

**ALEX VIDIGAL BASTOS**

**NSINKS: PROTOCOLO DE ROTEAMENTO PARA REDES DE  
SENSORES SEM FIO EM REDES COM VÁRIOS SORVEDOUROS**

Dissertação apresentada à Universidade Federal de Viçosa, como parte das exigências do Programa de Pós-Graduação em Ciência da Computação, para obtenção do título de *Magister Scientiae*.

VIÇOSA  
MINAS GERAIS - BRASIL  
2011

**Ficha catalográfica preparada pela Seção de Catalogação e  
Classificação da Biblioteca Central da UFV**

T

B327n Bastos, Alex Vidigal, 1978-  
2011 Nsinks : protocolo de roteamento para redes de sensores  
sem fio em redes com vários sorvedouros / Alex Vidigal  
Bastos. – Viçosa, MG, 2011.  
ix, 45f. : il. (algumas col.) ; 29cm.

Orientador: Carlos de Castro Goulart.  
Dissertação (mestrado) - Universidade Federal de Viçosa.  
Referências bibliográficas: f. 44-45-

1. Redes de sensores sem fio. 2. Redes de computadores -  
Protocolos. I. Universidade Federal de Viçosa. II. Título.

CDD 22. ed. 004.6

**ALEX VIDIGAL BASTOS**

**NSINKS: PROTOCOLO DE ROTEAMENTO PARA REDES DE  
SENSORES SEM FIO EM REDES COM VÁRIOS SORVEDOUROS**

Dissertação apresentada à Universidade Federal de Viçosa, como parte das exigências do Programa de Pós-Graduação em Ciência da Computação, para obtenção do título de *Magister Scientiae*.

APROVADA: 29 de julho de 2011.

---

Prof. Mauro Nacif Rocha  
(Coorientador)

---

Prof. Antônio Alfredo Ferreira Loureiro

---

Prof. Carlos de Castro Goulart  
(Orientador)

*Aos meus pais, a minha tia Eneida e a minha sogra, que me ajudaram a concretizar este sonho.*

*A minha esposa, que sempre acreditou em mim e esteve ao meu lado.*

*Ao professor Carlos Goulart pela sabedoria e ensino que foi doado durante o período do meu mestrado.*

# Sumário

Lista de Figuras	v
Lista de Tabelas	vi
Resumo	vii
Abstract	viii
<b>1 Introdução</b>	<b>1</b>
1.1 Problema da Comunicação em RSSF e sua importância . . . . .	2
1.2 Hipótese . . . . .	2
1.3 Objetivos . . . . .	3
1.4 Contribuições . . . . .	3
1.5 Organização do Texto . . . . .	3
<b>2 Redes de Sensores sem fio</b>	<b>5</b>
2.1 Caracterização de uma RSSF . . . . .	5
2.2 Arquitetura das RSSFs . . . . .	6
2.3 Requisitos de uma RSSFs . . . . .	7
2.4 Componentes Individuais dos Nós Sensores . . . . .	8
2.5 Camada de Rede . . . . .	10
2.5.1 Roteamento . . . . .	10
2.6 Protocolos de comunicação para RSSFs . . . . .	11
2.6.1 Protocolos da Camada de Rede . . . . .	12
2.6.2 Algumas características dos Protocolos da Camada de Rede . .	13
<b>3 PROTOCOLO NSINKS</b>	<b>17</b>
3.1 DSR ( <i>Dynamic Source Routing</i> ) . . . . .	17
3.1.1 Funcionamento do DSR . . . . .	18

3.2	ESDSR ( <i>Energy Saving Dynamic Source Routing</i> ) . . . . .	20
3.3	DESRO . . . . .	22
3.3.1	Funcionamento do DESRO . . . . .	23
3.4	NSINKS . . . . .	25
3.4.1	Características do protocolo NSINKS . . . . .	26
<b>4</b>	<b>Análise e Simulações com o NSINKS</b>	<b>28</b>
4.1	Simulador Sinalgo . . . . .	28
4.2	Metodologia . . . . .	29
4.3	Detalhes da Implementação . . . . .	30
4.4	Simulações . . . . .	32
4.5	Resultados . . . . .	34
4.5.1	Distribuição em grade . . . . .	34
4.5.2	Simulação do protocolo NSINKS com 03, 04 e 05 nós sorvedouros	37
<b>5</b>	<b>Conclusões e Trabalhos Futuros</b>	<b>42</b>
	<b>Referências Bibliográficas</b>	<b>44</b>

