



Universidade do Estado do Rio de Janeiro

Centro de Tecnologia e Ciências

Faculdade de Engenharia

Felipe da Rocha Henriques


Conservação de energia em  
redes de sensores sem fio

Rio de Janeiro

2010

Felipe da Rocha Henriques

**Conservação de energia em  
redes de sensores sem fio**



Dissertação apresentada, como requisito parcial para obtenção do título de Mestre em Ciências, ao Programa de Pós-Graduação em Engenharia Eletrônica, da Universidade do Estado do Rio de Janeiro. Área de concentração: Redes de Telecomunicações.

Orientador: Prof. Dr. Lisandro Lovisolo

Orientador: Prof. Dr. Marcelo Gonçalves Rubinstein

Rio de Janeiro

2010

CATALOGAÇÃO NA FONTE  
UERJ / REDE SIRIUS / BIBLIOTECA CTC/B

H518    Henriques, Felipe da Rocha.  
          Conservação de energia em redes de sensores sem fio / Felipe da  
          Rocha Henriques. - 2010.  
          117 f.

          Orientador (es): Lisandro Lovisolo. Marcelo Gonçalves  
          Rubinstein.

          Dissertação (Mestrado) – Universidade do Estado do Rio de  
          Janeiro, Faculdade de Engenharia.  
          Bibliografia: f. 110-117.

          1. Rede de sensores sem fio. 2. Conservação de energia. I.  
          Lovisolo, Lisandro. II. Universidade do Estado do Rio. III. Título.

CDU 681.586

Autorizo, apenas para fins acadêmicos e científicos, a reprodução total ou parcial desta tese, desde que citada a fonte.

---

Assinatura

---

Data

Felipe da Rocha Henriques

**Conservação de Energia em  
Redes de Sensores Sem Fio**

Dissertação apresentada, como requisito parcial para obtenção do título de Mestre em Ciências, ao Programa de Pós-Graduação em Engenharia Eletrônica, da Universidade do Estado do Rio de Janeiro. Área de concentração: Redes de Telecomunicações.

Aprovado em: \_\_\_\_\_

Banca Examinadora:



\_\_\_\_\_  
Prof. Lisandro Lovisolo, D.Sc. (Orientador)

Universidade do Estado do Rio de Janeiro



\_\_\_\_\_  
Prof. Marcelo Gonçalves Rubinstein, D.Sc. (Orientador)

Universidade do Estado do Rio de Janeiro



\_\_\_\_\_  
Prof. José Gabriel Rodriguez Carneiro Gomes, Ph.D.

Universidade Federal do Rio de Janeiro



\_\_\_\_\_  
Prof. Miguel Elias Mitre Campista, D.Sc.

Universidade Federal do Rio de Janeiro

Rio de Janeiro

2010

*Dedico esta obra aos meus avôs:  
Romeu José da Rocha (in memoriam)  
e Antônio Rony Henriques (in memoriam).*

## AGRADECIMENTO

Agradeço em primeiro lugar à Deus, meu Rei e Senhor, por sempre ter me iluminado, guardado e amado de forma incondicional. É só pela Tua infinita misericórdia que consegui chegar aqui hoje. Agradeço aos meus queridos pais, Jorge Luiz e Maria Lúcia, por todo o amor, carinho, apoio e incontáveis ensinamentos durante toda a minha vida. Aos meus irmãos Jorge Júnior, Anna Carolina e Thaynara, pelo amor e incentivo. Agradeço a minha querida e amada esposa, Joana, um anjo de Deus na minha vida, que me acompanhou nesta batalha, me deu forças, carinho e compreensão. Te amo.

Agradeço aos meus orientadores, Prof. Dr. Lisandro Lovisolo e Prof. Dr. Marcelo Gonçalves Rubinstein, por terem acreditado em mim, me respeitado e me dado todo o suporte necessário para que eu conseguisse concluir esta etapa. Sem os senhores, muito mais do que orientadores, este trabalho não teria sido concluído. Especialmente ao Prof. Lisandro, por toda a ajuda na minha decisão de entrar para a vida acadêmica. Seu incentivo me fez chegar até onde eu nunca imaginaria conseguir.

Agradeço ao Prof. José Gabriel Rodríguez Carneiro Gomes, Ph.D. e ao Prof. Miguel Elias Mitre Campista, D.Sc., por terem gentilmente aceitado participar da minha banca.

Agradeço aos meus amigos de Mestrado, Vinícius, Flávia e Dalbert, que estiveram comigo do início ao fim desta jornada, muito obrigado. Ao meu amigo Pedro Arlindo, por toda força e alegria. Agradeço à minha querida amiga Renata Caminha (da Graduação ao Mestrado), que sempre me ajudou com suas palavras de amizade e incentivo. Agradeço à galera do PROSAICO, meus grandes amigos Rafael “you know what i’m saying” Saraiva Campos e Flávio Considera, por todos os momentos de trabalho e descontração. Em especial ao Rafael, pelo brilhante exemplo de extrema dedicação ao estudo e trabalho, e por toda ajuda, sempre atuando com denodo e galhardia. Gostaria de agradecer aos meus amigos de Graduação, especialmente ao Daniel Morim, Diego Haddad e Raphael Melo Guedes.

Enfim, agradeço aos meus amigos professores da Coordenação de Telecomunicações, do CEFET/RJ (UnED Petrópolis), por todo apoio nos momentos que mais precisei. MUITO OBRIGADO!!!

*Combati o bom combate,  
terminei a minha carreira,  
guardei a fé.*

2 Tim 4, 7

## RESUMO

HENRIQUES, Felipe da Rocha. Conservação de energia em redes de sensores sem fio. 2010. 117f. Dissertação (Mestrado em Engenharia Eletrônica) - Faculdade de Engenharia, Universidade do Estado do Rio de Janeiro, 2010.

Esta dissertação tem por objetivo propor algoritmos para conservação de energia de uma Rede de Sensores Sem Fio (RSSF) aplicada ao monitoramento de um processo suave  $f(x, y, t)$ , que depende das coordenadas  $x$  e  $y$  dos nós sensores, e do tempo  $t$ , de forma a aumentar a autonomia da rede. Os algoritmos são desenvolvidos para redes de salto único e para redes de múltiplos saltos. Os algoritmos rodam na camada de aplicação de cada nó, e visam a economia de energia dos nós através do gerenciamento da necessidade de transmissões. Após a primeira amostra transmitida, apenas amostras com uma variação percentual maior do que um dado limiar são transmitidas. Além disso, cada nó pode permanecer inativo (economizando energia) entre essas transmissões. Em RSSFs de salto único, são propostos dois algoritmos: um baseado na fonte, onde cada nó é responsável por todo o processamento e tomada de decisões, e outro baseado no sorvedouro, onde todo o processamento e a tomada de decisões são realizados pelo sorvedouro. Além disso, uma extensão do algoritmo baseado na fonte é proposta, para RSSFs de múltiplos saltos. Através dos resultados obtidos, observa-se que os algoritmos conseguiram uma redução significativa da quantidade de transmissões, o que leva a um aumento no tempo de vida da rede e um consequente aumento na sua autonomia. Por fim, uma análise do compromisso entre o tempo de vida e o erro na reconstrução do processo é apresentada. Desta forma, pode-se conjugar a relação entre tempo de vida máximo e erro de reconstrução mínimo.

Palavras-chaves: Redes de Sensores Sem Fio. Energia. Limiar de inovação. Reconstrução. Tempo de vida. Camada de aplicação.



## ABSTRACT

This work aims at proposing algorithms for energy conservation in a Wireless Sensor Network (WSN) applied to monitoring a smooth process  $f(x, y, t)$ , modeled as function of sensor nodes coordinates  $x$  and  $y$ , and time  $t$ , to increase the network autonomy. The proposed algorithms are developed for single and multihop networks. The algorithms work in the application layer of each node, and aim at saving node energy by managing the need of transmissions. After the first transmitted sample, only samples with a percentual variation greater than an innovation threshold are transmitted. Furthermore, each node can stay in a sleep mode (saving energy) between these transmissions. For single hop WSNs, two algorithms are proposed: one source-based, in which sensor nodes make all data processing and decisions, and another sink-based, in which all data processing and decisions are performed by the sink node. Furthermore, an extension of the source-based algorithm is proposed for multihop WSNs. Simulation results show that the algorithms lead to a significant reduction in the amount of transmissions, that leads to an increase of the network lifetime and an increment in its autonomy. Finally, an analysis of the trade-off between network lifetime and reconstruction error is presented. In this way, one can conjugate the relationship between maximum lifetime and minimum reconstruction error.

Keywords: Wireless Sensor Networks. Energy. Innovation threshold. Reconstruction. Lifetime. Application layer.

## LISTA DE FIGURAS

Figura 1	- Nós em uma RSSF e suas medidas. ....	18
Figura 2	- RSSF com o sorvedouro dentro e fora da região de interesse. ....	22
Figura 3	- Unidades básicas de um nó sensor. ....	23
Figura 4	- Arquitetura genérica de camadas das RSSFs. ....	23
Figura 5	- Pilha de protocolos ZigBee. ....	26
Figura 6	- Estrutura do quadro de dados. ....	27
Figura 7	- Estrutura do <i>superquadro</i> . ....	27
Figura 8	- Topologias de redes ZigBee. ....	29
Figura 9	- Diagrama do modelo de energia baseado em estados. ....	31
Figura 10	- Diagrama do modelo de energia utilizado nesta Dissertação. ....	33
Figura 11	- Consumo no estado de Transmissão para o <i>hardware</i> TELOS [Fonte [1]].	34
Figura 12	- Taxonomia dos métodos de economia de energia. ....	35
Figura 13	- Definição de ciclo de trabalho. ....	36
Figura 14	- Métodos baseados no ciclo de trabalho. ....	37
Figura 15	- Protocolos de controle de topologia. ....	37
Figura 16	- Métodos baseados no gerenciamento de energia. ....	39
Figura 17	- Distância entre alguns nós e suas medidas. ....	42
Figura 18	- Métodos baseados nos Dados. ....	43
Figura 19	- Métodos baseados na Redução de Dados. ....	44
Figura 20	- Métodos baseados em Mobilidade dos nós. ....	47
Figura 21	- Exemplo de comunicação utilizando o algoritmo baseado na fonte. ....	52
Figura 22	- Exemplo de comunicação utilizando o algoritmo baseado no sorvedouro.	54
Figura 23	- Mudança de estado do nó sensor (Ativo $\rightarrow$ Inativo). ....	55
Figura 24	- Áreas de cobertura de RSSFs de único salto e de múltiplos saltos. ....	57
Figura 25	- Duas possíveis topologias para RSSFs de múltiplos saltos. ....	59
Figura 26	- Nós e seus períodos de inatividade. ....	60
Figura 27	- Superfície monitorada em $t = 0$ . ....	65
Figura 28	- Blocos do simulador TrueTime. ....	67
Figura 29	- Modelo de uma rede sem fio utilizando o TrueTime 1.5, no Simulink. ...	68

Figura 30 - Superfície monitorada em $t = 0$ (para uma área coberta de $80m \times 80m$ ).	70
Figura 31 - Aumento no tempo de vida da rede em função do limiar de inovação....	73
Figura 32 - Redução percentual das transmissões em função do limiar de inovação. .	74
Figura 33 - Períodos de inatividade em função do tempo (sete nós e limiares distintos).	79
Figura 34 - Períodos de inatividade em função do tempo (limiar de 1%).	79
Figura 35 - Taxa de variação do processo monitorado em função do tempo.	80
Figura 36 - Reconstrução do processo, a partir da interpolação de ordem zero. ....	81
Figura 37 - CDF do erro de reconstrução (quatro a sete nós e limiar de 1%).	82
Figura 38 - CDF do erro de reconstrução (limiares de 0,1%, 1%, 5% e 8% e sete nós).	83
Figura 39 - Análise do tempo de vida e do erro de reconstrução em função do limiar.	85
Figura 40 - Tempo de vida em função do limiar de inovação (inatividade inicial entre 0 e 0,5 segundos).	87
Figura 41 - Redução nas transmissões em função do limiar de inovação (inatividade inicial entre 0 e 0,5 segundos. ....	88
Figura 42 - CDF do erro de reconstrução (sete nós e limiar de 1%).	89
Figura 43 - CDF do erro de reconstrução (inatividade inicial de 0,2 segundos).	90
Figura 44 - Aumento no tempo de vida da rede em função do limiar de inovação (energias iniciais de 2 e 3 Joules).	91
Figura 45 - Redução nas transmissões em função do limiar de inovação (energias iniciais de 2 e 3 Joules).	92
Figura 46 - Taxa de entrega de pacotes em função do limiar de inovação. ....	96
Figura 47 - Aumento no tempo de vida da rede em função do limiar de inovação....	97
Figura 48 - Redução nas transmissões em função do limiar de inovação.	98
Figura 49 - CDF do erro de reconstrução (quinze nós).	101
Figura 50 - Análise do tempo de vida e do erro de reconstrução em função do limiar.	102
Figura 51 - Inverso do aumento no tempo de vida em função do erro médio.	103
Figura 52 - Logaritmo do inverso do aumento no tempo de vida em função do erro médio.	104

## LISTA DE TABELAS

Tabela 1	- Parâmetros Fixos de Simulação. ....	69
Tabela 2	- Cenários de simulação (quantidade de nós em função da área). ....	69
Tabela 3	- Taxa de entrega em função do limiar de inovação.....	72
Tabela 4	- Tempo de vida absoluto em função da escalabilidade (RSSFs de salto único).....	76
Tabela 5	- Tempo de vida absoluto normalizado em relação à quatro nós em função da escalabilidade (RSSFs de salto único). ....	76
Tabela 6	- Quantidade de transmissões em função da escalabilidade (RSSFs de salto único).....	77
Tabela 7	- Quantidade de transmissões normalizada em relação à quatro nós em função da escalabilidade (RSSFs de salto único). ....	78
Tabela 8	- Estatísticas do erro de reconstrução (limiar de 1%).....	83
Tabela 9	- Estatísticas do erro de reconstrução (sete nós). ....	84
Tabela 10	- Estatísticas do erro de reconstrução (limiar de 1%).....	89
Tabela 11	- Estatísticas do erro de reconstrução (inatividade inicial de 0,2 segundos). ....	89
Tabela 12	- Tempo de vida absoluto em função da energia inicial dos nós. ....	93
Tabela 13	- Tempo de vida absoluto normalizado em relação à 3 Joules em função da energia inicial dos nós. ....	93
Tabela 14	- Tempo de vida absoluto em função da escalabilidade (RSSFs de múltiplos saltos). ....	99
Tabela 15	- Tempo de vida absoluto normalizado em relação à sete nós em função da escalabilidade (RSSFs de múltiplos saltos).....	99
Tabela 16	- Quantidade de transmissões em função da escalabilidade (RSSFs de múltiplos saltos). ....	100
Tabela 17	- Quantidade de transmissões normalizada em relação à sete nós em função da escalabilidade (RSSFs de múltiplos saltos). ....	100
Tabela 18	- Estatísticas do erro de reconstrução (quinze nós). ....	101

## LISTA DE ABREVIATURAS

ACK	<i>Acknowledgment</i>
AODV	<i>Ad hoc On-demand Distance Vector</i>
AODV-E	<i>AODV-Energy</i>
APS	<i>APplication suport Sublayer</i>
ASCENT	<i>Adaptative Self-Configuring sEnsor Networks Topologies</i>
B-MAC	<i>Berkeley MAC</i>
BPSK	<i>Binary Phase Shift Keying</i>
BSN	<i>Body Sensor Networks</i>
CC-MAC	<i>spatial Correlation-based Collaborative MAC</i>
CDF	<i>Cumulative Distribution Function</i>
CSMA-CA	<i>Carrier Sense Medium Access with Collision Avoidance</i>
D-MAC	<i>Data Gathering MAC</i>
FFD	<i>Full Function Device</i>
FLAMA	<i>FLow-Aware Medium Access</i>
GAD	<i>Geographical ADaptative fidelity</i>
GeRaF	<i>Geographic Random Forwarding</i>
GTS	<i>Guaranteed Time Slot</i>
IEEE	<i>Institute of Eletrical and Eletronics Engineers</i>
I/O	<i>In/Out</i>
MAC	<i>Medium Access Control</i>
MPEG	<i>Moving Picture Experts Group</i>
NWK	<i>NetWork Layer</i>
OLSR	<i>Optimized Link State Routing</i>
O-QPSK	<i>Ortogonal Quadrature Phase Shift Keying</i>
P2P	<i>Peer-to-Peer</i>

PAQ	<i>Probabilistic Adaptable Query system</i>
PREMON	<i>PREdiction-based MONitoring in sensor Networks</i>
PTDMA	<i>Probabilistic TDMA</i>
PTW	<i>Pipeline Tone Wakeup</i>
RFD	<i>Reduced Function Device</i>
RSSF	<i>Redes de Sensores Sem Fio</i>
S-MAC	<i>Sensor MAC</i>
STEM	<i>Sparse Topology and Energy Management</i>
TDMA	<i>Time Division Multiple Access</i>
TRAMA	<i>TRaffic-Adaptative Medium Access protocol</i>
Wi-Fi	<i>Wireless-Fidelity</i>
WPAN	<i>Wireless Personal Area Network</i>
ZDO	<i>Zigbee Device Object</i>
Z-MAC	<i>Zebra MAC</i>

## SUMÁRIO

	<b>INTRODUÇÃO</b> .....	16
1	<b>REDES DE SENSORES SEM FIO</b> .....	21
1.1	<b>Características de RSSFs</b> .....	21
1.1.1	<u>Arquitetura genérica</u> .....	22
1.1.2	<u>Constituição dos nós</u> .....	22
1.1.3	<u>Gerência de RSSFs</u> .....	23
1.2	<b>ZigBee</b> .....	25
1.2.1	<u>A pilha de protocolos</u> .....	25
1.2.2	<u>Componentes de uma rede ZigBee</u> .....	28
1.2.3	<u>Topologias de uma rede ZigBee</u> .....	28
1.2.3.1	Topologia em estrela .....	28
1.2.3.2	Topologia P2P .....	28
1.3	<b>Conclusão</b> .....	29
2	<b>CONSUMO DE ENERGIA EM RSSFs</b> .....	30
2.1	<b>Modelos de Energia</b> .....	30
2.1.1	<u>Modelos baseados em pacotes</u> .....	30
2.1.2	<u>Modelos baseados em estados</u> .....	31
2.2	<b>Modelo de Energia Utilizado nesta Dissertação</b> .....	32
2.3	<b>Métodos de Economia de Energia</b> .....	34
2.4	<b>Métodos de Economia de Energia Baseados no Ciclo de Trabalho</b>	35
2.4.1	<u>Controle de topologia</u> .....	36
2.4.1.1	Protocolos baseados em localização.....	37
2.4.1.2	Protocolos baseados na conectividade .....	38
2.4.2	<u>Gerenciamento de energia</u> .....	38
2.5	<b>Métodos de Economia de Energia Baseados nos Dados</b> .....	42
2.5.1	<u>Redução de dados</u> .....	43
2.5.1.1	Protocolos baseados na Predição de Dados .....	43
2.5.1.2	Compressão de dados .....	45

2.5.1.3	Processamento interno (na rede).....	45
2.5.2	<u>Aquisição de dados com eficiência energética</u> .....	45
2.6	<b>Métodos de Economia de Energia Baseados na Mobilidade</b> .....	47
2.6.1	<u>Sorvedouro móvel</u> .....	47
2.6.2	<u>Encaminhador móvel</u> .....	48
2.7	<b>Conclusão</b> .....	48
3	<b>ALGORITMOS PARA ECONOMIA DE ENERGIA EM RSSFs</b> .	50
3.1	<b>Algoritmos para RSSFs de Salto Único</b> .....	50
3.1.1	<u>Algoritmo baseado na fonte</u> .....	50
3.1.2	<u>Algoritmo baseado no sorvedouro</u> .....	52
3.2	<b>Algoritmo para RSSFs de Múltiplos Saltos</b> .....	56
3.2.1	<u>Extensão do algoritmo baseado na fonte para RSSFs de múltiplos saltos</u> ...	58
3.3	<b>Conclusão</b> .....	62
4	<b>AVALIAÇÃO DE DESEMPENHO ATRAVÉS DE SIMULAÇÃO</b>	64
4.1	<b>O Processo Monitorado</b> .....	64
4.2	<b>Métricas Utilizadas na Avaliação</b> .....	64
4.3	<b>Ambiente de Simulação</b> .....	66
4.4	<b>Conclusão</b> .....	69
5	<b>RESULTADOS DOS ALGORITMOS PARA ECONOMIA DE ENERGIA EM RSSFs DE SALTO ÚNICO</b> .....	71
5.1	<b>Sobre a Taxa de Entrega de Pacotes</b> .....	71
5.2	<b>Sobre o Tempo de Vida da Rede</b> .....	72
5.3	<b>Sobre a Redução nas Transmissões</b> .....	74
5.4	<b>Sobre a Escalabilidade da Rede</b> .....	74
5.5	<b>Sobre os Períodos de Inatividade</b> .....	77
5.6	<b>Sobre o Erro na Reconstrução do Processo</b> .....	80
5.7	<b>Alguns Pontos Sobre o Paradigma do Sorvedouro</b> .....	84
5.7.1	<u>Análise do tempo de inatividade inicial</u> .....	86
5.7.2	<u>Análise da energia inicial dos nós</u> .....	88
5.8	<b>Conclusão</b> .....	94



6	<b>RESULTADOS DO ALGORITMO PARA ECONOMIA DE ENERGIA EM RSSFs DE MÚLTIPLOS SALTOS</b> .....	95
6.1	<b>Adaptações no Protocolo de Roteamento Utilizado</b> .....	95
6.2	<b>Sobre a Taxa de Entrega de Pacotes</b> .....	96
6.3	<b>Sobre o Tempo de Vida da Rede</b> .....	96
6.4	<b>Sobre a Redução nas Transmissões</b> .....	97
6.5	<b>Sobre a Escalabilidade da Rede</b> .....	98
6.6	<b>Sobre o Erro na Reconstrução do Processo</b> .....	100
6.7	<b>Avaliação Conjunta: Tempo de Vida da Rede e Erro na Reconstrução do Processo</b> .....	101
6.8	<b>Conclusão</b> .....	103
	<b>CONCLUSÃO</b> .....	105
	<b>REFERÊNCIAS</b> .....	110

## INTRODUÇÃO

Recentes avanços em áreas como a microeletrônica, a eletrônica digital e a comunicação sem fio possibilitaram o desenvolvimento de sensores de baixo custo, baixo consumo energético e de pequeno tamanho. Esses sensores possuem as funcionalidades de sensoriamento, processamento de dados e comunicação, e podem ser utilizados como nós em um tipo especial de rede *ad hoc* chamada Rede de Sensores Sem Fio (RSSF) [2].

As RSSFs possuem aplicações em diversas áreas, tais como: na área médica (para o monitoramento remoto de pacientes), na área militar (para o monitoramento de forças inimigas), na automação residencial e comercial, além do monitoramento de regiões de interesse, desde uma casa ou um prédio, até uma floresta [2] [3] [4] [5].

### RSSFs para o Monitoramento de Regiões

O monitoramento de regiões é uma aplicação relevante de RSSFs, onde os nós sensores podem ser espalhados aleatoriamente pela área a ser monitorada, de modo a coletar informações de grandezas físicas, como temperatura, umidade e pressão. Para esse fim, a rede deve possuir características de autoconfiguração e adaptação a falhas, além de poder ser composta por centenas, ou até milhares de nós. Segundo [6], esse tipo de aplicação, onde se utilizam sensores espalhados para o monitoramento de processos variáveis espaço-temporalmente é chamado de estimação de campo. Ou seja, usa-se uma RSSF para o monitoramento de um processo que pode ser modelado como uma superfície  $f(x, y, t)$  bidimensional, que varia com as coordenadas  $x$  e  $y$  dos nós sensores, e com o tempo  $t$ .

Importa observar que, quanto maior for a autonomia da rede, que pode ser medida através da maximização do tempo de vida da rede, maior será sua capacidade de monitoramento. Porém, em geral, os nós sensores possuem uma severa restrição, em termos energéticos. Cada sensor possui, além das unidades de sensoriamento e de comunicação, uma unidade de energia (uma bateria), e sua troca é na maioria das vezes inviável, já que os nós podem estar espalhados em regiões de difícil acesso. Portanto, a energia gasta pelos nós sensores e como economizar essa energia são aspectos fundamentais em uma RSSF, para estender sua autonomia.

De maneira geral, a tarefa mais custosa, em termos energéticos, de uma RSSF é a comunicação (transmissão/recepção), sendo seu consumo de energia muito maior do que aquele gasto no processamento dos dados [2] [7]. Verifica-se [8] que a energia gasta na transmissão de um único bit de informação é aproximadamente a mesma necessária para realizar milhares de operações em um nó sensor. Logo, é interessante utilizar métodos que minimizem a quantidade de transmissões, de modo a economizar energia. Pode-se verificar um exemplo desta estratégia em [9], que utiliza a aplicação de estimação de campo e propõe protocolos que têm por objetivo reduzir a quantidade de amostras enviadas ao sorvedouro.

Da mesma forma, de modo a diminuir o gasto energético, em [10] utiliza-se a correlação espaço-temporal entre amostras medidas pelos nós, com o intuito de reduzir a quantidade de transmissões. Além disso, nem todos os nós precisam estar ativos simultaneamente. Alguns podem passar para um estado de inatividade (ou economia de energia) [11] e, com isso economizar energia e assim prolongar o tempo de vida da rede. Outro mecanismo utilizado para o mesmo fim é a agregação de dados, método que explora a redundância dos dados coletados, de modo a economizar energia dos nós sensores [12].

Na utilização de RSSFs para a coleta de dados provenientes de uma superfície  $f(x, y, t)$  monitorada, cada nó sensor realiza medidas e as envia para um nó sorvedouro (ou estação base). Na Figura 1, os nós sensores (em branco) são espalhados na região de interesse e cada amostra medida pelos nós ( $X_i(j)$ ) é transmitida para o sorvedouro (em cinza). Em relação à amostra  $X_i(j)$ ,  $i$  representa um dado nó sensor e  $j$  a ordem sequencial das amostras medidas por esse sensor. De posse dos dados recebidos de cada nó sensor, o sorvedouro pode reconstruir a superfície, obtendo assim uma estimação do processo  $\hat{f}(x, y, t)$  monitorado. Portanto, o erro na reconstrução do processo é uma questão importante, já que pode ser considerado como métrica de avaliação da estratégia empregada para o monitoramento e a comunicação dos dados coletados.

## Objetivos

Esta dissertação tem por objetivo propor e avaliar algoritmos para a economia de energia de uma RSSF aplicada ao monitoramento de um processo suave, de forma a aumentar a sua autonomia e, por conseguinte, realizar sua tarefa por mais tempo. Os

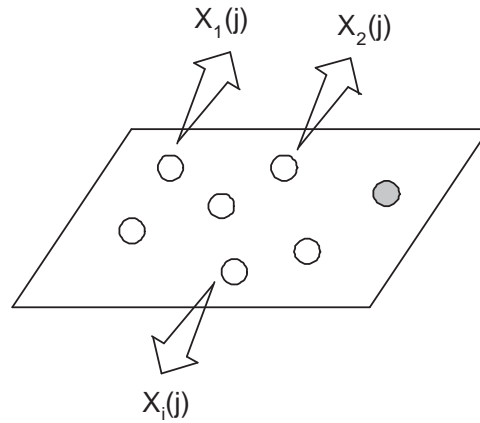


Figura 1 - Nós em uma RSSF e suas medidas.

algoritmos propostos no Capítulo 3 são aplicados diretamente na camada de aplicação, de forma a serem independentes de protocolos de roteamento. Outro objetivo vislumbrado no estudo dos algoritmos é a avaliação do erro de reconstrução do processo monitorado, de modo a conjugar o ganho do tempo de vida com a reconstrução do processo, obtida a partir das mensagens transmitidas pelos nós sensores. Além disso, no Capítulo 2, implementou-se um modelo de energia no ambiente de simulação utilizado, baseado em [13], onde além dos estados utilizados pelo autor citado, foram incluídos alguns modos de operação, de modo a tornar o modelo de energia mais realista.

### Os algoritmos para economia de energia

Nesta dissertação, foi considerado o monitoramento de um processo suave, tal como temperatura ou pressão. Os algoritmos propostos nesta dissertação são utilizados para gerenciar a necessidade de comunicação e com isso economizar energia [14] [15]. Inicialmente, dois algoritmos foram propostos, um baseado na fonte, e outro baseado no sorvedouro, ambos utilizados em RSSFs de salto único, ou seja, onde o sorvedouro está no raio de alcance de todos os nós sensores. Isso significa que, para transmitir uma mensagem para o sorvedouro, um dado nó sensor utiliza um único salto no processo de comunicação.

Nos algoritmos propostos, cada nó mede amostras do processo e as transmite para o sorvedouro. No algoritmo baseado na fonte, para cada nó sensor, após a primeira transmissão, o nó mede a próxima amostra e compara seu valor com o valor da amostra anteriormente transmitida. Caso esse valor seja maior que um limiar prefixado, aqui

chamado limiar de inovação, o nó transmite a amostra (pois é considerada uma amostra inovadora, ou com uma quantidade de inovação). Caso contrário, o nó mede a próxima amostra e a compara novamente. Além disso, os nós são postos em um estado de economia de energia entre transmissões inovadoras. Dessa forma, os nós economizam energia pela redução de transmissões e também pela utilização do estado de inatividade. Esse algoritmo é baseado na fonte, pois todo o processamento e tomada de decisões são feitos em cada nó sensor.

Já no algoritmo baseado no sorvedouro, os nós sensores não necessitam realizar nenhum processamento, tão pouco tomar decisões. Os nós apenas medem e transmitem amostras para o sorvedouro. Portanto, os nós utilizados podem ser mais simples e, com isso, mais baratos. Todo o processamento é realizado no sorvedouro que, a partir das amostras recebidas, calcula a variação percentual entre a amostra atual e a anteriormente recebida. Caso haja inovação (variação percentual entre as amostras maior do que o limiar), o nó calcula o período de inatividade, e manda que o respectivo nó transmissor durma (permaneça inativo) durante o período calculado. Após o nó acordar, ele mede e transmite novas amostras para o sorvedouro.

Uma extensão do algoritmo baseado na fonte é proposta, a qual utiliza um mecanismo que permite sua utilização em RSSFs de múltiplos saltos, onde o nó sorvedouro não está no raio de alcance de todos os nós sensores. Nessas RSSFs, se faz necessário o uso de comunicação de múltiplos saltos, através do encaminhamento das mensagens por nós intermediários (roteadores). No algoritmo aprimorado, quando um nó deseja transmitir uma mensagem, ele inclui um campo que contém o seu período de inatividade. Com isso, cada nó que encaminha mensagens de nós vizinhos atualiza um vetor que contém os períodos de inatividade. Assim, pode-se levar em conta os períodos de inatividade desses vizinhos, no cálculo do seu próprio. Isso permite com que o nó durma (economize energia), mas também acorde a tempo de encaminhar as possíveis mensagens de nós por ele roteadas.

Os algoritmos propostos são avaliados segundo métricas, como: a taxa de entrega de pacotes, o aumento do tempo de vida da rede, a redução da quantidade de transmissões e o erro de reconstrução do processo. O aumento do tempo de vida da rede e a redução da quantidade de transmissões são relativos às simulações sem o gerenciamento de energia. A partir dos resultados obtidos, pode-se verificar que os algoritmos conseguem significativos

aumentos no tempo de vida da rede e reduções na quantidade de transmissões. Em RSSFs de salto único, conseguiu-se um aumento percentual maior do que 190% no tempo de vida da rede. Já em RSSFs de múltiplos saltos, obteve-se um tempo de vida superior a vinte vezes o tempo de vida sem a utilização dos algoritmos propostos. Não obstante, uma tentativa de avaliar o compromisso entre o tempo de vida da rede e o erro de reconstrução do processo é apresentada.

### **Organização da Dissertação**

Esta dissertação está organizada da seguinte forma: no Capítulo 1, alguns aspectos acerca de RSSFs são apresentados, como características dos seus componentes e sua arquitetura. Além disso, a tecnologia de rede ZigBee é estudada, pelo fato dela ser utilizada na avaliação dos algoritmos propostos. O Capítulo 2 trata do consumo de energia em nós sensores. Inicialmente, dois modelos de consumo de energia encontrados na literatura são discutidos. Após isso, o modelo de energia utilizado nesta dissertação é apresentado. O Capítulo 3 apresenta os algoritmos propostos, para RSSFs de salto único e múltiplos saltos. Para RSSFs de salto único, são apresentados dois algoritmos: um baseado na fonte e outro baseado no sorvedouro. Já para RSSFs de múltiplos saltos, uma extensão do algoritmo baseado na fonte é proposta, assim como o mecanismo que permite sua utilização em RSSFs de múltiplos saltos. No Capítulo 4, a estratégia de testes utilizada nesta dissertação é apresentada, assim como o ambiente de simulação utilizado, e as métricas de avaliação são discutidas. O Capítulo 5 apresenta os resultados obtidos nas simulações dos algoritmos propostos, para RSSFs de salto único. Já no Capítulo 6 são apresentados os resultados obtidos, para o algoritmo proposto para múltiplos saltos. Enfim, o último capítulo traz as conclusões da dissertação e possíveis trabalhos futuros.

## 1 REDES DE SENSORES SEM FIO

Este capítulo trata de aspectos básicos sobre Redes de Sensores Sem Fios (RSSFs). Algumas particularidades das RSSFs são apresentadas, como as que as diferem de redes *ad hoc* tradicionais, além das características dos seus componentes e a arquitetura genérica da pilha de protocolos. Além disso, algumas aplicações de RSSFs são citadas. Será discutido um padrão muito utilizado em RSSFs, o ZigBee.

### 1.1 Características de RSSFs

Redes de Sensores Sem Fio são um tipo especial de redes *ad hoc* (redes onde os nós podem se comunicar diretamente, sem que haja a necessidade de uma infraestrutura, como uma estação base). As RSSFs possuem nós capazes de realizar tarefas de sensoria-mento, processamento de dados e comunicação [4]. Estes nós, que possuem sensores, tais como sísmicos, de calor, de umidade ou de pressão, possuem algumas características pró-prias, que não se observam em nós de redes *ad hoc* tradicionais, como tamanho reduzido, grandes restrições de energia, capacidades computacional e de memória restritas, além da necessidade de auto-organização e de adaptação a falhas. Além dessas peculiaridades intrínsecas aos nós sensores, algumas outras, mais gerais, corroboram a afirmação de que as RSSFs são “um tipo especial de redes *ad hoc*” [2]:

- a quantidade de nós em uma RSSF pode ser muito maior do que em uma rede *ad hoc*, já que os nós de uma RSSF devem possuir um custo menor do que os nós das redes *ad hoc*;
- os nós são dispostos densamente em uma região, onde a densidade pode ser definida como  $d = \text{quantidade de nós}/\text{área}$ ;
- a topologia de uma RSSF pode variar frequentemente, por exemplo, no caso de haver falhas ou falta de energia em alguns nós, ou ainda se a rede for composta por nós móveis.

As Redes de Sensores Sem Fio possuem larga faixa de aplicação, como na área militar, através do monitoramento de forças inimigas; na área médica, com o monitора-mento remoto de pacientes; na automação residencial e industrial; no monitoramento do

tráfego de veículos; em segurança e no monitoramento ambiental em áreas internas como prédios ou em áreas externas como florestas [4] [3] [5]. Outros exemplos são aplicações que se utilizam do fato destas redes poderem ser consideradas como um sistema descentralizado com nós autônomos, útil para a detecção distribuída de fenômenos [16], [17], e a inferência distribuída acerca de fenômenos detectados [18], [19]. No caso da inferência, os nós podem, a partir de dados coletados, estimar informações futuras acerca do fenômeno em questão.

### 1.1.1 Arquitetura genérica

Uma RSSF pode ser composta por dois tipos de nós, sensores ou sorvedouros. Os nós sensores, espalhados em uma região de interesse, coletam dados e os repassam para o sorvedouro. O sorvedouro pode estar posicionado dentro da área de interesse, ou fora dela (Figura 2), onde pode fazer o papel de um *gateway* para a Internet, por exemplo. Principalmente em casos onde o sorvedouro está fora do alcance de alguns nós sensores, esses nós podem utilizar comunicação de múltiplos saltos para que seus dados possam ser encaminhados até o sorvedouro.

