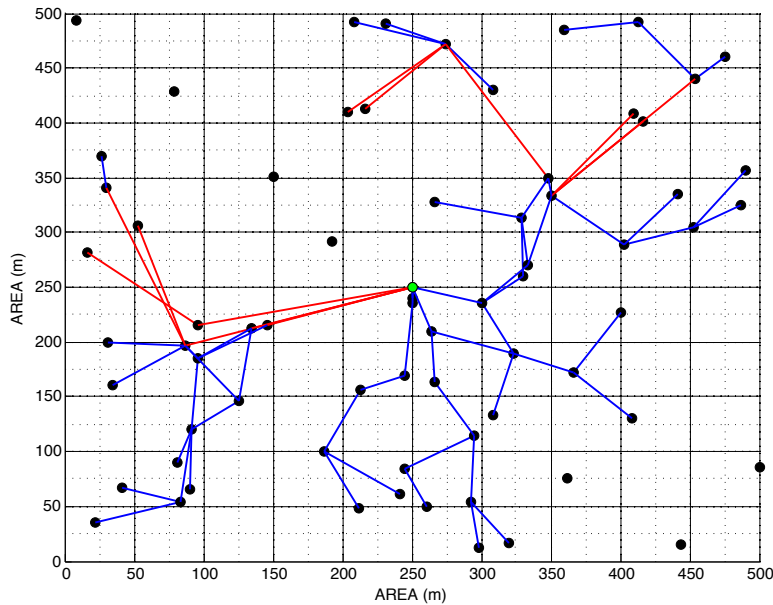


Figura 4.5: Grafo criado a partir da Figura 4.3 em que as arestas resultantes são as que promovem a menor quantidade de transmissões até o sorvedouro.

4.3 Resultados de Simulação

No capítulo anterior, foram propostas duas técnicas de controle de topologia em redes *ad hoc*: *CSC* e *CSC-HS*. A técnica *CSC* garante o caminho de menor custo para o sorvedouro e a técnica *CSC-HS* garante o caminho de menor custo possível para o sorvedouro de forma a manter as rotas eficientes em termos de transmissões necessárias para transpor um enlace. Neste capítulo, foram realizadas diversas simulações para comparar o desempenho do *CSC* e do *CSC-HS* com outras

técnicas de controle de topologia da literatura. Foram escolhidas para comparação as seguintes técnicas:

- *MST - Minimum Spanning Tree* [34]: Parte do grafo direto $\overline{G_P}$ (saída do passo 1 do *CSC*) e aplica o algoritmo *MST - Kruskal* para retirar as arestas menos eficientes em energia;
- *CoopSink* [5]: Parte do grafo G_P (saída do passo 2 do *CSC*) e cria um grafo G'_P que é feito de forma a manter as rotas eficientes em energia em relação ao nó sorvedouro. Num quarto momento, é feito o ajuste da potência de transmissão de modo a utilizar apenas a potência necessária para alcançar o nó vizinho mais distante e manter as arestas cooperativas.

Para avaliar a performance do *CSC* e do *CSC-HS* em relação às outras técnicas, foram utilizadas as seguintes métricas:

- **M1: Média de Saltos:** Consiste na média de saltos para um nó v_i alcançar o nó sorvedouro;
- **M2: Média de Transmissão por Enlace:** Consiste na média de transmissões necessárias para um nó v_i alcançar o sorvedouro;
- **M3: Potência Média da Rota:** Consiste na potência média da rota utilizada de um nó v_i para o nó sorvedouro na topologia final;
- **M4: Conectividade com o Sorvedouro:** Consiste na porcentagem de nós que possuem uma rota direcional para o nó sorvedouro;
- **M5: Potência Média de Transmissão por Nó:** Consiste na média das potências atribuídas a cada nó na topologia final;
- **M6: Número de Arestas CC:** Consiste em verificar quantas arestas CC foram criadas no grafo final de cada técnica.

A métrica M1 é justificada pois diminuir a quantidade de saltos para o sorvedouro é um dos objetivos do *CSC* e *CSC-HS*. Além disso é possível analisar como a quantidade de saltos influencia no atraso de fim-a-fim e qual o ganho da utilização destas técnicas em relação ao *CoopSink*. A métrica M2 é justificada pois diminuir a quantidade de transmissões para o sorvedouro é um dos objetivos do *CSC-HS*. Além disso, é possível avaliar nas métricas seguintes como diminuir a quantidade de transmissões afeta a quantidade de arestas cooperativas na topologia final e a potência média da rota. O objetivo da métrica M3 é avaliar o quão eficientes são as rotas geradas em termos de custo energético e esta métrica é justificada pois manter rotas eficientes para o nó sorvedouro é um dos objetivos do *CSC-HS*. A métrica M4 é utilizada para demonstrar o aumento de conectividade com a utilização de arestas CC e como essa conectividade cresce com o aumento da quantidade de nós no grafo. Como os dispositivos numa rede de sensores normalmente operam por baterias, é importante verificar o consumo energético por nó e para isto foi usada a métrica M5. Finalmente, na métrica M6 é feita uma comparação entre a quantidade de arestas cooperativas utilizadas na topologia final das técnicas apresentadas.

Para simulação, foram implementados os algoritmos descritos na Seção 4.2 no ambiente Matlab [6] e para efeitos de comparação, foram realizadas simulações para dois cenários com parâmetros diferentes. Os parâmetros de simulação dos dois cenários são apresentados a seguir:

1. **Cenário 1:** foi considerado um conjunto de nós n , numa área de 500×500 metros, cujo intervalo varia entre $[40, 120]$ nós com incrementos de 10 nós a cada iteração. Além disso, foi considerada uma potência máxima de transmissão de $P_{MAX} = 2500$ ou $-9,57\text{dBm}$, o que equivale a um $R_{MAX} = 50$ metros.
2. **Cenário 2:** foi considerado um conjunto de nós n , numa área de 500×500 metros, cujo intervalo varia entre $[20, 100]$ nós com incrementos de 10 nós a cada iteração. Além disso, foi considerada uma potência máxima de transmissão de $P_{MAX} = 4900$ ou $-3,73\text{dBm}$, o que equivale a um $R_{MAX} = 70$ metros.

O cenário 2 já foi utilizado em outras simulações de outras técnicas da literatura como o *CoopSink* [5] e *CoopBridges* [3]. Assim, para efeitos de comparação, também utilizamos os parâmetros do cenário 2. O cenário foi utilizado como meio de comparar o efeito da redução do alcance de transmissão no desempenho dos algoritmos analisados.