# Lista de Figuras

2.1	Camadas em RSSF . . . . .	6
2.2	Arquitetura de um nó sensor . . . . .	10
2.3	Agregação dos Dados . . . . .	13
2.4	Interconexão entre nós e usuários através da Internet e rede de satélites . .	14
3.1	Descoberta de Rotas no protocolo DSR . . . . .	19
3.2	Descobrimento de Rotas com o protocolo ESDSR . . . . .	22
4.1	Representação do nó sorvedouro . . . . .	35
4.2	Taxa de Entrega dos protocolos . . . . .	36
4.3	Média de Energia para o NSINKS . . . . .	36
4.4	Média de Energia para o DESRO . . . . .	37
4.5	Regiões que consumiram mais energia . . . . .	38
4.6	Representação do nó sorvedouro e os ambientes de simulação com n nós sorvedouros . . . . .	38
4.7	Taxa de Entrega conforme a variação do número de nós sorvedouros . . . .	39
4.8	Média de Energia para uma rede com três nós sorvedouros . . . . .	40
4.9	Média de Energia para uma rede com quatro nós sorvedouros . . . . .	40
4.10	Média de Energia para uma rede com cinco nós sorvedouros . . . . .	41

# Lista de Tabelas

4.1	Parâmetros utilizados nas simulações . . . . .	33
4.2	Pacotes Recebidos pelos Nós Sorvedouros . . . . .	41



# Resumo

Este trabalho descreve o funcionamento do protocolo NSINKS, originado à partir do protocolo DESRO, e que tem a característica de permitir a utilização de mais de um nó sorvedouro e a realização do gerenciamento de rotas de forma dinâmica em uma rede de sensores sem fio. Foi realizada uma comparação entre os protocolos NSINKS e DESRO, usando o simulador Sinalgo, que comprovam que o aumento de nós sorvedouros aumenta o tempo de vida da rede. Os resultados apresentados mostraram que o protocolo NSINKS obteve taxas melhores que o protocolo DESRO, devido à inserção de mais de um nó sorvedouro realizando o balanceamento do tráfego de pacotes entre os nós sorvedouros, dividindo a carga para os nós sensores vizinhos aos nós sorvedouros. Com esse balanceamento de tráfego, os nós sensores vizinhos aos nós sorvedouros esgotaram sua energia de forma mais lenta, evitando a perda de comunicação com o nó sorvedouro e o esgotamento total de sua energia, para os cenários analisados.

**Palavras-chave:** redes de sensores sem fio, protocolos de roteamento, simulador Sinalgo.

# Abstract

This work describes the operation of NSINKS, a protocol based on DESRO protocol, that has the feature of working with more than one sink node and allows dynamic route management in a Wireless Sensor Network. A comparison between protocols NSINKS and DESRO was performed, using the Sinalgo simulator, that showed that the use of multiple sink nodes can increase the network lifetime. The presented results showed that NSINKS protocol has a better performance than DESRO protocol, due to the use of more than one sink node that provided a packet traffic balance among the multiple sink nodes, splitting the load for sensors in the neighborhood of the sink nodes. With this balanced traffic, the sensor nodes in the neighborhood of sink nodes exhausted their energy more slowly, avoiding the loss of communication with the sink nodes and the total loss of their energy, for the analyzed scenarios.

**Keywords:** wireless sensor networks, routing protocols, Sinalgo simulator.

# Capítulo 1

## Introdução

A evolução que tem ocorrido nos meios de acesso à informação está transformando a forma como as pessoas interagem na sociedade. Setores voltados ao desenvolvimento de tecnologia estão fornecendo soluções que permitem acesso rápido à informação de qualquer lugar em qualquer momento. Diferentes infraestruturas de comunicação provêm serviços para diferentes setores que utilizam deste recurso para obter produtividade e acesso a informação.

Dentre as diferentes infraestruturas de comunicação, as Redes de Sensores sem Fio (RSSFs) são uma área que tem evoluído constantemente. Nos últimos anos vêm ocorrendo diversas pesquisas em torno desta área, devido às vantagens que a mesma oferece para aplicações que necessitam monitorar ou supervisionar um ambiente. Apesar das limitações deste tipo de rede, muitos setores tem se beneficiado desta tecnologia.

RSSFs são um tipo de redes *ad-hoc* onde os dispositivos trocam dados diretamente entre si, sem uma infraestrutura de comunicação (1). Com algumas particularidades, as RSSFs são redes específicas, podendo variar de acordo com a aplicação. Dentre algumas características destas redes podemos ressaltar: restrições de consumo de energia, baixa capacidade de processamento, restrições temporais e capacidade de auto-configuração e adaptação decorrentes de falhas na comunicação. Isto aumenta, dentro do contexto de infraestruturas de redes de comunicação, a necessidade de desenvolvimento de diferentes modelos, abordagens e algoritmos para o perfeito funcionamento destas redes.

RSSFs consistem em um grande número de nós sensores, em geral, densamente distribuídos em uma região de interesse em monitorar algum fenômeno. Os nós sensores têm como características: baixo poder de processamento, pouca memória e pouca energia. As características dos nós sensores influenciam diretamente na forma como a rede irá funcionar para monitorar e supervisionar fenômenos. RSSFs provêm uma conexão eficiente entre o mundo físico e o mundo computacional. Atualmente as RSSFs

estão presentes nas áreas de agricultura, ecologia, militar, saúde, em processos de monitoração e controle e em outros segmentos.