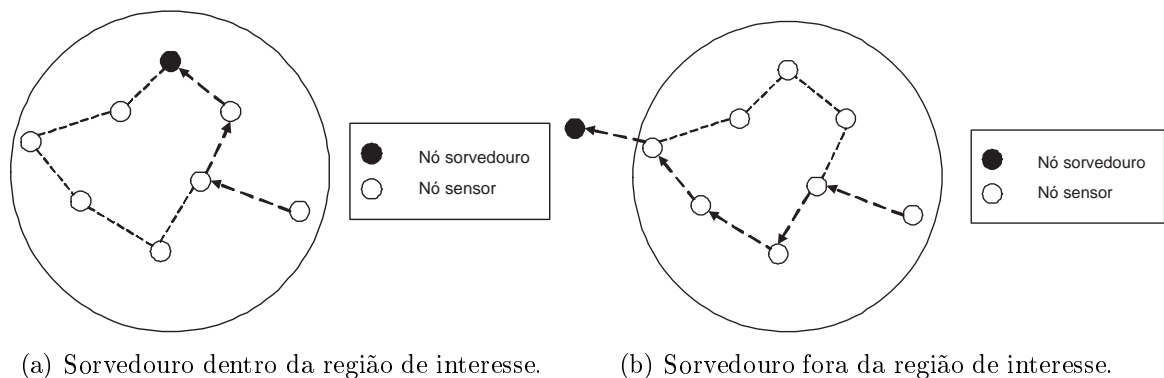


Figura 2 - RSSF com o sorvedouro dentro e fora da região de interesse.

### 1.1.2 Constituição dos nós

Em termos de *hardware*, cada nó sensor é composto por quatro unidades básicas, conforme visto na Figura 3 [2]:

- unidade de Sensoriamento, que permite aos nós sensores realizarem medições;



- unidade de Processamento, responsável pelo processamento dos dados dos nós;
- unidade de Comunicação, pela qual os nós podem transmitir e receber dados;
- unidade de Energia, uma bateria que alimenta as três unidades anteriores, possibilitando aos nós realizarem suas tarefas.

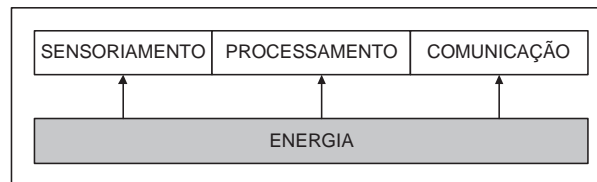


Figura 3 - Unidades básicas de um nó sensor.

### 1.1.3 Gerência de RSSFs

A arquitetura genérica de protocolos em RSSFs pode ser vista na Figura 4. Observa-se que as cinco camadas pertencentes ao modelo híbrido das redes de computadores [20] [21] trabalham junto com três planos de gerenciamento [2]. Algumas características do modelo em cinco camadas, e dos planos de gerenciamento serão descritas a seguir.

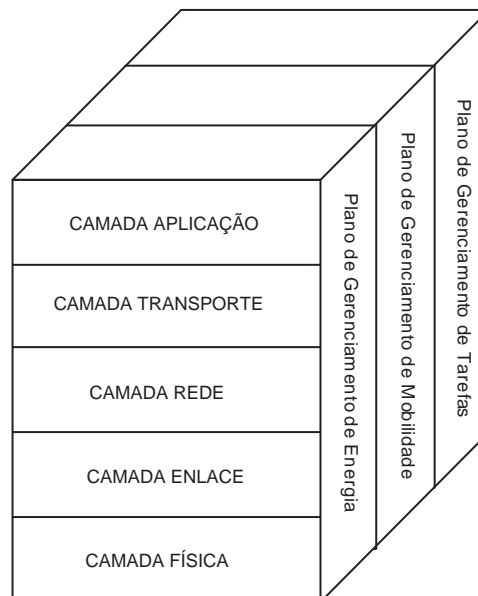


Figura 4 - Arquitetura genérica de camadas das RSSFs.

As camadas do modelo híbrido são:

- **camada Física:** Responsável pela transmissão e recepção de bits; define as interfaces elétricas e mecânicas e a especificação do meio físico;
- **camada de Enlace:** Realiza o controle de acesso ao meio compartilhado, o enquadramento dos dados, o endereçamento e a detecção de erros de bits; ainda realiza controle de fluxo;
- **camada de Rede:** Provê serviços de roteamento, controle de congestionamento e endereçamento;
- **camada de Transporte:** Provê serviços de entrega de dados ordenada e livre de erros e entrega de dados sem garantia de ordenação;
- **camada de Aplicação:** Provê aplicações de rede, como correio eletrônico e transferência de arquivos.

Os planos de gerenciamento são:

- **plano de Gerenciamento de Energia:** Define de que forma os nós podem gerenciar sua energia. Por exemplo, desligando seu receptor após receber mensagens de seus vizinhos;
- **plano de Gerenciamento de Mobilidade:** Este plano detecta e registra o movimento dos nós sensores. Desta forma, um nó irá conhecer a “posição” de seus vizinhos, mesmo que algum deles, ou o próprio nó, se movam;
- **plano de Gerenciamento de Tarefas:** Este plano gerencia as tarefas de cada nó em uma dada região; por exemplo, definindo que nem todos os nós de uma região monitorada devem realizar sensoriamento simultaneamente.

Ao utilizarem estes planos de gerenciamento, a rede e os nós podem trabalhar de forma colaborativa, já que podem, por exemplo, economizar energia, realizar roteamento em redes móveis e compartilhar recursos entre eles. Com isso, podem realizar suas tarefas com maior eficiência.

A próxima seção apresenta alguns aspectos de uma tecnologia de rede que, nos últimos anos, se tornou popular para RSSFs [22].

## 1.2 ZigBee

As abelhas (do inglês, *bees*) possuem um interessante método de comunicação. Ao encontrarem comida, elas dançam em zigue-zague para compartilhar informações de posição, distância e direção do alimento [23]. Esta peculiar característica desses insetos deu origem ao nome da tecnologia conhecida como ZigBee, formada a partir da união de dois grupos: a *ZigBee Alliance* [24] e o IEEE 802.15.4 [23].

A *ZigBee Alliance* é formada por grandes empresas, como Motorola, Phillips e Samsung, e seu objetivo é definir produtos de controle e monitoramento de baixo custo, baixo consumo e com a capacidade de comunicação sem fio.

Já o 802.15.4 é um grupo de IEEE que define padrões para as camadas Física e de Acesso ao Meio para redes pessoais sem fio (WPANs), cujas características principais são: baixa taxa de transmissão, baixa complexidade e necessidade de baixo consumo de energia [22].

### 1.2.1 A pilha de protocolos

A pilha de protocolos da arquitetura de rede ZigBee pode ser vista na Figura 5, onde a camada Física e a subcamada de Acesso ao Meio (MAC) foram definidas pelo grupo IEEE 802.15.4, e a *ZigBee Alliance* realizou a padronização da camada de Rede (NWK) e o arcabouço para a camada de Aplicação, que inclui uma subcamada de Suporte à Aplicação (APS), a definição dos Objetos dos Dispositivos ZigBee (ZDO) (porção da aplicação que define as funções dos dispositivos da rede), além de prover serviços de segurança [24].

A camada Física possibilita que os nós trabalhem em três bandas. A Europa e a América do Norte utilizam, respectivamente, as bandas 868 - 868,6 MHz e 902 - 928 MHz. Já o resto do mundo utiliza a banda 2,4 - 2,4835 GHz [25]. Segundo [22], as primeiras duas bandas trabalham com modulação BPSK. Na banda de 868 - 868,6 MHz, se utiliza uma taxa de transmissão de 20 kbps e há um canal disponível. Na banda de 902 - 928 MHz, se utiliza uma taxa de 40 kbps e há 10 canais. Já a banda de 2,4 GHz utiliza modulação O-QPSK, com 250 kbps de taxa e 16 canais disponíveis. Por fim, no padrão IEEE 802.15.4 utiliza-se a técnica de transmissão DSSS (Espalhamento Espectral por Sequência Direta) junto com a modulação.

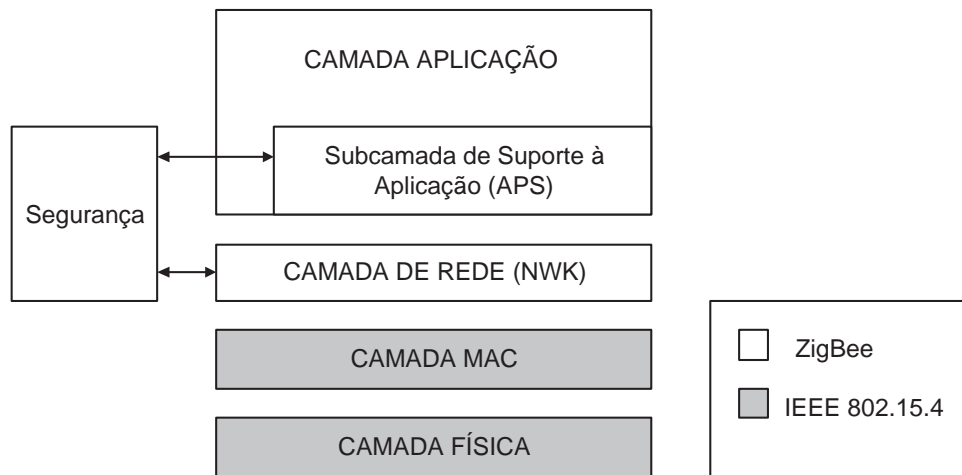


Figura 5 - Pilha de protocolos ZigBee.

Na Subcamada de Acesso ao Meio (MAC), pode-se usar o ARQ (*Automatic Repeat reQuest*), que é a retransmissão de quadros não reconhecidos. O formato do quadro de dados é apresentado na Figura 6, e possui os seguintes campos: Controle de Quadro (de 2 Bytes), que possui informações sobre o tipo de quadro, o formato do campo de endereço e o controle do uso de ACK; Número de sequência, de 1 Byte; Endereçamento (de 4 a 20 Bytes), que possui um identificador da rede destino, o endereço de destino, um identificador da rede fonte e o endereço de origem; Carga útil, com um tamanho máximo de 104 Bytes e CRC, para detecção de erros, com 2 Bytes.

Em relação ao acesso compartilhado ao meio, existem duas possibilidades: uma com uso de *beacon* e outra sem o uso de *beacon*. Sem o *beacon*, utiliza-se o mecanismo CSMA-CA (*Carrier Sense Multiple Access with Collision Avoidance*) [21]. No CSMA-CA, cada estação que deseja realizar uma transmissão espera um período de tempo aleatório. Após esse período de tempo, a estação escuta o canal. Caso o canal esteja livre, a estação pode transmitir. Caso contrário, a estação espera novamente por um período aleatório de tempo. Com o *beacon*, utiliza-se uma estrutura de *superquadro*, ilustrado na Figura 7. Os *beacons* são usados para sincronizar os dispositivos associados, para identificar a rede e para descrever a estrutura do *superquadro* [26]. O *superquadro* é composto por uma parte ativa e uma parte inativa. Na parte inativa, os dispositivos da rede podem entrar em um estado de economia de energia. A parte ativa é composta por 16 períodos fixos de tempo (*slots*), e é dividida em um período de acesso de contenção (CAP - *Contention Access Period*) e um período livre de contenção (CFP - *Contention Free Period*). O CAP deve

conter entre 8 e 16 *slots* e, como há acesso com contenção, utiliza-se o CSMA-CA. No CFP, não é utilizado o CSMA-CA. Em um CFP, o coordenador reserva um GTS (*Guaranteed Time Slot*) [25], mecanismo de garantia de alocação de *slots* de tempo. Durante um GTS, um dispositivo possui acesso exclusivo ao canal. Além disso, o comprimento de um GTS é um múltiplo inteiro de um *slot* de tempo.

A camada de Rede possui as seguintes atribuições: gerenciamento de entrada e saída de dispositivos na rede e provimento de roteamento aos quadros de dados até seus respectivos destinos. Também realiza a descoberta e manutenção de rotas entre dispositivos, descoberta dos vizinhos dentro da área de um salto, a inicialização da rede e é responsável pelo endereçamento dos dispositivos associados na rede.

Na camada Aplicação, a subcamada APS é responsável pela comunicação entre a camada de rede e a camada de aplicação, através de alguns serviços como, por exemplo, a conexão entre dois dispositivos, baseado em seus serviços e necessidades. Dois dispositivos que realizam os mesmos serviços podem ser conectados, e a subcamada APS pode transferir as mensagens que um deles receber diretamente para o outro [25].

As responsabilidades do ZDO incluem a definição da função dos dispositivos na rede, a descoberta de dispositivos na rede e a determinação de quais serviços de aplicação os dispositivos irão prover.

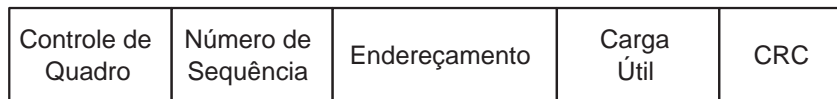


Figura 6 - Estrutura do quadro de dados.

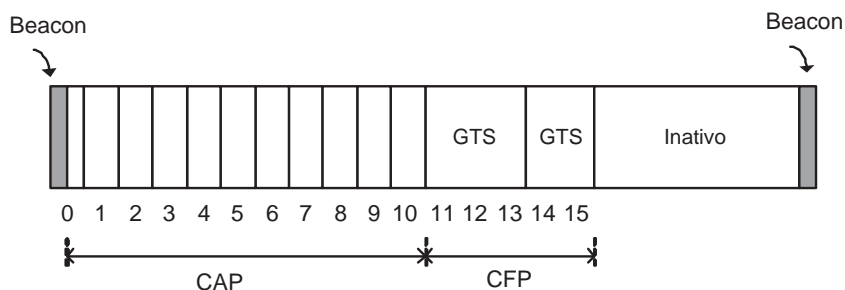


Figura 7 - Estrutura do *superquadro*.

### 1.2.2 Componentes de uma rede ZigBee

O padrão ZigBee [25] [24] [27] define dois tipos de dispositivos básicos, o Dispositivo de Função Completa (FFD) e o Dispositivo de Função Reduzida (RFD). Um FFD pode se comunicar tanto com outro FFD, quanto com um RFD. Já um RFD pode se comunicar apenas com um FFD. Uma rede deve possuir pelo menos um FFD, e neste caso, este possui a função de coordenador da rede. Por exemplo, em uma RSSF o sorvedouro pode ser o FFD, atuando como o coordenador da rede, e os nós sensores, RFDs.

### 1.2.3 Topologias de uma rede ZigBee

A arquitetura ZigBee suporta dois tipos de topologia, dependendo das características da aplicação que estiver em funcionamento, as topologias em estrela e par-a-par (P2P).

#### 1.2.3.1 Topologia em estrela

Nesta topologia, Figura 8 (a), a rede ZigBee é controlada por um elemento central, o Coordenador da rede. Este é responsável pela inicialização da rede e pelo controle dos dispositivos (nós) da rede, os quais se comunicam diretamente apenas com ele. Algumas aplicações que utilizam esta topologia são automação residencial, aplicações para periféricos de computadores, jogos e o monitoramento de sinais vitais de pacientes [25]. Como, na topologia em estrela, cada nó deve se comunicar diretamente com o coordenador, existe uma limitação de um salto para a comunicação.

#### 1.2.3.2 Topologia P2P

A Figura 8 (b) representa a topologia P2P. Diferentemente da topologia em estrela, nesta topologia não há a necessidade de um coordenador. Nela, quaisquer dispositivos (FFDs) podem comunicar-se entre si, desde que eles estejam dentro do raio de alcance um do outro. Uma rede P2P pode ser *ad hoc* e auto-organizável [4], e permite a comunicação em múltiplos saltos para rotear mensagens de um nó para qualquer outro na rede. Isso possibilita aplicações de Redes de Sensores Sem Fios, para monitoramento e coleta de dados em áreas maiores [25] [3] [5]. Em uma rede que utiliza comunicação com múltiplos

saltos, uma mensagem pode ser encaminhada através de diversos nós (roteadores) até chegar a um nó destino. Com isso, a distância entre os nós origem e destino pode ser aumentada, desde que haja nós suficientes para o encaminhamento das mensagens.

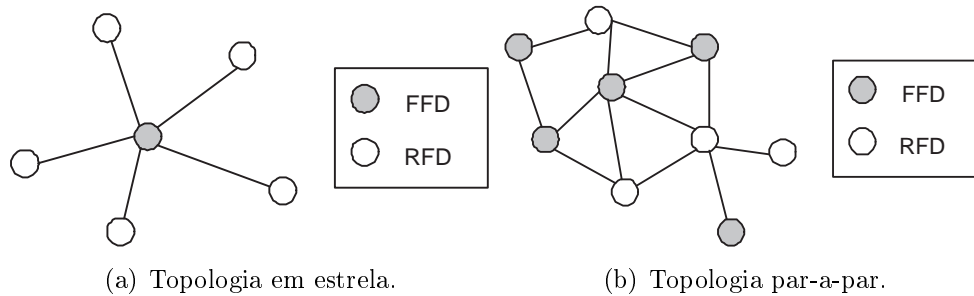


Figura 8 - Topologias de redes ZigBee.

### 1.3 Conclusão

Neste capítulo as Redes de Sensores Sem Fio (RSSFs) foram apresentadas. Seus principais aspectos e as características de seus nós foram descritos. Além disso, o padrão ZigBee foi apresentado, incluindo sua pilha de protocolos, seus componentes e os tipos de topologia. A partir das características da rede e de seus nós (apresentadas neste capítulo), pode-se representar o consumo de energia de cada nó da rede e definir métodos para economizar sua energia, o que será visto no próximo capítulo.

## 2 CONSUMO DE ENERGIA EM RSSFs

Este capítulo trata do consumo de energia pelos nós de uma RSSF. Em primeiro lugar, dois modelos de energia para redes *ad hoc* serão apresentados. Em seguida, será definido o modelo de energia utilizado nesta Dissertação. A partir do modelo de energia utilizado, pode-se caracterizar os métodos de economia de energia para RSSFs encontrados na literatura.

A modelagem do consumo de energia de um nó é importante, pois, a partir deste modelo, pode-se definir como economizar energia e, com isso, talvez aumentar a autonomia da rede.

### 2.1 Modelos de Energia

Um modelo de energia descreve como a energia de cada nó da rede é consumida no decorrer do tempo. Segundo [13], os modelos de energia para redes *ad hoc* podem ser divididos em dois grupos, os modelos de consumo por pacotes e os modelos de consumo por estados. No primeiro, a energia do nó é consumida (decrementada) a cada pacote recebido ou transmitido. Já no segundo grupo, a energia de cada nó é consumida em função do tempo durante o qual um nó permanece em um dado estado.

#### 2.1.1 Modelos baseados em pacotes

Nestes modelos, o consumo de energia é diretamente proporcional ao tamanho do pacote transmitido ou recebido; ou seja, tal consumo, chamado de  $E_c$  pode ser modelado como uma função linear do tamanho do pacote em Bytes, desta forma:

$$E_C = m \times tamanho + b. \quad (1)$$

Os parâmetros  $m$  e  $b$  são constantes em  $\mu\text{J}$ . Observa-se, na Equação (1), que a energia consumida na transmissão ou recepção de um pacote de dados possui duas parcelas, uma dependente do tamanho do pacote e outra independente (fixa). Isto significa que transmitir vários pacotes pequenos é menos eficiente que transmitir um pacote maior com informação equivalente [13].



### 2.1.2 Modelos baseados em estados

Nestes modelos, considera-se que cada nó da rede pode estar em um estado, definido pelo seu modo de operação atual, como um modo de transmissão ou de recepção. Ao permanecer em um dado estado (modo de operação) consome-se uma determinada quantidade de energia, que depende do estado e do tempo nesse estado. Esta energia consumida é função do estado atual do nó e do tempo de permanência nesse estado.

Em [13] é apresentado um modelo com quatro estados: Dormência, Ocioso, Tx e Rx, conforme a Figura 9. Com base nestes estados, pode-se estimar a energia  $E_m$  consumida, função do tempo em que um dado nó permanece em cada estado, conforme a Equação (2).

$$E_C = t_d \times P_d + t_{oc} \times P_{oc} + t_{rx} \times P_{rx} + t_{tx} \times P_{tx}. \quad (2)$$

Na Equação (2),  $t_d$ ,  $t_{oc}$ ,  $t_{rx}$  e  $t_{tx}$  são as parcelas de tempo em que os nós permanecem em cada estado, Dormência, Ocioso, Rx e Tx, respectivamente, e  $P_d$ ,  $P_{oc}$ ,  $P_{rx}$  e  $P_{tx}$  são as potências médias consumidas em cada estado, por unidade de tempo.

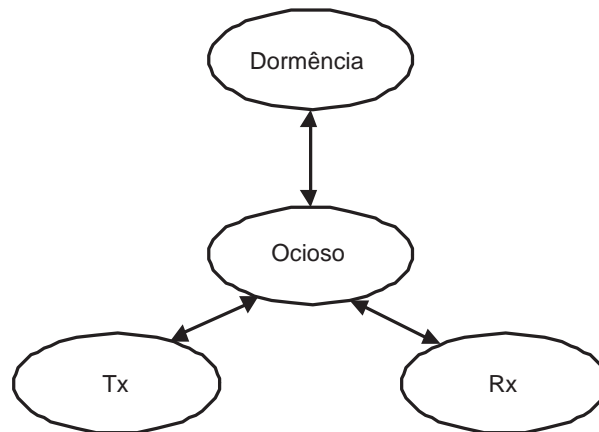


Figura 9 - Diagrama do modelo de energia baseado em estados.

É importante observar que outros modelos podem ser obtidos a partir deste (de quatro estados), com a inclusão de novos estados e/ou subestados, como por exemplo no modelo de energia utilizado nesta Dissertação, que será devidamente apresentado a seguir.

## 2.2 Modelo de Energia Utilizado nesta Dissertação

O modelo de energia utilizado nesta Dissertação pode ser considerado como híbrido. Além de empregar estados, considera também os tamanhos dos pacotes. O modelo é uma variante do modelo de quatro estados utilizado em [13], onde considera-se dois estados principais: o estado Inativo e o estado Ativo. O último é um macro-estado que engloba quatro subestados: Medição, Processamento, Transmissão e Recepção. Os dois últimos já foram considerados em [13]. O que motiva esta abordagem é a utilização de um modelo mais realista, onde um nó, além de realizar transmissões e recepções, e dormir, também consome energia quando processa dados ou quando realiza uma medição. Uma breve descrição de cada estado é apresentada a seguir:

1. **Ativo:** Neste estado os nós sensores estão ligados, não necessariamente realizando alguma tarefa. Este possui quatro modos de operação (subestados);
  - **medição:** O nó mede grandezas, como temperatura ou umidade do ambiente monitorado;
  - **processamento:** O nó pode processar informações medidas ou recebidas;
  - **transmissão:** O nó, neste estado, realiza transmissões;
  - **recepção:** O nó, operando neste modo, recebe dados advindos de algum outro nó.
2. **Inativo:** Modo de economia de energia. Neste estado, os nós estão dormindo (com o transceptor desligado) durante um intervalo de tempo. Após este intervalo, eles voltam ao estado Ativo e voltam a realizar suas tarefas.

Desta forma, cada nó pode estar associado a dois estados principais: Ativo ou Inativo. Quando Ativo, cada nó pode medir, processar, transmitir ou receber; ou ainda pode economizar energia, ao passar para o estado Inativo. A Figura 10 apresenta um diagrama ilustrativo dos estados existentes neste modelo de energia.

Em relação aos consumos de energia dos nós sensores, o modelo desta Dissertação se baseia em [1], que propõe um modelo empírico, obtido utilizando o *hardware* comercial TELOS [28]. Em [28] foi observado que, no estado de Transmissão, o consumo de energia e o tamanho da carga útil (*payload*), em Bytes, de um quadro de dados estão linearmente

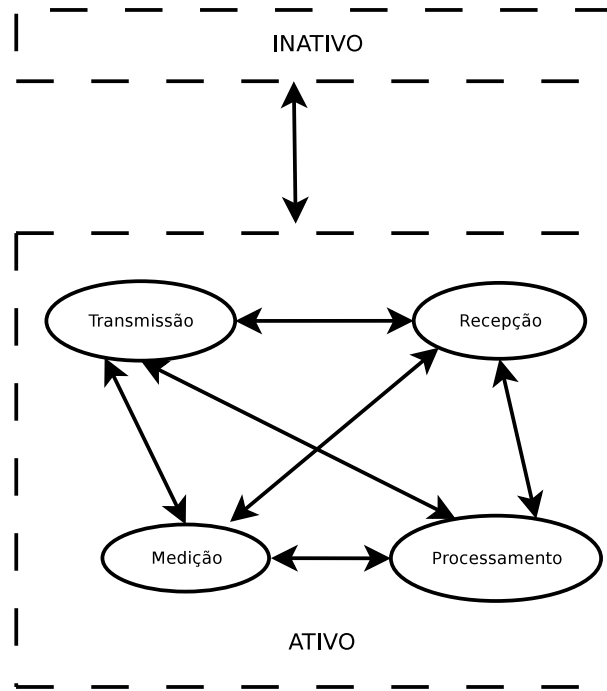


Figura 10 - Diagrama do modelo de energia utilizado nesta Dissertação.

relacionados. Essa relação pode ser vista na Figura 11. Já no estado de Recepção, verificou-se que o consumo de energia é independente do tamanho da carga útil do quadro de dados, assim como no estado Inativo.

Além disso, em [1], verificou-se que o consumo, na transmissão de 1 Byte de carga útil, é de 0,034 mJ, para uma potência de transmissão de -5 dBm. Observou-se também que esta transmissão leva cerca de 0,58 ms. Já os consumos nos estados de Recepção e Inativo, para cada um dos estados, são de 62,4 mJ/s e 1,8 mJ/s, respectivamente.

Nesta Dissertação, um custo adicional para o estado Ativo foi incluído. A utilização deste consumo é válida, pois os nós podem estar Ativos, porém sem realizar tarefas. Foi realizada a implementação deste modelo de energia proposto no ambiente de simulação usado (discutido na Seção 4.3), através da inclusão dos respectivos estados e subestados, cada qual com seu consumo de energia devidamente ajustado.

Pode-se estimar a energia consumida  $E_C$  em um nó, em função dos períodos de tempo em que o nó permanece em cada estado, segundo a equação:

$$E_C = t_I \times C_I + t_A \times C_A + t_M \times (C_A + C_M) + t_P \times (C_A + C_P) + t_R \times (C_A + C_R) + t_T \times (C_A + C_T). \quad (3)$$

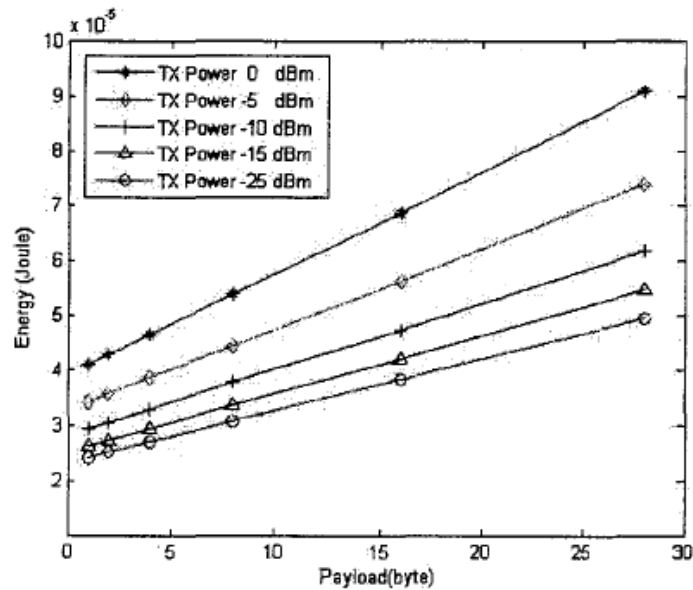


Figura 11 - Consumo no estado de Transmissão para o *hardware* TELOS [Fonte [1]].

Na Equação (3),  $t_I$ ,  $t_A$ ,  $t_M$ ,  $t_P$ ,  $t_R$  e  $t_T$  são, respectivamente, quanto tempo o nó permanece nos estados inativo, ativo, medição, processamento, recepção e transmissão. Além disso,  $C_I$ ,  $C_A$ ,  $C_M$ ,  $C_P$ ,  $C_R$  e  $C_T$  são os respectivos consumos nesses estados, por unidade de tempo. Pode-se observar que, quando um nó permanece nos estados de medição, processamento, recepção ou transmissão, os consumos relativos a cada um desses estados são somados ao consumo do estado ativo. Para que um nó realize uma transmissão, por exemplo, é necessário que ele esteja ativo. O mesmo ocorre para os outros estados. Portanto, há a soma do consumo de energia do estado ativo com o consumo de cada um dos quatro subestados.

### 2.3 Métodos de Economia de Energia

Os nós sensores de uma RSSF são alimentados por uma bateria. Além disso, dependendo da aplicação, RSSFs podem ser utilizadas em locais de difícil acesso [4], o que faz com que a troca manual de uma bateria se torne inviável. Logo, uma das questões importantes a respeito de RSSFs é a energia gasta pelos nós sensores; ou melhor, de que forma estes podem economizar energia e, portanto, aumentar a autonomia da rede.

A economia de energia dos nós sensores está diretamente relacionada com o conceito

de Tempo de Vida da rede. Este pode ser entendido como o tempo decorrido até a “morte” do primeiro nó sensor da rede, ou seja, o tempo em que sua bateria acaba [29] [30]. Portanto, mecanismos que permitam economizar a energia dos nós sensores e, com isso, aumentar o Tempo de Vida da rede, devem ser utilizados para aumentar a sua autonomia.

À primeira vista, uma maneira simples de economia de energia pelos nós sensores seria a utilização (ou o aproveitamento) da energia obtida do meio externo. Um exemplo é a utilização de células solares como fonte de energia [31]. Porém, a utilização das RSSFs em locais como mata fechada, cavernas e no interior de prédios ou casas faz com que esta fonte energética não esteja disponível em qualquer situação.

De maneira geral, a tarefa mais custosa, em termos energéticos, de uma RSSF é a comunicação (transmissão/recepção), sendo seu consumo de energia muito maior do que aquele gasto no processamento dos dados [2] [7]. Verifica-se [8] que a energia gasta na transmissão de um único *bit* de informação é aproximadamente a mesma necessária para realizar milhares de operações em um nó sensor.

De posse destas premissas, o restante deste capítulo apresenta uma classificação de métodos específicos de economia de energia em RSSFs encontrada na literatura. Esta Dissertação segue a classificação utilizada em [32], sendo ela dividida em três grupos de métodos: os baseados no ciclo de trabalho, os baseados nos dados e os baseados na mobilidade, conforme a Figura 12.

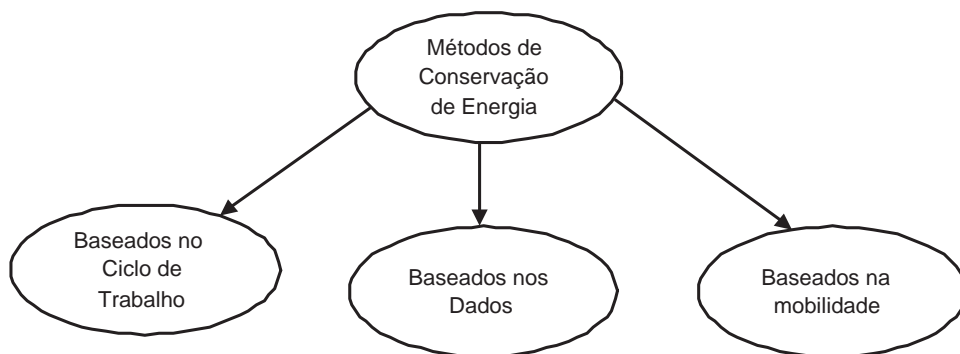


Figura 12 - Taxonomia dos métodos de economia de energia.

## 2.4 Métodos de Economia de Energia Baseados no Ciclo de Trabalho

Seja a função  $p(t)$ , representada pela Figura 13, cujo período é igual a  $T$  segundos. Suponha que, em um período, o pulso esteja ativo por  $T_a$  segundos. Pode-se dizer que o

ciclo de trabalho ( $CT$ ) de  $p(t)$  é a parcela do período em que o pulso está ativo, ou seja:

$$CT = \frac{T_a}{T} \cdot 100\% \quad (4)$$

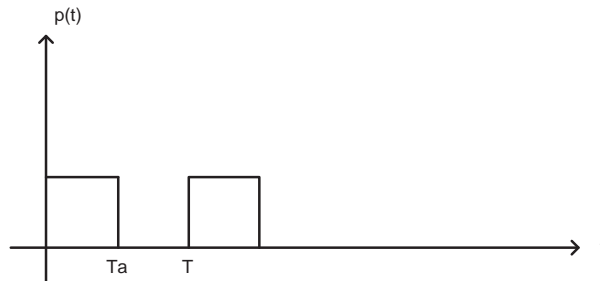


Figura 13 - Definição de ciclo de trabalho.

Os métodos de economia de energia baseados no ciclo de trabalho seguem este princípio: de que os nós sensores podem não ficar ativos durante todo o tempo, mas apenas durante seu ciclo de trabalho. Uma maneira de se economizar energia, utilizando esta idéia, é fazer com que um nó transite para um estado de economia de energia (*sleep mode*), quando estiver no período inativo [11]; ou seja, o ciclo de trabalho dos nós é dividido em duas etapas: Dormir e Acordar. Neste tipo de metodologia, algoritmos são desenvolvidos para gerenciar este escalonamento Dormir/Acordar.

Os métodos baseados no ciclo de trabalho podem ser divididos em duas subclasses, vide Figura 14. Segundo [32], o termo Controle de Topologia se refere a encontrar um subgrupo de nós ótimo (ativos), de modo a garantir a conectividade da rede. Por outro lado, os nós que não precisam estar ativos podem entrar em um estado de economia de energia, e este escalonamento entre o dormir e o acordar dos nós é chamado de Gerenciamento de Energia. Portanto, os dois métodos baseados no ciclo de trabalho são complementares, ambos implementam o ciclo de trabalho de maneiras distintas. Nesta Dissertação, o método de economia de energia empregado é baseado no ciclo de trabalho, na medida em que os nós utilizam o escalonamento Dormir/Acordar para economia de energia.

#### 2.4.1 Controle de topologia

Os protocolos de Controle de Topologia têm por objetivo selecionar um subgrupo de nós que permanecerão ativos na rede. Por outro lado, os nós que não são necessários podem economizar energia, dormindo. Isto é possível, principalmente quando houver alta



Figura 14 - Métodos baseados no ciclo de trabalho.

densidade de nós em uma RSSF [32]. Desta forma, estes protocolos garantem a operação da rede, enquanto minimizam a quantidade de nós ativos. Assim, tendem a prolongar o Tempo de Vida da rede.

Os protocolos de controle de topologia são divididos em: protocolos baseados na localização e protocolos baseados na conectividade, o que pode ser visto na Figura 15.

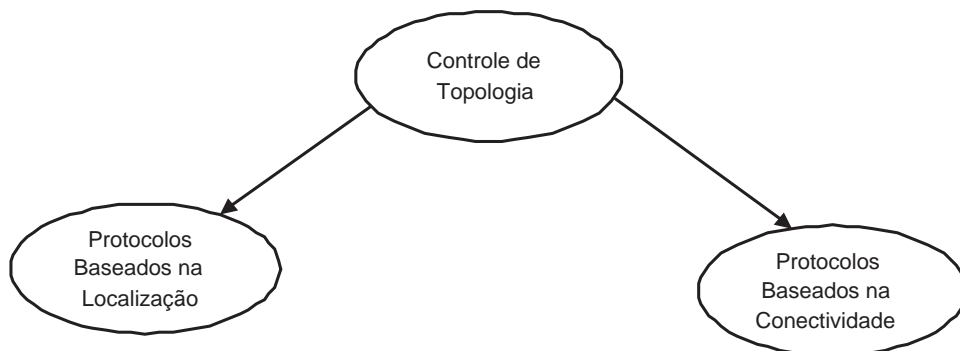


Figura 15 - Protocolos de controle de topologia.