A consolidação de todos os parâmetros de simulação dos dois cenários é apresentada na Tabela 4.1 a seguir:

Tabela 4.1: Parâmetros de simulação dos cenários 1 e 2

Parâmetros	Cenário 1	Cenário 2
<i>Intervalo</i>	$\{n = 40, 50, \dots, 120\}$	$\{n = 20, 30, \dots, 100\}$
<i>Posição do Sorvedouro (v_1)</i>	(250,250)	(250,250)
<i>PathLossExponent</i>	2	2
P_{MAX}	2500	4900
R_{MAX}	50m	70m
<i>CoopSink - t</i>	1,4	1,4
<i>Conectividade Mínima</i>	10%	10%
<i>Modelo de Propagação Utilizado</i>	<i>Modelo de Dois Raios</i>	<i>Modelo de Dois Raios</i>
<i>Potência de Transmissão Máxima</i>	$-9,57\text{dBm}$	$-3,73\text{dBm}$
P_r	-71dBm	-71dBm
h_t	1,5m	1,5m
h_r	1,5m	1,5m
G_t	1	1
G_r	1	1
<i>Intervalo de Confiança</i>	95%	95%

Considerações:

1. No cenário 1 optou-se iniciar com 40 nós pois é a quantidade mínima aproximada de nós para cobrir a área de 500x500m. Já no cenário 2 essa quantidade mínima aproximada de nós é igual a 20. O cálculo da quantidade mínima de nós para cobrir uma área é dada a seguir:

$$\text{Quantidade mínima de nós} = \frac{\text{Area}}{\pi R_{MAX}^2} \quad (4.5)$$

No caso de $R_{MAX} = 50$ temos:

$$\text{Quantidade mínima de nós} = \frac{500 \times 500}{\pi 50^2} = 31,83 \text{ nós} \quad (4.6)$$

No caso de $R_{MAX} = 70$ temos:

$$\text{Quantidade mínima de nós} = \frac{500 \times 500}{\pi 70^2} = 16,24 \text{ nós} \quad (4.7)$$

2. Devido ao fato de começarmos com uma quantidade de nós diferente em cada cenário, foi utilizado o intervalo de nós $n = 40, 50, \dots, 120$ no cenário 1 e $n = 20, 30, \dots, 100$ para cenário 2, de forma a obtermos a mesma quantidade de pontos nos dois cenários.
3. Como o principal objetivo do *CSC* e do *CSC-HS* é criar rotas com a menor quantidade de saltos e transmissões em direção ao sorvedouro, foi necessário estabelecer uma conectividade mínima. Sem esta conectividade, haveria casos em que não existiria nenhuma rota para o sorvedouro o que tornaria o resultados das métricas apresentadas não correspondentes com a realidade.

Para as duas simulações, variando esses parâmetros é feita uma média dos resultados com 400 simulações. Os resultados apresentados dizem respeito a utilização das técnicas *MST*, *CSC*, *CSC-HS* e *CoopSink*. As subseções a seguir expõem os resultados das simulações para cada uma das métricas propostas.

4.3.1 M1: Média de Saltos

Esta métrica consiste na média de saltos necessários para o um nó v_i alcançar o nó sorvedouro. De modo a calcular essa métrica foi utilizada a Definição 13 da Seção 4.2.2.3 em todas as técnicas analisadas. As Figuras 4.6 e 4.7 apresentam os resultados das simulações para a métrica M1 com $R_{MAX} = 50m$ e $R_{MAX} = 70m$, respectivamente. No eixo das abscissas, varia-se a quantidade de nós na topologia da rede e no eixo das ordenadas, apresenta-se a média de saltos e este formato de apresentação dos gráficos será mantido nas métricas seguintes. As Tabelas 4.2 e 4.3 mostram a relação do número de saltos com o atraso de fim-a-fim para diversas tecnologias das camadas de enlace e física no pior caso, $R_{MAX} = 50m$ e $n = 120$ nós. Estas tabelas foram computadas utilizando as informações da Seção 2.4 e [2].

Observa-se que:

- O *CSC* tem a menor média de saltos dentre todas as técnicas analisadas que utilizam CC e para os dois valores de R_{MAX} , obtendo uma média de saltos até 50% melhor se comparado com *CoopSink* quando n é maior que 70 nós. O *CSC-HS* tem a segunda menor média de saltos ficando bem próximo do *CSC*. Isto ocorre porque o *CSC* usa a própria Definição 13 no passo 3 do seu algoritmo e o *CSC-HS* também usa uma variação do algoritmo de menor caminho no passo 3 da sua descrição. O *CoopSink* tem a maior média de saltos para $R_{MAX} = 50m$ e a segunda pior média de saltos para $R_{MAX} = 70m$ e n maior que 70, ficando atrás apenas do *MST*;
- Comparando os resultados para $R_{MAX} = 50m$ e $R_{MAX} = 70m$, observamos um crescimento no número de saltos quando utilizamos um R_{MAX} menor, com exceção do *MST*. Isto ocorre pois quanto menor o alcance, maior deve ser a quantidade de saltos para um nó distante chegar ao sorvedouro;
- Como o *MST* apenas calcula a árvore geradora mínima baseado no custo energético das arestas e não utiliza comunicação cooperativa, era esperado que, a partir de uma quantidade de nós, a quantidade de saltos fosse maior que as técnicas *CSC* e *CSC-HS*;
- Como vimos na seção 2.4, a média de saltos está diretamente associada ao atraso fim-a-fim, ou seja, quanto maior a quantidade de saltos maior será o atraso. Como observado nas Tabelas 4.2 e 4.3, a utilização do *CSC* e *CSC - HS* diminui em, no mínimo, 45% o atraso de propagação fim-a-fim para $R_{MAX} = 50m$ e $n = 120$ nós se comparado ao *CoopSink*. No caso do *CSC* para $R_{MAX} = 70m$ e $n = 100$ nós, essa redução pode chegar a mais de 50%.

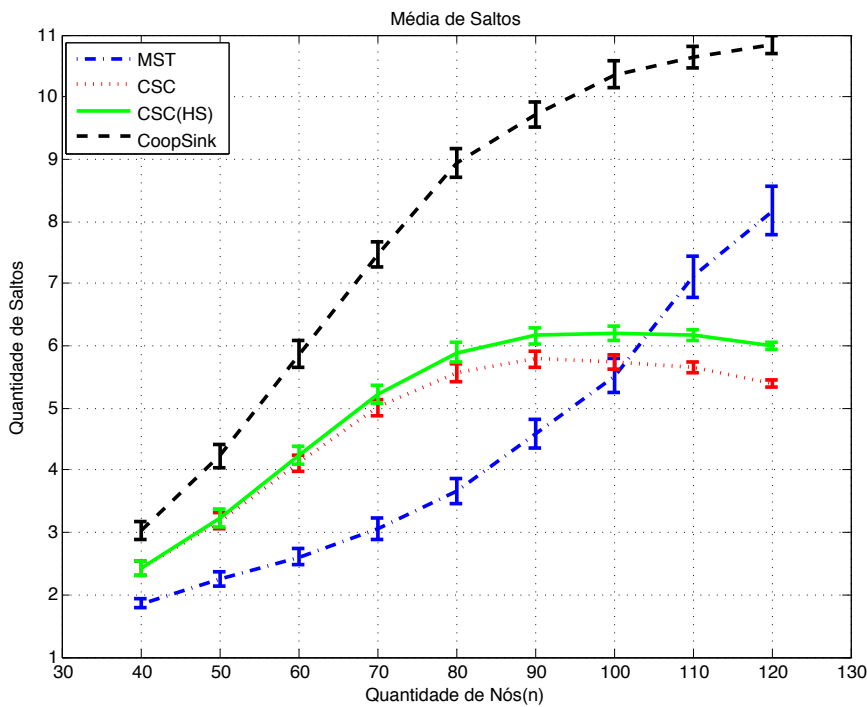


Figura 4.6: Média de Saltos para $R_{MAX} = 50$ metros.