## 1.1 Problema da Comunicação em RSSF e sua importância

Devido às limitações das RSSFs e as diversas aplicações que estas redes podem ser utilizadas, é necessário prover soluções dentro das características dos nós sensores os quais têm impactos diretos sobre a arquitetura e tomadas de decisão ao realizar a comunicação entre os nós sensores na rede.

Dentre os componentes individuais de um nó sensor, os rádios são componentes importantes por causa da energia dedicada ao enviar e receber mensagens, estes consideram três diferentes abstrações de camadas: camada física, camada de enlace e camada de rede, sendo a camada de rede responsável por estabelecer o caminho que a mensagem irá seguir através da rede até o destino, baseado no algoritmo de roteamento.

A forma de troca de dados, o chamado protocolo de roteamento, é a peça-chave para operação das redes de sensores: quanto mais eficiente for o protocolo, menos tempo cada nó sensor terá que ficar em funcionamento para transmitir seus dados e, portanto, irá economizar mais bateria.

Dentre os princípios seguidos para a construção de um protocolo de roteamento, devemos considerar: a eficiência ao utilizar a energia, a centralização dos dados e a facilidade de integrar as RSSFs com outras redes, como a Internet. Esta integração é realizada através dos nós sorvedouros, nós que têm como características, maior capacidade de energia, maior alcance de comunicação e não realizam sensoriamento.

Neste trabalho foi proposto um protocolo para diminuir os problemas ao realizar o roteamento de pacotes, levando em consideração uma melhor distribuição da energia em redes com vários nós sorvedouros, buscando uma comunicação mais eficiente e um tempo maior de vida útil da rede.

## 1.2 Hipótese

A utilização de vários nós sorvedouros pode melhorar a eficiência do uso de energia na comunicação entre os nós sensores em uma rede de sensores estacionários.(2)

## 1.3 Objetivos

O objetivo geral do trabalho é desenvolver um protocolo de roteamento para redes de sensores sem fio que explore a existência de mais de um nó sorvedouro, buscando aumentar o tempo de vida útil da rede.

Objetivos específicos:

a) Desenvolver o protocolo de roteamento levando em consideração os problemas relacionados ao consumo de energia dos nós sensores.

b) Realizar uma comparação entre o algoritmo que será utilizado como base e o algoritmo proposto.

## 1.4 Contribuições

Este trabalho contribui em alguns pontos que podemos destacar abaixo:

- o gerenciamento de rotas após a remoção de um nó sensor aumentou o tempo de vida da rede;
- com a adição de mais de um nó sorvedouro, foi realizado um melhor balanceamento de carga;
- o gerenciamento de rotas de forma dinâmica permitiu um melhor roteamento de pacotes.

## 1.5 Organização do Texto

Este documento está organizado como segue:

O capítulo 2 apresenta uma revisão bibliográfica sobre os fundamentos das RSSFs, seu funcionamento, suas características, os algoritmos de roteamento, seus principais problemas e descreve o simulador escolhido para ser utilizado nas simulações.

O capítulo 3 apresenta os algoritmos de roteamento que serviram de base para o desenvolvimento do NSINKS e apresenta o algoritmo NSINKS que é o objeto deste documento.

O capítulo 4 apresenta os resultados obtidos com as simulações entre os algoritmos utilizados na simulação, apresentando as vantagens e desvantagens do algoritmo proposto.

O capítulo 5 apresenta as conclusões e as considerações finais sobre o trabalho.

# Capítulo 2

## Redes de Sensores sem fio

RSSFs são um tipo de redes *ad-hoc* onde os dispositivos trocam dados diretamente entre si, sem uma infraestrutura de comunicação (1). Com algumas particularidades, as RSSFs são redes específicas, podendo variar de acordo com a aplicação. Dentre algumas características destas redes podemos ressaltar: restrições de consumo de energia, baixa capacidade de processamento, restrições temporais e capacidade de auto-configuração e adaptação decorrentes de falhas na comunicação. Isto torna, dentro do contexto de infraestruturas de redes de comunicação, a necessidade de desenvolvimento de diferentes modelos, abordagens e algoritmos para o perfeito funcionamento destas redes.

### 2.1 Caracterização de uma RSSF

Segundo (3) os recentes avanços que vêm ocorrendo nas comunicações *wireless* e nos dispositivos eletrônicos tem permitido o desenvolvimento de RSSFs para diversas finalidades. Uma rede de sensores é composta por um número grande de nós sensores que estão densamente implantados dentro do fenômeno ou bem próximos a eles. A posição dos nós sensores não necessariamente precisa ser arquitetada ou pré-determinada. Existem situações em que os nós sensores precisam ser implantados em áreas inacessíveis ou de difícil acesso, não sendo possível determinar sua posição.