#### 2.4.1.1 Protocolos baseados em localização

Estes definem os nós que estarão ativos e quando isto irá ocorrer, baseados na localização de cada nó sensor (assumida como conhecida), e tem como exemplos:

- *Geographical Adaptive Fidelity* (GAF) [33], um protocolo em que a área monitorada é dividida em pequenas grades virtuais. Os nós dentro de uma mesma grade são considerados equivalentes (para o roteamento) e apenas um deles permanece ativo para realizar esta tarefa;

- *Geographic Random Forwarding* (GeRaF) é um protocolo que explora a localização, e também o ciclo de trabalho, para economizar energia. Os nós seguem um dado ciclo de trabalho para alternar entre os estados Ativo e Inativo. Além disso, dependendo da área de atuação, um esquema de prioridade é utilizado para definir quais nós estarão ativos em um período de tempo [34].

Uma restrição deste tipo dos Protocolos baseados em localização é a necessidade de se conhecer (*a priori*) a localização dos nós. Isto tende a limitar a aplicabilidade destes protocolos.

#### 2.4.1.2 Protocolos baseados na conectividade

Os protocolos baseados na conectividade ativam e desativam os nós da rede, garantindo que a conectividade da rede se mantenha estável, como em:

- *Span* [35], um protocolo que, adaptativamente, elege coordenadores para a rede. Estes, permanecem Ativos e realizam roteamento das mensagens transmitidas. Os outros nós dormem e, periodicamente, verificam se há necessidade de acordar e se tornar um coordenador;
- *Adaptative Self-Configuring sEnsor Networks Topologies* (ASCENT) [36] que, diferentemente do Span, não depende de algum protocolo de roteamento. No ASCENT, um nó decide o tempo de inatividade baseado em informações sobre a conectividade da rede e perdas de pacotes, que são verificadas localmente pelo próprio nó.

A questão da conectividade é considerada no método de economia proposto nesta Dissertação, para RSSFs de múltiplos saltos.

#### 2.4.2 Gerenciamento de energia

Os protocolos baseados no gerenciamento de energia podem ser implementados de duas formas, uma independente da camada MAC (geralmente na camada Rede ou Aplicação) e outra integrada com os protocolos específicos da camada MAC, vide Figura 16.

##### 1. Protocolos Dormir/Acordar



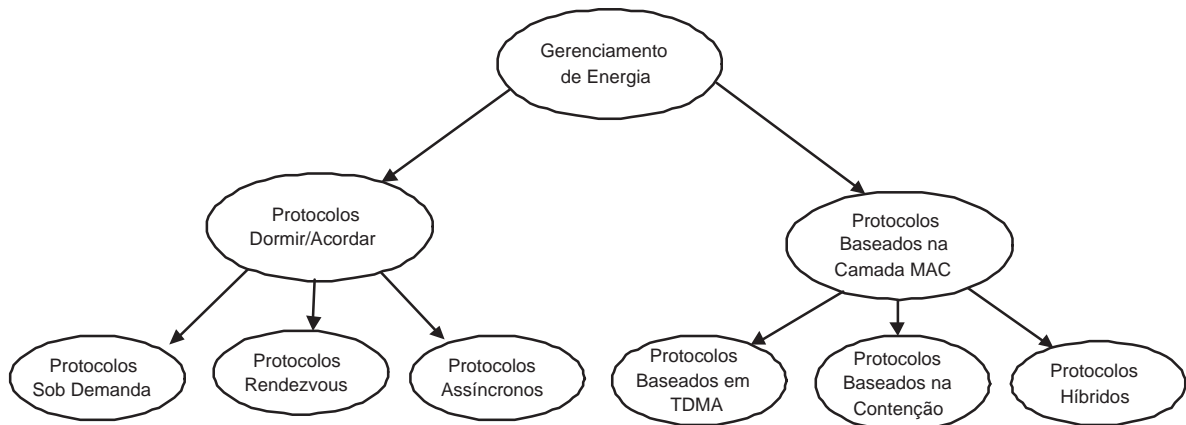


Figura 16 - Métodos baseados no gerenciamento de energia.

Os chamados Protocolos Dormir/Acordar, que são independentes dos protocolos da Camada de Acesso ao Meio são divididos em três categorias: sob demanda, *rendezvous* e assíncronos.

(a) Protocolos Sob Demanda

Nesta categoria, utiliza-se a idéia de que um nó deve acordar apenas quando outro nó deseja se comunicar com ele. Porém, não se sabe *a priori* de que forma um nó Inativo (dormindo)  $S_i$  pode saber quando outro nó  $S_j$  deseja se comunicar com ele. Uma solução é utilizar múltiplos rádios nos nós sensores, como em [37]. Outras opções vistas na literatura que usam protocolos Dormir/Acordar sob demanda são:

- *Sparse Topology and Energy Management* (STEM) [38], que usa dois rádios, um para verificar quando acordar e o outro para transmissão de dados. Com isso, através de um sinal (tom de atividade), os nós podem acordar seus vizinhos;
- *Pipeline Tone Wakeup* (PTW). A diferença entre este protocolo e o STEM, é que um nó transmissor apenas envia um tom de atividade quando um evento é detectado, enquanto o receptor acorda periodicamente [39].

(b) Protocolos *Rendezvous*

Nos protocolos *Rendezvous*, cada nó deve acordar junto com seus vizinhos. Os nós vizinhos, então, permanecem acordados durante um determinado período para poderem se comunicar. Uma vantagem é que, quando um nó acorda,

existe a garantia de que seus vizinhos também estarão acordados. Por outro lado, isto requer que seus relógios estejam devidamente sincronizados, o que não é trivial. Um exemplo da utilização de protocolos *Rendezvous* pode ser visto em [40].

(c) Protocolos Assíncronos

Por fim, nos protocolos assíncronos um nó pode acordar de forma independente de seus vizinhos. Desta forma, não há restrições quanto ao sincronismo dos relógios dos nós, o que poderia limitar a aplicação utilizada. Um exemplo pode ser visto em [41], onde os nós podem acordar de forma aleatória. Outros dois exemplos da utilização de protocolos assíncronos podem ser vistos em [42], [43]. Essa característica assíncrona é utilizada nos protocolos propostos nesta Dissertação.

## 2. Protocolos MAC

Os protocolos Dormir/Acordar que são integrados à Camada de Acesso ao Meio implementam um esquema de baixo ciclo de trabalho para o gerenciamento de energia. Estes podem ser divididos em três tipos: baseados em TDMA, baseados em Contenção e Híbridos.

(a) Protocolos baseados em TDMA

No Acesso Múltiplo por Divisão de Tempo (TDMA) [44], o tempo é dividido em parcelas, chamadas de *slots*. A partir daí, cada nó  $S_i$  pode acessar o meio durante uma ou  $n$  parcelas do tempo total, previamente definidas.

Nestes protocolos, os nós permanecem ativos durante suas parcelas de tempo, economizando energia no restante do tempo, como em:

- TRAMA (*T*Raffic-*A*daptative *M*edium *A*ccess protocol), um protocolo que divide o tempo em duas porções, cada qual formada por *slots*. Na primeira porção, o acesso aos *slots* de tempo é feito através da disputa pelo meio. Na porção seguinte, existem *slots* reservados para cada nó [45];
- FLAMA (*F*low-*A*ware *M*edium *A*ccess) é um protocolo derivado do TRAMA, e é utilizado em aplicações para monitoramento periódico. A idéia é utilizar fluxos periódicos, onde os dados apenas são transferidos quando solicitados. Com isso, pretende-se minimizar a sobrecarga do tráfego [46].

## (b) Protocolos baseados na Contenção

Tais protocolos integram o escalonamento Dormir/Acordar com um período de disputa pelo meio. Neste período, métodos de acesso ao meio compartilhado, como o CSMA, são utilizados, para que os nós não realizem transmissões ao mesmo tempo, o que causaria colisão das mensagens. Alguns exemplos são:

- B-MAC (*Berkeley MAC*) é um protocolo que possui um mecanismo de acesso ao meio eficiente (energeticamente), através da implementação de um controle de acesso ao meio (com *backoff*, estimação de canal e confirmações opcionais) integrado a um esquema de Dormir/Acordar assíncrono [47]. Outras variantes são: S-MAC (*Sensor-MAC*) [48] e D-MAC (*Data Gathering MAC*) [49];
- o padrão IEEE 802.15.4 [25], que define a camada física e a subcamada de acesso ao meio para redes que utilizam dispositivos com restrições de energia e com baixa taxa de transmissão, como as RSSFs. Na sua camada MAC (camada de acesso ao meio), o IEEE 802.15.4 pode utilizar dois tipos de acesso ao meio compartilhado: um período de contenção, onde os nós utilizam métodos de acesso ao meio compartilhado (no caso, o CSMA-CA); e um período livre de contenção, onde cada nó pode acessar ao meio por uma parcela de tempo predefinida.

## (c) Protocolos Híbridos

Estes protocolos se adaptam de acordo com o nível de contenção da rede, ou seja, se há mais ou menos disputa pelo meio. Caso este nível seja baixo (com uma disputa menor), o protocolo se comporta como um protocolo baseado em Contenção. Com uma menor disputa pelo meio, há uma menor probabilidade de haver colisões. Caso contrário (quando há mais disputa pelo meio), o protocolo se comporta como, por exemplo, um esquema TDMA, onde cada nó possui sua parcela de tempo para transmitir. Dois protocolos Híbridos encontrados na literatura são:

- PTDMA (*Probabilistic TDMA*) [50], que usa um esquema probabilístico para ajustar o protocolo MAC para os esquemas TDMA ou CSMA, de acordo com o nível de contenção da rede;

- Z-MAC (*Zebra MAC*) [51], que possui o objetivo de garantir que as transmissões de um nó para seus vizinhos de um salto não irão interferir nas transmissões para seus vizinhos de dois saltos. No Z-MAC, a distribuição de *slots* de tempo é feita de maneira que dois nós (vizinhos de dois saltos) não utilizem o mesmo *slot*.

## 2.5 Métodos de Economia de Energia Baseados nos Dados

Quando se realiza o monitoramento de alguma região com RSSFs, grandezas como temperatura, pressão ou umidade são medidas pelos nós sensores da rede. Tais amostras, geralmente possuem grande correlação espacial e/ou temporal [10]. A Figura 17 ilustra uma rede com alguns nós, onde os índices  $S_i$  (ou  $S_j$ ) representam os nós, as grandezas medidas pelos nós da rede são representadas por  $y_i$  (ou  $y_j$ ) e a distância entre dois nós  $S_i$  e  $S_j$  é representada por  $d_{ij}$ . Nessa figura, os nós sensores  $S_i$  estão medindo as grandezas  $y_i$ . Portanto o sensor  $S_i$  mede uma amostra  $y_i(t)$ , onde  $t$  representa o instante de tempo em que a medida é realizada. Além disso,  $d_{ij}$  é a distância entre os nós  $S_i$  e  $S_j$ . A correlação espacial diz respeito à similaridade entre amostras medidas por um sensor  $S_i$  e amostras medidas por outro sensor  $S_j$ . Quanto menor a distância entre os nós, a princípio, maior será a similaridade entre  $y_i(t)$  e  $y_j(t)$ . Já a correlação temporal se dá entre amostras consecutivas, de um mesmo sensor ou entre sensores distintos; ou seja, é a redundância entre  $y_i(t)$  e  $y_i(t - \Delta t)$  ou com  $y_j(t - \Delta t)$ .

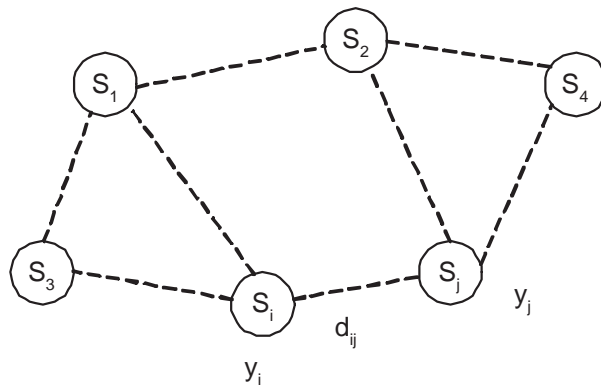


Figura 17 - Distância entre alguns nós e suas medidas.

Segundo [32], amostras correlacionadas significam informações redundantes. Com isso, pode-se conservar energia reduzindo a quantidade de amostras transmitidas; ou seja,

os métodos baseados nos dados conservam a energia dos nós sensores, através do decréscimo da quantidade de transmissões realizadas, observando características espaço-temporais das amostras medidas.

Os métodos baseados nos dados são divididos em duas classes: Redução de Dados (a serem transmitidos) e Aquisição de Dados com Eficiência Energética, que podem ser vistas na Figura 18.

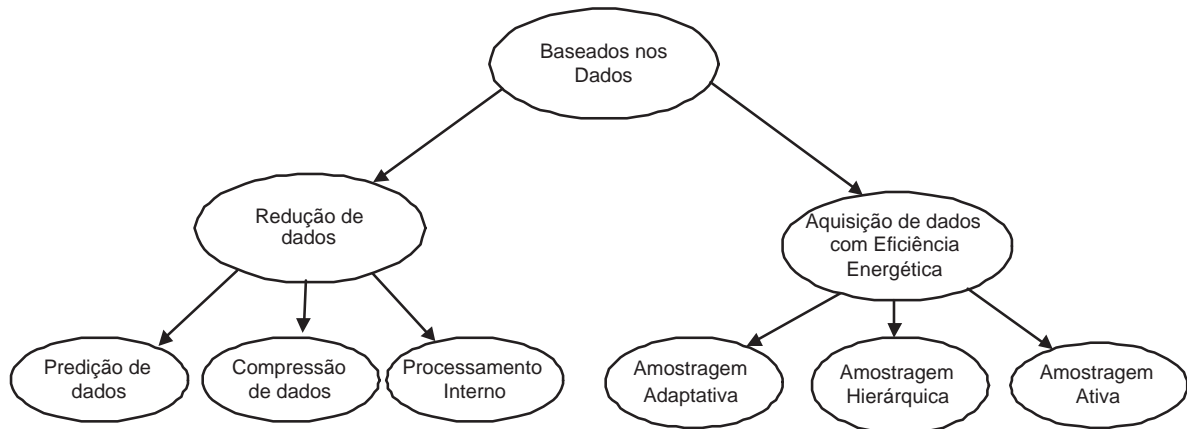


Figura 18 - Métodos baseados nos Dados.

### 2.5.1 Redução de dados

Os métodos baseados na redução de dados podem ser divididos em três subclasses: Predição de Dados, Compressão de Dados e Processamento Interno, conforme a Figura 19, e têm por objetivo reduzir a quantidade de dados a ser entregue ao sorvedouro.

#### 2.5.1.1 Protocolos baseados na Predição de Dados

A ideia destes protocolos é construir modelos do fenômeno/processo sensoreado e, a partir destes modelos, responder questões acerca do processo sem que haja necessidade de medição dos dados.

Nestes protocolos um modelo do processo é construído, a partir das amostras recebidas pelo sorvedouro, advindas de cada nó sensor. Desta forma, o modelo do sorvedouro pode ser utilizado para estimar possíveis transmissões provenientes dos sensores. Por exemplo, caso um sensor esteja dormindo, as amostras não transmitidas por ele, nesse período, podem ser estimadas, a partir das amostras recebidas pelo sorvedouro. Com isso,

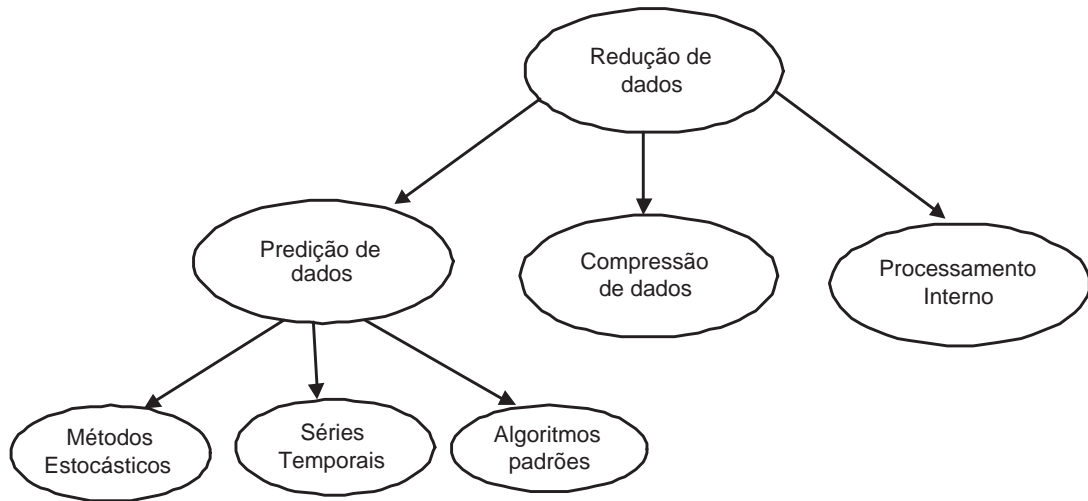


Figura 19 - Métodos baseados na Redução de Dados.

espera-se reduzir a quantidade de transmissões por parte dos nós sensores, economizando energia.

Protocolos que usam Predição de Dados encontrados na literatura são os baseados em Métodos Estocásticos, os baseados em Séries Temporais e os baseados em Algoritmos padrões [32].

#### 1. Protocolos baseados em Métodos Estocásticos

Estas técnicas utilizam caracterizações estatísticas do processo monitorado, por exemplo utilizado a Função Densidade de Probabilidade do processo estocástico para extração de informações. Estas informações estatísticas podem ser usadas para criar um modelo do processo e, com isso, poder reduzir a quantidade de transmissões na rede.

Um exemplo de protocolo baseado em métodos estocásticos é a solução de Ken [52], onde existe uma quantidade de modelos do processo monitorado e, cada um destes é replicado no nó sensor e no servidor. Como cada modelo é probabilístico, após uma fase de treinamento, a função densidade de probabilidade (pdf) do processo pode ser estimada. Além disso, pode-se explorar características do fenômeno modelado, como sua correlação temporal, através de processos de *Markov*.

#### 2. Protocolos baseados em Séries Temporais

Nestes, a predição de amostras futuras é realizada através de um histórico de amostras, utilizando uma Série Temporal, por exemplo com um modelo Auto-

Regressivo [53]. A partir deste histórico, pode-se estimar o comportamento do processo monitorado (ao longo do tempo). Um exemplo de protocolo baseado em séries temporais é o PAQ (*Probabilistic Adaptable Query system*) [54], baseado em um modelo Auto-Regressivo de baixa ordem, que tem por objetivo reduzir o processamento computacional realizado pelos nós sensores.

### 3. Protocolos baseados em Algoritmos padrões

Finalmente, nesta última classe de técnicas de Predição de Dados, os protocolos desenvolvidos são baseados em algoritmos já existentes, e utilizam suas características para gerar métodos de economia de energia. Como exemplo, pode-se citar o PRE-MON (*Prediction-based Monitoring in Sensor Networks*) [55], que é uma técnica de predição inspirada nos conceitos da codificação MPEG.

#### 2.5.1.2 Compressão de dados

Protocolos que usam Compressão de Dados podem ser usados para reduzir a quantidade de informação a ser transmitida pelos nós sensores. Processos como codificação dos dados, nos nós sensores, e a decodificação, no sorvedouro, são envolvidos. Com isso, pode-se reduzir a quantidade de dados a ser transmitida por um nó sensor. Métodos de compressão, utilizados em RSSFs, podem ser vistos em [56] e [57].

#### 2.5.1.3 Processamento interno (na rede)

Esta abordagem utiliza a agregação de dados em nós intermediários entre o nó fonte e o sorvedouro. Com isso, a quantidade de informação é reduzida no percurso interno da rede, o que gera economia de energia [12] [58]. Com o processamento interno, pode-se realizar a fusão de dados em nós intermediários. Por exemplo, se a informação de dois nós forem agregadas em um nó intermediário, pode-se realizar apenas uma transmissão no nó agregador, ao invés de duas. Assim, tende-se a reduzir a quantidade de transmissões pelos nós.

#### 2.5.2 Aquisição de dados com eficiência energética

Estes métodos se baseiam na premissa de que, ao reduzir a quantidade de amostras pelos nós sensores, haverá redução na quantidade de comunicação requerida e, por

consequente, economia de energia. A Figura 18 mostra a subdivisão destes métodos.

### 1. Amostragem adaptativa

O objetivo destes protocolos é conseguir estimar a taxa de amostragem para a medição, de forma adaptativa, a partir das características dos dados medidos. Em [59], os autores propõem um esquema de amostragem adaptativa usado no monitoramento de neve, para previsão de avalanches; em [60], os autores utilizam a variação do processo monitorado para decidir quais nós estarão ativos em uma determinada região; por fim, em [61], criou-se um protocolo colaborativo baseado na correlação entre as amostras medidas, chamado CC-MAC.

### 2. Amostragem Hierárquica

Estes esquemas se utilizam de nós equipados com vários tipos de sensores, cada qual com características específicas, uns com maior acurácia e consumo de energia mais elevado, e outros com consumo menor, porém com menor acurácia. A idéia é, a partir da aplicação, verificar que tipo de sensor utilizar para realizar aquela medida específica. Com isso, pode-se ter um compromisso entre acurácia na medida e consumo de energia. Exemplos podem ser vistos em [62] e em [63].

### 3. Modelos Baseados em Amostragem Ativa

Estes esquemas conseguem reduzir a quantidade de amostras através de modelos computacionais. Alguns exemplos podem ser vistos em:

- O método *Barbie-Q* [64], que gera um modelo probabilístico a partir de uma determinada quantidade de amostras. A partir deste modelo, pode-se utilizar a função densidade de probabilidade para conseguir as correlações espacial e temporal do fenômeno monitorado;
- Outro método que usa Amostragem Ativa é *Priority Aware Situation Awareness* [65]. Este, divide a rede em *clusters* e, dentro de um *cluster* apenas alguns nós realizam medidas de amostras do ambiente monitorado e as enviam para o sorvedouro.

O controle de amostragem das medidas, realizado pelos nós, é utilizado pelos algoritmos propostos nesta Dissertação, já que nem todas as amostras medidas são transmitidas pelos nós sensores.



## 2.6 Métodos de Economia de Energia Baseados na Mobilidade

Segundo [32], recentemente, a mobilidade tem sido considerada um solução alternativa interessante para a economia de energia em RSSFs. Porém, uma das questões intrigantes acerca desta perspectiva é como conceder mobilidade aos nós. Uma das soluções seria equipar apenas uma parcela dos nós com mobilidade, ao invés de equipar todos eles, o que poderia diminuir o custo. Outra idéia seria não prover mobilidade aos nós, mas sim posicioná-los em objetos móveis como carros, ou em animais.

Porém, se faz necessário identificar de que forma pode-se economizar energia utilizando mobilidade. Geralmente, em uma RSSF os pacotes percorrem a rede, da fonte até o sorvedouro, através de um caminho com múltiplos saltos. Quando a posição do sorvedouro é fixa, algumas rotas podem acabar mais carregadas, dependendo da topologia da rede e da taxa de geração de dados nos nós sensores. A utilização de nós móveis pode alterar o fluxo do tráfego, criando rotas alternativas, o que pode auxiliar no balanceamento energético da rede.

Os métodos baseados na Mobilidade podem ser classificados de duas formas, de acordo com a características do nó móvel: Sorvedouro Móvel ou Encaminhador Móvel, o que pode-se ver na Figura 20.

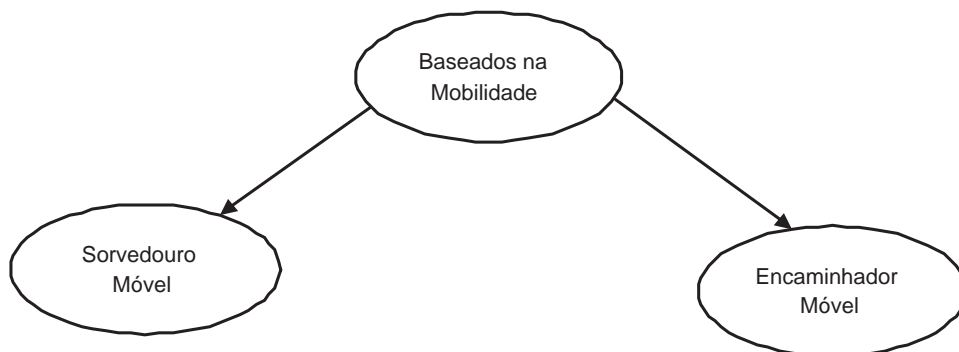


Figura 20 - Métodos baseados em Mobilidade dos nós.

### 2.6.1 Sorvedouro móvel

Como o próprio nome diz, nestes métodos o nó sorvedouro possui mobilidade, e esta peculiaridade é explorada para que haja um balanceamento do consumo de energia na rede e, por conseguinte, o seu Tempo de Vida seja aumentado.

Em [30], propõem-se um modelo onde o sorvedouro pode se mover para uma quantidade limitada de locais para se comunicar com algum sensor. Em [66], múltiplos sorvedouros móveis são utilizados e ganhos de cinco a dez vezes no tempo de vida da rede foram observados, em relação à utilização de um sorvedouro fixo.

### 2.6.2 Encaminhador móvel

Encaminhadores Móveis são nós especiais que são introduzidos na rede para oferecerem serviços de encaminhamento de mensagens [32]. Estes passeiam pela rede e coletam dados que devem ser encaminhados até o destino. Encaminhadores móveis podem ser entendidos como uma infraestrutura de comunicação móvel que tende a acomodar a transferência de dados em uma RSSF. Dois exemplos de estratégias para utilizar encaminhadores móveis podem ser vistas em [67] e [68].

## 2.7 Conclusão

Este capítulo tratou de aspectos sobre o consumo de energia pelos nós em uma RSSF. Foram descritos dois modelos de energia para redes *ad hoc* existentes na literatura e, em seguida, o modelo de energia utilizado nesta Dissertação foi apresentado. A partir de um dado modelo de energia, que representa a forma como a energia de um nó é consumida no decorrer do tempo, pode-se verificar como economizar a energia dos nós da rede. Desta forma, uma taxonomia de métodos de economia de energia foi apresentada.

Os algoritmos propostos nesta dissertação podem ser classificados de duas maneiras, segundo a taxonomia vista na Seção 2.3, pois utilizam características de duas classes de forma colaborativa. Podem ser classificados como algoritmos baseados no escalonamento Dormir/Acordar (Assíncronos), pois utilizam a transição entre os estados Ativo e Inativo para economizar energia. Além disso, são ainda baseados na Amostragem Adaptativa, pois utilizam a variação do processo monitorado para definir quando e por quanto tempo cada nó permanecerá Inativo dado um erro aceitável na medida, além de reduzir a quantidade de transmissões pelos nós sensores. Por fim, os algoritmos ainda consideram a conectividade da rede, e a necessidade de transmissão.

A partir do conhecimento do modelo de consumo de energia empregado pelos nós da rede e do estudo dos principais métodos de economia de energia existentes na literatura,

no próximo capítulo serão apresentados os algoritmos para economia de energia propostos nesta dissertação.

### 3 ALGORITMOS PARA ECONOMIA DE ENERGIA EM RSSFs

Este capítulo apresenta os algoritmos para economia de energia, para RSSFs que utilizam comunicação de salto único e de múltiplos saltos, propostos pelo autor em [14] e em [15], respectivamente. Inicialmente serão apresentados os algoritmos (um baseado na fonte e o outro no sorvedouro) para RSSFs de salto único. Após essa explanação, será apresentado um aprimoramento do algoritmo baseado na fonte, de modo a permitir sua utilização em RSSFs de múltiplos saltos.

#### 3.1 Algoritmos para RSSFs de Salto Único

Nas redes de salto único, o nó sorvedouro está dentro do raio de alcance de cada nó sensor. Um exemplo de aplicação que utiliza comunicação de salto único é a chamada Rede de Sensores Para o Corpo Humano (BSN - *Body Sensor Networks*, do inglês), utilizadas para o monitoramento remoto de pacientes. Nas BSNs, os sensores podem ser dispostos no corpo ou na roupa do paciente, com o objetivo de coletar dados biométricos, como pressão arterial ou saturação de oxigênio. Neste tipo de aplicação, é comum que o sorvedouro (ou estação base) se encontre a poucos metros do paciente, por exemplo em um quarto de hospital [69] [70].

##### 3.1.1 Algoritmo baseado na fonte

Ao considerar uma RSSF como um sistema distribuído (descentralizado), pode-se dizer que esta rede é composta por nós autônomos e cada qual realiza suas tarefas de forma independente. O algoritmo aqui apresentado se baseia nesta premissa e utiliza o conceito de Inovação na fonte, ou seja, em cada nó sensor. Isto significa que a verificação da Inovação ocorre em cada nó sensor.

Inicialmente, cada nó sensor realiza a primeira medida e transmite a amostra para o destino (sorvedouro). Chamaremos as medidas de  $x_i(j)$ , onde  $i$  representa o sensor e  $j$ , a ordem sequencial das medidas. Então, a partir da segunda medida de cada sensor, esta será comparada com a anterior. O sensor então, irá transmitir apenas as amostras para as quais a variação percentual entre a amostra atual e a anteriormente transmitida seja maior do que um limiar prefixado, chamado de limiar de inovação. Assim, o sensor  $i$

apenas transmite uma nova amostra ( $j+n$ ), caso:

$$\frac{|x_i(j+n) - x_i(j)|}{x_i(j)} > \text{limiar}. \quad (5)$$

onde  $x_i(j)$  é a última amostra transmitida pelo sensor.

Logo, apenas amostras com um determinado nível de inovação serão transmitidas. Desta forma, espera-se reduzir a quantidade de transmissões de cada sensor e, com isso, aumentar o tempo de vida da rede.

Além disso, cada nó sensor, de forma autônoma, passa para o estado Inativo após transmissões inovadoras, de modo a aumentar a economia de energia. Uma estratégia em que os nós transitam para o estado Inativo após as transmissões é abordada em [71].

Se  $\Delta t$  for definido como o intervalo de tempo entre as duas últimas transmissões (com inovação), o nó permanecerá no estado Inativo pelos próximos  $PI_i$  segundos, que pode ser definido como:

$$PI_i = \frac{\Delta t}{2}. \quad (6)$$

O período de inatividade utiliza o intervalo  $\Delta t/2$  para que, caso o processo varie mais rapidamente, por exemplo no caso de uma explosão, para um monitoramento de temperatura, os nós possam voltar ao estado Ativo a tempo, de modo a “relatar” mais rápido este “incidente”. Este período de inatividade não é fixo, e pode variar ao longo do tempo, de acordo com a variação do processo monitorado. Após o período de inatividade, o nó retorna ao estado Ativo, realiza nova medida e compara esta com a anterior (com inovação). Este processo é realizado até que a energia do nó  $S_i$  acabe.

Nesta abordagem, baseada na fonte, a partir do processamento local das amostras medidas, os nós são autônomos para decidir por transmitir ou não, passar para o estado Inativo e por quanto tempo permanecer nesse estado.

Na Figura 21, que ilustra um exemplo da utilização do algoritmo em um processo de comunicação, um nó sensor  $S_i$  mede uma primeira amostra  $x_i(1)$  e a transmite para o sorvedouro (no instante de tempo  $t_i(1)$ ). Após isto,  $S_i$  mede uma segunda amostra ( $x_i(2)$ ), mas não a transmite, pois a variação percentual entre esta amostra e a primeira foi menor do que o limiar de inovação. O mesmo ocorre com a medida  $x_i(3)$ . Porém, suponha que na próxima comparação (entre as amostras medidas  $x_i(4)$  e  $x_i(1)$ ), a variação percentual entre

elas seja maior do que o limiar. Portanto  $x_i(4)$  será transmitida (no instante de tempo  $t_i(4)$ ) e  $S_i$  passa para o modo Inativo, no qual economiza energia durante  $\Delta t = \frac{t_i(4) - t_i(1)}{2}$ . Após  $\Delta t$  segundos, o nó  $S_i$  acorda, ou seja, volta ao estado Ativo, mede uma nova amostra  $x_i(5)$  e a compara com a anteriormente transmitida ( $x_i(4)$ ). Este processo termina quando a energia do nó  $S_i$  acaba. Já na parte referente ao sorvedouro, pode-se ver que este apenas recebe as amostras medidas com inovação e transmitidas pelos nós sensores. Importa observar que as medidas em cinza são aquelas transmitidas (inovadoras) pelo nó sensor em questão e, por conseguinte, recebidas pelo sorvedouro.

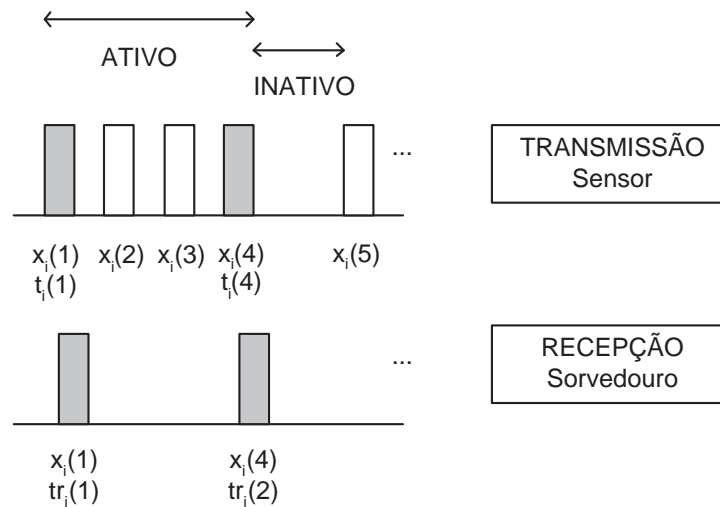


Figura 21 - Exemplo de comunicação utilizando o algoritmo baseado na fonte.

A seguir, pode-se ver, para cada nó  $S_i$ , os passos do algoritmo baseado na fonte.

Deve-se observar que não se sabe, inicialmente, o tempo de inatividade de cada nó. Portanto, no começo do algoritmo, este é definido como  $PI_i \leftarrow 0$ . Logo, deve-se realizar medidas até encontrar este valor, a partir da diferença entre os instantes de transmissão de medidas com inovação.

### 3.1.2 Algoritmo baseado no sorvedouro

Este algoritmo utiliza outro paradigma. Ao contrário do algoritmo baseado na fonte, é um algoritmo centralizado, mais especificamente no sorvedouro, que realiza todo o processamento (no que diz respeito à verificação de amostras inovadoras) e a tomada de decisões (no que tange às transições para o estado Inativo e por quanto tempo permanecer neste modo). Dessa forma, pode-se usar nós sensores mais simples (em termos

---

**Algoritmo 1** Algoritmo baseado na fonte
 

---

```

 $j \leftarrow 1$ 
 $PI_i \leftarrow 0$ 
while  $energia_i > 0$  do
   $S_i$  mede  $x_i(j)$ 
  if  $j = 1$  then
     $S_i$  transmite  $x_i(j)$ 
     $Tx_i \leftarrow x_i(j)$ 
     $t_i(j) \leftarrow$  instante de transmissão
     $tr \leftarrow t_i(j)$ 
  else
    if  $\frac{|x_i(j) - Tx_i|}{Tx_i} > limiar$  then
       $S_i$  transmite  $x_i(j)$ 
       $Tx_i \leftarrow x_i(j)$ 
       $t_i(j) \leftarrow$  instante de transmissão
       $PI_i \leftarrow \frac{t_i(j) - tr}{2}$ 
       $tr \leftarrow t_i(j)$ 
       $S_i$  Inativo durante  $PI_i$  segundos
       $S_i$  acorda após  $PI_i$  segundos
    end if
  end if
   $j \leftarrow j + 1$ 
end while

```

---

computacionais), já que esses nós não necessitam realizar nenhum tipo de processamento, que é feito diretamente no sorvedouro. Isso motiva o estudo deste algoritmo.

Nesta abordagem, os nós sensores simplesmente medem as amostras e as transmitem para o sorvedouro. Este último é o responsável pela tomada de decisões. Para cada sensor, o sorvedouro, a partir da segunda amostra recebida, calcula a variação percentual, conforme a Equação (5), bastando trocar  $x$  por  $y$ , que são as coordenadas da posição de cada nó sensor dentro da região monitorada. Seja  $y_i(j)$  o primeiro valor recebido pelo sorvedouro, ou seja, a amostra  $j$  do sensor  $i$ . A partir da próxima amostra recebida, o sorvedouro calcula a variação percentual entre a amostra atual ( $y_i(j + n)$ ) e a primeira. Caso esta variação seja maior do que o limiar de inovação, ou seja, caso haja inovação, o sorvedouro calcula o intervalo de tempo  $\Delta t$  entre as respectivas recepções e manda que o respectivo nó transmissor durma durante metade deste intervalo, portanto,  $PI_i = \frac{\Delta t}{2}$  segundos. Após este período de inatividade, que pode variar com  $\Delta t$ , o nó sensor em questão volta ao estado Ativo e continua a transmitir amostras medidas para o sorvedouro.