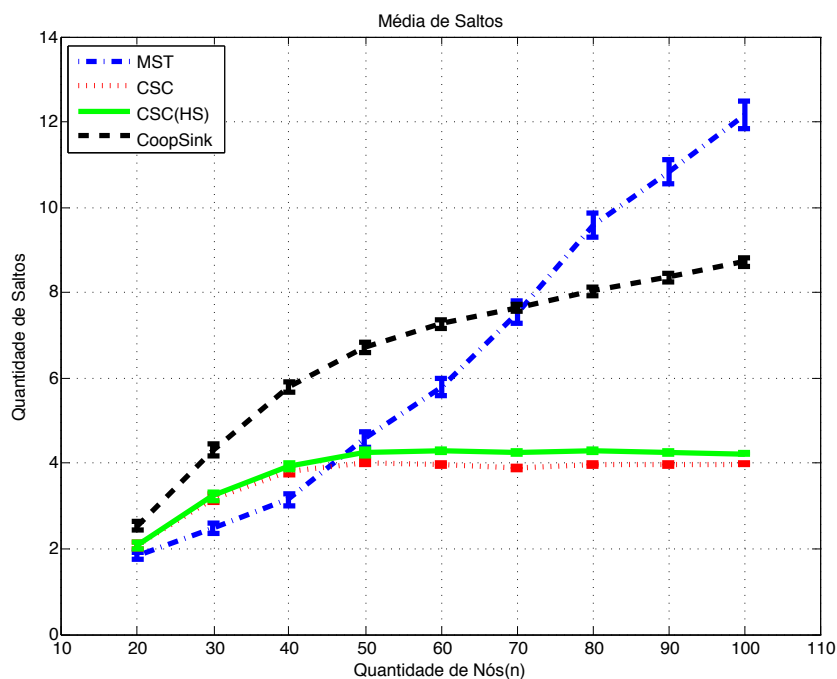


Figura 4.7: Média de Saltos para $R_{MAX} = 70$ metros.

Tabela 4.2: Atraso de Fim-a-Fim utilizando o *CSC* e o *CSC-HS* com quantidade de saltos = 6, utilizando CSMA/CA com RTS/CTS, $R_{MAX} = 50m$ e $n = 120$

Esquema	Taxa	Atraso		
		MSDU=512 bytes	MSDU=1024 bytes	MSDU=1480 bytes
FHSS	1Mbps	35,93ms	61,27ms	83,84ms
FHSS	2Mbps	22,41ms	35,08ms	46,37ms
DSSS	1Mbps	35,46ms	60,04ms	81,92ms
DSSS	2Mbps	22,36ms	34,64ms	45,59ms
HR - DSSS	5,5Mbps	14,02ms	18,49ms	22,47ms
HR - DSSS	11Mbps	11,63ms	13,87ms	15,86ms
OFDM	6Mbps	6,12ms	10,21ms	13,86ms
OFDM	12Mbps	3,69ms	5,73ms	7,56ms
OFDM	24Mbps	2,49ms	3,52ms	4,43ms
OFDM	54Mbps	1,81ms	2,27ms	2,67ms

Tabela 4.3: Atraso de fim-a-fim utilizando o *CoopSink* e com quantidade de saltos = 11, utilizando CSMA/CA com RTS/CTS, $R_{MAX} = 50m$ e $n = 120$

Esquema	Taxa	Atraso		
		MSDU=512 bytes	MSDU=1024 bytes	MSDU=1480 bytes
FHSS	1Mbps	65,86ms	112,33ms	153,71ms
FHSS	2Mbps	41,09ms	64,32ms	85,01ms
DSSS	1Mbps	65,01ms	110,07ms	150,19ms
DSSS	2Mbps	40,99ms	63,51ms	83,58ms
HR - DSSS	5,5Mbps	25,70ms	33,89ms	41,19ms
HR - DSSS	11Mbps	21,33ms	25,43ms	29,07ms
OFDM	6Mbps	11,02ms	18,73ms	25,41ms
OFDM	12Mbps	6,76ms	10,51ms	13,86ms
OFDM	24Mbps	4,57ms	6,45ms	8,12ms
OFDM	54Mbps	3,32ms	4,15ms	4,90ms

4.3.2 M2: Média de Transmissão por Enlace

Esta métrica consiste na média de transmissões necessária para o um nó v_i alcançar o nó sorvedouro. De modo a calcular essa métrica foi utilizada a Definição 14 da Seção 4.2.3.1 em todas as técnicas analisadas. As Figuras 4.8 e 4.9 apresentam os resultados das simulações para a métrica M2 com $R_{MAX} = 50m$ e $R_{MAX} = 70m$ respectivamente.

Observa-se que:

- O *CSC-HS* tem a menor média de transmissões por enlace dentre todas as técnicas analisadas que utilizam CC e para os dois valores de R_{MAX} e obtêm resultado 33,3% melhor que o *CoopSink* quando $R_{MAX} = 50m$ e $n = 120$ nós. O *CSC* tem a segunda menor média de transmissões ficando bem próximo do *CSC-HS*. Como o *CSC-HS* utiliza, no Passo 3 do seu algoritmo, a Definição 14 da Seção 4.2.3.1 que propõem a quantidade de transmissões necessárias para transpor um enlace, era esperado que este obtivesse o melhor resultado nesta métrica. O *CoopSink* tem a maior média de transmissões dentre todas as técnicas analisadas para $R_{MAX} = 50m$ e a segunda pior média para $R_{MAX} = 70m$ e n maior que 70 nós, ficando atrás apenas do *MST*;
- Comparando os resultados para $R_{MAX} = 50m$ e $R_{MAX} = 70m$, observamos um crescimento no número de transmissões quando utilizamos um R_{MAX} menor, com exceção do *MST*. Isso ocorre pois quanto menor o alcance, maior deve ser a quantidade de transmissões necessárias para um nó distante chegar ao sorvedouro;
- No caso de $R_{MAX} = 50m$ e $n = 120$ nós, o *CSC-HS* obtêm uma média de transmissões por enlace 33% menor que a do *CoopSink*. Já no caso de $R_{MAX} = 70m$ e $n = 100$ nós, o resultado do *CSC-HS* é ainda melhor, apresentando uma média de transmissões por enlace 44% menor que a do *CoopSink*.

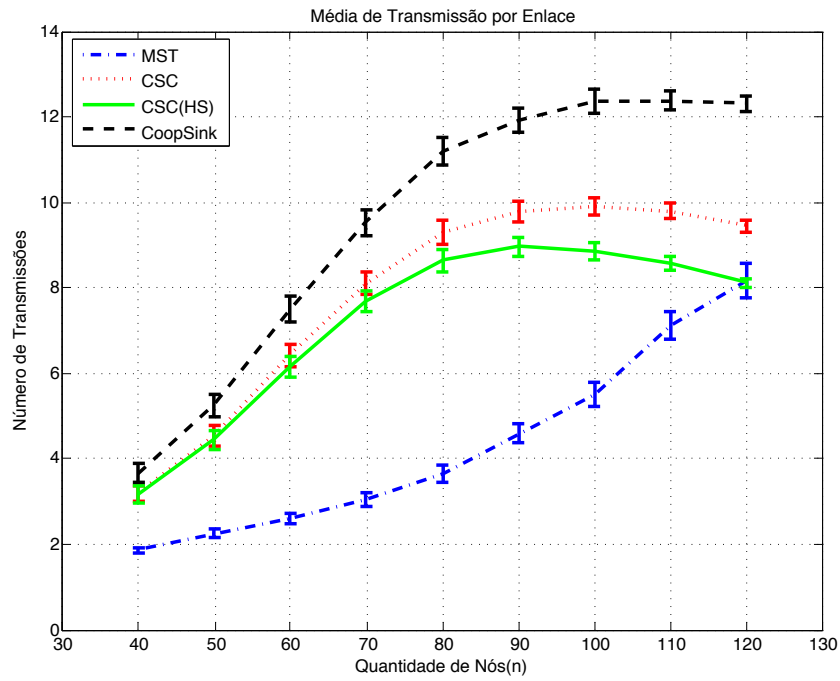


Figura 4.8: Média de Transmissão por Enlace para $R_{MAX} = 50$ metros.