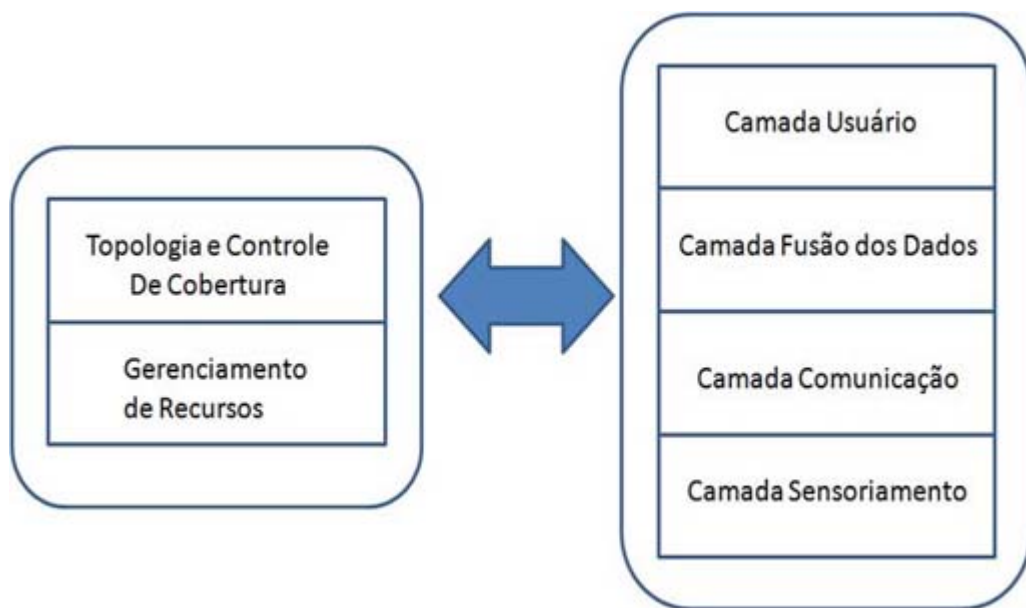
RSSFs é um tipo de rede *ad-hoc*, ou seja, redes que não necessitam de uma infraestrutura de comunicação entre os nós, como por exemplo uma estação base servindo de intermediária, para existir comunicação entre eles. Os nós destas redes possuem a capacidade de autoconfiguração através de algoritmos de roteamento e disseminação de informação. Apesar das RSSFs possuírem características das redes *ad-hoc*, algumas alterações são necessárias na sua arquitetura para suportar as restrições referentes às limitações e especificidades das RSSFs.

RSSFs são redes que consistem de um número de nós sensores, equipados com um poder computacional, de comunicação, de armazenamento e de sensoriamento.

O número de características das RSSFs tem impacto direto sobre a arquitetura, sendo que diferentes tipos de sensores e aplicações distintas requerem alterações nos componentes individuais dos nós sensores permitindo utilizar estas redes para diferentes situações.

## 2.2 Arquitetura das RSSFs

Segundo (4), a organização e arquitetura de uma RSSF deve ser designada ou adaptada para tarefas especiais de modo a otimizar a performance da rede, maximizar o tempo de vida das operações e minimizar o custo. A Figura 2.1 apresenta as camadas de uma arquitetura em RSSF.



**Figura 2.1.** Camadas em RSSF

Dentre as funções das camadas, podemos destacar:

- a camada de sensoriamento executa o trabalho de coleta dos dados gerados nos eventos.
- a camada de comunicação executa as tarefas de correlação dos dados, compressão dos dados, disseminação dos dados e roteamento. A função desta camada é entregar os resultados observados para o nó sorvedouro.



- a camada de fusão de dados processa os dados recebidos da camada de comunicação e combina eles usando vários processamento de sinais, fusão de dados, inteligência artificial e outras técnicas de tomada de decisão.
- a camada do usuário prove uma interface homem máquina com exibição e funções de interação presente nos resultados finais para os humanos e/ou sistemas de computadores onde existem diferentes formas de requisição.

Blocos funcionais adicionais fornecem várias outros processos de apoio e operações tais como gerenciamento de recursos e cobertura/monitoramento da topologia e controle. O módulo de gerenciamento de recursos monitora os recursos disponíveis (tais como energia, memória, e unidade de armazenamento) e balanceia o consumo de energia entre os nós sensores. A topologia/cobertura ajusta a topologia da rede, e harmoniza as operações de sensoriamento entre os vários sensores.

Conforme as funcionalidades de cada camada e necessário adaptar algumas características dos nós sensores para ajustar a arquitetura das RSSF e alguns requisitos são determinados.

## 2.3 Requisitos de uma RSSFs

Devido às particularidades das aplicações e as restrições dos dispositivos sensores que compõem as redes de sensores sem fio, os seguintes requisitos devem ser considerados para a definição da arquitetura e/ou projeto (5):

**Pequeno Tamanho Físico:** a redução do tamanho físico do nó sensor sempre foi uma das características observadas nos projeto dos sensores. Portanto, o objetivo é fornecer um processador com capacidade de processamento razoável, com memória, rádio e outros componentes, onde deve-se manter um tamanho aceitável para a aplicação onde será utilizado o sensor.