A Figura 22 representa um exemplo referente ao algoritmo descrito. Nesta Figura, um sensor  $S_i$  transmite a sua primeira medida ( $y_i(1)$ ), no instante  $t_i(1)$ , para o sorvedouro

$S_r$ . Como é a primeira medida recebida, esta foi ilustrada com tom cinza. Após esta transmissão/recepção inicial,  $S_i$  continua realizando medidas e as transmitindo para  $S_r$ . Este, a cada amostra recebida, compara a atual com a primeira. No exemplo em questão, as amostras  $y_i(2)$  e  $y_i(3)$  foram comparadas com  $y_i(1)$  e as variações percentuais calculadas foram menores do que o limiar de inovação. Contudo, ao receber a amostra  $y_i(4)$  e compará-la com a primeira, verificou-se uma variação maior do que o limiar, ou seja, observou-se a inovação. Repare que esta amostra também possui tom cinza. De posse dos tempos de recepção de cada amostra inovadora ( $tr_i(4)$  e  $tr_i(1)$ ), o sorvedouro calcula o tempo de inatividade  $\Delta t = \frac{tr_i(4) - tr_i(1)}{2}$  e manda que o respectivo nó sensor durma (entre no estado Inativo) por  $\Delta t$  segundos. Após este tempo,  $S_i$  acorda (volta para o estado Ativo), mede  $y_i(5)$  e transmite esta medida para o sorvedouro. Este, recebe a amostra e a compara com a amostra anterior com inovação ( $y_i(4)$ ), para verificar o próximo período de inatividade do respectivo nó sensor. Este processo continua até que a energia de  $S_i$  acabe.

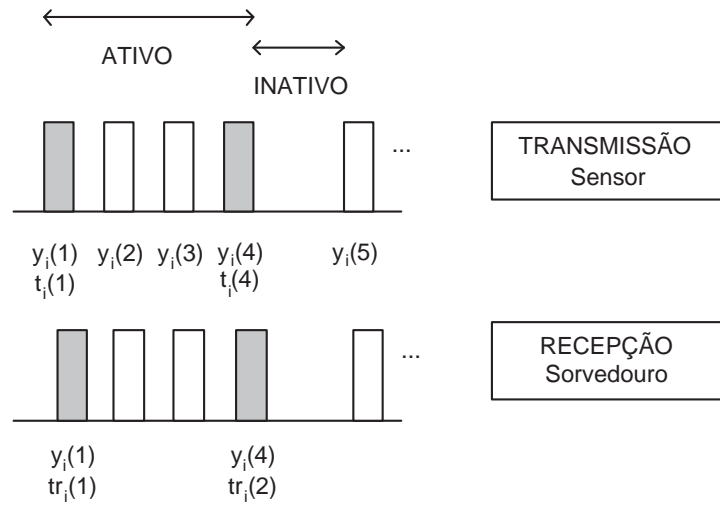


Figura 22 - Exemplo de comunicação utilizando o algoritmo baseado no sorvedouro.

A seguir é apresentado o algoritmo, para o sorvedouro, chamado de  $S_r$ .

Nesse algoritmo, inicialmente, o período de inatividade é definido como  $PI_i \leftarrow 0$ , pelos mesmos motivos apresentados para o algoritmo baseado na fonte. Neste caso, o sorvedouro deve receber amostras até encontrar este valor, para cada nó sensor.

Cumprir observar que, após o sorvedouro verificar uma amostra inovadora (caso a variação percentual entre a amostra atual e a anterior seja maior do que o limiar), e calcular o período de inatividade do respectivo nó sensor  $S_i$ , ele envia uma mensagem



---

**Algoritmo 2** Algoritmo baseado no sorvedouro
 

---

```

 $\hat{j} \leftarrow 1$ 
 $PI_i \leftarrow 0$ 
while  $energia_i > 0$  do
   $S_r$  recebe  $y_i(j)$ 
  if  $j = 1$  then
     $Ry_i \leftarrow y_i(j)$ 
     $t_i(j) \leftarrow$  instante de recepção
     $tr \leftarrow t_i(j)$ 
  else
     $t_i(j) \leftarrow$  instante de recepção
    if  $\frac{|y_i(j) - Ry_i|}{Ry_i} > limiar$  then
       $tInativo_i \leftarrow \frac{t_i(j) - tr}{2}$ 
       $tr \leftarrow t_i(j)$ 
       $Ry_i \leftarrow y_i(j)$ 
       $S_i$  Inativo durante  $PI_i$  segundos
       $S_i$  acorda após  $PI_i$  segundos
    end if
  end if
   $j \leftarrow j + 1$ 
end while

```

---

para o nó sensor com o período de inatividade calculado ( $tInativo_i$ ). Após  $S_i$  receber a mensagem, este permanece Inativo durante o período calculado. A Figura 23 ilustra a mudança de estado (Ativo  $\rightarrow$  Inativo) do nó  $S_i$ , para o caso em que o sorvedouro verifica recepção inovadora. Além disso, caso a amostra recebida não seja inovadora, o sorvedouro espera até receber a próxima amostra, e faz o cálculo novamente.

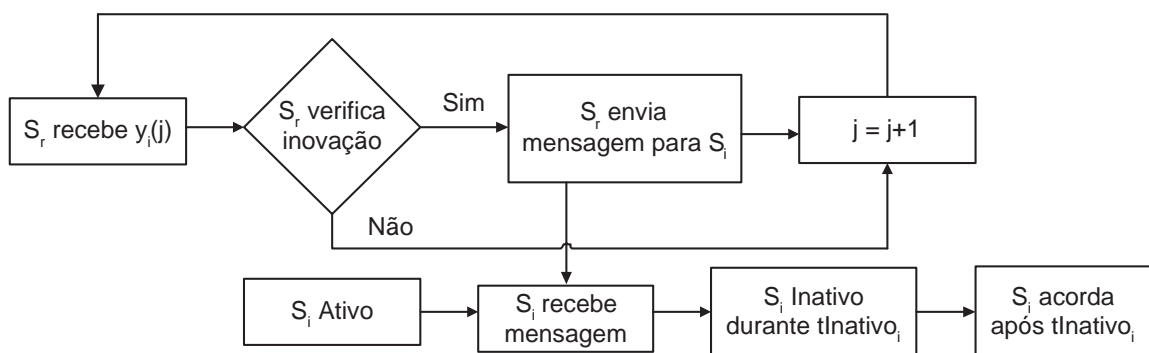


Figura 23 - Mudança de estado do nó sensor (Ativo  $\rightarrow$  Inativo).

### 3.2 Algoritmo para RSSFs de Múltiplos Saltos

Nas RSSFs de salto único, objeto de discussão da Seção 3.1, o nó sorvedouro está posicionado dentro raio de alcance de todos os nós da rede. Desta forma, cada nó sensor pode transmitir mensagens diretamente para o sorvedouro. Porém, o alcance de cada nó da rede é função de alguns parâmetros, como a distância  $d_{ir}$  entre dois nós  $S_i$  e  $S_r$ , a potência de transmissão do rádio do nó  $P_{txi}$  e a sensibilidade do receptor  $S_{rxr}$ . Além disso, fatores ambientais, como a existência de obstáculos entre os dois nós, também são fatores limitantes para que estes consigam se comunicar diretamente. Ou seja, um dado nó  $S_i$  só pode se comunicar diretamente com um outro  $S_r$ , caso eles estejam dentro do mesmo raio de alcance. Isto, obviamente, limita a área de cobertura de uma rede. Por exemplo, suponha uma RSSF que esteja sendo utilizada para realizar o monitoramento de uma região e a transmissão dos dados coletados para uma estação base, através de comunicação de único salto. Por maior que seja a quantidade de nós em uma rede, a sua área de cobertura sempre será limitada pelo alcance de cada rádio.

Já nas RSSFs de múltiplos saltos, um nó destino  $S_r$  não precisa estar dentro da área de cobertura do nó fonte  $S_i$ . Neste tipo de rede, as mensagens são encaminhadas nó-a-nó até o destino. Ou seja, os nós encaminham os dados (da fonte até o destino) utilizando um protocolo de roteamento.

A Figura 24 apresenta um exemplo da área de cobertura para uma RSSF de salto único, na Figura 24 (a), e para uma RSSF de múltiplos saltos, na Figura 24 (b). Suponha que um nó sensor  $S_i$  deseje se comunicar com um nó sorvedouro  $S_r$ , e que a distância entre os dois nós seja  $d_{ir}$ . A área de cobertura dessa rede estará sempre limitada à área de cobertura do nó  $S_r$ , ou seja, a distância máxima entre os dois nós deve ser menor do que o alcance do rádio de  $S_r$ , o que pode-se verificar na Figura 24 (a). Se definirmos um alcance  $A(S_r)$  para o nó  $S_r$ , a área de cobertura da rede, ou seja, a distância  $d_{ir}$  entre os nós  $S_i$  e  $S_r$  pode ser definida da seguinte forma:

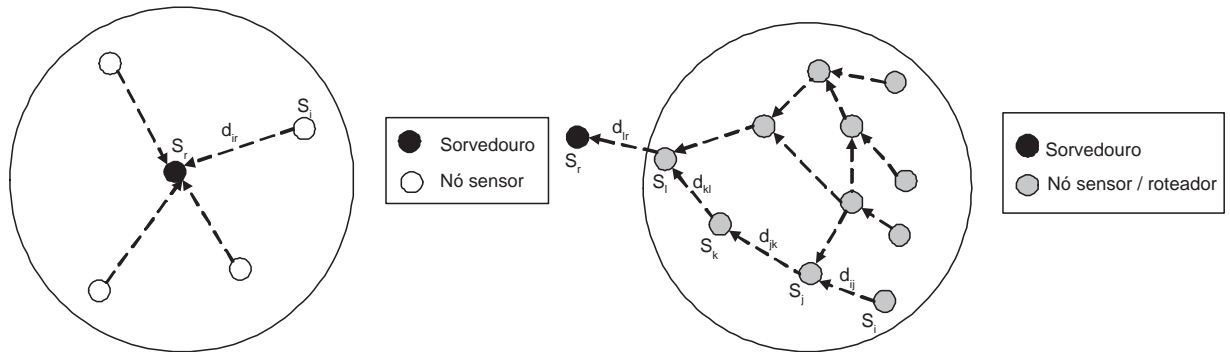
$$d_{ir} \leq A(S_r). \quad (7)$$

Já em RSSFs de múltiplos saltos, pode-se verificar, na Figura 24 (b), que a área de cobertura da rede pode ser bastante estendida. Suponha que um nó sensor  $S_i$  deseje se comunicar com um nó sorvedouro  $S_r$ , porém utilizando comunicação de múltiplos saltos,

ou seja, nós intermediários podem encaminhar as mensagens da fonte até o destino. No exemplo da Figura 24 (b), os nós  $S_j$ ,  $S_k$  e  $S_l$  são responsáveis pelo encaminhamento das mensagens da fonte ( $S_i$ ) até o destino ( $S_r$ ). Se os alcances dos nós  $S_i$ ,  $S_j$ ,  $S_k$  e  $S_l$  forem definidos como  $A(S_i)$ ,  $A(S_j)$ ,  $A(S_k)$  e  $A(S_l)$ , respectivamente, pode-se definir a área de cobertura da RSSF, em termos da distância  $d_{ir}$  entre a fonte e o destino, da seguinte forma:

$$d_{ir} \leq A(S_i) + A(S_j) + A(S_k) + A(S_l). \quad (8)$$

Como exemplo prático, se os alcances dos nós  $S_i$ ,  $S_j$ ,  $S_k$  e  $S_l$  forem definidos como  $40m$ , a área de cobertura da RSSF (para múltiplos saltos) será igual a  $d_{ir} < 160m$ , e a área (para único salto) será igual a  $d_{ir} < 40m$ .



(a) Área de cobertura de uma rede de único salto. (b) Área de cobertura de uma rede de múltiplos saltos.

Figura 24 - Áreas de cobertura de RSSFs de único salto e de múltiplos saltos.

Alguns exemplos de protocolos de roteamento para redes *ad hoc* e para RSSFs são:

- **AODV** (*Ad hoc On-demand Distance Vector*) [72] [73] é um protocolo de roteamento em que as rotas são criadas sob demanda. Ou seja, quando um nó necessita transmitir uma mensagem para um dado destino, inicia-se um processo de descoberta de rotas que inunda a rede com mensagens de requisição de rotas. Caso um nó (destino ou intermediário) possua uma rota válida, esta é informada ao nó fonte através de uma mensagem de resposta. Este protocolo é usado em redes *ad hoc*, mas também em RSSF;
- Já no **OLSR** (*Optimized Link State Routing*) [74], um dado nó fonte seleciona, entre seus vizinhos de um salto, um subconjunto de nós que irão encaminhar suas

mensagens. Caso algum nó, que não faça parte deste subconjunto, receba uma mensagem do nó fonte, ele não deverá encaminhar esta mensagem. Porém, se o nó conhecer a rota, ele encaminhará a mensagem. Desta forma, o algoritmo visa diminuir a quantidade de mensagens de controle da rede. Este protocolo é muito utilizado em redes *ad hoc*;

- No protocolo *Directed Diffusion* [75] [76], os dados são classificados segundo algum atributo. Uma tarefa de sensoriamento é disseminada pela rede, segundo o interesse por um dado classificado. Essa disseminação estabelece gradientes pela rede destinados a coletar os eventos (dados que atendem ao interesse). Os eventos, então, começam a fluir pela rede, em direção aos nós originadores dos interesses, utilizando múltiplos caminhos. Esse protocolo é utilizado em RSSFs.

### 3.2.1 Extensão do algoritmo baseado na fonte para RSSFs de múltiplos saltos

Em redes de múltiplos saltos, os nós encaminham mensagens utilizando um protocolo de roteamento, como o AODV. Logo, rotas são formadas entre as fontes de informação (nós sensores) e o nó sorvedouro. Uma vez estabelecida uma dada rota, os dados podem ser encaminhados (pelos nós intermediários) até o sorvedouro. Portanto, é razoável esperar que os nós roteadores, ou seja, aqueles que são responsáveis pelo encaminhamento das mensagens, devem permanecer acordados (Ativos) todo o tempo. Estando acordados, os roteadores podem encaminhar mensagens advindas de nós que o usam como rota sempre que necessário. Caso algum nó roteador esteja dormindo, mensagens podem não ser encaminhadas quando necessário, causando problemas à conectividade e cobertura da rede. Nesse caso, em que os roteadores permanecem acordados todo o tempo, uma topologia possível é ilustrada na Figura 25 (a), onde os nós sensores são posicionados próximos a uma região de interesse (coletando dados), e os nós roteadores apenas encaminham as mensagens até o sorvedouro.

Nesta subseção, será apresentada uma extensão do algoritmo baseado na fonte, que permite sua utilização em RSSFs de múltiplos saltos. Porém, apenas o algoritmo baseado na fonte foi estendido. Como será visto no Capítulo 5, os resultados obtidos a partir desse algoritmo se mostram melhores do que os resultados do algoritmo baseado no sorvedouro, por razões que serão mostradas no mesmo capítulo. Logo, escolheu-se o algoritmo baseado na fonte para ser estendido e utilizado em RSSFs de múltiplos saltos.

O algoritmo baseado na fonte proposto nesta subsecção considera uma situação mais geral. Em cenários reais, nós sensores são espalhados em uma região de interesse, como uma floresta, de forma a monitorar um dado processo, por exemplo umidade e temperatura nessa região. A ideia que permeia este algoritmo é que todos os nós podem ser fontes de informação (ao coletarem amostras medidas do processo), e também podem ser roteadores (ao encaminharem um pacote). Dessa forma, caso ocorra um fenômeno, como um incêndio na floresta, qualquer nó sensor poderá coletar e transmitir esta mensagem (agindo como uma fonte), e encaminhá-la através da rede (agindo como um roteador). A Figura 25 (b) ilustra uma topologia onde todos os nós são sensores e roteadores.

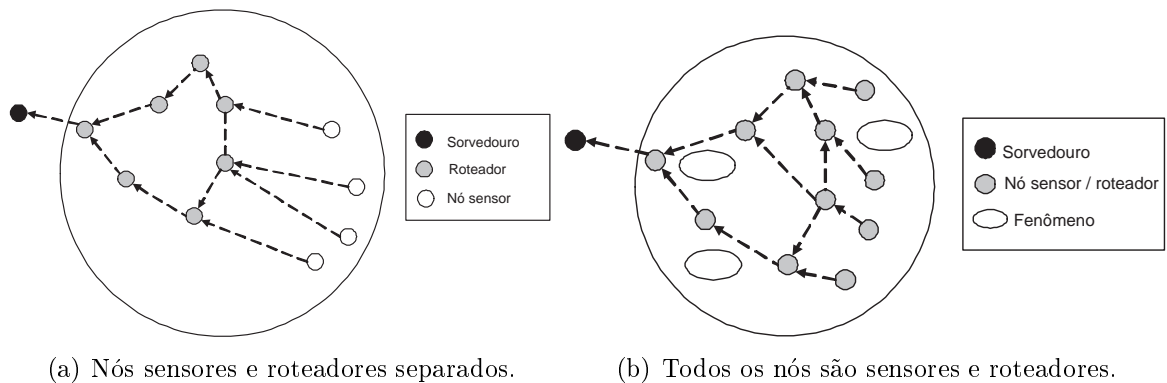


Figura 25 - Duas possíveis topologias para RSSFs de múltiplos saltos.

No algoritmo proposto, cada nó sensor pode entrar no estado Inativo (para economizar energia) entre transmissões inovadoras, da mesma forma que ocorria no algoritmo baseado na fonte apresentado na Seção 3.1.1. Porém, cada nó deve garantir que a rota (uma vez estabelecida pelo protocolo de roteamento) não será afetada, quando um nó está inativo.

Assim como o algoritmo apresentado na Seção 3.1.1 3, um sensor  $S_i$  apenas irá transmitir amostras para as quais a variação percentual entre a amostra atual e a anteriormente transmitida seja maior do que o limiar de inovação. Assim, o sensor  $i$  apenas transmite uma nova amostra  $(j+n)$ , caso:

$$\frac{|x_i(j+n) - x_i(j)|}{x_i(j)} > \text{limiar}. \quad (9)$$

onde  $x_i(j)$  é a última amostra transmitida pelo sensor. A extensão do algoritmo baseado na fonte utiliza o seguinte mecanismo: quando um nó  $S_i$  deseja transmitir uma mensagem,

ele inclui o período de inatividade ( $PI_i$ ), período de tempo no qual o nó permanecerá inativo, dentro da mensagem. Ou seja, um campo  $\langle P_{inatividade_i} \rangle$  é incluído na mensagem a ser transmitida. Assim, todos os nós conhecem os períodos de inatividade dos seus vizinhos. Os nós, então, calculam seus próprios períodos de inatividade baseados nos períodos de inatividade dos seus vizinhos, de forma a aumentar a probabilidade de estarem acordados para encaminhar suas mensagens.

Na Figura 26, é apresentado um exemplo ilustrativo do mecanismo proposto. Os nós  $S_1$ ,  $S_2$  e  $S_3$  são fontes de informação, coletando dados de um dado processo. Suponha que, dentro de suas mensagens transmitidas,  $S_1$  e  $S_2$  incluem os seus respectivos períodos de inatividade  $PI_1$  e  $PI_2$ . O nó  $S_3$  mantém um vetor períodos de inatividade ( $V_{PI_3}$ ), que contém os períodos de inatividade de seus vizinhos ( $[PI_1, PI_2]$ ). Quando  $S_3$  tem que dormir, ele calcula seu próprio período de inatividade  $PI_3$ , mas ele só pode dormir durante um período de tempo igual à metade do valor mínimo entre  $PI_1$ ,  $PI_2$  e  $PI_3$ , de forma a aumentar a probabilidade dele estar acordado para encaminhar os pacotes de  $S_1$  e de  $S_2$ . Então, o nó  $S_3$  irá permanecer inativo durante  $P_{in3}$  segundos.

$$P_{in3} = \frac{1}{2} \times \min(PI_1, PI_2, PI_3). \quad (10)$$

Quando  $S_3$  acorda, ele realiza novas medidas, calcula novamente seu período de inatividade  $P_{in3}$ , recebe e encaminha as mensagens de seus vizinhos, se for o caso. Cumpre observar que o nó  $S_3$  deve estar acordado para encaminhar as mensagens. Logo, para aumentar a probabilidade de  $S_3$  estar acordado, o valor mínimo do vetor calculado é multiplicado por  $\frac{1}{2}$ .

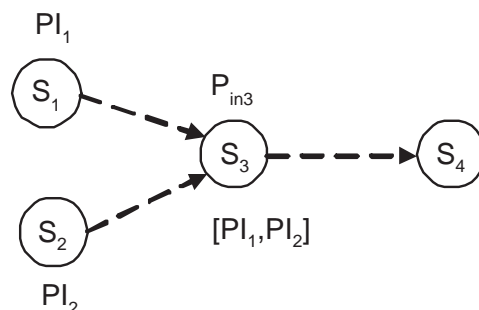


Figura 26 - Nós e seus períodos de inatividade.

Suponha  $PI_1 = 2$  segundos,  $PI_2 = 1$  segundo e que o nó  $S_3$  calcule seu período

de inatividade como  $PI_3 = 2$  segundos. Portanto, o vetor períodos de inatividade do nó  $S_3$  é composto pelos seguintes valores  $V_{PI_3} = [2, 1, 2]$  segundos. Porém, a verificação dos períodos de inatividade por cada nó não se dá ao mesmo tempo, já que os relógios dos nós não estão sincronizados. Portanto, caso o nó  $S_3$  permaneça Inativo pelo período de tempo mínimo entre os períodos do vetor, ele iria acordar após 1 segundo e o nó  $S_2$  já estaria acordado. Dessa forma, não há garantias de que o nó  $S_3$  consiga encaminhar alguma possível mensagem transmitida pelo nó  $S_2$ . Logo, o nó  $S_3$  irá permanecer Inativo durante  $P_{in3}$  segundos, ou seja,

$$\begin{aligned} P_{in3} &= \frac{1}{2} \times \min(2, 1, 2) \\ P_{in3} &= 0,5 \text{ segundo.} \end{aligned} \tag{11}$$

Dessa forma, aumenta-se a probabilidade de o nó  $S_3$  estar acordado para encaminhar uma possível mensagem transmitida por  $S_2$ . Portanto, o algoritmo permite que o nó  $S_3$  economize energia, que permanecerá inativo durante  $P_{in3}$  segundos, o força a estar acordado para encaminhar as mensagens de seus vizinhos de um salto. Esse processo é realizado por todos os nós sensores, minimizando possíveis danos à conectividade da rede.

Cumpramos observar que a utilização do vetor períodos de inatividade torna o algoritmo proposto para múltiplos saltos uma generalização do algoritmo baseado na fonte para salto único. Pode-se observar que os dois algoritmos possuem as mesmas características: apenas amostras inovadoras, ou seja, com uma variação percentual maior do que um limiar são transmitidas; e os nós permanecem Inativos entre essas transmissões. Porém, no algoritmo para múltiplos saltos, cada nó leva em conta os períodos de inatividade de seus vizinhos que o utilizam como rota, para dormir, conforme visto na Equação (10). Contudo, se o vetor períodos de inatividade de cada nó for definido como nulo, as informações dos seus vizinhos não serão utilizadas quando estes forem dormir, o que implica no algoritmo baseado na fonte para RSSFs de salto único.

Importa observar que o algoritmo, assim como os algoritmos apresentados neste capítulo, opera diretamente na camada de Aplicação em cada nó. Do exemplo da Figura 26, quando o nó  $S_3$  recebe uma mensagem proveniente de um dos seus vizinhos, ele passa a mensagem para a sua camada de aplicação para atualizar o vetor  $V_{PI_3}$ . Além

disso, quando esse nó mede amostras do processo monitorado, o teste de inovação ocorre, segundo a Equação (9), e todo o processamento é realizado nessa camada. A ideia de todo o processo ser realizado na camada de aplicação, é tornar o algoritmo o mais genérico possível, independentemente do protocolo de roteamento utilizado.

Os passos do algoritmo aprimorado baseado na fonte são apresentados para cada nó  $S_i$ .

---

**Algoritmo 3** Algoritmo baseado na fonte estendido

---

```

 $j \leftarrow 1$ 
 $PI_i \leftarrow 0$ 
while  $energia_i > 0$  do
   $S_i$  mede  $x_i(j)$ 
  if  $j = 1$  then
     $S_i$  transmite  $x_i(j)$ 
     $Tx_i \leftarrow x_i(j)$ 
     $t_i(j) \leftarrow$  instante de transmissão
     $tr \leftarrow t_i(j)$ 
  else
    if  $\frac{|x_i(j) - Tx_i|}{Tx_i} > limiar$  then
       $S_i$  transmite  $x_i(j)$ 
       $Tx_i \leftarrow x_i(j)$ 
       $t_i(j) \leftarrow$  instante de transmissão
       $PI_i \leftarrow \frac{t_i(j) - tr}{2}$ 
       $tr \leftarrow t_i(j)$ 
      if  $V_{PI_i}$  (está vazio) then
         $S_i$  Inativo durante  $PI_i$  segundos
      else
         $S_i$  encaminha as mensagens de seus vizinhos
         $V_{PI_i}[i] \leftarrow PI_i$ 
         $P_{ini} \leftarrow \frac{1}{2} \times \arg \min(V_{PI_i})$ 
         $S_i$  Inativo durante  $P_{ini}$  segundos
      end if
       $S_i$  acorda após  $P_{ini}$  segundos
    end if
  end if
   $j \leftarrow j + 1$ 
end while

```

---

### 3.3 Conclusão

Neste capítulo, foram apresentados os algoritmos para economia de energia propostos. Inicialmente, foram apresentados dois algoritmos usados em RSSFs de salto único, um baseado na fonte, e outro baseado no sorvedouro. A ideia que permeia a base de de-



envolvimento dos algoritmos é o gerenciamento da necessidade de comunicação, pelos nós sensores. Os algoritmos têm por objetivo reduzir a quantidade de transmissões pelos nós, e ainda utilizar um estado de inatividade entre transmissões inovadoras, para aumentar a economia de energia pelos nós sensores. Por fim, apresenta-se uma extensão do algoritmo baseado na fonte, que permite a sua utilização em redes de múltiplos saltos.

O próximo capítulo trata da caracterização do problema discutido nesta dissertação, da apresentação das métricas utilizadas na avaliação de desempenho dos algoritmos, e do ambiente de simulação utilizado.

## 4 AVALIAÇÃO DE DESEMPENHO ATRAVÉS DE SIMULAÇÃO

Os principais tópicos vistos neste capítulo são: a descrição do problema analisado, a apresentação da forma como será realizada esta análise e como serão feitas as simulações necessárias, além de discussão de que métricas serão utilizadas na avaliação do que é proposto.

### 4.1 O Processo Monitorado

Suponha que se deseje monitorar uma dada região de interesse, por exemplo, uma floresta, uma caverna ou um prédio. Pode-se realizar este monitoramento através de uma RSSF, cujos nós sensores são espalhados nessa região e utilizados para medir grandezas, tais como temperatura, umidade e pressão. Nesta dissertação, tal região monitorada será representada por uma superfície que varia em função das posições  $x$  e  $y$  dos nós sensores e do tempo  $t$ . A ideia é que cada sensor é fonte de informação, ou seja, mede amostras do processo (região) monitorado  $f(x, y, t)$  e as envia para o sorvedouro. Na avaliação de desempenho dos algoritmos, considera-se um processo suave, que pode ser descrito por:

$$f(x, y, t) = \left[ e^{-\frac{(x-m_x)^2}{2 \cdot dp_x^2}} + e^{-\frac{(y-m_y)^2}{2 \cdot dp_y^2}} \right] \times \left[ e^{-\frac{(t-m_t)^2}{2 \cdot dp_t^2}} \right] + C \quad (12)$$

No processo em questão,  $m_x = m_y = 0$  e  $m_t = 40$ , representam as médias em  $x$ ,  $y$  e  $t$ ;  $dp_x = dp_y = dp_t = 20$  são, respectivamente, os desvios padrões em  $x$ ,  $y$  e  $t$ , e  $C = 5$  é uma constante. Tal função foi utilizada, para que os algoritmos propostos sejam avaliados de maneira generalizada. A Figura 27 ilustra a superfície no instante de tempo  $t = 0$ . Esta superfície foi utilizada nas simulações referentes às RSSFs de salto único.

### 4.2 Métricas Utilizadas na Avaliação

As métricas utilizadas na avaliação dos algoritmos propostos são apresentadas a seguir:

- **escalabilidade:** Com esta métrica, pode-se verificar o impacto da quantidade de nós da rede no desempenho dos algoritmos propostos. Para RSSFs de salto único,

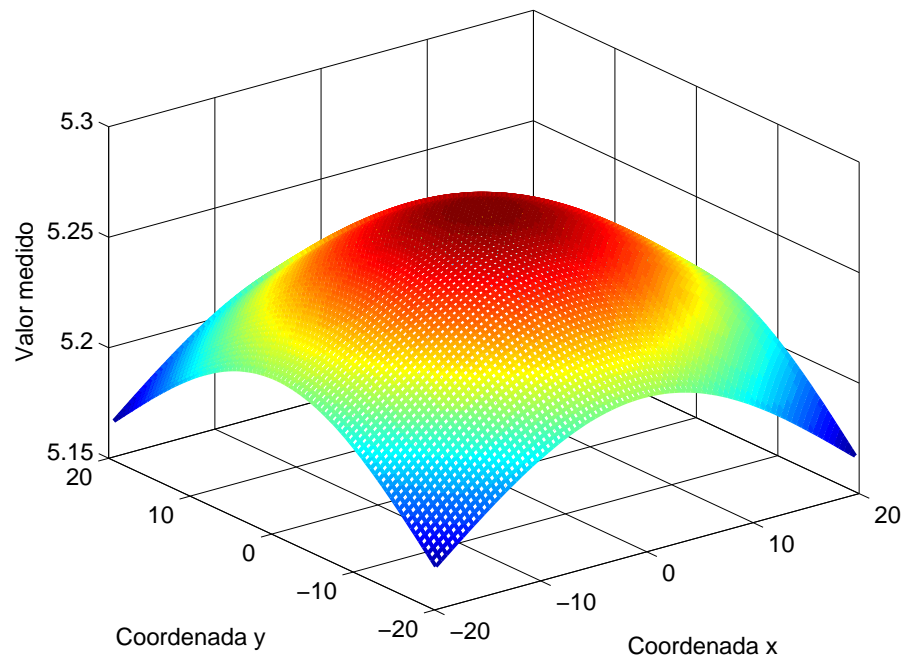


Figura 27 - Superfície monitorada em  $t = 0$ .

a quantidade de nós utilizada irá variar de quatro a sete (para uma área coberta de  $40 \text{ m} \times 40 \text{ m}$ ). Já nas RSSFs de múltiplos saltos, serão utilizadas quantidades de sete, quinze, trinta e cinquenta nós, para as respectivas áreas:  $80 \text{ m} \times 80 \text{ m}$ ,  $140 \text{ m} \times 140 \text{ m}$ ,  $200 \text{ m} \times 200 \text{ m}$  e  $250 \text{ m} \times 250 \text{ m}$ ;

- **tempo de vida da rede:** Esta métrica diz respeito ao tempo decorrido até a morte do primeiro sensor da rede, ou seja, quando acaba a energia de sua bateria, podendo ser definido como:

$$TdV = \arg \min_i (E_i(t) = 0) \quad (13)$$

onde  $E_i(t)$  é a energia de um nó  $S_i$  no tempo  $t$ ;

- **redução nas transmissões:** A redução na quantidade de transmissões implica em conservação de energia por parte dos nós sensores. Podemos definir a redução nas transmissões como:

$$R_{tx} = \frac{T_{x_{Tot}} - T_{x_{alg}}}{0.01 \times T_{x_{Tot}}}, \quad (14)$$

em [%], onde  $T_{x_{Tot}}$  é a quantidade total de transmissões na rede (sem a utilização dos algoritmos para economia de energia), e  $T_{x_{alg}}$  é a quantidade de transmissões

(com a utilização dos algoritmos propostos);

- **erro de reconstrução:** Este é o erro de reconstrução do processo monitorado, e pode ser definido como:

$$e(x_i, y_i, t) = f(x_i, y_i, t) - \hat{f}(x_i, y_i, t), \quad (15)$$

onde  $f(x_i, y_i, t)$  é o processo monitorado, e  $\hat{f}(x_i, y_i, t)$  é o processo reconstruído, para as coordenadas fixas  $(x_i, y_i)$  dos nós sensores, e para o tempo  $t$  contínuo. Com esta métrica, deseja-se avaliar a qualidade da reconstrução do processo monitorado;

- **taxa de entrega:** É a razão entre a quantidade de pacotes recebidos e transmitidos, definida como:

$$TE = \frac{\text{pacotes recebidos}}{\text{pacotes transmitidos}}. \quad (16)$$

### 4.3 Ambiente de Simulação

Para o desenvolvimento das simulações, foi utilizado o TrueTime 1.5, um ambiente de simulação em “tempo-real” baseado no MATLAB/Simulink [77] [78] [79]. O TrueTime é um simulador baseado em blocos, que representam computadores, nós em uma rede e interfaces de redes. Logo, a conexão entre os blocos do TrueTime e os blocos do Simulink formam um sistema de controle em tempo real. Os blocos constituintes do simulador podem ser vistos na Figura 28.

O bloco *kernel* modela um computador com, por exemplo, tarefas de entrada e saída (I/O), algoritmos de controle e interfaces de rede. Os blocos de rede (com e sem fios) simulam as camadas Física e de Acesso ao Meio de redes como *Ethernet*, *Wi-Fi* e *ZigBee*. Os blocos *ttSendMsg* e *ttGetMsg* simulam sistemas simples em rede sem a necessidade de *kernels*. Por fim, o bloco de bateria simula o consumo de energia dos nós ou de um computador. Este bloco foi modificado, já que foram incluídos novos estados referentes ao consumo de energia dos nós, de forma a poder-se calcular o consumo de energia de um dado nó a partir da Equação (3), referente ao modelo apresentado na Seção 2.2.

Um exemplo da interligação entre os blocos principais (da biblioteca de blocos) do TrueTime e os blocos do Simulink pode ser visto na Figura 29, que ilustra uma rede sem

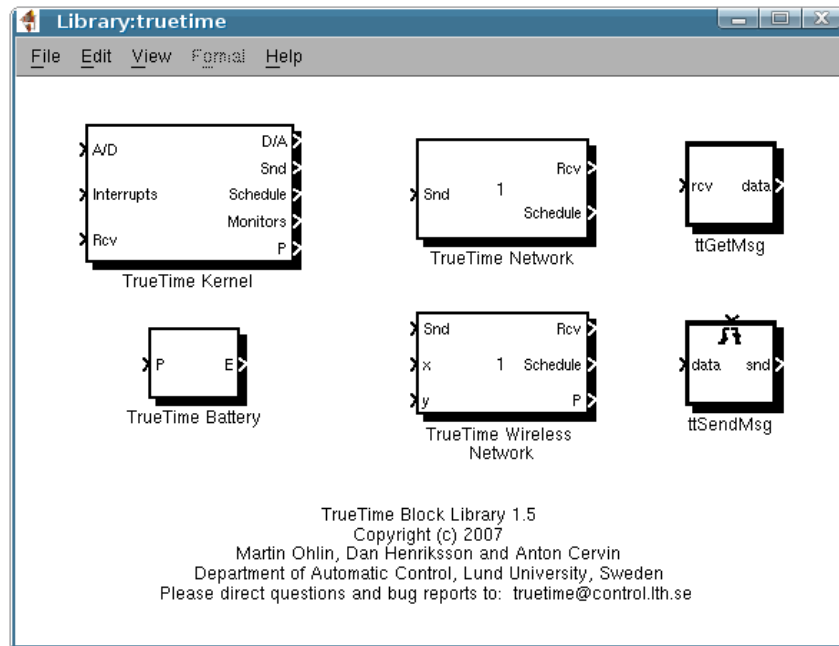


Figura 28 - Blocos do simulador TrueTime.

fo com sete nós.