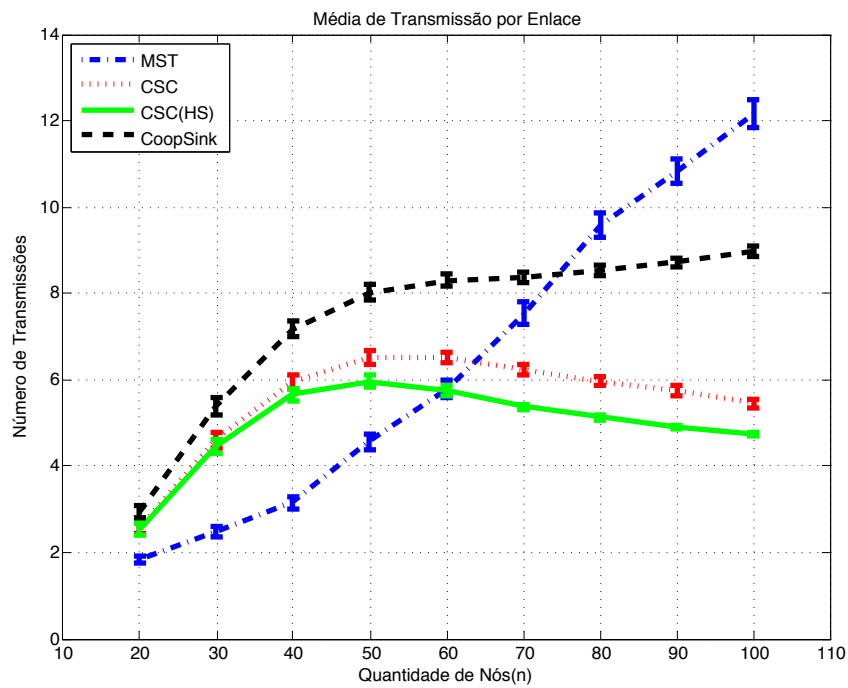


Figura 4.9: Média de Transmissão por Enlace para $R_{MAX} = 70$ metros.

4.3.3 M3: Potência Média da Rota

A potência média da rota é a média de todos os custos de menor caminho (Definição 8) para o sorvedouro. Para o cálculo desta métrica, foi aplicada a definição de custo de menor caminho na topologia final de todas as técnicas avaliadas. As Figuras 4.10 e 4.11 apresentam os resultados das simulações para a métrica M3 com $R_{MAX} = 50\text{m}$ e $R_{MAX} = 70\text{m}$, respectivamente.

- Para $R_{MAX} = 50\text{m}$, o *MST* apresenta a menor potência média da rota entre todas as técnicas apresentadas, isto ocorre pois o *MST* não usa arestas cooperativas, o que diminui o custo energético total. Já para $R_{MAX} = 70\text{m}$ e acima de 80 nós, o *MST* tem um resultado semelhante ao *CSC-HS* e pior do que *CoopSink*. Isso ocorre por causa do ajuste de potência do grafo que o *CoopSink* e o *CSC-HS* fazem para favorecer apenas conexões direcionais com o nó sorvedouro;
- Para ambos os cenários apresentados, o *CSC* tem a maior média de potência da rota apesar de esta técnica apresentar o menor caminho possível para o sorvedouro. Isso ocorre pois, esta técnica utiliza mais arestas cooperativas aumentando seu custo energético.
- O *CSC-HS*, para ambos os cenários apresentados, tem uma média de potência da rota próxima do *CoopSink*. Assim, apesar de ter uma potência média de transmissão por nó alta, a diminuição do número de saltos e transmissões consegue, na média, manter a potência média da rota quase tão eficiente quanto a do *CoopSink*. O *CSC-HS* consegue diminuir a quantidade de saltos e transmissões para alcançar o sorvedouro, usa um algoritmo mais simplificado e consegue ter uma eficiência energética por rota muito próxima do *CoopSink*, cujo algoritmo apenas dá ênfase ao aumento da conectividade e da eficiência das rotas.

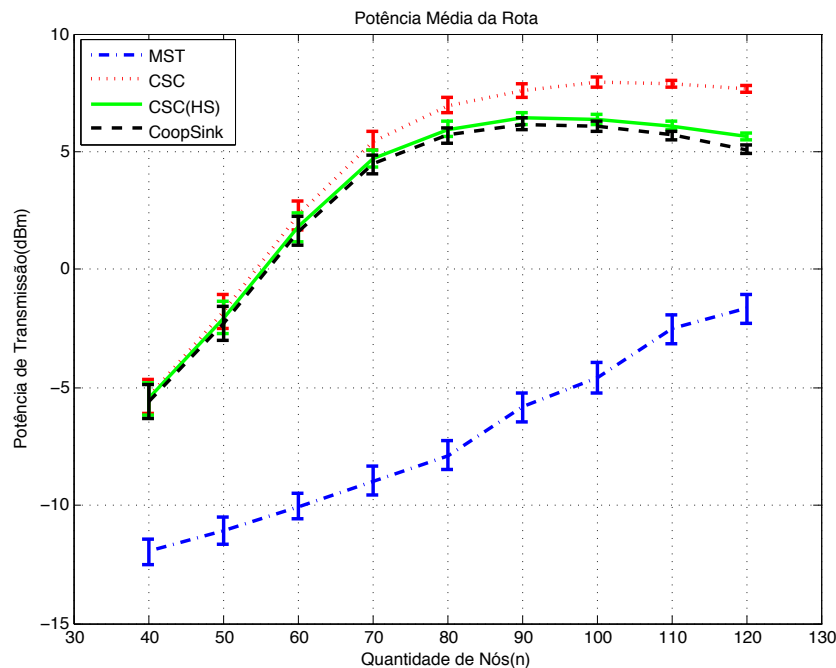


Figura 4.10: Potência Média da Rota para $R_{MAX} = 50$ metros.

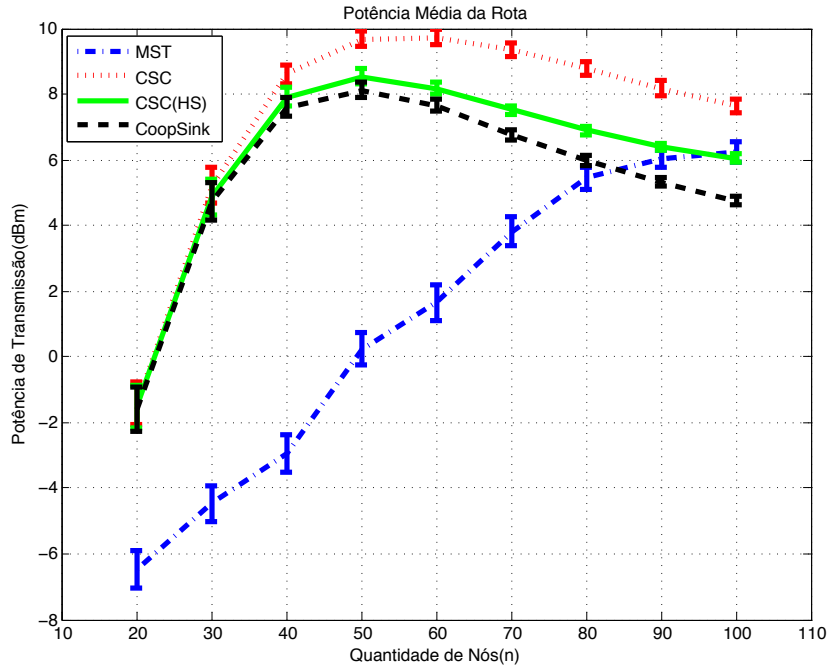


Figura 4.11: Potência Média da Rota para $R_{MAX} = 70$ metros.