**Baixo consumo de energia:** a capacidade de vida, e de desempenho dos sensores são limitados pela energia. Os sensores devem ser capazes de funcionar por um tempo razoavelmente longo sem recarregar a bateria, pois a manutenção é cara e, às vezes, impossível de ser realizada devido ao local onde está sendo feito o sensoriamento.

**Operações Simultâneas Intensivas:** a necessidade de alcançar um melhor desempenho, requer que os dados sejam coletados e ao mesmo tempo processados, comprimidos e enviados para a rede em paralelo, ao invés das ações em sequência. Duas

abordagens conceituais atendem a esse requisito: (1) particionamento do processador em várias unidades em que cada uma é atribuída responsabilidade para uma tarefa específica, e (2) redução do tempo de troca de contexto.

**Diversidade no Modelo e Utilização:** cada nó deve ser pequeno no tamanho e ter baixo consumo de energia. Os nós sensores tendem a ser específicos devido às aplicações onde serão utilizados. Entretanto, os diferentes sensores têm necessidades diferentes, por exemplo, câmeras e termômetros simples são dois extremos em termos de funcionalidade e complexidade. Portanto, é importante que o projeto facilite a escolha entre custo, reutilização e eficiência.

**Operações Robustas:** como os sensores serão implantados ao longo de um grande ambiente e por vezes hostil (floresta, uso militar, o corpo humano), deve ser tolerante a falhas e erros. Portanto, nós sensores devem ter capacidade de auto-teste, auto-calibração e auto-reparo.

**Segurança e privacidade:** cada nó sensor deve ter mecanismos de segurança suficientes para evitar o acesso não autorizado, ataques e danos involuntários da informação dentro do nó. Além disso, os mecanismos de privacidade adicionais devem ser incluídos de acordo com cada aplicação e modelo do sensor.

**Compatibilidade:** é importante ser capaz de reutilizar o código legado através da compatibilidade binária ou tradução binária. c **Flexibilidade:** é necessário acomodar funcionalidades e mudanças com o decorrer do tempo. Flexibilidade deve ser alcançada através de dois meios: (1) programação (2) reconfiguração.

## 2.4 Componentes Individuais dos Nós Sensores

Segundo (5), os nós sensores são compostos por cinco componentes: processador, unidade de armazenamento, bateria, sensores e atuadores e rádio

**Processador:** o processador é responsável pelo processamento dos dados sensorizados. É específico para cada projeto de um nó sensor, trabalha em baixas frequências para economizar energia.

**Unidade de Armazenamento:** dependendo da estrutura da RSSFs, o requisito de armazenamento em termos de rapidez e memória não volátil para cada nó, pode ser diferente. Como exemplo, podemos comparar um modelo de arquitetura que toda in-

formação é instantaneamente enviada para um nó central, então existe muito pouca necessidade para armazenamento local no nó individualmente, enquanto que é mais provável em cenários que existe a necessidade de minimizar a comunicação e enviar montantes significantes por cada nó sensor, requisitando neste caso maior unidade de armazenamento.

**Bateria:** pode ser considerada uma das principais restrições tecnológicas dos nós sensores. Para muitas aplicações, bateria não é uma opção, ou seja, devem ser usadas. Muitos tipos de baterias são avaliados no mundo todo para evitar custos de hardware associados à rede elétrica, mas estes também requerem designs consideráveis.

É importante que um projeto que envolva a utilização de baterias, consiga avaliar o tempo de vida da bateria, gerar um alarme quando a bateria estiver baixa, auxiliando desta forma o funcionamento dos nós sensores.

**Sensores e Atuadores:** a proposta dos sensores não é para computar ou para comunicar, mas sensoriar. O componente de sensoriamento, o nó sensor é o atual gargalo tecnológico, pois estas tecnologias não estão evoluindo rapidamente como os semicondutores.

Outro desafio dos nós sensores é referente ao tipo, a quantidade de sensores e a colocação dos nós sensores em uma dada região. Esta tarefa é difícil por causa do número de tipos de sensores que tem propriedades diferentes tais como resolução, custo, exatidão, tamanho e consumo de energia. Tais características devem estar de acordo com a aplicação que o nó sensor será utilizado, para existir interação entre o sensor e o ambiente.

Várias outras tarefas são associadas ao projeto de um nó sensor, incluindo tolerância a falhas, controle de erros, calibração e sincronização de tempo.

**Rádio:** componentes de comunicação a rádio de curto alcance são excepcionalmente importantes porque parte da energia utilizada pelos nós sensores é para envio e recebimento de mensagens. Durante o projeto e seleção do rádio, deve-se considerar pelo menos três camadas de abstrações diferentes: física, enlace e rede. A camada física é responsável por estabelecer o *link* entre o transceptor e um ou mais receptores. O compartilhamento dos meios de comunicação é facilitado pela camada de enlace e a camada de rede é responsável por estabelecer o caminho que a mensagem deve seguir pela rede da fonte até o destino.