Os principais parâmetros fixos utilizados nas simulações estão apresentados na Tabela 1. Além disso, todas as simulações utilizaram uma base de tempo de 0,1 segundos. Foi utilizada, nesta dissertação, a tecnologia ZigBee [24] [23], na banda de 2,4 GHz e com uma taxa de 250 kbps. E em relação aos resultados, foi considerado um intervalo de confiança de 95%. O intervalo de confiança [80] da média de  $n$  amostras pode ser definido da seguinte forma:

$$IC = \bar{X} \pm t \times \frac{S}{\sqrt{n}}, \quad (17)$$

onde,  $\bar{X}$  é a média das amostras;  $t$  é um valor obtido através de uma tabela de quantis da distribuição  $t$ , chamado Valor Crítico;  $S$  é o desvio padrão das amostras; e  $n$  é a quantidade de amostras. Como, nas simulações, utilizou-se  $n < 30$ , considera-se a distribuição  $t$ -Student para o cálculo do intervalo de confiança [80].

Por fim, cumpre observar que os resultados foram obtidos através da comparação entre as simulações utilizando os algoritmos propostos e simulações onde não se utilizou nenhum tipo de gerenciamento. Neste caso, cada nó, periodicamente, realiza medidas e as transmite para o sorvedouro, sem utilização de nenhum tipo de método de economia de energia.

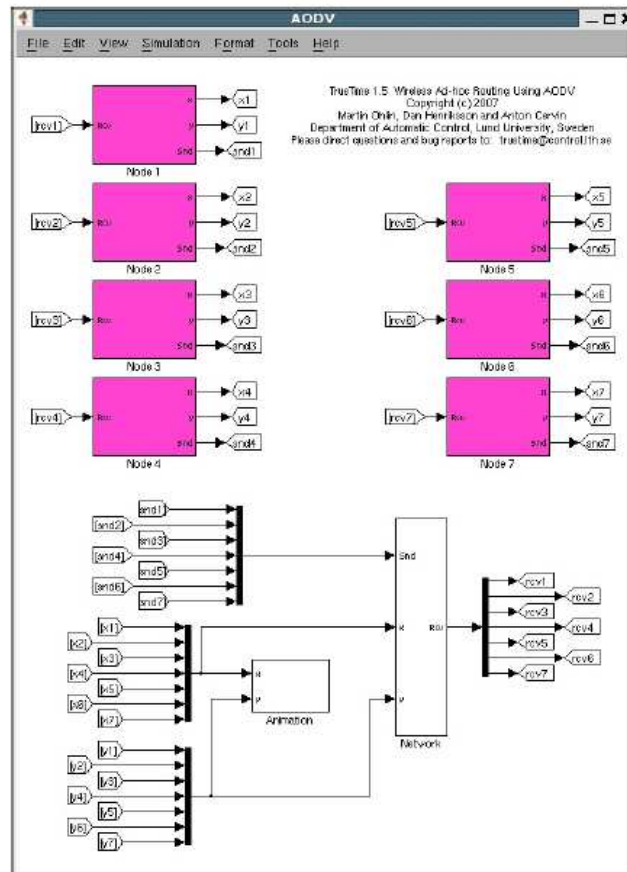


Figura 29 - Modelo de uma rede sem fio utilizando o TrueTime 1.5, no Simulink.

Em relação à topologia, a área da região de interesse variou segundo o tipo de RSSF. Nas simulações referentes aos algoritmos propostos para RSSFs de salto único, a área monitorada foi de  $40 \text{ m} \times 40 \text{ m}$ , com uma quantidade de nós entre quatro e sete. Nesta topologia, o sorvedouro foi disposto no centro na região de interesse, na coordenada  $(0,0)$ , e os nós sensores foram posicionados aleatoriamente em torno dele. Portanto, as coordenadas  $x$  e  $y$  da região monitorada variaram entre  $-20 \text{ m}$  e  $20 \text{ m}$ , conforme ilustrado na Figura 27, com as médias de  $x$  e  $y$  definidas como  $m_x = m_y = 0$ .

Já nas simulações referentes ao algoritmo proposto para RSSFs de múltiplos saltos, para sete nós, utilizou-se uma área de  $80 \text{ m} \times 80 \text{ m}$ . Para quinze nós, utilizou-se uma área de  $140 \text{ m} \times 140 \text{ m}$ . Para trinta nós, uma área de  $200 \text{ m} \times 200 \text{ m}$  foi monitorada e, por fim, simulações com cinquenta nós foram realizadas em uma região de interesse de  $250 \text{ m} \times 250 \text{ m}$ . A Tabela 2 apresenta a variação da área monitorada em função da quantidade de nós na rede. Nestas simulações, o sorvedouro (nó de número sete) foi posicionado

Tabela 1 - Parâmetros Fixos de Simulação.

Energia inicial dos nós (J)	2
Potência de Tx (dBm)	-5
Sensibilidade de Rx (dBm)	-66
Alcance do rádio (m)	40
$C_I$ : Consumo no estado inativo (mJ/s)	1,8
$C_A$ : Consumo no estado ativo (mJ/s)	10
$C_M$ : Consumo no modo medição (mJ/s)	18
$C_P$ : Consumo no modo processamento (mJ/s)	18
$C_R$ : Consumo no modo recepção (mJ/s)	62,4
$C_T$ : Consumo no modo transmissão (mJ/s)	58,62
Tamanho da carga útil do pacote (Byte)	1

à direita e no centro vertical da área (fixo) e os sensores, à sua esquerda, dispostos de forma aleatória dentro da área monitorada. A cada simulação realizada, a posição dos nós sensores é sorteada, sempre à esquerda do sorvedouro. Portanto, as coordenadas  $x$  e  $y$  da região monitorada variaram entre 0 e o valor máximo (de acordo com a quantidade de nós), por exemplo, de 0 até 140 m, para uma quantidade de quinze nós. Assim, para as simulações referentes à RSSF de múltiplos saltos, os seguintes parâmetros para a superfície descrita pela Equação (12) foram considerados:  $m_x = m_y = m_t = 40$  são as médias de  $x$ ,  $y$ , e  $t$ ;  $dp_x = dp_y = dp_t = 20$  são os valores de desvio padrão para  $x$ ,  $y$ , e  $t$ , e  $C = 5$  é uma constante. A Figura 30 ilustra o processo monitorado, para  $t = 0$  e uma quantidade fixa de sete nós, cujas médias de  $x$  e  $y$  foram definidas como  $m_x = m_y = 40$ , já que a área monitorada é de 80 m  $\times$  80 m.

Tabela 2 - Cenários de simulação (quantidade de nós em função da área).

7 nós	80 m $\times$ 80 m
15 nós	140 m $\times$ 140 m
30 nós	200 m $\times$ 200 m
50 nós	250 m $\times$ 250 m

#### 4.4 Conclusão

Neste capítulo, caracterizou-se a superfície monitorada pela rede (problema avaliado na dissertação), apresentou-se as métricas utilizadas na avaliação de desempenho

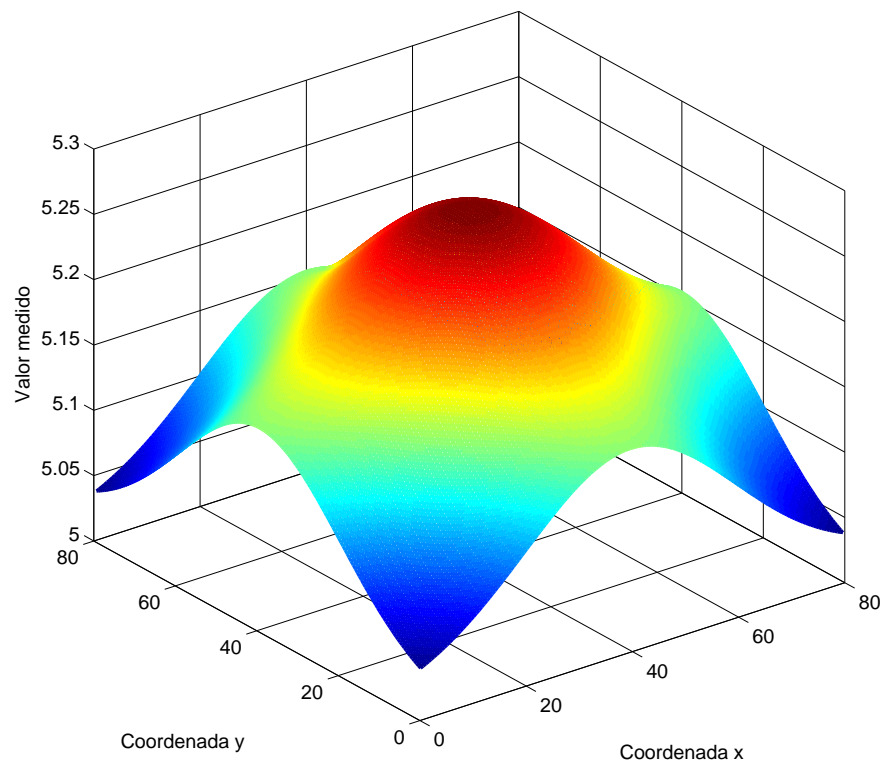


Figura 30 - Superfície monitorada em  $t = 0$  (para uma área coberta de  $80m \times 80m$ ).

(escalabilidade, tempo de vida da rede, redução das transmissões, erro na reconstrução do processo, e a taxa de entrega). Por fim, o ambiente de simulação utilizado foi apresentado, assim como os principais parâmetros de simulação.

O próximo capítulo apresenta os resultados das simulações dos algoritmos propostos, para RSSFs de salto único.



## 5 RESULTADOS DOS ALGORITMOS PARA ECONOMIA DE ENERGIA EM RSSFs DE SALTO ÚNICO

Neste capítulo serão apresentados os resultados dos algoritmos para economia de energia propostos, para RSSFs de salto único. Serão apresentados os resultados das simulações para os algoritmos baseados na fonte e no sorvedouro. Após a explanação dos resultados do algoritmo que utiliza o paradigma do sorvedouro, duas análises serão feitas, de modo a estudar o desempenho desse algoritmo: uma análise do tempo de inatividade inicial e uma análise da energia inicial dos nós.

Nas simulações para redes de salto único, utilizou-se uma quantidade máxima de sete nós, com seis nós sensores e um nó sorvedouro. Em aplicações como Redes de Sensores para o Corpo Humano, pode-se utilizar poucos nós, já que estes podem ficar dispostos sobre o corpo ou na roupa dos pacientes [70] [81]. Logo, é justificável a utilização de uma rede com até sete nós, desde que haja garantia de conectividade, através, por exemplo, da análise de métricas como a taxa de entrega de pacotes (TE) [23], definida pela Equação (16).

### 5.1 Sobre a Taxa de Entrega de Pacotes

A Tabela 3 apresenta a taxa de entrega em função do limiar de inovação (LI), para os dois algoritmos, onde  $TE_F$  e  $TE_S$  são as taxas de entrega para os algoritmos baseados na fonte e no sorvedouro, respectivamente. Pode-se observar que a conectividade da rede se manteve, já que as taxas de entregas obtidas, em ambos os algoritmos, são maiores do que 94%. Além disso, pode-se verificar uma redução, mesmo que pequena, na taxa de entrega com o aumento da quantidade de nós. Com a escalabilidade da rede, há um aumento da disputa pelo meio, o que pode levar a uma maior quantidade de colisões, reduzindo a taxa de entrega, conforme visto nos resultados dessa Tabela. Porém, como não há um aumento grande na quantidade de nós (apenas um aumento entre quatro e sete nós), não se observa uma redução grande na taxa de entrega.

Tabela 3 - Taxa de entrega em função do limiar de inovação.

Taxa de entrega para 4 nós				
LI	0,1%	1%	5%	8%
$TE_F$	98,85% ± 1,03%	99,00% ± 0,90%	97,60% ± 1,00%	98,23% ± 1,00%
$TE_S$	98,53% ± 1,00%	98,00% ± 0,20%	97,32% ± 0,35%	97,2% ± 1,00%
Taxa de entrega para 5 nós				
LI	0,1%	1%	5%	8%
$TE_F$	98,00% ± 1,00%	97,79% ± 1,72%	97,60% ± 0,50%	97,51% ± 0,53%
$TE_S$	97,83% ± 1,00%	97,60% ± 0,50%	97,40% ± 0,42%	97,00 ± 1,00%
Taxa de entrega para 6 nós				
LI	0,1%	1%	5%	8%
$TE_F$	95,90% ± 1,00%	96,05% ± 1,44%	97,00% ± 1,40%	96,50% ± 1,11%
$TE_S$	96,93% ± 1,00%	95,00% ± 0,50%	96,22% ± 1,00%	95,90% ± 1,00%
Taxa de entrega para 7 nós				
LI	0,1%	1%	5%	8%
$TE_F$	95,85% ± 1,00%	98,90% ± 1,00%	94,66% ± 1,00%	96,57% ± 1,90%
$TE_S$	95,53% ± 0,89%	95,69% ± 1,00%	95,64% ± 1,00%	95,62% ± 1,90%

## 5.2 Sobre o Tempo de Vida da Rede

A Figura 31 apresenta os resultados relativos ao aumento do tempo de vida da rede, em função do limiar de inovação, para as quantidades de quatro, cinco, seis e sete nós (todas com um sorvedouro). A Figura 31 (a) diz respeito ao algoritmo baseado na fonte, e a Figura 31 (b) apresenta o ganho no tempo de vida da rede, utilizando-se o algoritmo baseado no sorvedouro. Os limiares de inovação utilizados em todas as simulações foram de 0,1%, 1%, 5% e 8%. Além disso, em todos os gráficos apresentados, a “barra” vertical em cada ponto das curvas representa o Intervalo de Confiança utilizado. O tempo de vida está normalizado em relação às simulações sem a utilização de nenhum tipo de economia de energia, onde cada nó simplesmente mede amostras do processo e as envia para o sorvedouro. Utilizou-se um simulador discreto e nesse cenário (sem gerenciamento de energia), a tarefa de medição e transmissão das amostras foi realizada a cada 0,1 segundos; ou seja, utilizou-se uma base de tempo de 0,1 segundos. O aumento do tempo de vida ( $TdV_A$ ) pode ser definido da seguinte forma:

$$TdV_A = \frac{TdV_{alg} - TdV_{sg}}{0,01 \times TdV_{sg}}, \quad (18)$$

em [%], onde,  $TdV_{alg}$  é o tempo de vida da rede com a utilização dos algoritmos; e  $TdV_{sg}$  é o tempo de vida da rede sem a utilização do gerenciamento.

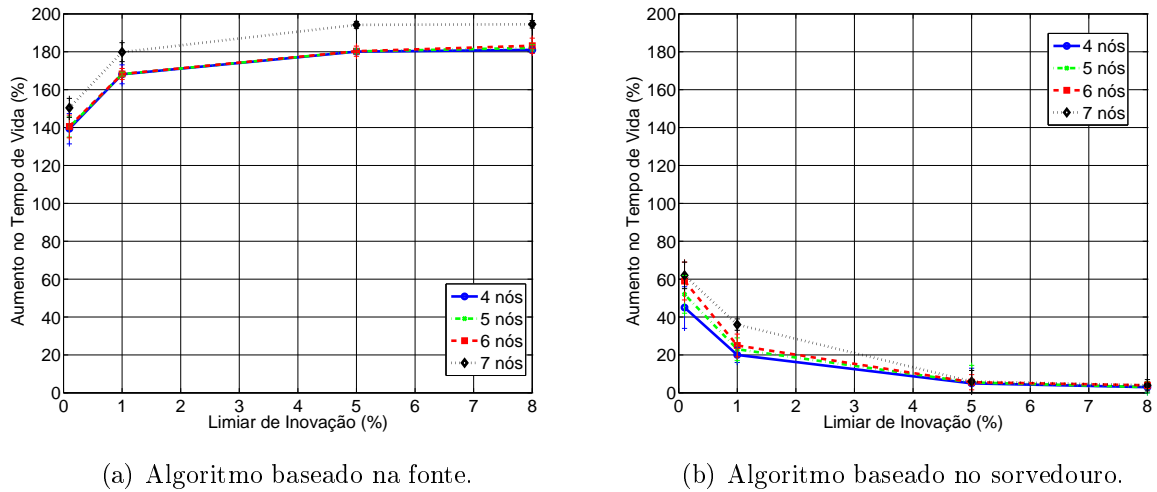


Figura 31 - Aumento no tempo de vida da rede em função do limiar de inovação.

No algoritmo baseado na fonte, verifica-se um aumento no tempo de vida da rede com o aumento do limiar. Isto porque o aumento do limiar faz com que os nós apenas transmitam amostras medidas com variações percentuais maiores. Além disso, os nós permanecem inativos por tempos maiores, entre transmissões inovadoras. Ou seja, a quantidade de transmissões é reduzida, o que gera uma economia de energia maior.

Já no algoritmo baseado no sorvedouro, observa-se um comportamento diferente. Neste, os nós transmitem as amostras sem verificar se estas possuem, ou não, inovação. A verificação da inovação é realizada no próprio sorvedouro que, ao receber as amostras provenientes de cada nó sensor, decide, de forma centralizada, quando os nós ficarão Inativos e por quanto tempo. Portanto, quanto maior o limiar, mais mensagens o sorvedouro terá que receber para que se identifique a variação na medida e se decida os intervalos de inatividade dos nós, já que assume-se que o processo não possui variações rápidas. Logo, não houve muita redução na quantidade de transmissões e, portanto, não se economizou tanta energia quanto no método baseado na fonte.

Observa-se um ganho máximo no tempo de vida de 194,5% para o algoritmo baseado na fonte (para um limiar de inovação de 8%) e um ganho máximo de 62% para o algoritmo baseado no sorvedouro (para um limiar de inovação de 0,1%)

### 5.3 Sobre a Redução nas Transmissões

A Figura 32 apresenta o impacto do aumento do limiar de inovação na redução percentual da quantidade de transmissões, para os dois algoritmos propostos, em relação às simulações sem a utilização do gerenciamento de energia. A Figura 32 (a) apresenta os resultados do algoritmo baseado na fonte, e a Figura 32 (b), apresenta a redução nas transmissões para o algoritmo que utiliza o paradigma do sorvedouro.

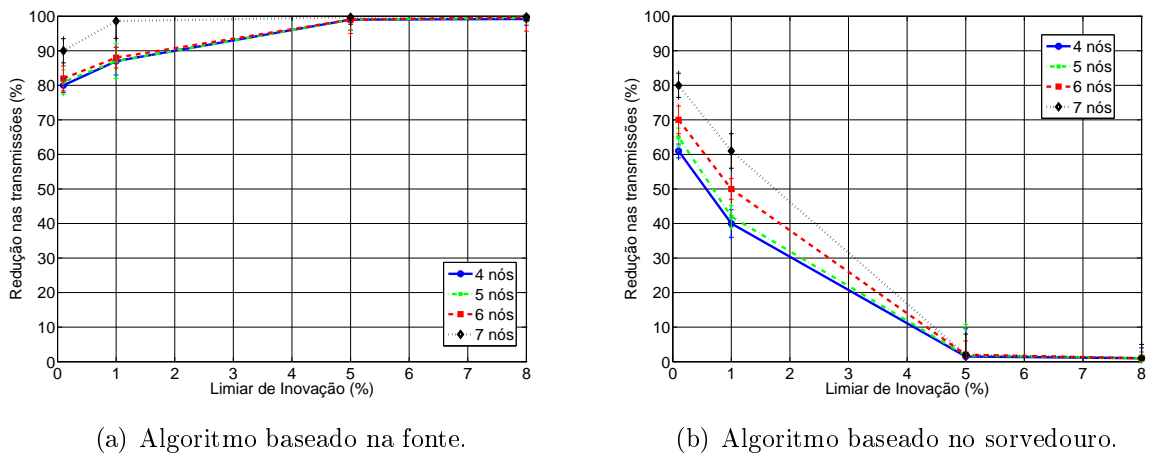


Figura 32 - Redução percentual das transmissões em função do limiar de inovação.

Em relação ao algoritmo baseado na fonte, com o aumento do limiar, há uma redução significativa na quantidade de transmissões em relação às simulações sem o gerenciamento de energia. Entretanto, com um limiar muito grande, a reduzida quantidade de transmissões pode ser insuficiente para a reconstrução do processo, no sorvedouro, o que será discutido na Seção 5.6.

Já no método baseado no sorvedouro, observa-se um aumento na quantidade de transmissões, pois com limiares maiores, o sorvedouro deve receber mais mensagens para que se verifique o nível de variação da medida esperado.

### 5.4 Sobre a Escalabilidade da Rede

Em relação à escalabilidade, verifica-se na Figura 31, que há um aumento no tempo de vida da rede com o aumento da quantidade de nós. Para tentar explicar o motivo desse aumento no tempo de vida é apresentada a Tabela 4, que contém os tempos de vida absolutos (em segundos), para cada quantidade de nós. Cada tempo de vida absoluto foi

obtido através da média das  $n_s$  simulações realizadas. Na tabela,  $Q_n$  é a quantidade de nós;  $TdV_{SG}$  é o tempo de vida absoluto da rede sem o gerenciamento de energia;  $TdV_{L01}$  é o tempo de vida absoluto para um limiar de inovação de 0,1%;  $TdV_{L1}$  é o tempo de vida absoluto para um limiar de inovação de 1%;  $TdV_{L5}$  é o tempo de vida absoluto para um limiar de inovação de 5% e  $TdV_{L8}$  é o tempo de vida absoluto para um limiar de inovação de 8%. A Tabela 4 apresenta os tempos de vida para o algoritmo baseado na fonte e para o algoritmo baseado no sorvedouro. Para melhor visualização da relação dos tempos de vida absolutos para cada quantidade de nós, apresenta-se a Tabela 5, onde os tempos são normalizados em relação às simulações com quatro nós.

Pode-se observar, na Tabela 4 e na Tabela 5, uma redução no tempo de vida absoluto em função do aumento da quantidade de nós da rede (para as simulações sem os algoritmos propostos). Uma possível explicação é o fato de que, quando aumenta-se a quantidade de nós, aumenta a disputa pelo meio, o que pode causar mais colisões. Isso pode levar a um consumo maior por parte dos nós sensores, devido a retransmissões de quadros perdidos. Porém, em relação às simulações com os algoritmos, a escalabilidade da rede não causa muita variação do tempo de vida, pois a quantidade de transmissões é menor. Logo, o ganho (percentual) do tempo de vida (com os algoritmos) parece ser causado principalmente pela redução do tempo de vida absoluto das simulações com o aumento da quantidade de nós sem o gerenciamento de energia.

Na prática, mais importante do que aumentar ou não o tempo de vida da rede (com o seu escalamento) é verificar que o algoritmo continua funcionando, mesmo para redes maiores. Os tempos de vida absolutos diminuem com a complexidade da rede (para as simulações sem o gerenciamento de energia), porém os algoritmos conseguem “contornar” esse problema, na medida em que conseguem levar a um ganho no tempo de vida (em relação às simulações sem o gerenciamento) e fazer com que os nós acompanhem a variação do processo monitorado, o que será melhor discutido na Seção 5.5.

Além disso, pode-se observar na Figura 32 que há uma maior redução na quantidade de transmissões com a escalabilidade da rede, para limiares de inovação menores do que 5%. Porém, esse comportamento não é verificado para limiares a partir desse valor. Pode-se observar na Figura 32 (a) e na Figura 32 (b) que, para limiares altos (a partir de 5%), há um “limite” máximo e outro mínimo para a redução na quantidade de transmissões, onde não há variação significativa na redução obtida, ainda que rede seja escalada.

Tabela 4 - Tempo de vida absoluto em função da escalabilidade (RSSFs de salto único).

Algoritmo baseado na fonte (seg)					
Qn	$TdV_{SG}$	$TdV_{L01}$	$TdV_{L1}$	$TdV_{L5}$	$TdV_{L8}$
4	71,02 ± 0,26	170,02 ± 3,14	190,40 ± 3,19	198,92 ± 1,94	199,56 ± 3,90
5	70,30 ± 0,54	168,93 ± 2,76	188,54 ± 0,60	196,98 ± 0,97	198,31 ± 0,81
6	70,16 ± 0,49	168,80 ± 2,50	188,24 ± 3,18	196,65 ± 0,60	198,69 ± 1,21
7	67,34 ± 0,66	168,61 ± 0,12	188,41 ± 0,22	198,11 ± 0,20	198,31 ± 1,45
Algoritmo baseado no sorvedouro (seg)					
Qn	$TdV_{SG}$	$TdV_{L01}$	$TdV_{L1}$	$TdV_{L5}$	$TdV_{L8}$
4	71,02 ± 0,26	102,95 ± 1,35	85,20 ± 4,12	74,55 ± 1,75	73,13 ± 3,23
5	70,30 ± 0,54	106,85 ± 4,08	86,46 ± 5,75	74,16 ± 6,34	72,41 ± 0,52
6	70,16 ± 0,49	111,55 ± 5,45	87,70 ± 3,82	74,09 ± 6,85	72,96 ± 1,13
7	67,34 ± 0,66	109,09 ± 2,84	91,58 ± 3,51	71,38 ± 0,19	70,03 ± 0,31

Tabela 5 - Tempo de vida absoluto normalizado em relação à quatro nós em função da escalabilidade (RSSFs de salto único).

Algoritmo baseado na fonte					
Qn	$TdV_{SG}$	$TdV_{L01}$	$TdV_{L1}$	$TdV_{L5}$	$TdV_{L8}$
$\frac{Qn_5}{Qn_4}$	1,0000	1,0000	1,0000	1,0000	1,0000
$\frac{Qn_6}{Qn_4}$	0,9874	0,9930	0,9902	0,9902	0,9937
$\frac{Qn_7}{Qn_4}$	0,9854	0,9928	0,9887	0,9886	0,9956
$\frac{Qn_8}{Qn_4}$	0,9458	0,9917	0,9895	0,9959	0,9937
Algoritmo baseado no sorvedouro					
Qn	$TdV_{SG}$	$TdV_{L01}$	$TdV_{L1}$	$TdV_{L5}$	$TdV_{L8}$
$\frac{Qn_5}{Qn_4}$	1,0000	1,0000	1,0000	1,0000	1,0000
$\frac{Qn_6}{Qn_4}$	0,9874	1,0379	1,0148	0,9948	0,9902
$\frac{Qn_7}{Qn_4}$	0,9854	1,0855	1,0293	0,9938	0,9977
$\frac{Qn_8}{Qn_4}$	0,9458	1,0596	1,0749	0,9575	0,9576

Para tentar explicar o motivo do aumento da redução percentual da quantidade de transmissões com a escalabilidade da rede, será apresentada a Tabela 6 e também a Tabela 7. Estas tabelas contêm as quantidades totais de transmissões obtidas nas simulações sem o gerenciamento de energia e com os algoritmos propostos, para cada quantidade de nós. Os valores correspondentes às quantidades de transmissões foram obtidos através da média das  $n_s$  simulações realizadas. Nas tabelas,  $Qn$  é a quantidade de nós;  $Tx_{SG}$  é a quantidade de transmissões sem o gerenciamento de energia;  $Tx_{L01}$  é a quantidade de transmissões para um limiar de inovação de 0,1%;  $Tx_{L1}$  é a quantidade de transmissões para um limiar de inovação de 1%;  $Tx_{L5}$  é a quantidade de transmissões para um limiar de inovação de

5% e  $Tx_{L8}$  é a quantidade de transmissões para um limiar de inovação de 8%. Na Tabela 7, os valores de quantidades de transmissões foram normalizados em função dos valores obtidos para uma quantidade de quatro nós.

Argumentos equivalentes aos que embasaram a explicação do aumento do tempo de vida da rede com sua escalabilidade podem aqui ser utilizados. Pode-se verificar na Tabela 6 e na Tabela 7 que, para às simulações sem o gerenciamento de energia, há um aumento na quantidade de transmissões com o aumento dos nós, conforme esperado. Com a utilização dos algoritmos, para limiares menores do que 5%, observa-se pouca variação das quantidades totais de transmissões. Logo, o aumento na redução percentual de transmissões com a escalabilidade da rede se deve pelo aumento das quantidades totais de transmissões, obtidas nas simulações sem a utilização dos algoritmos, ao escalarmos a rede. Já para limiares a partir desse valor, há uma variação significativa das quantidades totais de transmissões com o aumento da quantidade dos nós, nesse caso havendo pouca variação na redução percentual das transmissões, conforme visto na Figura 32 (a) e na Figura 32 (b).

Tabela 6 - Quantidade de transmissões em função da escalabilidade (RSSFs de salto único).

Algoritmo baseado na fonte					
Qn	$Tx_{SG}$	$Tx_{L01}$	$Tx_{L1}$	$Tx_{L5}$	$Tx_{L8}$
4	2120 ± 16	424 ± 20	275 ± 15	21 ± 3	16 ± 3
5	2828 ± 89	537 ± 14	367 ± 18	28 ± 2	14 ± 2
6	3233 ± 65	581 ± 27	387 ± 20	32 ± 3	12 ± 4
7	4139 ± 39	413 ± 35	206 ± 5	20 ± 2	12 ± 2
Algoritmo baseado no sorvedouro					
Qn	$Tx_{SG}$	$Tx_{L01}$	$Tx_{L1}$	$Tx_{L5}$	$Tx_{L8}$
4	2120 ± 16	826 ± 39	1272 ± 24	2088 ± 28	2098 ± 20
5	2828 ± 89	989 ± 55	1640 ± 93	2777 ± 32	2799 ± 40
6	3233 ± 65	969 ± 50	1616 ± 62	3168 ± 48	3200 ± 21
7	4139 ± 39	827 ± 11	1241 ± 31	4056 ± 21	4097 ± 27

## 5.5 Sobre os Períodos de Inatividade

A Figura 33 apresenta os períodos de inatividade de um dado nó sensor ao longo do tempo, de forma a avaliar o comportamento dos algoritmos propostos (para uma

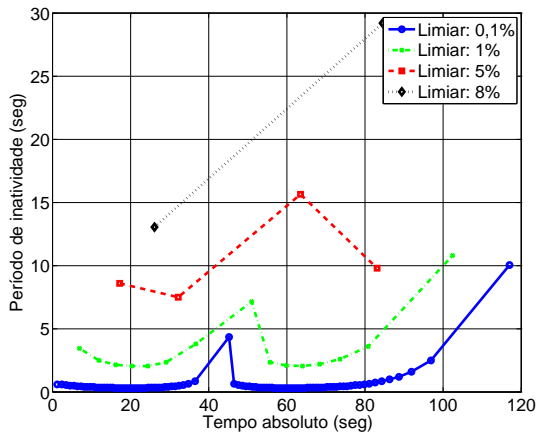
Tabela 7 - Quantidade de transmissões normalizada em relação à quatro nós em função da escalabilidade (RSSFs de salto único).

Algoritmo baseado na fonte					
Qn	$Tx_{SG}$	$Tx_{L01}$	$Tx_{L1}$	$Tx_{L5}$	$Tx_{L8}$
$Qn_4$	1,0000	1,0000	1,0000	1,0000	1,0000
$\frac{Qn_5}{Qn_4}$	1,3340	1,2665	1,3345	1,3333	0,8750
$\frac{Qn_6}{Qn_4}$	1,5250	1,3703	1,4073	1,5238	0,7500
$\frac{Qn_7}{Qn_4}$	1,9524	0,9741	0,7941	0,9524	0,7500
Algoritmo baseado no sorvedouro					
Qn	$Tx_{SG}$	$Tx_{L01}$	$Tx_{L1}$	$Tx_{L5}$	$Tx_{L8}$
$Qn_4$	1,0000	1,0000	1,0000	1,0000	1,0000
$\frac{Qn_5}{Qn_4}$	1,3340	1,1973	1,2893	1,3300	1,3341
$\frac{Qn_6}{Qn_4}$	1,5250	1,1731	1,2704	1,5172	1,5253
$\frac{Qn_7}{Qn_4}$	1,9524	1,0012	0,9756	1,9425	1,9528

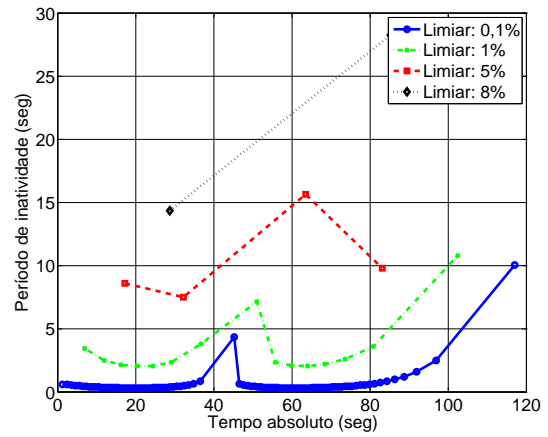
quantidade fixa de sete nós), em relação ao monitoramento do processo  $f(x, y, t)$ . Cada curva ilustra os períodos de inatividade em função de um limiar de inovação, para uma rodada de simulação. Os limiares usados foram de 0,1%, 1%, 5% e 8%. Com o aumento do limiar de inovação, há um aumento nos períodos de inatividade, conforme esperado, já que, aumentar o limiar significa a necessidade de variações percentuais maiores para que haja transmissões inovadoras. Com isso, quando um nó realiza uma transmissão inovadora, ele permanece inativo por um período mais longo. Para a curva correspondente ao limiar de 0,1%, quando o processo monitorado varia mais rápido, os nós permanecem inativos por tempos menores e os pontos com períodos de inatividade maiores são relativos aos momentos em que o processo varia lentamente. Porém, para limiares mais altos, é possível que os nós demorem para encontrar a inovação, por isso verifica-se na curva correspondente períodos de inatividade maiores.

A Figura 34 apresenta os períodos de inatividade de um dado nó ao longo do tempo, cada curva para uma quantidade fixa de nós entre quatro e sete, e todas com um limiar de inovação de 1%, para uma rodada de simulação. A Figura 34 (a) ilustra os períodos de inatividade para o algoritmo baseado na fonte, e a Figura 34 (b) apresenta os períodos de inatividade para o algoritmo baseado no sorvedouro. Pode-se observar que os períodos de inatividade se comportam da mesma forma, para cada quantidade de nós, apresentando valores próximos, já que se utilizou o mesmo limiar de inovação.



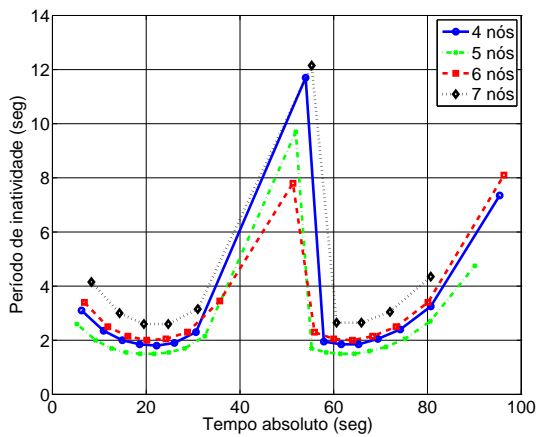


(a) Algoritmo baseado na fonte.

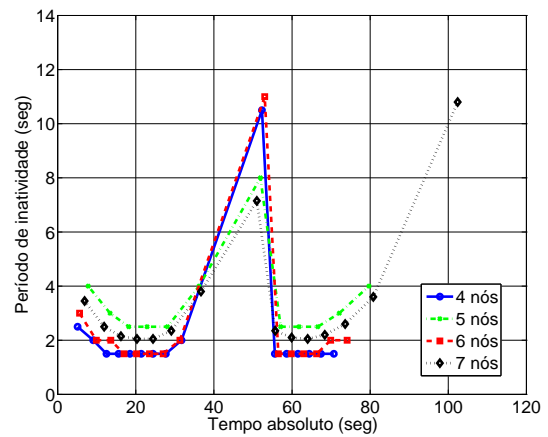


(b) Algoritmo baseado no sorvedouro.

Figura 33 - Períodos de inatividade em função do tempo (sete nós e limiares distintos).



(a) Algoritmo baseado na fonte.



(b) Algoritmo baseado no sorvedouro.

Figura 34 - Períodos de inatividade em função do tempo (limiar de 1%).