4.3.4 M4: Conectividade com o Sorvedouro

As Figuras 4.12 e 4.13 apresentam os resultados das simulações para a métrica M1 com $R_{MAX} = 50$ m e $R_{MAX} = 70$ m, respectivamente. Esta métrica é utilizada para verificar o ganho em conectividade quando arestas cooperativas são utilizadas.

Observa-se que:

- O *CSC* e o *CSC-HS* apresentam conectividade igual ao *CoopSink* em todos os pontos observados tanto para $R_{MAX} = 50$ quanto para $R_{MAX} = 70$. Isto ocorre pois o *CSC*, o *CSC-HS* e o *CoopSink* utilizam o mesmo mecanismo de cálculo dos enlaces cooperativos;
- A conectividade do *CSC* e o *CSC-HS* superou a do *MST* em todos os pontos observados para as duas opções de R_{MAX} ;
- Para o cenário 1, a partir de 100 nós, mais de 80% dos nós possuem uma rota para o sorvedouro tanto no *CSC* quanto no *CSC-HS*. Para o cenário 2, a partir de 80 nós, mais de 90% dos nós possuem uma rota para o sorvedouro tanto no *CSC* quanto no *CSC-HS*. Devido ao fato de que o cenário usa um R_{MAX} maior, era esperado que fosse atingida uma conectividade maior com uma menor quantidade de nós.

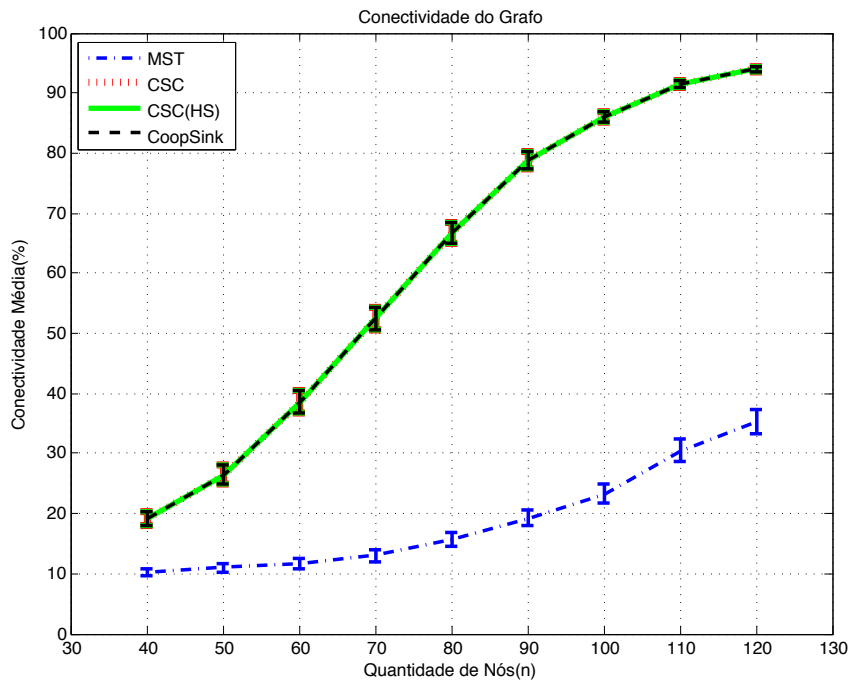


Figura 4.12: Conectividade com o Sorvedouro para $R_{MAX} = 50$ metros.

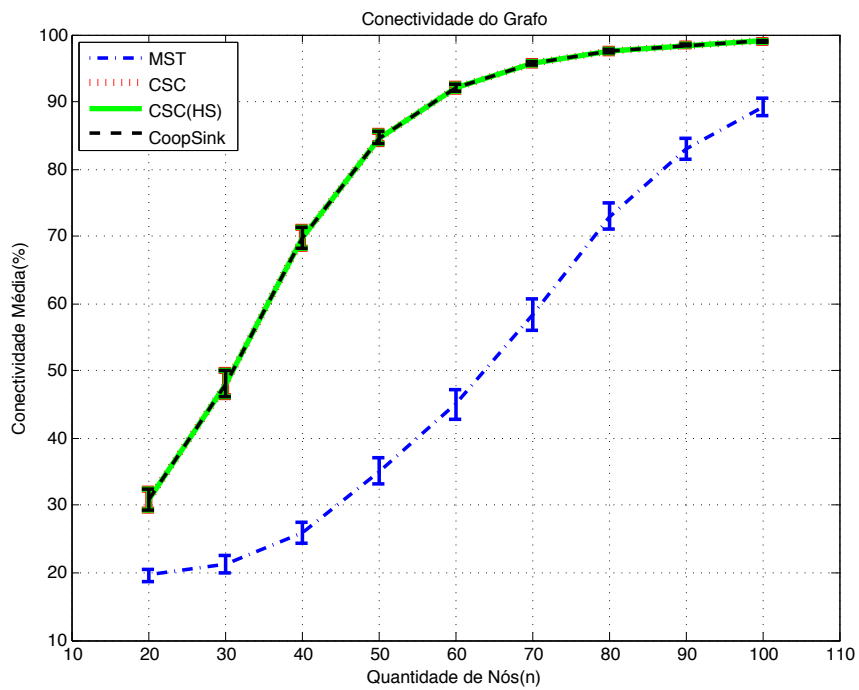


Figura 4.13: Conectividade com o Sorvedouro para $R_{MAX} = 70$ metros.

4.3.5 M5: Potência Média de Transmissão por Nó

As Figuras 4.14 e 4.15 apresentam os resultados das simulações para a métrica M5 com $R_{MAX} = 50m$ e $R_{MAX} = 70m$, respectivamente. Como os dispositivos numa rede de sensores normalmente operam por baterias, esta métrica foi utilizada para mostrar a eficiência energética por nó.

Observa-se que:

- Para se obter um alcance maior, é necessário uma potência de transmissão maior. Desta forma, a potência de transmissão para $R_{MAX} = 70m$ é maior em todos os pontos de todas as técnicas apresentadas do que a potência de transmissão para $R_{MAX} = 50m$.
- 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, para o caso de $R_{MAX} = 70m$, o *CoopSink* tem um custo energético menor que o do *MST*. Isso ocorre por causa do ajuste de potência do grafo para favorecer apenas conexões direcionais com o nó sorvedouro;
- Como as técnicas *CSC* e *CSC-HS* não levam em conta o custo energético na poda do grafos, era esperado que estas tivessem uma potência média de transmissão por nó maior;
- Dentre as técnicas propostas neste trabalho, o *CSC-HS* tem uma potência média de transmissão por nó ligeiramente menor do que a do *CSC* nos dois cenários apresentados. Isso ocorre pois o *CSC-HS* usa menos arestas cooperativas que o *CSC*. Apesar da diferença da potência média de transmissão por nó entre o *CSC* e *CSC-HS* ser pequena, essa diferença ajuda a reduzir o consumo energético o que é essencial já que os nós operam por baterias.