Segundo (4), em termos de componentes dos nós, as RSSFs podem ser classificadas em

geral sobre duas categorias: redes de sensores homogêneas e redes de sensores heterogêneas. Em uma rede de sensores homogênea, os nós sensores tem capacidades idênticas e funcionalidades com respeito a vários aspectos de sensoriamento, comunicação e restrições de recursos. Em redes de sensores heterogêneas, cada nó pode ter capacidades diferentes e executar funções diferentes. Uma típica arquitetura dos nós sensores é apresentada na Figura 2.2.

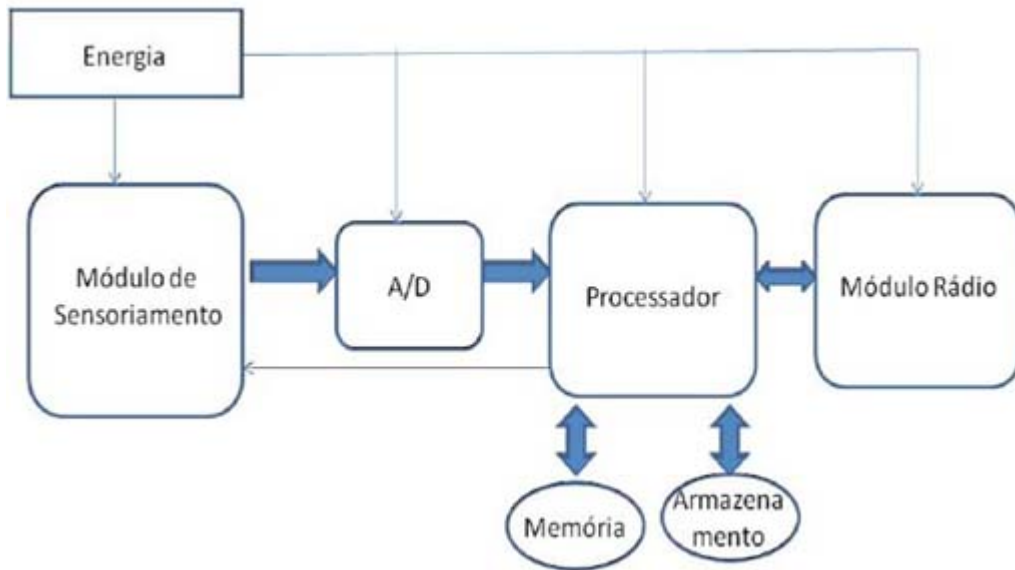


Figura 2.2. Arquitetura de um nó sensor

## 2.5 Camada de Rede

Dentro do modelo OSI (*Open Systems Interconnection*), a camada de rede controla as operações de rede e roteamento de pacotes. No entanto, a operação de roteamento em um RSSFs representa desafios significantes para o projeto destas redes, devido à necessidade de ciclos de funcionamento baixos nos nós sensores. Estes ciclos de funcionamento exigem limites admissíveis de *overhead* para sincronização, negociação, coordenação e outras atividades da camada de rede.

### 2.5.1 Roteamento

Para que os dados monitorados pelos nós sensores se propaguem até os nós sorvedouros, é necessário que sejam estabelecidas rotas entre a origem e o destino. Os protocolos que têm como objetivo determinar uma rota entre a origem e o destino são chama-

dos protocolos de roteamento. Existem diversos algoritmos de roteamento para redes cabeadas (1), sendo que para redes *ad-hoc* estes algoritmos não são eficazes devido à natureza da comunicação sem fio, pois a banda é bastante limitada e a quantidade de erros é alta. O meio sem fio é suscetível a interferências de outras fontes de energia. Outra limitação é que a maiorias dos nós possuem restrições de energia. Desta forma, algoritmos de roteamento eficientes devem levar em consideração estas limitações da RSSFs.

As RSSFs se assemelham às redes sem fio pelo fato de o caminho entre a origem e o destino ser, na maioria das vezes *multihop*. Entretanto, os requisitos dessas redes diferem em alguns aspectos das redes sem fio. Em primeiro lugar, a forma típica de comunicação em uma RSSF é unidirecional no sentido dos nós sensores para o nó sorvedouro. Segundo, como os dados dos nós vizinhos podem se referir a fenômenos comuns pode haver redundância de dados. Terceiro, normalmente os nós sensores possuirão pouca ou nenhuma mobilidade. Finalmente, como principal restrição, a energia disponível para os nós sensores é bastante limitada. Estas diferenças devem ser levadas em consideração no momento de desenvolver um protocolo de roteamento.