A Figura 35 apresenta a derivada parcial do processo monitorado em relação ao do tempo, cuja unidade é  $U/\text{seg}$ , onde  $U$  é a unidade específica do processo. A derivada parcial tem a ver com a taxa de variação do processo, e é definida como:

$$f'(x, y, t) = \frac{\partial f(x, y, t)}{\partial t}. \quad (19)$$

Pode-se verificar pela análise da Figura 33 e da Figura 35 que os sensores conseguem rastrear a taxa de variação do processo. Observa-se que, os menores períodos de inatividade ocorrem aproximadamente nos pontos de maior variação do processo moni-

torado. Além disso, no fim, os períodos de inatividade aumentam, pois o processo varia mais lentamente. Cumpre observar que consegue-se rastrear o processo para limiares pequenos. Para limiares maiores, como 5% e 8% (Figura 33), o rastreamento não ocorre pois tolera-se um certo erro.

Na prática, pode haver um período de inatividade máximo e, portanto, um mecanismo de disparo que acorde os sensores, para que não percam variações caso estejam dormindo. Observa-se, por exemplo, que para limiares altos, os sensores podem demorar muito até encontrar amostras inovadoras e, caso encontrem, podem permanecer inativos durante tempos muito elevados, o que prejudicaria o rastreamento da variação do processo.

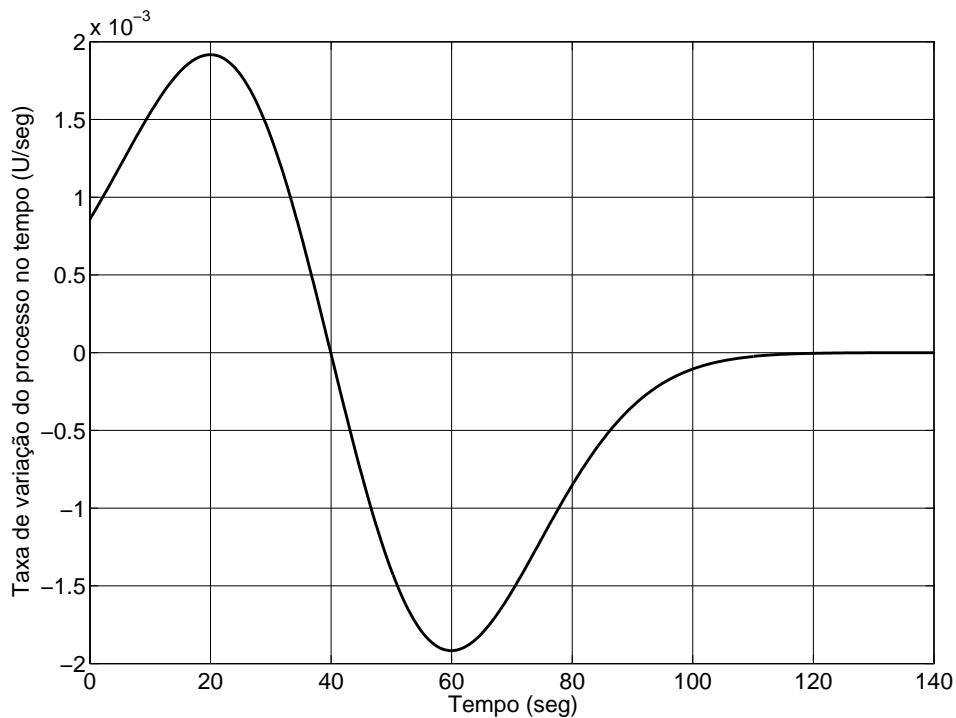
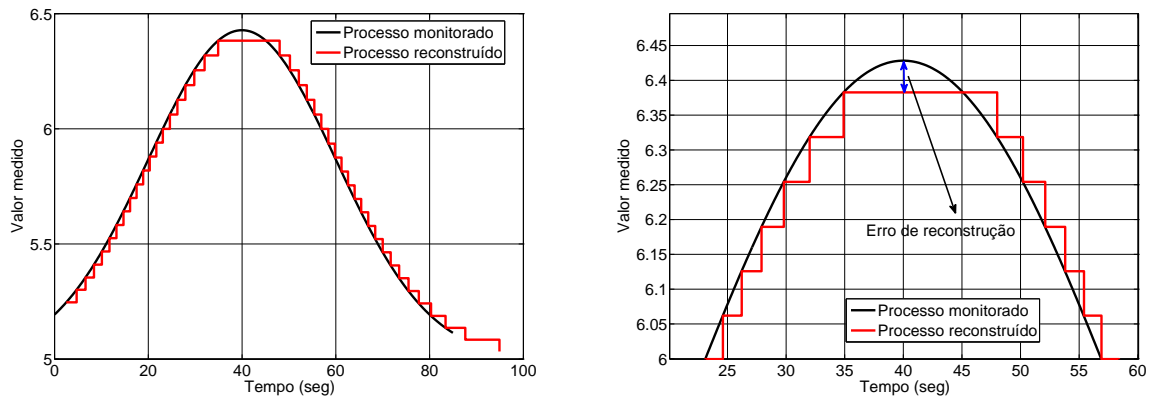


Figura 35 - Taxa de variação do processo monitorado em função do tempo.

## 5.6 Sobre o Erro na Reconstrução do Processo

Esta seção trata de aspectos estatísticos acerca do erro de reconstrução do processo monitorado  $f(x, y, t)$ , onde o erro é definido pela Equação (15). Para o cálculo do erro de reconstrução do processo monitorado, utiliza-se uma modelagem a partir da interpolação do processo, para sua reconstrução. Especificamente, utiliza-se uma interpolação de ordem zero, onde consegue-se reconstruir o processo  $\hat{f}(x_i, y_i, t)$  a partir das amostras recebidas

pelo sorvedouro. A Figura 36 apresenta o processo monitorado  $f(x_i, y_i, t)$  (em preto) para coordenadas  $x$  e  $y$  fixas e ao longo de todo o tempo; e o processo reconstruído (em vermelho), onde apenas as amostras inovadoras são utilizadas no processo de reconstrução. A Figura 36 (a) apresenta os dois processos para uma quantidade fixa de quinze nós e um limiar de inovação de 1% e a Figura 36 (b) mostra uma visão aproximada dos processos, além do erro de reconstrução, que é dado pela diferença entre as curvas, representado pela seta em azul, calculado de acordo com a Equação (15).



(a) Processos monitorado e reconstruído para quinze nós e limiar de 1%. (b) Processos monitorado e reconstruído e erro de reconstrução.

Figura 36 - Reconstrução do processo, a partir da interpolação de ordem zero.

A Figura 37 e a Figura 38 apresentam as Funções Distribuição Cumulativa de probabilidades (CDFs) do erro de reconstrução. A CDF avalia a probabilidade do erro de reconstrução ( $e$ ) ser menor ou igual a um valor  $E$ , ou seja,

$$P(e \leq E) = p. \quad (20)$$

Para o cálculo da CDF do erro, consideram-se fixas as coordenadas  $x$  e  $y$  dos nós sensores,  $x_i$  e  $y_i$ , mas considera-se o tempo  $t$  contínuo, já que a distribuição de probabilidades do erro é calculada ao longo de todo o tempo. A Figura 37 ilustra a CDF do erro para quantidades de nós entre quatro e sete, e um limiar de inovação fixo de 1%; a Figura 38 apresenta a CDF do erro para uma quantidade fixa de sete nós e valores de limiar de inovação de 0,1%, 1%, 5% e 8%. A Figura 37 (a) e a Figura 38 (a) apresentam a CDF do erro, utilizando-se o algoritmo baseado na fonte. Já a Figura 37 (b) e a Figura 38 (b) apresentam a CDF do erro relativa ao algoritmo que utiliza o paradigma do sorvedouro.

Em ambos os algoritmos, verifica-se um aumento no erro máximo de reconstrução em função do aumento do limiar. O aumento do limiar provoca um decréscimo da quantidade de transmissões inovadoras, o que afeta o erro, a partir do momento em que há menos informação para se utilizar na reconstrução do processo. Além disso, observa-se um aumento do erro de reconstrução com o aumento da quantidade de nós. Para um limiar fixo de 1% (Figura 37), há um aumento na redução percentual de transmissões, o que pode ser visto na Figura 32 e explicado na Seção 5.4. Essa redução percentual na quantidade de transmissões causa um aumento no erro de reconstrução do processo monitorado, já que há uma menor quantidade de informação a ser utilizada na reconstrução.

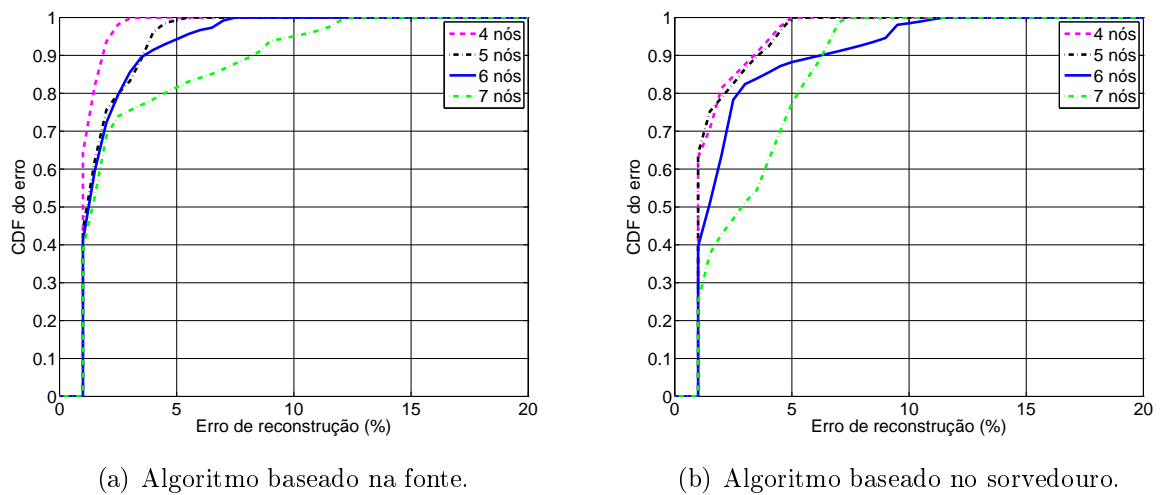


Figura 37 - CDF do erro de reconstrução (quatro a sete nós e limiar de 1%).

A Tabela 8 e a Tabela 9 apresentam algumas estatísticas acerca do erro de reconstrução do processo  $f(x, y, t)$ , onde considerou-se as coordenadas  $x$  e  $y$  dos nós sensores fixas e o tempo contínuo  $t$ . Na Tabela 8 e na Tabela 9,  $Q_n$  é a quantidade de nós; EMedR é o erro médio de reconstrução; EMaxR é o erro máximo de reconstrução; e LI é o limiar de inovação. Além disso, EMedR e EMaxR são os erros médios e máximos nas  $n$  rodadas de simulação. Na Tabela 8, considera-se um limiar de inovação fixo de 1% e quantidades de nós entre quatro e sete. Já na Tabela 9, considera-se uma quantidade fixa de sete nós e limiares de inovação de 0,1%, 1%, 5% e 8%, vistos na coluna LI.

Verifica-se, na Tabela 8, que o erro, em ambas as abordagens (paradigma da fonte e do sorvedouro), aumenta um pouco com o aumento da quantidade de nós. Uma possível explicação é o fato de que, quanto maior a quantidade de nós na rede, mais mensagens

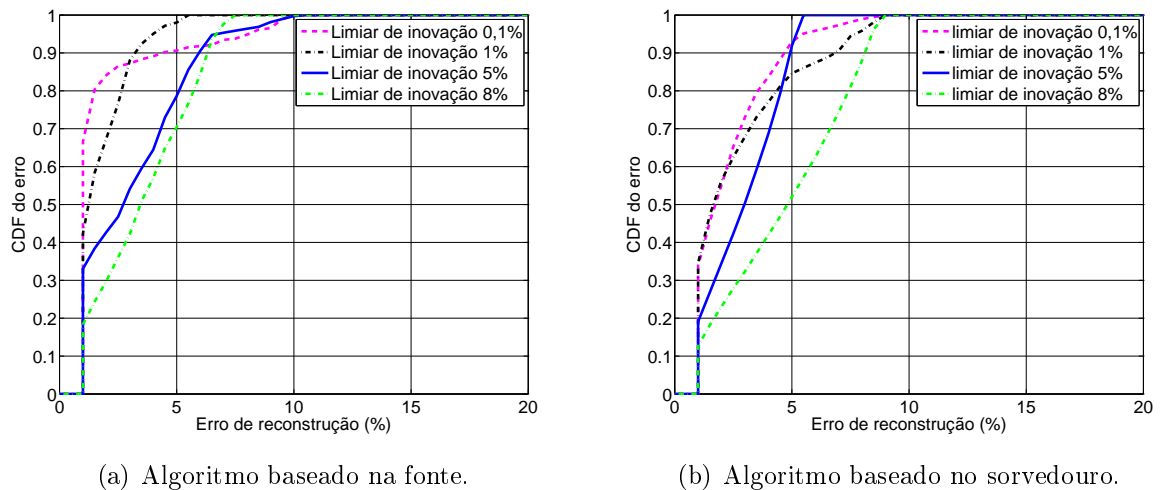


Figura 38 - CDF do erro de reconstrução (limiares de 0,1%, 1%, 5% e 8% e sete nós).

Tabela 8 - Estatísticas do erro de reconstrução (limiar de 1%).

Algoritmo baseado na fonte		
Qn	EMedR	EMaxR
4	0,9%	2,3%
5	1,5%	2,9%
6	1,6%	3,5%
7	2,6%	4,7%
Algoritmo baseado no sorvedouro		
Qn	EMedR	EMaxR
4	1,22%	3,8%
5	1,3%	3,1%
6	2,2%	4,6%
7	3,0%	5,0%

são transmitidas, e maior é a probabilidade de haver colisões. Pode-se ver, na Tabela 3, que a taxa de entrega de pacotes diminui, mesmo que em pequena proporção, com o aumento da quantidade de nós. Na Tabela 9 avaliou-se os erros médios e máximos para uma quantidade fixa de sete nós. Pode-se observar um aumento do erro médio em função do aumento do limiar, conforme visto também na Figura 38. Além disso, verifica-se que, na relação EMedR/LI, erros percentuais maiores ocorrem para pequenos limiares. Enfim, observa-se, na coluna EMaxR/LI, que quanto maior o limiar, o erro máximo tende a ficar mais próximo do limiar de inovação.

Pelos resultados do algoritmo baseado na fonte, mostrados na Figura 31 vê-se que,

Tabela 9 - Estatísticas do erro de reconstrução (sete nós).

Algoritmo baseado na fonte				
LI	EMedR	EMaxR	EMedR/LI	EMaxR/LI
0,1%	1,4%	3,3%	14	33
1%	2,6%	4,7%	2.6	4.7
5%	2,9%	6,8%	0,58	1,36
8%	3,4%	8,6%	0,42	1,07
Algoritmo baseado no sorvedouro				
LI	EMedR	EMaxR	EMedR/LI	EMaxR/LI
0,1%	2,15%	6%	25	60
1%	3%	5,7%	3	5,7
5%	3,8%	5,3%	0,76	1,06
8%	4,6%	8,8%	0,57	1,1

quanto maior o limiar de inovação, maior é a economia de energia pelos nós sensores e, portanto, maior é o ganho no tempo de vida da rede, por conta da redução da quantidade de transmissões. Porém, aumentar indefinidamente o limiar de inovação pode não ser a melhor solução, pois esse aumento pode impactar o erro na reconstrução do processo monitorado. De forma a elucidar esta questão, uma análise (assintótica) do tempo de vida da rede (para o algoritmo baseado na fonte) é representada na Figura 39, para uma quantidade fixa de sete nós. Os limiares de inovação utilizados nessa análise foram de 0,1%, 1%, 5%, 8%, 10%, 15% e 20%, para melhor visualização do comportamento das curvas.

O eixo Y à esquerda apresenta o tempo de vida da rede e o eixo Y à direita mostra o erro máximo na reconstrução do processo. Nesta figura, verifica-se que, assintoticamente, após um limiar de 5%, não há aumento significativo no tempo de vida da rede. Porém, a curva do erro não para de crescer, e quanto maior o Limiar de Inovação estipulado, maior será o erro máximo.

### 5.7 Alguns Pontos Sobre o Paradigma do Sorvedouro

Pode-se observar na Figura 31 e na Figura 32 que o algoritmo baseado no sorvedouro não possui desempenho tão bom quanto o que usa o paradigma da fonte. Mais do que isto, para limiares altos, como os acima de 5%, praticamente não há redução de transmissões e, por conseguinte, pouco ganho no tempo de vida da rede. Para limiares

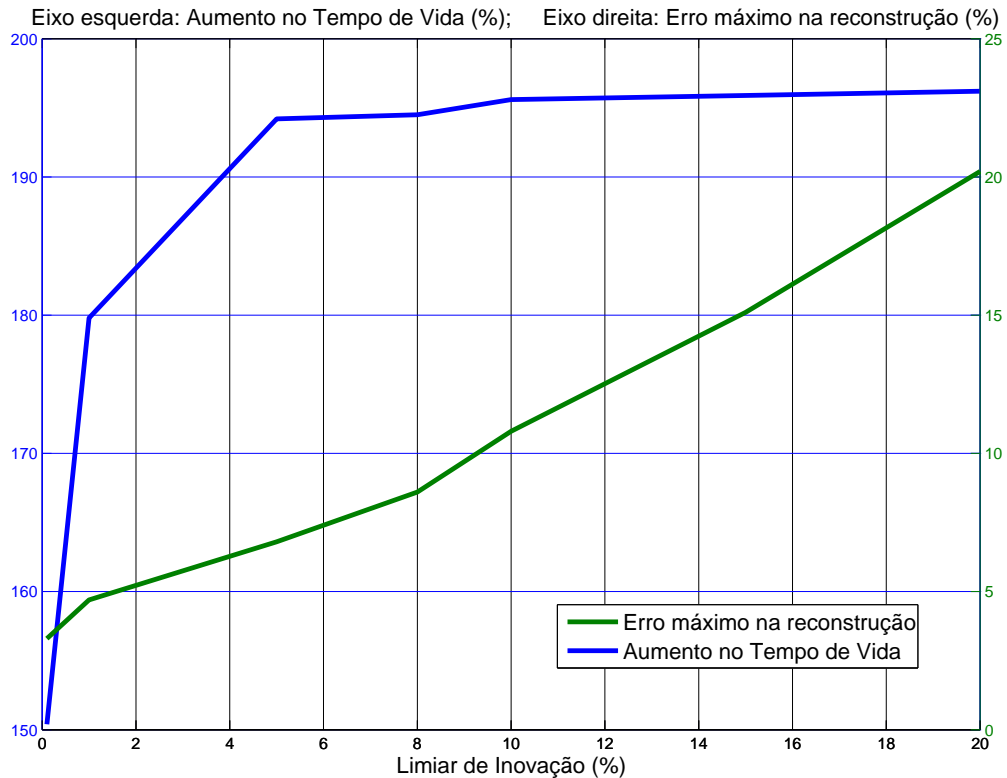


Figura 39 - Análise do tempo de vida e do erro de reconstrução em função do limiar.

altos, o sorvedouro tem que receber muitas amostras para que, então, possa encontrar mensagens com inovação. Isto significa que a convergência do algoritmo pode levar um tempo grande para ocorrer; ou seja, nesta situação, caso o sorvedouro demore muito para encontrar amostras com inovação (se encontrar), os nós podem não dormir e, portanto não economizarão energia.

A partir dessa observação, duas análises serão realizadas, com o objetivo de estudar o desempenho do algoritmo discutido: a análise do tempo de inatividade inicial e a análise da energia inicial dos nós. A justificativa para realizar estas análises apenas no algoritmo baseado no sorvedouro diz respeito à tentativa de aproximar seu desempenho ao desempenho do algoritmo baseado na fonte; ou pelo menos melhorá-lo. As simulações relativas à estas análises utilizam uma quantidade fixa de sete nós.

### 5.7.1 Análise do tempo de inatividade inicial

Como não se sabe *a priori* o tempo de inatividade de cada nó, este parâmetro foi definido como zero no início de cada simulação ( $tInativo_i \leftarrow 0$ ). Porém, se um tempo de inatividade inicial fosse definido, caso não haja inovação, ou seja, caso a variação percentual entre duas amostras seja menor do que o limiar, o nó em questão poderá permanecer Inativo durante este tempo predefinido e, portanto poderá economizar energia, mesmo sem inovação. Isto quer dizer que os nós, nesse caso, dormem não só porque o sorvedouro encontrou inovação e mandou-os dormir, mas também permanecem Inativos por um período pequeno definido previamente. Quando o sorvedouro encontra inovação, os nós permanecem Inativos pelo período calculado pelo sorvedouro, assim como visto anteriormente. Assim, analisamos o impacto da utilização do tempo de inatividade inicial nas simulações e a influência da inserção deste parâmetro nas métricas já avaliadas (tempo de vida da rede, redução na quantidade de transmissões e erro máximo na reconstrução do processo). Os resultados apresentados na Figura 40, na Figura 41 e na Figura 42 possuem 6 curvas, cada qual representando os seguintes tempos de inatividade iniciais: 0 (a mesma curva dos resultados já apresentados); 0,1 segundos; 0,2 segundos; 0,3 segundos; 0,4 segundos; e 0,5 segundos.

A Figura 40 ilustra as curvas para a variação do tempo de vida da rede, normalizado em relação à não utilização da economia de energia (pela Equação (18)), em função do limiar de inovação.

Observa-se que, com a utilização de um tempo de inatividade inicial, há uma nítida melhora no desempenho do algoritmo, principalmente em relação ao problema da pouca economia de energia (para limiares altos) descrita anteriormente. Conseguiu-se um acréscimo de mais de 60 pontos percentuais de ganho no tempo de vida (para o limiar de 0,1%) e de mais de 100 pontos percentuais de ganho para o limiar de 8%.

A mudança de tendência verificada entre os limiares de 0,1% e 1% pode ser explicada da seguinte forma. Com 0,1% de limiar, pequenas variações do sinal recebido pelo sorvedouro levam à inovação. Com isto, conseguiu-se, no algoritmo original, reduzir muito a quantidade de transmissões. Para limiares próximos a 1%, a curva apresenta um crescimento pois soma-se a redução de transmissões devido às mensagens inovadoras, onde os nós dormem durante os tempos calculados, e a redução de transmissões devido ao tempo de inatividade inicial adicionado ao algoritmo.



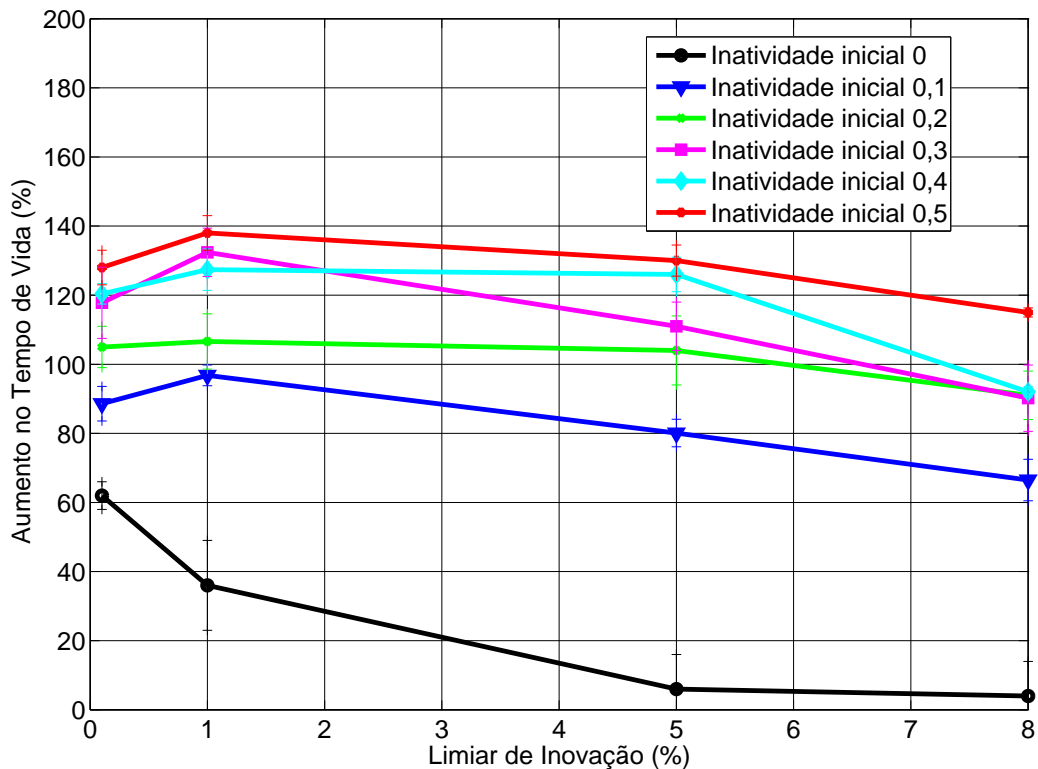


Figura 40 - Tempo de vida em função do limiar de inovação (inatividade inicial entre 0 e 0,5 segundos).

A mesma tendência pode ser observada em relação à redução na quantidade de transmissões (Figura 41), onde obteve-se um acréscimo de até 90 pontos percentuais na redução, em relação às simulações sem o tempo de inatividade inicial (para o limiar de 8%)

A Figura 42 apresenta a CDF do erro de reconstrução do processo, para um limiar de inovação fixo de 1%, onde cada curva apresenta um tempo de inatividade inicial, entre 0,1 e 0,5 segundos. Os resultados, apresentados na Figura 42 e na Tabela 10 (onde  $T_{in}$  é o tempo de inatividade inicial, em segundos), mostram um aumento do erro com o aumento do tempo de inatividade inicial. Caso não haja inovação o nó transmissor dorme durante este período (tempo de inatividade inicial). Porém, para tempos de inatividades maiores, o nó pode dormir, e não realizar uma transmissão inovadora. Isto pode causar um aumento do erro, já que quando o nó está dormindo, o processo continua variando, ao longo do tempo.

A Figura 43 apresenta a CDF do erro de reconstrução para um período de inati-

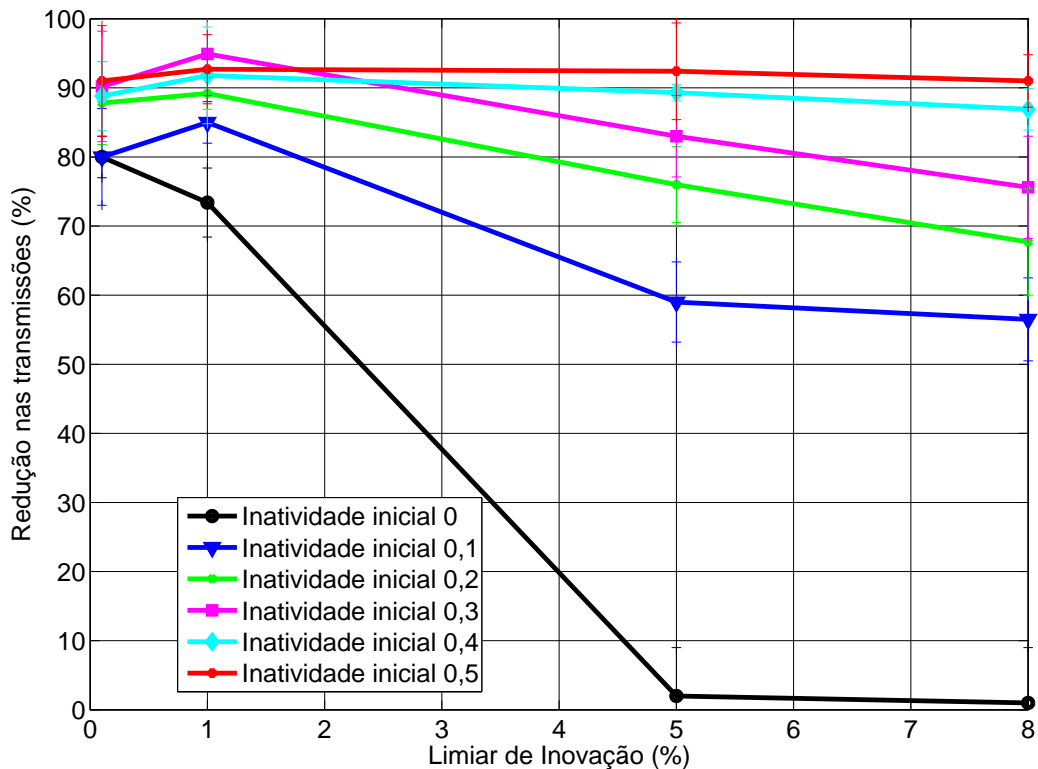


Figura 41 - Redução nas transmissões em função do limiar de inovação (inatividade inicial entre 0 e 0,5 segundos).

vidade inicial fixo de 0,2 segundos, e cada curva diz respeito aos limiares de inovação de 0,1%, 1%, 5% e 8%. O aumento do limiar provoca uma redução de mensagens inovadoras, o que leva a uma menor quantidade de informação utilizada na reconstrução do processo pelo sorvedouro, causando um acréscimo do erro. A Tabela 11 apresenta as estatísticas de erro médio e máximo do processo. Verifica-se, na coluna EMed/LI, que erros percentuais maiores ocorrem para menores valores do limiar de inovação. Além disso, na coluna EMaxR/LI, pode-se verificar que quanto maior o limiar, o erro máximo é mais próximo ao LI.

### 5.7.2 Análise da energia inicial dos nós

A utilização de limiares altos no algoritmo baseado no sorvedouro pode levar a uma lenta convergência do algoritmo para encontrar mensagens inovadoras, já que, nesta situação, o sorvedouro necessita de muitas mensagens para que se encontre uma amostra inovadora. Caso haja uma demora para que se encontre amostras com inovação, quando

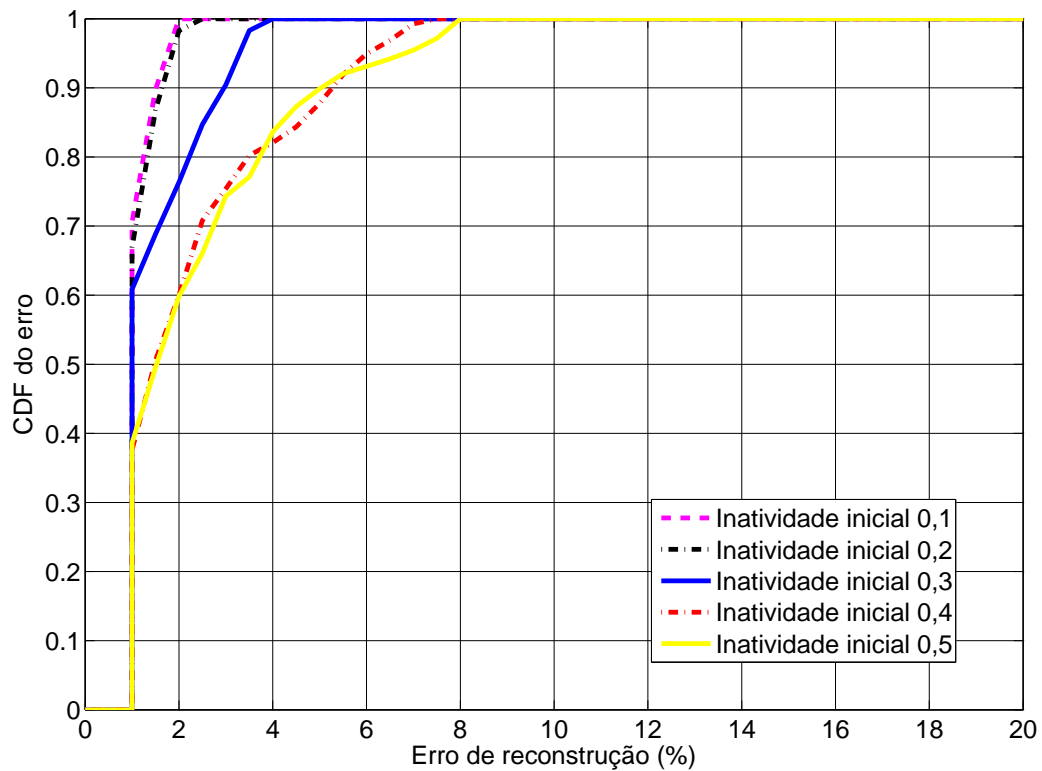


Figura 42 - CDF do erro de reconstrução (sete nós e limiar de 1%).

Tabela 10 - Estatísticas do erro de reconstrução (limiar de 1%).

Tin	EMedR	EMaxR
0,1 seg	0,76%	1,25%
0,2 seg	1,2%	2,7%
0,3 seg	1,9%	3,84%
0,4 seg	2,1%	4,7%
0,5 seg	2,15%	4,78%

Tabela 11 - Estatísticas do erro de reconstrução (inatividade inicial de 0,2 segundos).

LI	EMedR	EMaxR	EMedR/LI	EMaxR/LI
0,1%	0,55%	1,1%	5,5	11
1%	1,2%	2,7%	1,2	2,7
5%	3,8%	6,1%	0,76	1,22
8%	4,5%	8,3%	0,56	1,04

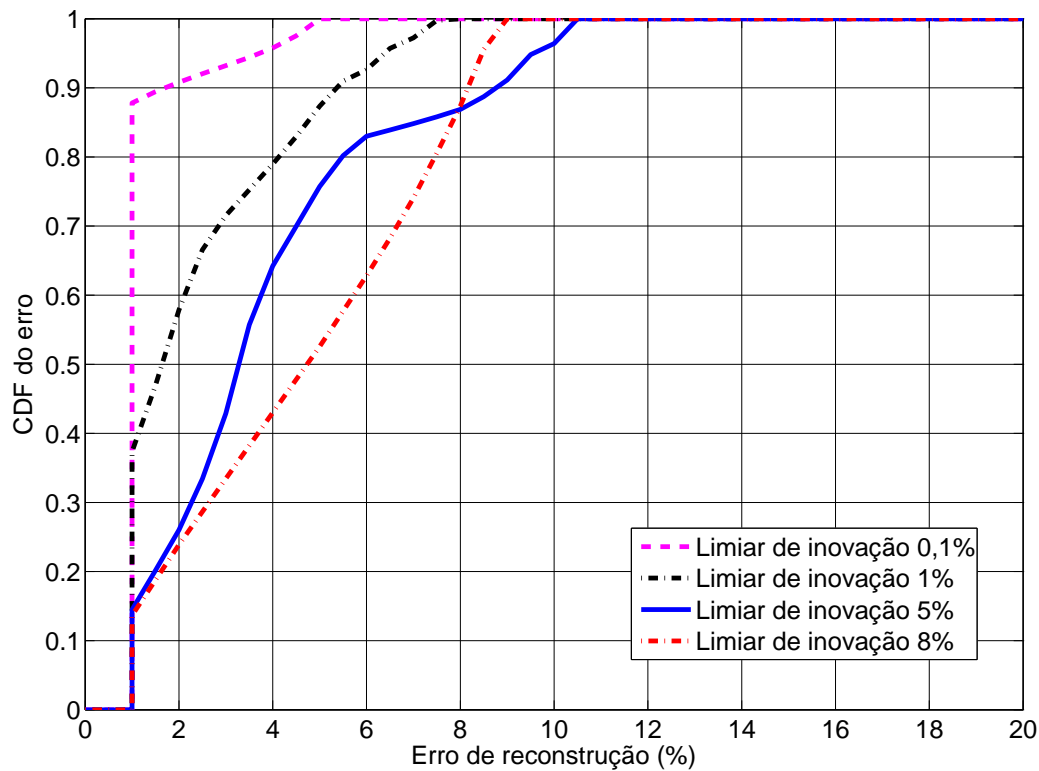


Figura 43 - CDF do erro de reconstrução (inatividade inicial de 0,2 segundos).

isto ocorrer, um dado nó sensor pode já não ter energia armazenada suficiente para que o escalonamento Dormir/Acordar consiga um ganho significativo no tempo de vida da rede.

Uma análise que pode contribuir para entender esta questão é a avaliação do impacto da energia inicial dos nós sensores, mesmo sabendo-se não ser esta muito prática. De fato, utilizar nós sensores com capacidade energética maior pode ser mais custoso, em termos financeiros. Para estudar o impacto da energia inicial dos nós sensores, foram feitas simulações com energia inicial de 3 Joules. Os resultados que serão descritos, possuem duas curvas, uma com a energia usual deste trabalho (2 Joules) e a outra com a alternativa (3 Joules) e as mesmas métricas para análise do tempo de inatividade inicial serão utilizadas.