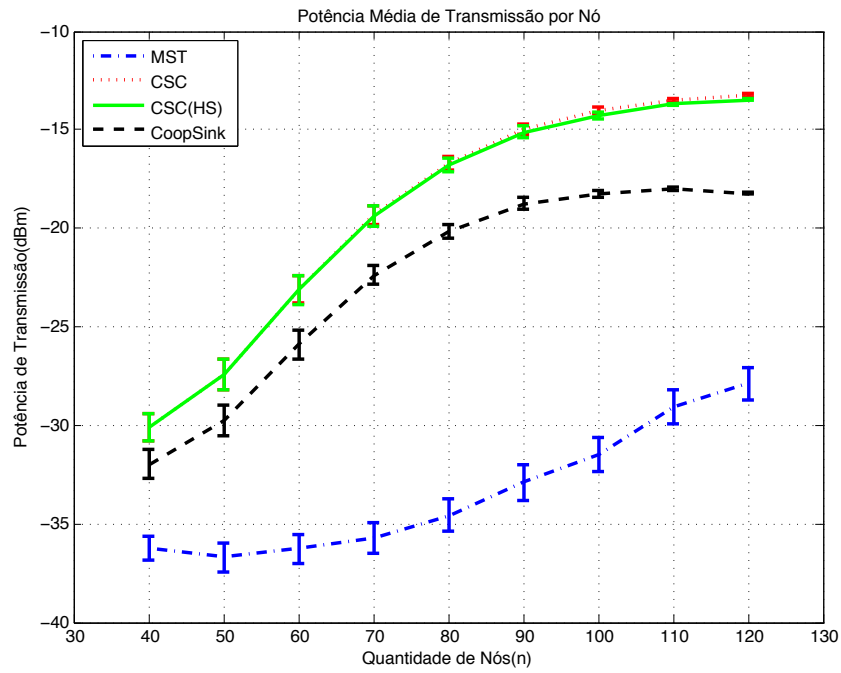


Figura 4.14: Potência Média de Transmissão por Nó para $R_{MAX} = 50$ metros.

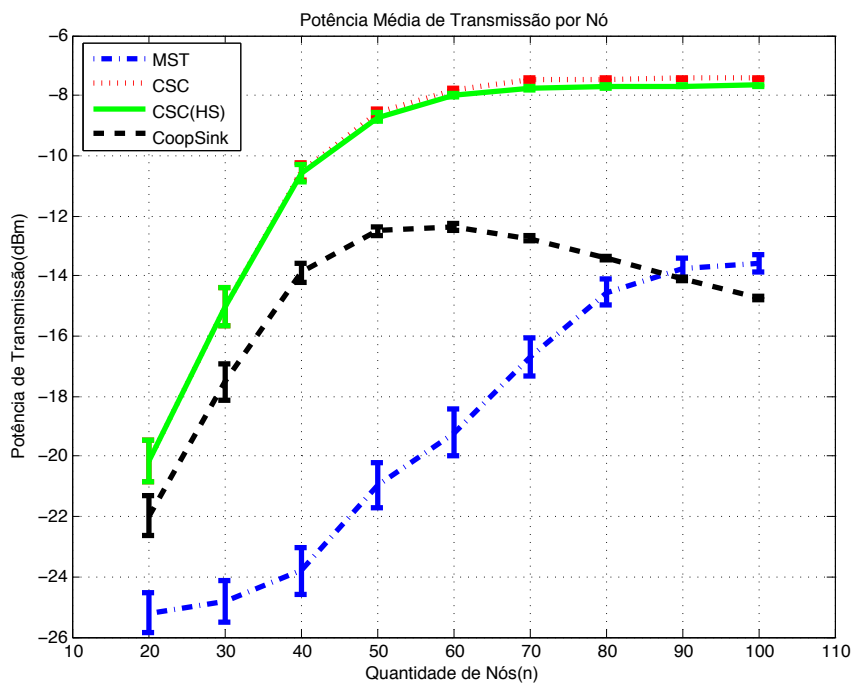


Figura 4.15: Potência de Transmissão por Nó para $R_{MAX} = 70$ metros.

4.3.6 M6: Número de Arestas CC

As Figuras 4.16 e 4.17 apresentam os resultados das simulações para a métrica M6. Como o *MST* não cria arestas cooperativas, esta técnica não será mostrada nas Figuras 4.16 e 4.17.

Observa-se que:

- Nos dois cenários, o *CSC* apresenta o maior número de arestas CC obtendo, aproximadamente, 77% mais arestas cooperativas do que o *CoopSink* quando $R_{MAX} = 50\text{m}$ e $n = 120$. Isto ocorre pois esta técnica tem como prioridade encontrar o menor caminho para o sorvedouro baseando-se na Definição 13 da Seção 4.2.2.3 e não observa o custo energético para tal. Assim, era esperado que esta técnica utiliza-se mais arestas CC;
- O *CSC-HS* também tem como prioridade encontrar o menor caminho para o sorvedouro. Entretanto, esta técnica observa a Definição 14 da Seção 4.2.3.1 na realização da poda do grafo e, por isso, a utilização de arestas cooperativas na topologia final é menor do que no *CSC*. Entretanto, esta técnica ainda utiliza, aproximadamente, 71% mais arestas cooperativas que o *CoopSink* para $R_{MAX} = 50\text{m}$ e $n = 120$ nós;
- O *CoopSink* tem menos arestas CC do que as técnicas *CSC* e *CSC-HS* nos dois cenários apresentados. Este resultado está condizente com os da métrica M5, onde a potência média de transmissão por nó foi menor do que nestas outras técnicas, justamente porque existem menos arestas CC no seu grafo resultante;
- A quantidade de arestas CC num grafo pode ser diretamente relacionada com o aumento no custo energético, já que estas arestas tem um custo energético maior do que as arestas diretas;
- Na Figura 4.17, é possível observar que quando a quantidade de nós aumenta e, conseqüentemente, a quantidade de arestas diretas também aumenta, o algoritmo do *CSC* e do *CSC-HS* tende a utilizar as arestas diretas já que estas tem um menor custo baseado nas métricas de cada algoritmo.

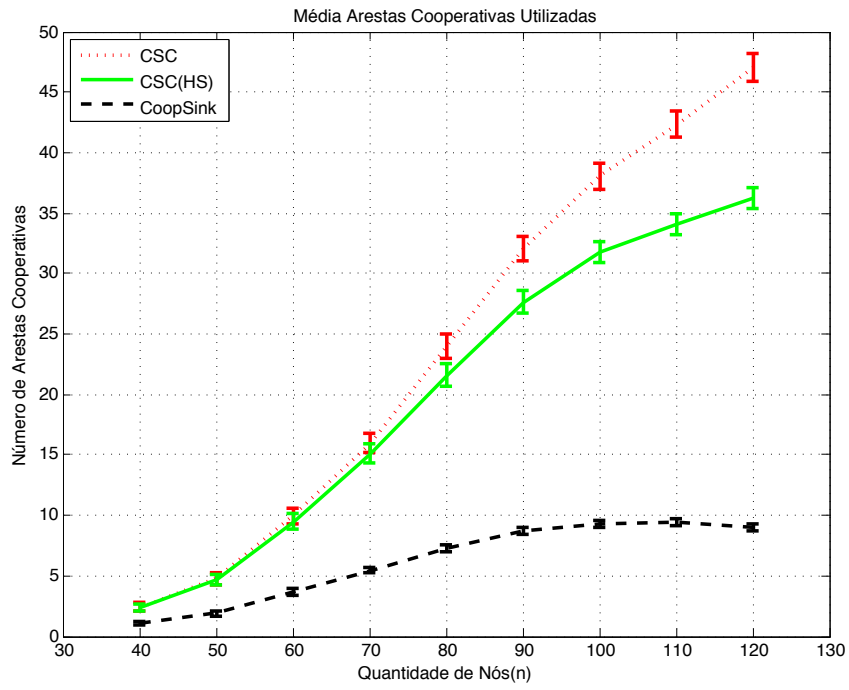


Figura 4.16: Número de Arestas CC para $R_{MAX} = 50$ metros.