## 2.6 Protocolos de comunicação para RSSFs

Segundo (6), os protocolos de comunicação para redes de sensores podem influenciar as capacidades de esforços coletivos para fornecer usuários com aplicações especializadas. Estes protocolos podem fundir, extrair e agregar dados de sensores em campo. Além que, eles podem auto-organizar os nós sensores em *clusters* para completar suas tarefa ou superar obstáculos.

Embora os nós sensores comuniquem através de rede sem fio, protocolos e algoritmos propostos para redes *ad-hoc*, estes podem não ser adequados para RSSFs. RSSFs são aplicações específicas, e os nós sensores trabalham colaborativamente juntos. Além, os nós sensores são restringidos de energia em comparação com os dispositivos das redes *ad-hoc* tradicionais. As diferenças entre redes sensores e redes *ad-hoc* são:

- o número de nós sensores em uma rede de sensores pode ser várias ordens de magnitude maior que os nós em uma rede *ad-hoc*;
- nós sensores são densamente implantados;
- nó sensores são propensos a falhas;
- a topologia de uma rede de sensores muda frequentemente;

- nós sensores usam principalmente o paradigma da comunicação em *broadcast* considerando que redes *ad-hoc* são baseadas em comunicação ponto-a-ponto;
- nós sensores são limitados em energia, poder computacional e memória;
- nós sensores podem não ter uma identificação global (ID) devido à grande quantidade de *overhead* e o número grande de nós sensores;
- redes sensores são implantadas com uma específica aplicação de sensoriamento em mente; redes *ad-hoc* são principalmente construídas para efeito de comunicação.

Com estas diferenças, os projetos para protocolos de comunicação para RSSFs exigem uma atenção específica.

### 2.6.1 Protocolos da Camada de Rede

Segundo (6), os nós sensores podem ser densamente dispersos em uma área para observar os fenômenos. Como resultado, eles podem ficar muito próximos um do outro. Em tal cenário, comunicação *multihop* pode ser uma boa escolha para redes de sensores com estritos requisitos de consumo de energia e níveis de transmissão de energia. Em comparação com comunicação sem fio para longas distâncias, comunicação *multihop* pode ser eficaz, como forma de superar algumas propagações de sinal. Além que, os nós sensores consomem menos energia quando for transmitir uma mensagem, porque as distâncias entre os nós sensores são curtas.

Como resultado, a Camada de Rede de uma rede de sensores é usualmente designada de acordo com os seguintes princípios:

- eficiência de energia é sempre uma consideração importante;
- redes de sensores são na sua maioria centrada em dados;
- uma rede de sensores ideal tem atributos baseados em endereçamento e conhecimento do local;
- agregação de dados é útil somente quando não prejudicar o esforço de colaboração dos nós sensores;
- o protocolo de roteamento deve ser facilmente integrado com outras redes, ex; Internet.

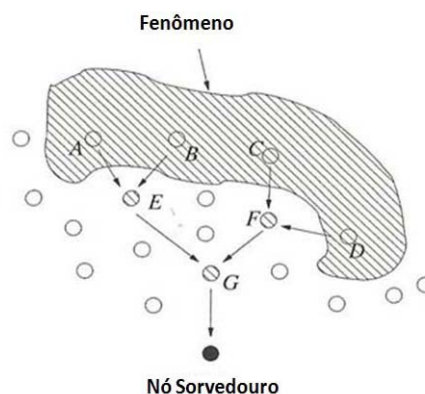
O princípio dos projetos serve como guia para projetar protocolos de roteamento para rede de sensores.

## 2.6.2 Algumas características dos Protocolos da Camada de Rede

Algumas características são importantes para os protocolos nas diferentes camadas. Na camada de rede podemos destacar alguns pontos que são importantes para o funcionamento correto e eficiente destes protocolos, onde podemos destacar: centralização dos dados e eficiência em energia.

### 2.6.2.1 Centralização dos dados

Em RSSFs informação ou dados podem ser descritos usando atributos. A fim de integrar perfeitamente com a informação ou dados, os protocolos de roteamento podem ser projetados de acordo com técnicas de centralização de dados. Os protocolos de roteamento centrado em dados requerem atributos baseados em nomeação, que é usado para realizar consultas por atributos de um fenômeno. Em essência, os usuários estão mais interessados em recolher dados dos fenômenos nas RSSFs ao invés de um único nó sensor. Eles consultam as RSSFs usando atributos do fenômeno que querem observar. Na Figura 2.3, é representado um nó consultando uma rede de sensores para observar as condições ambientais de um fenômeno, onde os nós enviam uma mensagem de consulta onde "buscam encontrar áreas onde a temperatura é mais de 70° F", sendo que o nó sensor E agrega os dados de A e B, enquanto o nó sensor F agrega os dados de C e D, e o nó G agrega os dados de E e F sendo depois os dados enviados do nó G para o nó sorvedouro, conforme (6).