Em relação ao tempo de vida da rede, utilizado energia inicial de 3 Joules, conseguiu-se ganhos de 86 pontos percentuais em relação às simulações com energia inicial de 2 Joules (para um limiar de 0,1%) e de 13 pontos percentuais (para um limiar de 8%), vistos na Figura 44. O mesmo ocorreu na Redução das transmissões, onde conseguiu-se melhoras de até 19 pontos percentuais (para um Limiar de 8%). Este resultado pode ser visto na

Figura 45. Os ganhos de tempo de vida e de redução nas transmissões já eram esperados, já que para nós com uma energia inicial maior, após a convergência do algoritmo, ainda haveria energia suficiente para que o escalonamento Dormir/Acordar cause uma economia energética dos nós.

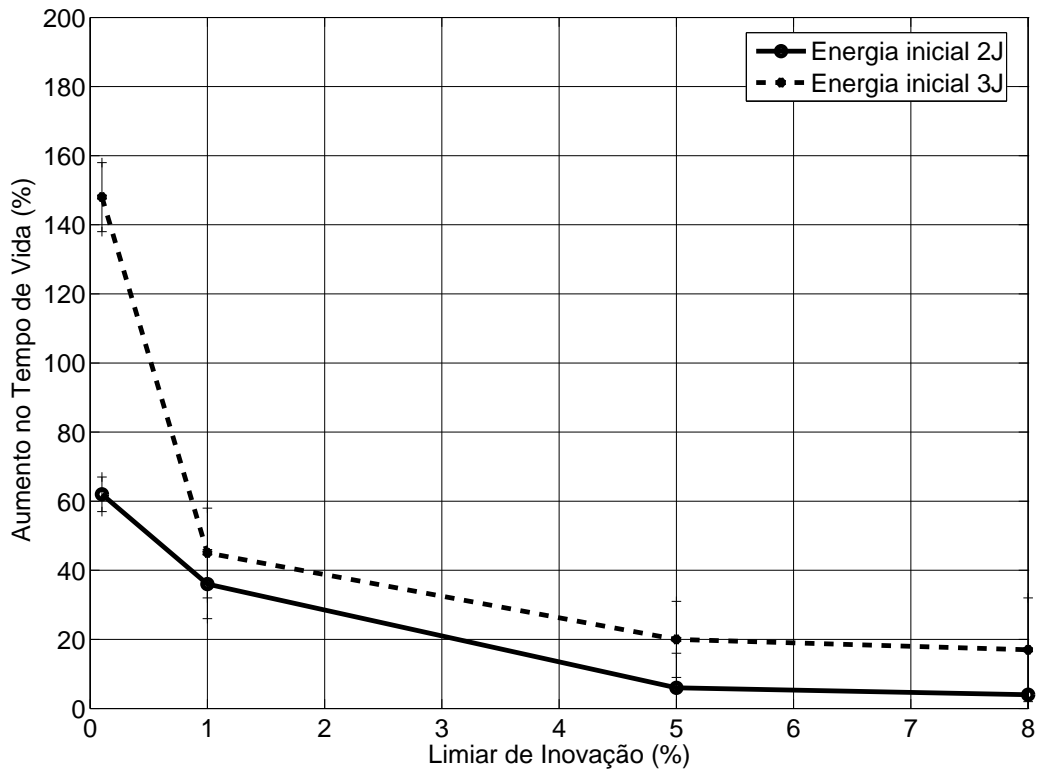


Figura 44 - Aumento no tempo de vida da rede em função do limiar de inovação (energias iniciais de 2 e 3 Joules).

A Tabela 12 apresenta os tempos de vida absolutos (em segundos) para energias iniciais de 2 Joules e de 3 Joules, de modo a investigar se há alguma linearidade nos tempos de vida absolutos obtidos quando se aumenta a energia inicial dos nós. Cada tempo de vida absoluto foi obtido através da média das  $n_s$  simulações realizadas. Na tabela, EI é a energia inicial utilizada, em Joules;  $TdV_{SG}$  é o tempo de vida absoluto da rede sem o gerenciamento de energia;  $TdV_{L01}$  é o tempo de vida absoluto para um limiar de inovação de 0,1%;  $TdV_{L1}$  é o tempo de vida absoluto para um limiar de inovação de 1%;  $TdV_{L5}$  é o tempo de vida absoluto para um limiar de inovação de 5% e  $TdV_{L8}$  é o tempo de vida absoluto para um limiar de inovação de 8%. Para melhor visualização da relação dos tempos de vida absolutos para cada valor de energia inicial, apresenta-se a

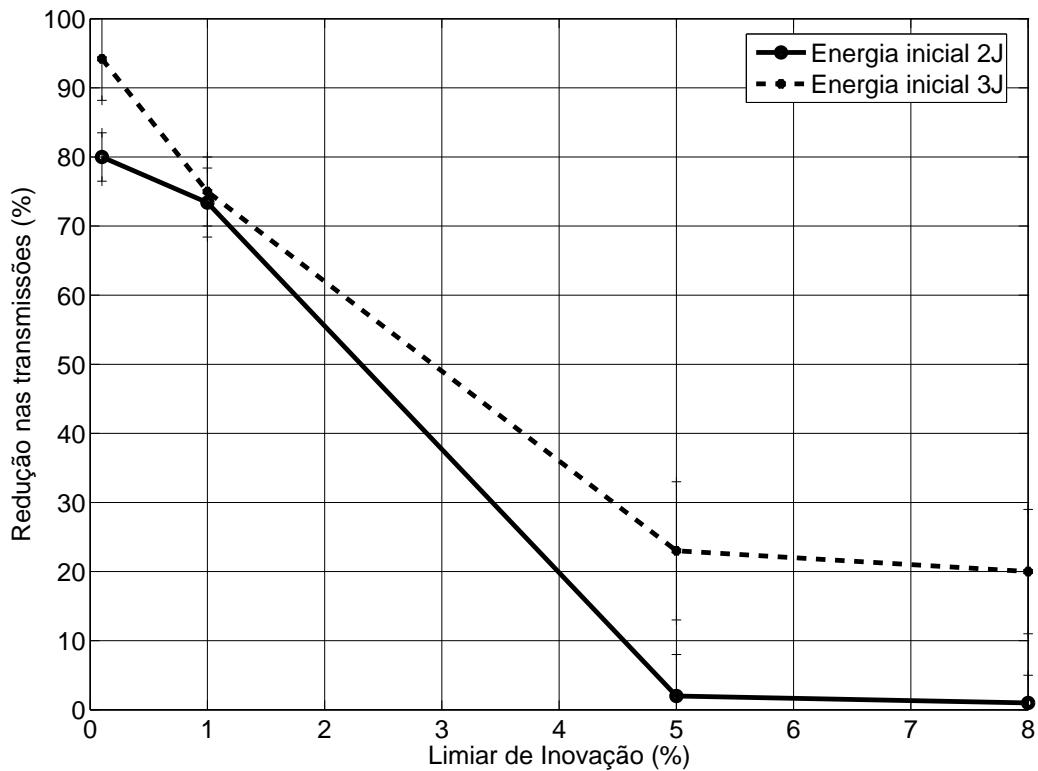


Figura 45 - Redução nas transmissões em função do limiar de inovação (energias iniciais de 2 e 3 Joules).

Tabela 13, onde os tempos são normalizados em relação às simulações com uma energia inicial de 2 Joules.

Conforme esperado, verifica-se na Tabela 12 uma redução nos tempos de vida absolutos com o aumento do limiar de inovação, com a utilização do algoritmo e um aumento nos tempos de vida absolutos quando se aumenta a energia inicial dos nós. Já na Tabela 13, pode-se observar que nas simulações sem o gerenciamento de energia, há um aumento praticamente linear do tempo de vida absoluto em função do aumento da energia inicial dos nós. Verifica-se um aumento de 1,4999 vezes no tempo de vida absoluto para uma energia inicial de 3 Joules, em relação à utilização de nós com 2 Joules de energia inicial nas simulações sem o gerenciamento de energia, o que pode se considerar o mesmo que aumentar a energia inicial de 2 Joules para 3 Joules, ou seja, 1,5 vezes. Porém, com a utilização do algoritmo, pode-se observar que o aumento no tempo de vida absoluto com o aumento da energia inicial dos nós se mostra mais acentuado para limiares pequenos. Na Figura 31, na Figura 44 e na Tabela 4, verifica-se que os melhores resultados em termos do

tempo de vida, para o algoritmo baseado no sorvedouro, ocorrem para limiares pequenos. Já para limiares maiores, pode-se observar, na Tabela 13, que a razão entre os tempos de vida com energias de 3 Joules e de 2 Joules, para cada valor de limiar, é mais próxima a 1,5.

Contudo, para tornar a análise apresentada mais conclusiva, se faz necessário a obtenção de mais resultados, com a utilização de outros valores de energias iniciais.

Tabela 12 - Tempo de vida absoluto em função da energia inicial dos nós.

Algoritmo baseado no sorvedouro (seg)					
EI(J)	$TdV_{SG}$	$TdV_{L01}$	$TdV_{L1}$	$TdV_{L5}$	$TdV_{L8}$
2	$67,34 \pm 0,66$	$109,09 \pm 2,84$	$91,58 \pm 3,51$	$71,38 \pm 0,19$	$70,03 \pm 0,31$
3	$101,00 \pm 1,26$	$250,48 \pm 5,78$	$146,45 \pm 9,53$	$121,20 \pm 19,97$	$118,17 \pm 18,39$

Tabela 13 - Tempo de vida absoluto normalizado em relação à 3 Joules em função da energia inicial dos nós.

Algoritmo baseado no sorvedouro					
EI	$TdV_{SG}$	$TdV_{L01}$	$TdV_{L1}$	$TdV_{L5}$	$TdV_{L8}$
$EI_2$	1,0000	1,0000	1,0000	1,0000	1,0000
$\frac{EI_3}{EI_2}$	1,4999	2,2961	1,5991	1,6980	1,6874

Uma observação importante que deve ser feita diz respeito ao tipo de *hardware* de nós sensores que melhor se aplica a cada algoritmo. No algoritmo baseado na fonte, o processamento é realizado nos próprios nós, que são responsáveis por processar dados e tomar decisões. Portanto, necessitam de um *hardware* mais complexo e, por conseguinte, mais caro. Sendo assim, a utilização de um tempo de inatividade inicial igual a zero é mais viável. Por outro lado, no algoritmo que utiliza o paradigma do sorvedouro, os nós podem ser mais simples, já que estes não realizam “processamento”, e não tomam decisões. Logo, *hardwares* mais baratos podem ser utilizados. Não obstante, pode-se ajustar o tempo de inatividade inicial, de modo que se obtenha um melhor desempenho no algoritmo.

Já a análise da energia inicial dos nós não possui viabilidade prática, já que este não é um parâmetro ajustável dos nós. Sendo assim, a utilização de nós com energia inicial elevada se restringe ao custo na obtenção de *hardwares* com este tipo de especificação.

## 5.8 Conclusão

Este capítulo teve por objetivo apresentar resultados do comportamento dos algoritmos propostos (para RSSFs de único salto), através de simulações. Além das métricas apresentadas no Capítulo 4, duas análises foram feitas para o algoritmo que usa o paradigma do sorvedouro: a análise do tempo de inatividade inicial e a análise da energia inicial dos nós.

No próximo capítulo, serão apresentados os resultados das simulações referentes ao algoritmo proposto (estendido) para RSSFs de múltiplos saltos. Além disso, uma avaliação conjunta entre o tempo de vida da rede e o erro na reconstrução do processo será feita.



## 6 RESULTADOS DO ALGORITMO PARA ECONOMIA DE ENERGIA EM RSSFs DE MÚLTIPLOS SALTOS

Neste capítulo serão apresentados os resultados das simulações do algoritmo (para economia de energia) baseado na fonte estendido, apresentado na Subseção 3.2.1. Inicialmente, serão apresentadas adaptações em dois parâmetros do protocolo de roteamento utilizado na avaliação do algoritmo, de modo a tornar fixas as rotas por ele estabelecidas. Após esta primeira seção, os resultados serão apresentados segundo as seguintes seções: tempo de vida da rede, redução nas transmissões e erro de reconstrução do processo. Por fim, uma análise do compromisso entre o tempo de vida da rede e o erro de reconstrução do processo monitorado será realizada.

### 6.1 Adaptações no Protocolo de Roteamento Utilizado

O protocolo de roteamento usado nas simulações referentes à RSSF de múltiplos saltos foi o AODV [72]. Para avaliar o algoritmo proposto, rotas fixas (estabelecidas pelo protocolo de roteamento) foram utilizadas, para se evitar os gastos energéticos devido ao roteamento. Para isso, dois parâmetros do protocolo AODV foram modificados: *hello interval* e *active route timeout* [72]. O primeiro é um parâmetro usado para a verificação da conectividade de uma dada rota. Periodicamente, cada nó envia (em *broadcast*) uma mensagem chamada de *hello*. Quando um vizinho de um salto (de um dado nó) recebe uma mensagem de *hello*, ele envia um reconhecimento (ACK) confirmando que está ativo na rede. Caso um nó transmissor não receba o ACK, ele transmite o *hello* novamente. Se, após algumas tentativas, o nó não receber o reconhecimento do *hello*, ele assume uma falha no respectivo enlace e envia uma mensagem de erro de rota (RERR) aos seus vizinhos até que os nós fontes sejam avisados, e uma nova rota é estabelecida. Já o *active route timeout* é um parâmetro que define se uma rota está ativa. Caso não haja pacotes sendo transmitidos por uma determinada rota após este intervalo de tempo, a rota é desabilitada. Portanto, para tornar as rotas estabelecidas pelo protocolo de roteamento fixas, esses dois parâmetros foram definidos com um valor muito alto.

## 6.2 Sobre a Taxa de Entrega de Pacotes

Na topologia utilizada nas simulações, a posição do sorvedouro se manteve fixa, porém, para cada simulação, as posições dos nós sensores foram sorteadas de forma aleatória, dentro das áreas especificadas na Tabela 2. Foram utilizadas dez rodadas de simulação para a obtenção de todos os resultados

A Figura 46 apresenta os resultados de taxa de entrega de pacotes (TE) em função do limiar de inovação. Pode-se observar, na Figura 46, uma redução na taxa de entrega, com o aumento do limiar. Com o aumento do limiar de inovação, há uma redução maior na quantidade de transmissões, o que pode impactar a conectividade da rede.

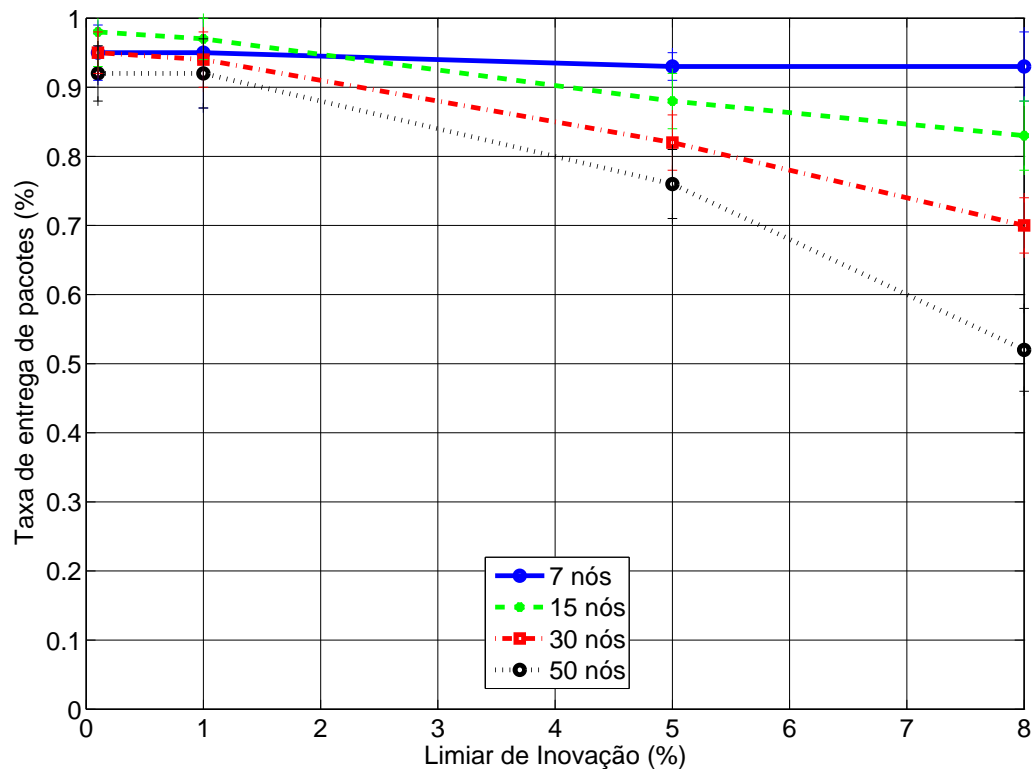


Figura 46 - Taxa de entrega de pacotes em função do limiar de inovação.

## 6.3 Sobre o Tempo de Vida da Rede

A Figura 47 apresenta os resultados referentes ao aumento percentual do tempo de vida da rede (em relação às simulações sem a utilização de gerenciamento de energia) em função do limiar de inovação. Pode-se observar um aumento do tempo de vida, com o

aumento do limiar. Quanto maior o limiar de inovação, maior deve ser a variação percentual entre amostras, para que haja transmissão. Portanto, a quantidade de transmissões na rede é reduzida, o que gera economia de energia por parte dos nós sensores e o consequente aumento no tempo de vida. Além disso, os nós permanecem mais tempo no estado inativo.

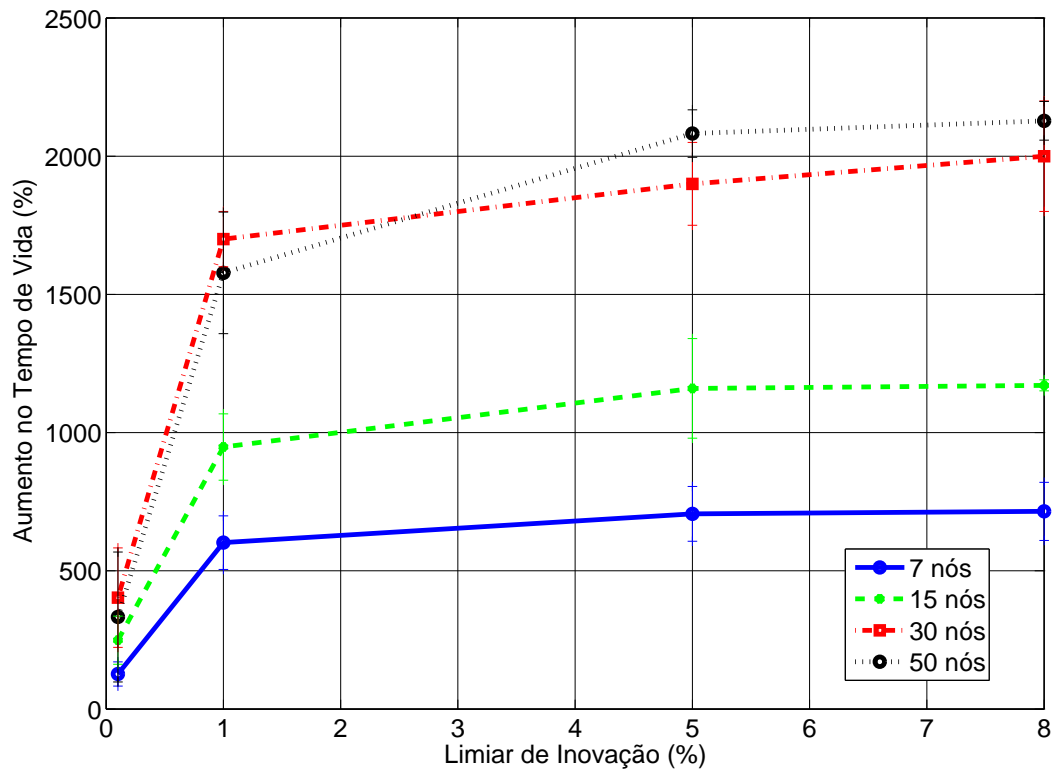


Figura 47 - Aumento no tempo de vida da rede em função do limiar de inovação.

#### 6.4 Sobre a Redução nas Transmissões

O mesmo comportamento observado na avaliação do tempo de vida da rede ocorre em relação à redução de transmissões (em relação às simulações sem a utilização do gerenciamento de energia), cujo comportamento é apresentado na Figura 48. Com o aumento do limiar de inovação, percebe-se uma maior redução na quantidade de transmissões, que, associada à permanência dos nós no estado inativo, leva a uma maior economia de energia pelos nós sensores.

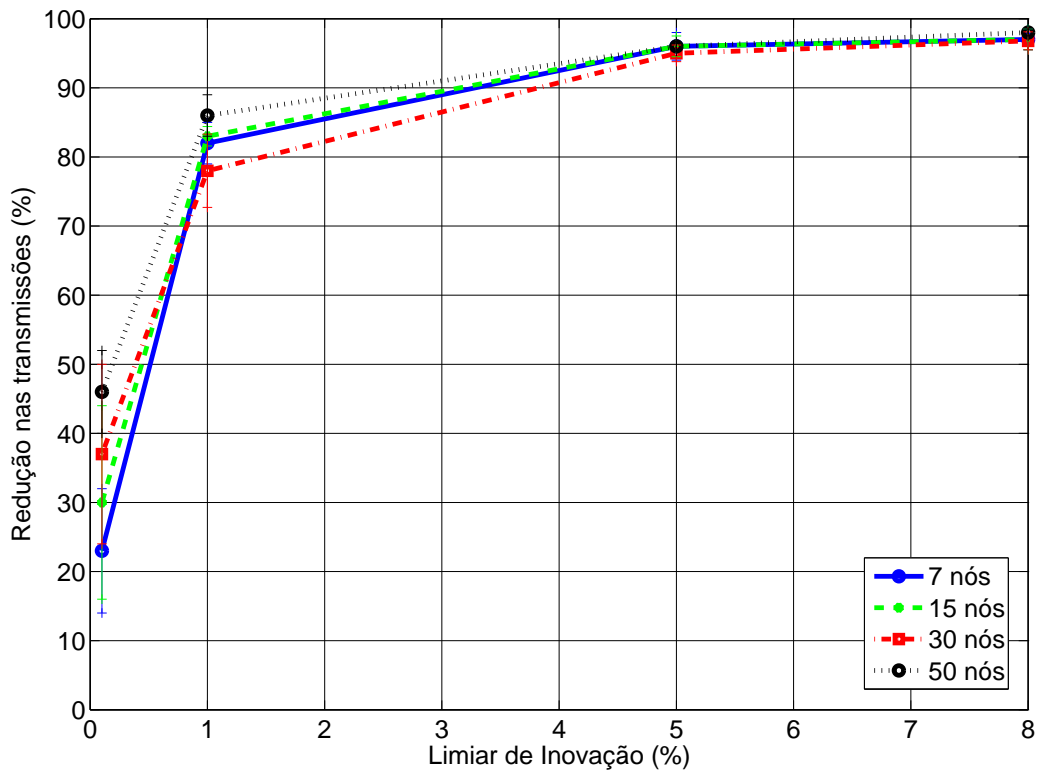


Figura 48 - Redução nas transmissões em função do limiar de inovação.

### 6.5 Sobre a Escalabilidade da Rede

Em relação à escalabilidade, pode-se verificar, na Figura 47, que há um aumento no tempo de vida da rede em função do aumento da quantidade de nós. Os mesmos argumentos utilizados na Seção 5.2 para explicar este fato podem se aplicar para RSSFs de múltiplos saltos. Da mesma forma, os tempos de vida absolutos (em segundos) da rede são apresentados na Tabela 14, onde  $Q_n$  é a quantidade de nós;  $TdV_{SG}$  é o tempo de vida absoluto da rede sem o gerenciamento de energia;  $TdV_{L01}$  é o tempo de vida absoluto para um limiar de inovação de 0,1%;  $TdV_{L1}$  é o tempo de vida absoluto para um limiar de inovação de 1%;  $TdV_{L5}$  é o tempo de vida absoluto para um limiar de inovação de 5% e  $TdV_{L8}$  é o tempo de vida absoluto para um limiar de inovação de 8%. Cada tempo de vida absoluto foi obtido através da média das  $n_s$  simulações realizadas, onde  $n = 10$ . Pode-se ver, na Tabela 14, que o tempo de vida absoluto diminui (para as simulações sem o gerenciamento de energia). Porém, com o algoritmo proposto, simplesmente consegue-se acompanhar melhor o processo monitorado e “contornar” esse problema, já que o algoritmo

leva a um ganho no tempo de vida da rede. Além disso, a Tabela 15 é apresentada, para melhor visualização da relação entre os tempos de vida absolutos (para cada quantidade de nós). Na Tabela 15, os tempos absolutos são normalizados em relação aos tempos de vida absolutos para uma quantidade fixa de sete nós.

Tabela 14 - Tempo de vida absoluto em função da escalabilidade (RSSFs de múltiplos saltos).

Algoritmo baseado na fonte					
Qn	$TdV_{SG}$	$TdV_{L01}$	$TdV_{L1}$	$TdV_{L5}$	$TdV_{L8}$
7	24,76 ± 3,13	56,15 ± 3,58	173,81 ± 14,65	199,56 ± 3,65	201,74 ± 0,77
15	15,94 ± 2,04	55,63 ± 4,95	167,05 ± 30,20	200,84 ± 4,76	202,59 ± 5,00
30	10,07 ± 0,52	50,65 ± 2,83	181,26 ± 32,26	201,40 ± 6,10	213,98 ± 3,04
50	8,56 ± 0,22	37,45 ± 6,54	145,14 ± 34,14	188,74 ± 3,48	192,72 ± 3,79

Tabela 15 - Tempo de vida absoluto normalizado em relação à sete nós em função da escalabilidade (RSSFs de múltiplos saltos).

Algoritmo baseado na fonte (seg)					
Qn	$TdV_{SG}$	$TdV_{L01}$	$TdV_{L1}$	$TdV_{L5}$	$TdV_{L8}$
$Qn_7$	1,0000	1,0000	1,0000	1,0000	1,0000
$\frac{Qn_{15}}{Qn_7}$	0,6438	0,9907	0,9611	1,0064	1,0042
$\frac{Qn_{30}}{Qn_7}$	0,4067	0,9020	1,0429	1,0092	1,0607
$\frac{Qn_{50}}{Qn_7}$	0,3457	0,6670	0,8350	0,9458	0,9553

Assim como discutido na Seção 5.4, pode-se verificar na Figura 48 que há uma maior redução percentual na quantidade de transmissões da rede com a sua escalabilidade, o que ocorre de forma mais acentuada para limiares abaixo de 5%. Observa-se, na Tabela 16 e na Tabela 17, que esse comportamento ocorre devido ao aumento das quantidades totais de transmissões obtidas nas simulações sem o gerenciamento de energia, pelos mesmos motivos apresentados para as RSSFs de salto único.

Nestas tabelas, os valores correspondentes às quantidades de transmissões foram obtidos através da média das  $n_s$  simulações realizadas. Nas tabelas, Qn é a quantidade de nós;  $Tx_{SG}$  é a quantidade de transmissões sem o gerenciamento de energia;  $Tx_{L01}$  é a quantidade de transmissões para um limiar de inovação de 0,1%;  $Tx_{L1}$  é a quantidade de transmissões para um limiar de inovação de 1%;  $Tx_{L5}$  é a quantidade de transmissões para um limiar de inovação de 5% e  $Tx_{L8}$  é a quantidade de transmissões para um limiar

de inovação de 8%. Na Tabela 17, os valores de quantidades de transmissões foram normalizados em função dos valores obtidos para uma quantidade de sete nós.

Tabela 16 - Quantidade de transmissões em função da escalabilidade (RSSFs de múltiplos saltos).

Algoritmo baseado na fonte					
Qn	$Tx_{SG}$	$Tx_{L01}$	$Tx_{L1}$	$Tx_{L5}$	$Tx_{L8}$
7	1489 ± 187	1146 ± 151	268 ± 63	59 ± 11	44 ± 5
15	2283 ± 137	1598 ± 146	388 ± 61	91 ± 13	68 ± 8
30	2933 ± 134	1847 ± 154	645 ± 77	146 ± 14	93 ± 18
50	4263 ± 71	2302 ± 102	596 ± 67	170 ± 26	85 ± 11

Tabela 17 - Quantidade de transmissões normalizada em relação à sete nós em função da escalabilidade (RSSFs de múltiplos saltos).

Algoritmo baseado na fonte					
Qn	$Tx_{SG}$	$Tx_{L01}$	$Tx_{L1}$	$Tx_{L5}$	$Tx_{L8}$
$Qn_7$	1,0000	1,0000	1,0000	1,0000	1,0000
$\frac{Qn_{15}}{Qn_7}$	1,5332	1,3944	1,4478	1,5424	1,5455
$\frac{Qn_{30}}{Qn_7}$	1,9698	1,6117	2,4067	2,4746	2,1136
$\frac{Qn_{50}}{Qn_7}$	2,8630	2,0008	2,2239	2,8814	1,9318

## 6.6 Sobre o Erro na Reconstrução do Processo

A Figura 49 apresenta a CDF do erro de reconstrução do processo monitorado  $f(x, y, t)$ , onde cada curva diz respeito a um valor de limiar de inovação (para uma quantidade fixa de quinze nós). No cálculo da CDF, considerou-se um tempo contínuo  $t$ . A Tabela 18 apresenta algumas características estatísticas acerca do erro de reconstrução, sendo LI o limiar de inovação, EMedR o erro médio de reconstrução, e EMaxR o erro máximo de reconstrução do processo.

Na Figura 49, conforme esperado, pode ser observado um aumento do erro de reconstrução em função do aumento do limiar de inovação. Além disso, verifica-se, nas colunas EMedR/LI e EMaxR/LI da Tabela 18, que erros percentuais maiores ocorrem para menores valores de limiar de inovação. Enfim, pode-se verificar (na coluna EMedR/LI) que quanto maior o limiar, o erro médio de reconstrução tende a ser mais próximo a LI.

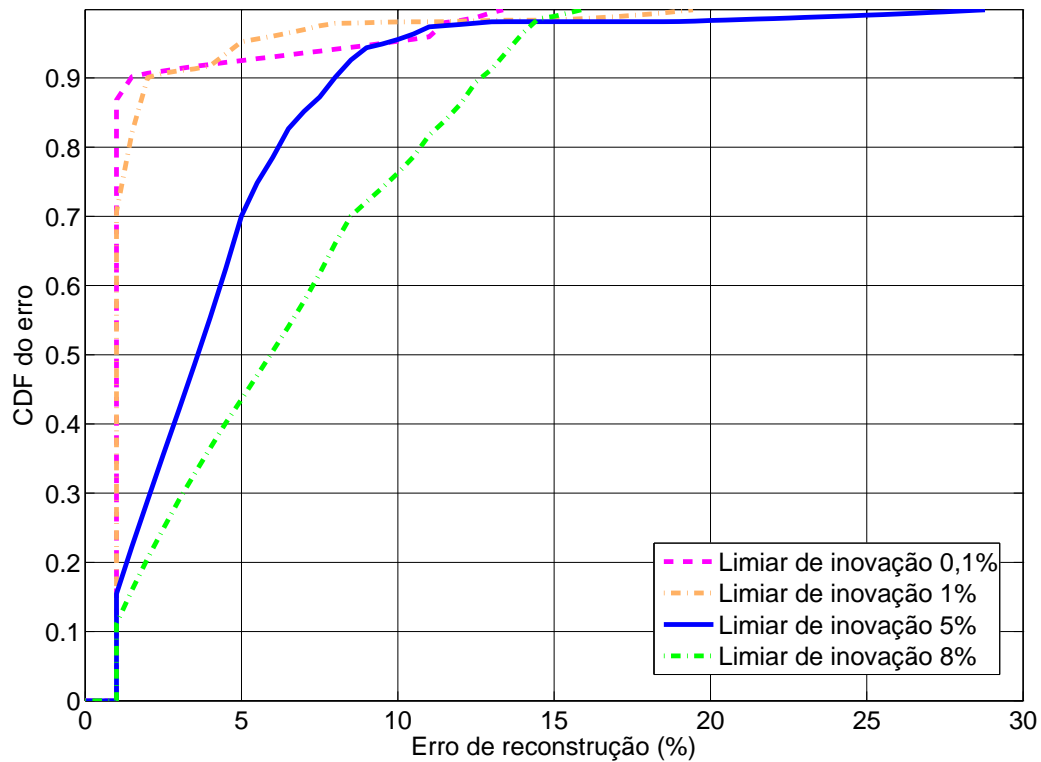


Figura 49 - CDF do erro de reconstrução (quinze nós).

Tabela 18 - Estatísticas do erro de reconstrução (quinze nós).

LI	EMedR	EMaxR	EMedR/LI	EMaxR/LI
0,1%	0,9%	1,9%	9	19
1%	1,3%	3%	1,3	3
5%	4,2%	6%	0,84	1,2
8%	6,3%	9,4%	0,79	1,17

### 6.7 Avaliação Conjunta: Tempo de Vida da Rede e Erro na Reconstrução do Processo

Nesta dissertação, usa-se uma RSSF para se realizar um monitoramento (com eficiência energética) de um processo representado por uma superfície  $f(x, y, t)$ . Logo, uma avaliação importante a ser realizada é o compromisso entre o tempo de vida da rede e o erro de reconstrução do processo monitorado, visto na Figura 50, onde o eixo Y à esquerda representa o tempo de vida da rede, o eixo Y à direita ilustra o erro médio de reconstrução, e como esses dois parâmetros variam em função do limiar de inovação

(eixo X), para uma quantidade fixa de 15 nós. Verifica-se, assim como na análise feita na RSSF de único salto, que após 5% de limiar, o tempo de vida tende à uma assíntota, com aumentos pouco significativos. Porém, quanto maior o limiar de inovação, maior é o erro na reconstrução do processo.

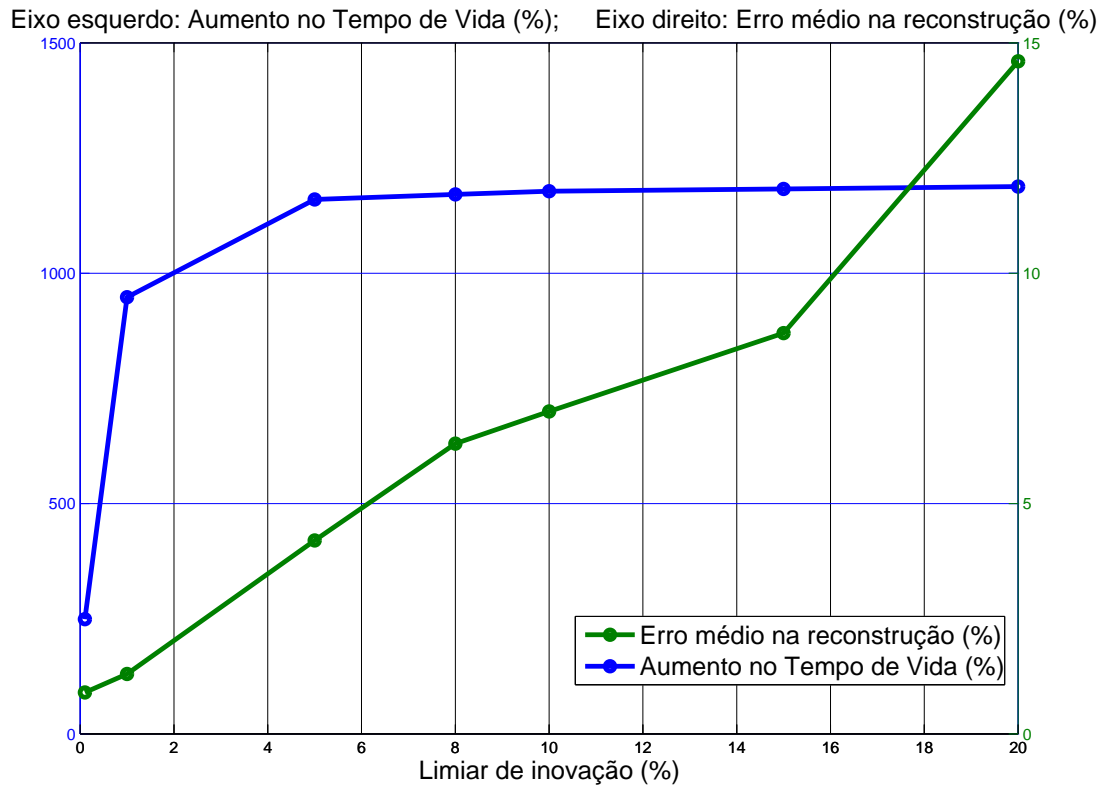


Figura 50 - Análise do tempo de vida e do erro de reconstrução em função do limiar.