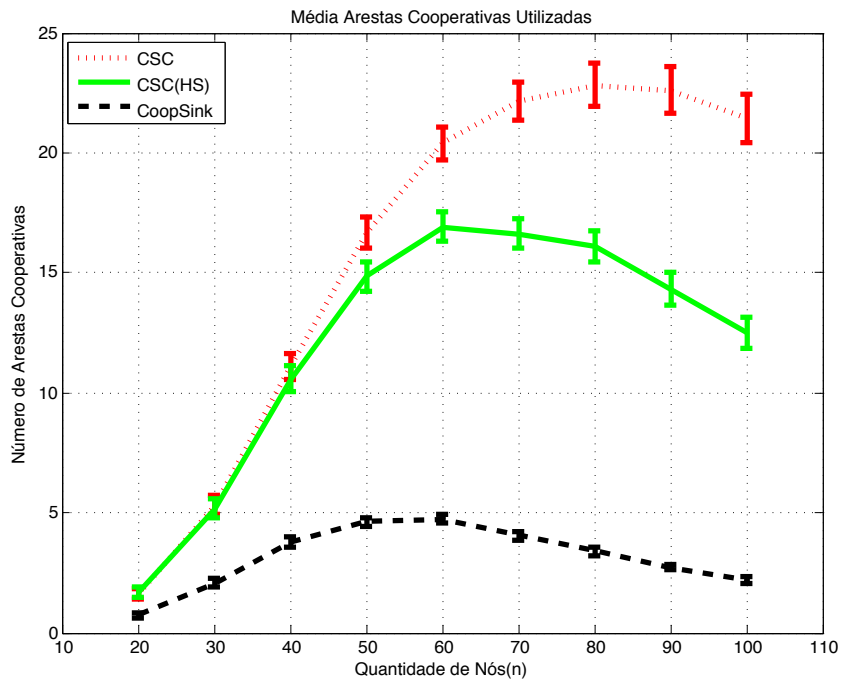


Figura 4.17: Número de Arestas CC para $R_{MAX} = 70$ metros.

4.4 Discussão

Nesse capítulo, foram apresentadas as duas propostas deste trabalho. O *CSC* é uma técnica que utiliza comunicação cooperativa e controle de topologia para aumentar a conectividade e diminuir o atraso fim-a-fim numa rede de sensores. O *CSC-HS* é uma técnica que utiliza comunicação cooperativa e controle de topologia para aumentar a conectividade e diminuir o atraso fim-a-fim mantendo a eficiência energética. Primeiramente, as duas técnicas foram descritas e seus algoritmos apresentados e, em seguida, as técnicas foram implementadas e comparadas com outras técnicas da literatura por meio de simulação. Como principais resultados temos que o *CSC* reduz em até 50% o atraso fim-a-fim se comparado com outras técnicas da literatura e o *CSC-HS* diminui em até 45% do atraso fim-a-fim mantendo uma eficiência energética próxima a do *CoopSink*. Além disso, as duas técnicas propostas, por utilizarem comunicação cooperativa, conseguem aumentar significativamente a conectividade com o sorvedouro se comparadas com técnicas da literatura que não utilizam CC.

Capítulo 5

Considerações Finais

A proposta deste trabalho consiste em utilizar comunicação cooperativa e controle de topologia para aumentar a conectividade, reduzir o atraso fim-a-fim mantendo a eficiência energética em redes *ad hoc*. Inicialmente, foram abordados os principais tópicos relacionados ao trabalho: redes *ad hoc*, modelos de propagação em ambientes sem fio, vazão máxima teórica de redes *ad hoc*, comunicação cooperativa e controle de topologia. Num segundo momento, foram apresentadas duas propostas: *CSC* e *CSC-HS*. Cada proposta foi detalhada e, por fim, simulações foram feitas para comparar as propostas com outras técnicas da literatura e avaliar o desempenho de cada técnica em métricas específicas.

A técnica *CSC* promove o aumento da conectividade com a utilização de comunicação cooperativa entre os nós e utiliza controle de topologia para redução do atraso fim-a-fim. Esta técnica apresentou um resultado até 50% superior, no melhor cenário, do que outras técnicas da literatura no quesito redução de atraso e demonstrou um aumento significativo da conectividade. Entretanto, a principal desvantagem desta técnica é o elevado custo energético. A técnica *CSC-HS* é uma melhoria da técnica anterior, o *CSC*. Nesta proposta, a comunicação cooperativa é utilizada para aumentar a conectividade dos nós sensores com o sorvedouro e o controle de topologia é utilizado para reduzir o atraso fim-a-fim mantendo as rotas para o sorvedouro as mais eficientes energeticamente possíveis. O *CSC-HS* obtêm resultados de até 45% de redução no atraso fim-a-fim se comparado com outras técnicas da literatura, apresenta uma eficiência energética das rotas próxima do *CoopSink* e aumenta a conectividade na mesma proporção que o *CSC*.

As duas técnicas apresentadas tem o seu nicho de uso. Por exemplo, pode-se usar o *CSC* quando existe a necessidade do nó sensor transmitir a informação com maior rapidez para o nó sorvedouro e utilizar o *CSC-HS* quando a eficiência energética das rotas for prioridade.