Porém, para fins práticos, deve-se avaliar os dois parâmetros (Tempo de Vida e erro na reconstrução) de modo a maximizar a relação entre eles. A Figura 51 apresenta a variação do inverso do aumento do tempo de vida da rede ( $\frac{1}{TdV}$ ) em função do erro médio de reconstrução. Na Figura 51, uma quantidade fixa de quinze nós é utilizada, e cada ponto da curva diz respeito a um limiar de inovação. Os limiares utilizados foram de: 0,1%, 1%, 5%, 8%, 10%, 15% e 20%. Dessa forma, o ponto mais próximo à origem representa a melhor conjugação entre o tempo de vida máximo e o erro de reconstrução mínimo. Para melhor visualização do aumento do tempo de vida, utiliza-se, no eixo y da Figura 52, o logaritmo do inverso do aumento do tempo de vida, ou seja,  $\log(\frac{1}{TdV})$ . A partir da Figura 51 e da Figura 52, pode-se observar o compromisso entre o tempo de vida da rede e o erro de reconstrução, além de incluir uma melhor visualização do limiar de



inovação utilizado, já que esse parâmetro é representado por pontos nas figuras. Porém, tal compromisso depende do que se quer, em termos práticos, se a intenção for possuir um monitoramento durante um tempo maior, mas que permita um erro de reconstrução maior; ou se for desejável um erro menor, porém com um período de monitoramento menor. Pode-se ver, em ambas as figuras, que, dentre os pontos usados, aquele que é relativo ao limiar de 1% está mais próximo à origem. Portanto, este poderia ser um bom limiar, para que haja uma boa conjugação entre tempo de vida máximo e erro de reconstrução mínimo.

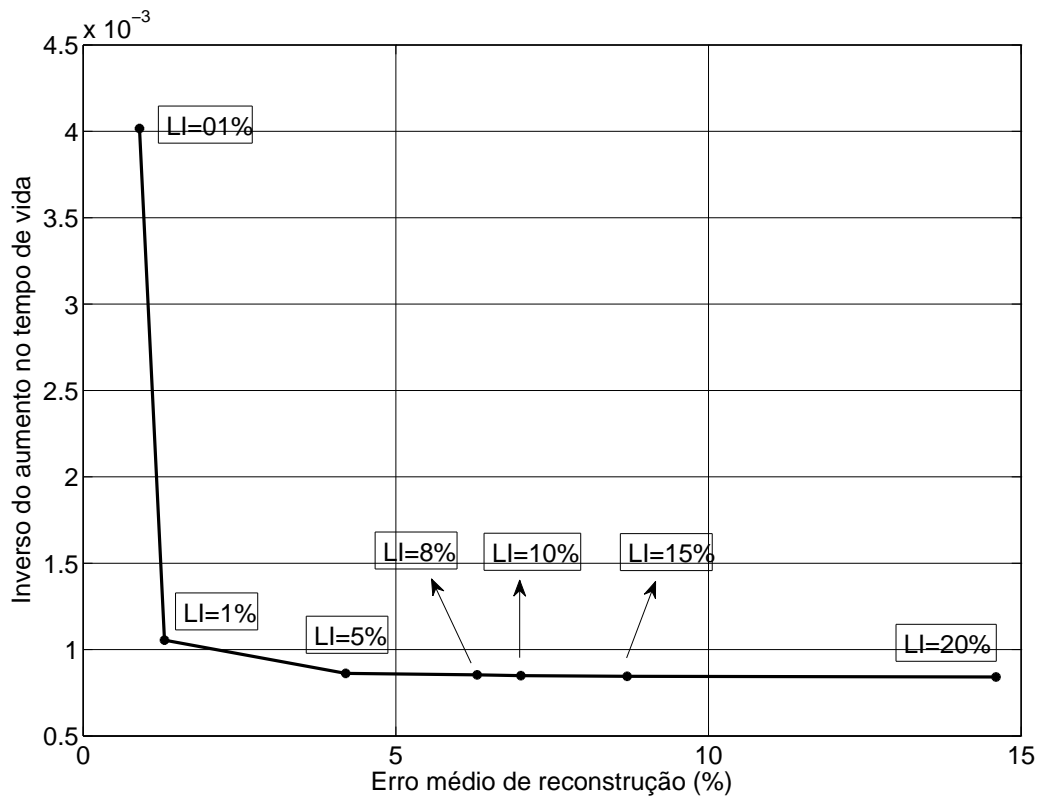


Figura 51 - Inverso do aumento no tempo de vida em função do erro médio.

## 6.8 Conclusão

Este capítulo teve por objetivo apresentar resultados do comportamento do algoritmo proposto, para RSSFs de múltiplos saltos, através de simulações. Realizou-se também uma avaliação conjunta entre o tempo de vida da rede e o erro na reconstrução do processo.

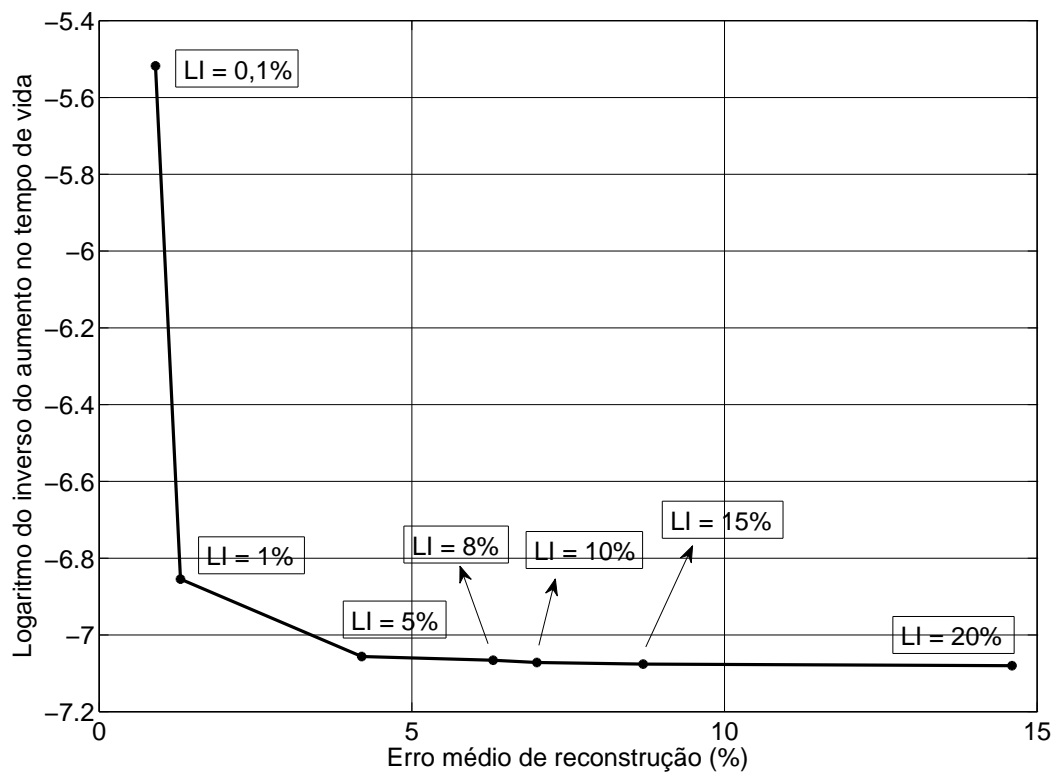


Figura 52 - Logaritmo do inverso do aumento no tempo de vida em função do erro médio.

No próximo capítulo, serão apresentadas as conclusões desta dissertação e discutidos alguns possíveis trabalhos futuros.

## CONCLUSÃO

Nesta dissertação, foram propostos e avaliados algoritmos (na camada de aplicação) para a economia de energia em Redes de Sensores Sem Fio aplicadas ao monitoramento de um processo  $f(x, y, t)$ . Os algoritmos propostos têm por objetivo aumentar a autonomia da rede e, assim, prolongar seu tempo de vida. Inicialmente, implementou-se um modelo de energia baseado em estados no ambiente de simulação. O modelo utilizado é uma extensão de um modelo de energia encontrado na literatura, onde incluiu-se alguns modos de operação dos nós, como o modo de medição e o modo de processamento, de forma a tornar o modelo mais realista.

Os algoritmos propostos se valem da ideia de que, para algumas aplicações, não há a necessidade de se transmitir todas as amostras medidas pelos nós sensores. Apenas amostras consideradas com uma quantidade de inovação serão transmitidas. Além disso, explora-se a utilização de um estado de inatividade pelos nós sensores (estado de economia de energia), de modo a aumentar a economia de energia pelos nós. No desenvolvimento dos algoritmos, considerou-se RSSFs que utilizam dois tipos de “coberturas”: uma RSSF que usa comunicação de salto único, onde o sorvedouro está posicionado dentro do raio de alcance dos nós sensores; e uma RSSF que usa comunicação de múltiplos saltos, através do roteamento dos pacotes transmitidos pelos nós sensores até o sorvedouro. Os algoritmos foram avaliados segundo algumas métricas, como: o aumento no tempo de vida da rede, a redução nas transmissões pelos nós sensores, o erro de reconstrução do processo monitorado e a taxa de entrega de pacotes.

### RSSFs de Salto Único

Em RSSFs de salto único foram propostos dois algoritmos: um baseado na fonte e outro baseado no sorvedouro. Em relação ao *hardware*, os nós sensores utilizados no algoritmo baseado na fonte necessitam de algum poder de processamento, já que esses são responsáveis por realizar cálculos e tomar decisões, diferentemente do baseado no sorvedouro. Não foi considerado no escopo desta dissertação quantificar o poder computacional necessário. Já em relação ao algoritmo baseado no sorvedouro, todo o processamento e tomada de decisões são realizados diretamente no sorvedouro, o que possibilita a utilização

de nós sensores mais simples em termos computacionais.

A partir de agora, nesta subseção, os tópicos serão apresentados conforme a seguinte divisão: sobre os algoritmos baseados na fonte e sobre os algoritmos baseados no sorvedouro.

#### Sobre o algoritmo baseado na fonte

Como nesse algoritmo, os nós apenas transmitem as amostras, tais que sua variação percentual entre a amostra atual e a anteriormente transmitida seja maior do que um limiar de inovação, há uma redução na quantidade de transmissões em função (em relação à não utilização de nenhum tipo de gerenciamento de energia) do aumento do limiar de inovação. Essa redução da quantidade de transmissões leva a um aumento no tempo de vida da rede, já que economiza-se energia com a diminuição das transmissões, e também por os nós permanecem inativos por períodos de tempo maiores.

Em relação à escalabilidade da rede, observamos que há um aumento do tempo de vida com o aumento da quantidade de nós da rede. Vimos que este aumento se deve principalmente à redução do tempo de vida (absoluto) com o aumento da quantidade dos nós (nas simulações sem gerenciamento), devido a um aumento das colisões e, conseqüentemente, das retransmissões realizadas. Porém, o algoritmo proposto consegue “contornar” essa questão, já que leva a um aumento no tempo de vida e consegue acompanhar a variação do processo monitorado, mesmo em redes mais complexas. Esse raciocínio também pode ser aplicado para os outros algoritmos propostos (algoritmo baseado no sorvedouro e algoritmo para RSSFs de múltiplos saltos).

Vimos ainda que os períodos de inatividade calculados pelos nós conseguem rastrear a taxa de variação do processo. Quando o processo varia mais rapidamente, observamos que os nós dormem por períodos de tempo menores, para que se consiga acompanhar a variação e não se percam informações importantes acerca do processo. A partir desse estudo, verificamos a viabilidade de haver um período máximo de inatividade, e um mecanismo de disparo para que os sensores possam detectar possíveis variações mais bruscas do processo. Vimos que a análise da taxa de variação do processo também se aplica ao algoritmo baseado no sorvedouro.

Além disso, foi analisada a distribuição de probabilidades do erro de reconstrução

do processo. Quando um nó está dormindo, o processo pode variar, o que leva a um erro na reconstrução do processo monitorado. Vimos que, para limiares altos, o tempo de vida tende a uma assíntota (apresentando variações pouco significativas). Porém, o erro continua a aumentar, com o aumento do limiar. Ainda foi verificado que erros maiores ocorrem para valores menores de limiar, e que para limiares maiores, o erro se aproxima desse parâmetro. Não obstante, para limiares grandes, há uma redução muito grande na quantidade de transmissões na rede, o que contribui para a elevação do erro.

Por fim, verificou-se uma taxa de entrega de pacotes alta (acima de 90%), em ambos os algoritmos, o que pode mostrar que a utilização dos algoritmos não afeta a conectividade da rede.

#### Sobre o algoritmo baseado no sorvedouro

No algoritmo baseado no sorvedouro, verificou-se um comportamento contrário, em relação ao tempo de vida da rede e da redução nas transmissões. Para limiares maiores, o sorvedouro precisa receber muitas mensagens para que encontre uma mensagem inovadora e, com isso, mande o respectivo nó dormir. Logo, com o aumento do limiar, observou-se um aumento na quantidade de transmissões na rede, o que levou a uma redução no seu tempo de vida. Em relação ao erro na reconstrução, a mesma tendência vista no algoritmo baseado na fonte foi observada, já que esse nó utiliza as recepções inovadoras para o cálculo do erro, e com o aumento do limiar, é necessário que haja maior variação percentual para que ocorra a inovação.

Porém, vimos que o desempenho (em termos do tempo de vida) do algoritmo baseado no sorvedouro foi pior do que o encontrado no algoritmo baseado na fonte, principalmente para limiares maiores, já que não há redução significativa de transmissões. Com isso, um estudo foi realizado nesse algoritmo, através de duas avaliações: do tempo de inatividade inicial, e da energia inicial dos nós. Na primeira avaliação, quando o sorvedouro calcula e não encontra a inovação, ou seja, quando a variação percentual entre recepções é menor do que o limiar, o nó sensor dorme por um período pequeno predefinido. Observou-se melhora no tempo de vida da rede, inclusive com um pequeno aumento desse parâmetro para limiares entre 0,1% e 1%, em relação à não utilização do tempo de inatividade inicial. Porém, vimos um aumento do erro de reconstrução, com o aumento

do tempo de inatividade inicial, já que o nó pode estar dormindo enquanto há variações no processo. Na segunda análise, com a utilização de nós com energia inicial maior, vimos que há uma demora na convergência do algoritmo, o que significa que ele demora para encontrar a primeira amostra com inovação. Caso o nó possua capacidade energética maior, após a convergência do algoritmo, o nó ainda possuirá energia suficiente para que o escalonamento Dormir/Acordar consiga um ganho significativo no tempo de vida da rede. Verificou-se um ganho no tempo de vida da rede, e um aumento pequeno nos erros médios e máximos de reconstrução do processo.

### **RSSFs de Múltiplos Saltos**

Em RSSFs de múltiplos saltos, usa-se um protocolo de roteamento para que as mensagens transmitidas sejam encaminhadas do nó origem até o nó destino (sorvedouro). Nesta dissertação, foi proposta uma extensão do algoritmo baseado na fonte, em que cada nó sensor é fonte de informação, pois coleta dados da superfície monitorada, mas também é roteador, pois encaminha as mensagens de seus nós vizinhos.

#### Sobre a extensão do algoritmo baseado na fonte

Nesse algoritmo, insere-se um mecanismo que permita sua utilização em RSSFs de múltiplos saltos. Os nós permanecem inativos por um período mínimo, tal que consigam acordar em tempo de encaminhar as mensagens de seus vizinhos de um salto. As métricas utilizadas na avaliação foram as mesmas utilizadas na avaliação dos algoritmos para RSSFs de salto único: o tempo de vida da rede, a redução na quantidade de transmissões, o erro de reconstrução do processo e a taxa de entrega de pacotes.

Quanto ao tempo de vida da rede, vimos que esse parâmetro aumenta em função do aumento do limiar de inovação. Com o aumento do limiar, há uma maior redução na quantidade de transmissões pelos nós, e isso leva ao aumento no tempo de vida da rede, assim como observado no algoritmo baseado na fonte, em RSSFs de único salto.

Em relação ao erro na reconstrução do processo, observou-se que esse aumenta em função do crescimento do limiar de inovação, pelos mesmos motivos apresentados para o algoritmo baseado na fonte em RSSFs de salto único.

Observou-se também uma redução da taxa de entrega em função do aumento do limiar. O aumento do limiar leva a uma redução da quantidade de transmissões. Para limiares altos, onde verifica-se grande redução nas transmissões, pequenas perdas de pacotes podem possuir um grande impacto na taxa de entrega. Além disso, vimos que essa métrica se mostrou pior para limiares mais altos (conforme comentado), e ainda mais presente para redes com maior quantidade de nós.

### **Possíveis Trabalhos Futuros**

Futuramente, pretende-se estudar e desenvolver um modelo analítico para o comportamento do erro de reconstrução e o limiar de inovação. O objetivo é conseguir calibrar o limiar, a partir de um erro máximo aceitável (predefinido). Além disso, pretende-se avaliar o algoritmo baseado na fonte para RSSFs de múltiplos saltos, utilizando outros protocolos de roteamento, como o AODV-E (*Energy-Aware AODV*) [82], uma variante do protocolo AODV, que considera melhor o consumo de energia dos nós. Não obstante, pretende-se avaliar os algoritmos através da utilização do roteamento dinâmico, onde as rotas estabelecidas pelo protocolo podem variar ao longo do tempo. Desta forma, uma nova avaliação da energia pode ser utilizada, onde o conceito de tempo de vida da rede pode ser estendido para além da morte do primeiro nó sensor. Uma ideia é analisar a quantidade de nós mortos ao longo do tempo, e de que forma isto pode afetar, por exemplo, a conectividade da rede.

## REFERÊNCIAS

- [1] PANTHACHAI, Y.; KEERATIWINTAKORN, P. An Energy Model for Transmission in Telos-Based Wireless Sensor Networks. In: *Proceedings of the 4th International Joint Conference on Computer Science and Software Engineering*. [S.l.: s.n.], 2007.
- [2] AKYILDIZ, I. F. et al. Wireless Sensor Networks: A Survey. *Computer Networks*, v. 38, p. 393–422, 2002.
- [3] BECKWITH, R.; TEIBEL, D.; BOWMEN, P. Pervasive Computing and Proactive Agriculture. In: *2nd International Conference on Pervasive Computing*. Áustria: [s.n.], 2004.
- [4] LOUREIRO, A. A. F. Redes de Sensores Sem Fio - Minicurso. In: *Simpósio Brasileiro de Redes de Computadores*. Rio Grande do Norte, Brasil: [s.n.], 2003. p. 179–226.
- [5] MAINWARING, A. et al. Wireless Sensor Networks for Habitat Monitoring. In: *ACM International Workshop on Wireless Sensor Networks and Applications*. EUA: [s.n.], 2002. p. 88–97.
- [6] CUNHA, D. O. *Redes Sem Fio de Múltiplos Saltos: Protocolos Específicos para Aplicações e Roteamento com Suporte à Diversidade Cooperativa*. Tese (Doutorado) — Universidade Federal do Rio de Janeiro - COPPE, 2008.
- [7] ARAÚJO, R. B.; VILLAS, L. A.; BOUKERCHE, A. Uma Solução de QoS com Processamento Centrado para Redes de Atuadores e Sensores Sem Fio. In: *Simpósio Brasileiro de Redes de Computadores*. Pará, Brasil: [s.n.], 2007. p. 309–322.
- [8] POTTIE, G.; KAISER, W. Wireless Integrated Sensor Networks. *Communication of ACM*, v. 43, n. 5, p. 51–58, 2000.
- [9] CUNHA, D. O. et al. Bio-Inspired Field Estimation With Wireless Sensor Networks. In: *12th International Conference on Telecommunications ICT'2005*. África do Sul: [s.n.], 2005.
- [10] AKYILDIZ, I. F.; VURAN, M. C.; AKAN, O. B. On Exploiting Spatial and Temporal Correlation in Wireless Sensor Networks. In: *Proc. WiOpt'04: Modeling and*



- Optimization in Mobile, Ad Hoc and Wireless Sensor Networks*. Reino Unido: [s.n.], 2004.
- [11] KOSTIN, S.; AMORIM, C. L. Controle de Potência de Transmissão para Redes de Sensores em Cenários com Obstáculos. In: *Simpósio Brasileiro de Redes de Computadores*. Pará, Brasil: [s.n.], 2007. p. 337–350.
- [12] PINTO, A. J. G.; STOCHERO, J.; REZENDE, J. F. Descoberta de Rotas e Mecanismo de Agregação em Redes de Sensores. In: *Simpósio Brasileiro de Redes de Computadores*. Rio Grande do Sul, Brasil: [s.n.], 2004. p. 576–570.
- [13] CUNHA, D. O. *Conservação de Energia em Redes Ad Hoc*. Dissertação (Mestrado) — Universidade Federal do Rio de Janeiro - COPPE, 2004.
- [14] HENRIQUES, F. R.; LOVISOLO, L.; RUBINSTEIN, M. G. Algoritmos para Aumentar o Tempo de Vida de Redes de Sensores Sem Fio Utilizando Inovação. In: *XXVII Simpósio Brasileiro de Telecomunicações SBrT 2009*. Santa Catarina, Brasil: [s.n.], 2009.
- [15] HENRIQUES, F. R.; LOVISOLO, L.; RUBINSTEIN, M. G. An Innovation-Based Algorithm for Energy Conservation in Multihop Wireless Sensor Networks. In: *IEEE International Telecommunications Symposium (ITS 2010)*. Manaus, Brasil: [s.n.], 2010.
- [16] CHAMBERLAND, J.; VEERAVALLI, V. V. Wireless Sensors in Distributed Detection Applications. *IEEE Signal Processing Magazine*.
- [17] ERMIS, E. B. Distributed Detection in Sensor Networks With Limited Range Multimodal Sensors. *IEEE Transactions on Signal Processing*, v. 58, n. 2, p. 843–858, Fevereiro 2010.
- [18] PREED, J. B.; KULKARNI, S. R.; POOR, H. V. Distributed Learning in Wireless Sensor Networks. *IEEE Signal Processing Magazine*, p. 56–69, 2006.
- [19] CETIN, M. et al. Distributed Fusion in Sensor Networks. *IEEE Signal Processing Magazine*, p. 56–69, 2006.
- [20] TANENBAUM, A. S. *Computer Networks*. [S.l.]: Prentice Hall, 2003.
- [21] KUROSE, J. F.; ROSS, K. W. *Redes de Computadores e a Internet*. [S.l.]: Pearson, 2007.

- [22] PETROVA, M. Performance Study of IEEE 802.15.4 Using Measurements and Simulations. In: *IEEE Wireless Communications and Networking Conference (WCNC 2006)*. EUA: [s.n.], 2006. p. 487–492.
- [23] ZHENG, J.; LEE, M. J. Will IEEE 802.15.4 Make Ubiquitous Networking a Reality?: A Discussion on a Potencial Low Power, Low Bite Rate Standard. *IEEE Communications Magazine*, p. 140–146, 2004.
- [24] ALLIANCE, Z. *Zigbee Specification*. [S.l.]: Zigbee Standards Organization, 2005.
- [25] 802.15.4. *Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*. [S.l.]: IEEE Computer Society, 2003.
- [26] SANTOS, S. T. *Redes de Sensores Sem Fio em Monitoramento e Controle*. Dissertação (Mestrado) — Universidade Federal do Rio de Janeiro - COPPE, 2007.
- [27] FARAHANI, S. *ZigBee Wireless Networks and Transceivers*. [S.l.]: Elsevier, 2008.
- [28] POLSATRE, J.; SZEWCZYK, R.; CULLER, D. Telos: Enabling Ultra-Low Power Wireless Research. In: *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*. EUA: [s.n.], 2005.
- [29] AIOFFI, W. M.; MATEUS, G. R. Otimização do Controle de Densidade Com Soredouros Móveis em Redes de Sensores. In: *Simpósio Brasileiro de Redes de Computadores*. Pará, Brasil: [s.n.], 2007. p. 323–336.
- [30] WANG, Z. M. et al. Exploiting Sink Mobility for Maximizing Sensor Networks Lifetime. In: *Proc. 38th Annual Hawaii International Conference on System Sciences (HICSS'05)*. Havaí: [s.n.], 2005.
- [31] PARADISO, J. A.; STARNER, T. Energy Scavenging for Mobile and Wireless Electronics. *IEEE Pervasive Computing*, v. 4, n. 1, p. 18–27, 2005.
- [32] ANASTASI, G. et al. Energy Conservation in Wireless Sensor Networks: A Survey. *Ad Hoc Networks*, v. 7, p. 537–568, 2009.
- [33] XU, Y.; HEIDEMANN, J.; ESTRIN, D. Geography-Informed Energy Conservation for Ad Hoc. In: *Proc. ACM MobiCom*. Itália: [s.n.], 2001. p. 70–84.

- [34] CESARI, P. et al. A Detailed Simulation Study of Geographic Random Forwarding (GeRaF) in Wireless Sensor Networks. In: *Proc. Military Communications Conference*. EUA: [s.n.], 2005. p. 17–20.
- [35] CHEN, B. et al. Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Networks. *ACM Wireless Networks*, v. 8, n. 5, 2002.
- [36] CERPA, A.; ESTRIN, D. Ascent: Adaptive Self-Configuring Sensor Network Topologies. In: *Proc. IEEE Infocom*. EUA: [s.n.], 2002.
- [37] BOLONI, L.; TURGUT, D. Should I Send Now Or Send Later? A Decision-Theoretic Approach to Transmission Scheduling in Sensor Networks with Mobile Sinks. *Special Issue of Wiley's Wireless Communications and Mobile Computing Journal (WCMC) on Mobility Management and Wireless Access*, 2007.
- [38] SCHURGERS, C. et al. Optimizing Sensor Networks in the Energy-Latency-Density Design Space. *IEEE Transactions on Mobile Computing*, v. 1, n. 1, p. 70–80, 2002.
- [39] YANG, X.; VAIDYA, N. A Wakeup Scheme for Sensor Networks: Achieving Balance Between Energy Saving and End-to-End Delay. In: *IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2004)*. Canadá: [s.n.], 2004. p. 19–26.
- [40] KESHAVARZIAN, A.; LEE, H.; VENKATRAMAN, L. Wakeup Scheduling in Wireless Sensor Networks. In: *Proc. ACM MobiHoc*. EUA: [s.n.], 2006. p. 322–333.
- [41] PARUCHURI, V. et al. Random Asynchronous Wakeup Protocol for Sensor Networks. In: *Proc. IEEE International Conference on Broadband Networks*. EUA: [s.n.], 2004.
- [42] TSENG, Y.; HSU, C.; HSIEH, T. Power Saving Protocols for IEEE 802.11 Ad Hoc Networks . In: *Proc. IEEE Infocom*. EUA: [s.n.], 2002.
- [43] ZHENG, R.; HOU, J.; SHAH, L. Asynchronous Wakeup for Ad Hoc Networks. In: *Proc. ACM Mobicom*. EUA: [s.n.], 2003. p. 35–45.
- [44] HAYKIN, S.; MOHER, M. *Sistemas Modernos de Comunicações Wireless*. [S.l.]: Bookman, 2008.

- [45] RAJENDRAN, V.; OBRACZA, K.; ACEVES, J. J. G.-L. Energy-Efficient, Collision-Free Medium Access Control For Wireless Sensor Networks. In: *Proc. ACM SenSys*. EUA: [s.n.], 2003.
- [46] RAJENDRAN, V.; ACEVES, J. G.-L.; OBRACZA, K. Efficient Application-Aware Medium Access For Sensor Networks. In: *Proc. 2nd IEEE Conference on Mobile Ad-Hoc and Sensor Systems*. EUA: [s.n.], 2005.
- [47] POLASTRE, J.; HILL, J.; CULLER, D. Versatile Low Power Media Access For Sensor Networks. In: *Proc. Second ACM Conference on Embedded Networked Sensor Systems*. EUA: [s.n.], 2004.
- [48] YE, W.; HEIDEMANN, J. Medium Access Control With Coordinated Adaptive Sleeping For Wireless Sensor Networks. *IEEE ACM Transactions on Networking*, v. 12, n. 3, p. 493–506, 2004.
- [49] LU, G.; KRISHNAMACHARI, B. An Adaptive Energy-Efficient and Low-Latency Mac For Data Gathering in Wireless Sensor Networks. In: *Proc. 18th International Parallel and Distributed Processing Symposium*. Novo México: [s.n.], 2004. p. 26–30.
- [50] EPHREMIDES, A.; MOWAFI, O. Analysis of a Hybrid Access Scheme For Buffered User Probabilistic Time Division. *IEEE Transactions on Software Engineering*, v. 8, n. 1, p. 52–61, 1982.
- [51] RHEE, I. et al. Z-MAC: A Hybrid MAC For Wireless Sensor Networks. In: *Proc. ACM SenSys*. EUA: [s.n.], 2005.
- [52] CHU, D. et al. Approximate Data Collection in Sensor Networks Using Probabilistic Models. In: *Proc. 22nd International Conference on Data Engineering (ICDE06)*. EUA: [s.n.], 2006.
- [53] HAYKIN, S. *Adaptive Filter Theory*. [S.l.]: Prentice Hall, 2001.
- [54] TULONE, D.; MADDEN, S. PAQ: Time Series Forecasting For Approximate Querying Answering in Sensor Networks. In: *Proc. 3rd European Conference on Wireless Sensor Networks (EWSN06)*. Suíça: [s.n.], 2006.

- [55] GOEL, S.; IMIELINSKI, T. Prediction-Based Monitoring in Sensor Networks: Taking Lessons from MPEG. *ACM Computer Communication Review*, v. 31, n. 5, 2001.
- [56] WU, M.; CHEN, C. H. Multiple Bit Stream Image Transmission Over Wireless Sensor Networks. *IEEE and Wiley Interscience*, p. 677–687, 2006.
- [57] XIONG, Z.; LIVERIS, A. D.; CHENG, S. Distributed Source Coding for Sensor Networks. *IEEE Signal Processing Magazine*, v. 21, n. 5, p. 80–94, 2004.
- [58] FASOLO, E. et al. In-Network Agregation Techniques for Wireless Sensor Networks: A Survey. *IEEE Wireless Communications*, v. 14, n. 2, 2004.
- [59] ALIPPI, C. et al. Adaptative Sampling for Energy Conservation in Wireless Sensor Networks for Snow Monitoring Applications. In: *Proc. IEEE International Workshop on Mobile Ad Hoc and Sensor Systems for Global Homeland Security (MASS-GHS 2007)*. Itália: [s.n.], 2007.
- [60] WILLETT, R.; MARTIN, A.; NOWAK, R. Broadcasting: Adaptative Sampling for Sensor Networks. In: *Proc. Third International Symposium on Information Processing in Sensor Networks (IPSN 2004)*. EUA: [s.n.], 2004. p. 124–133.
- [61] VURAN, M. C.; AKYILDIZ, I. F. Spatial Correlation-Based Colaborative Medium Access Control in Wireless Sensor Networks. *IEEE/ACM Transactions on Networking*, v. 14, n. 2, p. 316–329, 2006.
- [62] TSENG, Y. C. et al. iMouse: An Integrated Mobile Surveillance and Wireless Sensor System. *IEEE Computer*, v. 40, n. 6, p. 60–66, 2007.
- [63] KIJEWLSKI-CORREA, T.; HAENGGI, M.; ANTSAKLIS, P. Wireless Sensor Networks for Structural Health Monitoring: A Multiscale Aproach. In: *ASCE Structures 2006 Congress*. EUA: [s.n.], 2006.
- [64] DESHPANDE, A. et al. Model-Driven Data Acquisition in Sensor Networks. In: *Proc. 30th International Conference on Very Large Data Bases (VLDB 2004)*. Canadá: [s.n.], 2004.
- [65] GEDIK, B.; LIU, L.; YU, P. S. ASAP: An Adaptative Sampling Aproach to Data Collection in Sensor Networks. *IEEE Transactions on Parallel Distributed Systems*, v. 18, n. 12, 2007.

- [66] GANDHAM, S. R. et al. Energy Efficient Schemes for Wireless Sensor Networks With Multiple Mobile Base Stations. In: *Proc. IEEE Globecom 2003*. EUA: [s.n.], 2003.
- [67] JAIN, S. et al. Exploiting mobility for Energy Efficient Data Collection in Wireless Sensor Networks. *ACM/Springer Mobile Networks and Applications*, v. 11, p. 327–339, 2006.
- [68] SHAH, R. C. et al. Data MULEs: Modeling a Three-Tier Architecture for Sparse Sensor Networks. In: *Proc. IEEE International Workshop on Sensor Network Protocols And Applications (SNPA 2003)*. [S.l.: s.n.], 2003. p. 30–41.
- [69] PANDIAN, P. S. et al. Wireless Sensor Networks for Wearable Physiological Monitoring. *Journal of Networks*, v. 3, n. 5, p. 21–29, 2008.
- [70] VIRONE, G. et al. An Advanced Wireless Sensor Network for Health Monitoring. In: *Proceedings of the First Distributed Diagnosis and Home Healthcare Conference*. EUA: [s.n.], 2006. p. 95–100.
- [71] SAUSEN, P. S. et al. Dynamic Power Management with Scheduled Switching Modes in Wireless Sensor Networks. In: *Proceedings of the 15th Annual Meeting of the IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'07)*. Turquia: [s.n.], 2007. p. 1–9.
- [72] PERKINS, C.; BELDING-ROYER, E.; DAS, S. Ad Hoc On Demand Distance Vector Routing (AODV). In: *RFC 3561*. [S.l.: s.n.], 2003.
- [73] GOMEZ, C. et al. Adapting AODV for IEEE 802.15.4 Mesh Sensor Networks: Theoretical Discussion and Performance Evaluation in a Real Environment. In: *Proceedings of the 2006 International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'06)*. EUA: [s.n.], 2006.
- [74] CLAUSEN, T. et al. Optimized Link State Routing Protocol. In: *IEEE International Multi Topic Conference (INMIC)*. [S.l.: s.n.], 2001. p. 62–68.
- [75] TEIXEIRA, I.; REZENDE, J. F. de; PEDROZA, A. C. P. Wireless Sensor Networks: Improving the Network Energy Consumption. In: *XXVII Simpósio Brasileiro de Telecomunicações SBrT 2004*. Brasil: [s.n.], 2004.

- [76] INTANAGONWIWAT, C.; GOVINDAN, R.; ESTRIN, D. Directed Diffusion: a Scalable and Robust Communication Paradigm for Sensor Networks. In: *Proceedings of ACM MobiCon'00*. EUA: [s.n.], 2000. p. 56–67.
- [77] CERVIN, A. et al. How Does Control Timing Affect Performance? *IEEE Control Systems Magazine*, v. 23, n. 3, p. 16–30, 2003.
- [78] ANDERSSON, M. et al. Simulation of Wireless Networked Control Systems. In: *Proceedings of the 4th IEEE Conference on Decision and Control and European Control Conference ECC 2005*. Espanha: [s.n.], 2005.
- [79] ARZÉN, K.-E. et al. Holistic Simulation of Mobile Robot and Sensor Network Applications Using TrueTime. In: *Proceedings of the European Control Conference ECC 2007*. Grécia: [s.n.], 2007.
- [80] JAIN, R. *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. [S.l.]: Wiley, 2001.
- [81] SENE, I. G. et al. Monitoração de Temperatura Corporal Baseada em uma Rede de Sensores Sem Fios. In: *XVIII Congresso Brasileiro de Informática em Saúde (CBIS)*. Santa Catarina, Brasil: [s.n.], 2006.
- [82] CUNHA, D. O.; COSTA, L. H. M. K.; DUARTE, O. C. M. B. An Energy-Aware Routing Mechanism for Mobile Ad Hoc Networks. In: *The 3rd IEEE Latin American Network Operations and Management Symposium - LANOMS'2003*. Brasil: [s.n.], 2003. p. 47–54.