Como descrito, a comunicação cooperativa têm dois momentos: num primeiro momento, o nó fonte transmite o dado a ser transmitido para o nós auxiliares e no segundo momento o nó fonte e os nós auxiliares transmitem, em sincronismo, o dado para o nó de destino. Para o sincronismo entre o nó fonte e os nós auxiliares existirem, é necessário um protocolo de controle. Como trabalho futuro, sugere-se analisar o impacto de um protocolo de controle no atraso fim-a-fim e na eficiência energética das técnicas *CSC* e *CSC-HS* apresentadas neste trabalho.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] ALOMARI, S. A.; SUMARI, P. *An Overview of Mobile Ad Hoc Networks For the Existing Protocols and Applications*. 2010. 87-110 p.
- [2] JUN, J.; PEDDABACHAGARI, P.; SICHITIU, M. Theoretical Maximum Throughput of IEEE 802.11 and its Applications. In: *NCA '03: Proceedings of the Second IEEE International Symposium on Network Computing and Applications*. Washington, DC, USA: IEEE Computer Society, 2003. p. 249+. ISBN 0-7695-1938-5. Disponível em: <<http://portal.acm.org/citation.cfm?id=825340>>.
- [3] YU, J. et al. Cooperative bridges: Topology control in cooperative wireless ad hoc networks. In: *INFOCOM, 2010 Proceedings IEEE*. [S.l.: s.n.], 2010. p. 1–9. ISSN 0743-166X.
- [4] ZHOU, Z. et al. Energy-efficient cooperative communication based on power control and selective single-relay in wireless sensor networks. *Wireless Communications, IEEE Transactions on*, v. 7, n. 8, p. 3066–3078, August 2008. ISSN 1536-1276.
- [5] NEVES, T.; BORDIM, J. Topology control in cooperative ad hoc wireless networks. *Electronic Notes in Theoretical Computer Science*, v. 302, n. 0, p. 29 – 51, 2014. ISSN 1571-0661. Proceedings of the {XXXIX} Latin American Computing Conference (CLEI 2013). Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1571066114000206>>.
- [6] GRANT, M.; BOYD, S. *CVX: Matlab Software for Disciplined Convex Programming, version 2.1*. mar. 2014. <http://cvxr.com/cvx>.
- [7] LI, N.; HOU, J.; SHA, L. 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*. [S.l.: s.n.], 2003. v. 3, p. 1702–1712 vol.3. ISSN 0743-166X.
- [8] BASAGNI, S.; CONTI, M.; GIORDANO, S. *Mobile Ad Hoc Networking*. Wiley-IEEE Press, 2004. Hardcover. ISBN 0471373133. Disponível em: <<http://www.amazon.com/gp/product/0471373133>>.
- [9] JUBIN, J.; TORNOW, J. The darpa packet radio network protocols. *Proceedings of the IEEE*, v. 75, n. 1, p. 21–32, Jan 1987. ISSN 0018-9219.
- [10] BURATTI, C. et al. An overview on wireless sensor networks technology and evolution. *Sensors*, v. 9, n. 9, p. 6869–6896, 2009. ISSN 1424-8220. Disponível em: <<http://www.mdpi.com/1424-8220/9/9/6869>>.

- [11] RAPPAPORT, T. S. *Wireless Communication: Principles and Practice*. [S.l.]: Prentice Hall, 2002.
- [12] PERKINS, C. E.; BHAGWAT, P. Highly dynamic destination-sequenced distance-vector routing (dsv) for mobile computers. *SIGCOMM Comput. Commun. Rev.*, ACM, New York, NY, USA, v. 24, n. 4, p. 234–244, out. 1994. ISSN 0146-4833. Disponível em: <<http://doi.acm.org/10.1145/190809.190336>>.
- [13] CLAUSEN, T.; JACQUET, P. (Ed.). *Optimized Link State Routing Protocol (OLSR)*. United States: RFC Editor, 2003.
- [14] BELLMAN, R. On a Routing Problem. *Quarterly of Applied Mathematics*, v. 16, p. 87–90, 1958.
- [15] JOHNSON, D. B.; MALTZ, D. A.; BROCH, J. Ad hoc networking. In: . Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2001. cap. DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks, p. 139–172. ISBN 0-201-30976-9. Disponível em: <<http://dl.acm.org/citation.cfm?id=374547.374552>>.
- [16] PERKINS, C.; BELDING-ROYER, E.; DAS, S. *Ad Hoc On-Demand Distance Vector (AODV) Routing*. United States: RFC Editor, 2003.
- [17] PERKINS, C. E.; ROYER, E. M. Ad-hoc on-demand distance vector routing. In: *Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications*. Washington, DC, USA: IEEE Computer Society, 1999. (WMCSA '99), p. 90–. ISBN 0-7695-0025-0. Disponível em: <<http://dl.acm.org/citation.cfm?id=520551.837511>>.
- [18] IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE P802.11-REVmb/D12, November 2011 (Revision of IEEE Std 802.11-2007, as amended by IEEEs 802.11k-2008, 802.11r-2008, 802.11y-2008, 802.11w-2009, 802.11n-2009, 802.11p-2010, 802.11z-2010, 802.11v-2011, 802.11u-2011, and 802.11s-2011)*, p. 1–2910, March 2012.
- [19] BIANCHI, G.; FRATTA, L.; OLIVERI, M. 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*. [S.l.: s.n.], 1996. v. 2, p. 392–396 vol.2.
- [20] SADEK, A.; SU, W.; LIU, K. Multinode cooperative communications in wireless networks. *Signal Processing, IEEE Transactions on*, v. 55, n. 1, p. 341–355, Jan 2007. ISSN 1053-587X.
- [21] XIAO, Y. IEEE 802.11n: enhancements for higher throughput in wireless lans. *Wireless Communications, IEEE*, v. 12, n. 6, p. 82–91, Dec 2005. ISSN 1536-1284.
- [22] PAULRAJ, A. et al. An overview of MIMO communications - a key to gigabit wireless. *Proceedings of the IEEE*, v. 92, n. 2, p. 198–218, Feb 2004. ISSN 0018-9219.

- [23] CARDEI, M.; WU, J.; YANG, S. Topology control in ad hoc wireless networks using cooperative communication. *Mobile Computing, IEEE Transactions on*, v. 5, n. 6, p. 711–724, June 2006. ISSN 1536-1233.
- [24] LANEMAN, J.; TSE, D.; WORNELL, G. W. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *Information Theory, IEEE Transactions on*, v. 50, n. 12, p. 3062–3080, Dec 2004. ISSN 0018-9448.
- [25] KOTOBI, K. *Energy Conservation of Cooperative Communications over Composite Channel*. Tese (Masters in Electrical Engineering) — Delft University of Technology, June 2011.
- [26] LANEMAN, J.; WORNELL, G. W.; TSE, D. An efficient protocol for realizing cooperative diversity in wireless networks. In: *Information Theory, 2001. Proceedings. 2001 IEEE International Symposium on*. [S.l.: s.n.], 2001. p. 294–.
- [27] BHATNAGAR, M.; HJORUNGNES, A.; SONG, L. Amplify-and-forward cooperative communications using double-differential modulation over nakagami-m channels. In: *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*. [S.l.: s.n.], 2008. p. 350–355. ISSN 1525-3511.
- [28] BLETSAS, A.; SHIN, H.; WIN, M. Cooperative communications with outage-optimal opportunistic relaying. *Wireless Communications, IEEE Transactions on*, v. 6, n. 9, p. 3450–3460, September 2007. ISSN 1536-1276.
- [29] COVER, T.; GAMAL, A. Capacity theorems for the relay channel. *Information Theory, IEEE Transactions on*, v. 25, n. 5, p. 572–584, Sep 1979. ISSN 0018-9448.
- [30] YEH, C.-H. et al. Compressed-and-forward: Compressive sensing for cooperative communication. In: *Intelligent Signal Processing and Communications Systems (ISPACS), 2012 International Symposium on*. [S.l.: s.n.], 2012. p. 319–322.
- [31] SANTI, P. Topology control in wireless ad hoc and sensor networks. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 37, n. 2, p. 164–194, jun. 2005. ISSN 0360-0300. Disponível em: <<http://doi.acm.org/10.1145/1089733.1089736>>.
- [32] CHEN, W.-T.; HUANG, N.-F. The strongly connecting problem on multihop packet radio networks. *Communications, IEEE Transactions on*, v. 37, n. 3, p. 293–295, Mar 1989. ISSN 0090-6778.
- [33] CORMEN, T. H. et al. *Introduction to Algorithms, Third Edition*. 3rd. ed. [S.l.]: The MIT Press, 2009. ISBN 0262033844, 9780262033848.
- [34] KRUSKAL, J. B. On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. In: *Proceedings of the American Mathematical Society*, 7. [S.l.: s.n.], 1956.
- [35] DIJKSTRA, E. W. A note on two problems in connexion with graphs. *Numerische Mathematik*, v. 1, p. 269–271, 1959.