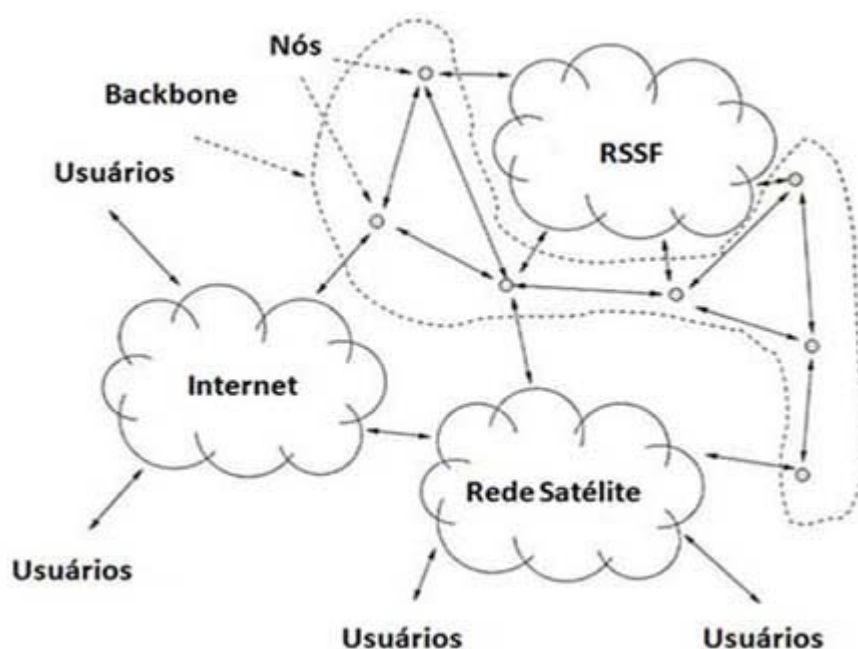


**Figura 2.3.** Agregação dos Dados

Agregação de dados pode ser percebida como um conjunto de métodos automatizados para combinar dados de vários nós sensores sobre um conjunto de informações significativas. Em respeito, sabe-se que agregação de dados é uma fusão de dados,

onde cuidados devem ser tomados quando agregar dados específicos, pois alguns dados específicos são necessários para algumas aplicações.

Um dos princípios de projeto para a camada de redes é permitir facilidade de integração com outras redes, seja uma rede de satélites ou a Internet. São representados na Figura 2.4, os nós que são a base de comunicação que serve como um *gateway* para as outras redes. Os usuários podem consultar a rede de sensores através da Internet ou da Rede de Satélites, dependendo da proposta de consulta ou do tipo de aplicação que estiver rodando.



**Figura 2.4.** Interconexão entre nós e usuários através da Internet e rede de satélites

Alguns esquemas de rede foram propostos para RSSFs para melhorar a eficiência no roteamento de pacotes e aumentar a expectativa de vida da rede, dentre eles temos o SMECN (7) que cria um sub-gráfico da rede de sensores que contém o caminho mínimo de energia; o LEACH (8) que forma *clusters* para minimizar a dissipação de energia; o SAR (9) que cria várias árvores, onde a raiz de cada árvore é vizinha de um nó sorvedouro e seleciona uma árvore de dados para rotear a volta do nó sorvedouro de acordo com os recursos de energia e métricas de QoS; o *Flooding* que envia os dados em broadcast para todos os vizinhos, independente se eles receberam antes ou não; o *Gossiping* (10) que envia os dados para um vizinho selecionado de forma randômica; o SPIN (11) que envia os dados para os nós sensores somente se eles interessarem, tendo três tipos de mensagens: ADV, REQ e DATA e o *Direct Diffusion* (12) que configura



os gradientes para o fluxo de dados da fonte para o nó sorvedouro durante interesses de disseminação.

Devido a diferentes aplicações requererem diferentes tipos de protocolos da camada de rede, muitos avanços nos protocolos de roteamento centrado em dados são necessários.

### 2.6.2.2 Eficiência de energia nos protocolos da camada de rede

A camada de redes em RSSFs é responsável por entregar os dados da fonte para o destino através de uma rota selecionada (13). Devido às características das RSSFs, muitos dos protocolos da camada de rede designados para convencionais redes *ad-hoc* não podem ser utilizados nas RSSFs, devido aos requerimentos destas redes.

Para reduzir o consumo de energia na comunicação, protocolos da camada de rede têm atraído a atenção. Muitos fatores influenciam na proposta dos protocolos da camada de rede, e um amplo número de esquemas tem sido proposto. Algumas propostas de classificação de eficiência-energia (E2) dos protocolos da camada de rede servem para ajudar no estudo da eficiência de energia dos protocolos da camada de rede. Podemos classificar os protocolos da camada de rede nas RSSF de acordo com:

- a posição da estação base;
- se a implantação dos sensores é pré-determinada, auto regulamentada ou a distribuição é randômica;
- de acordo com as propriedades dos nós se é homogênea ou heterogênea;
- com relação à dinâmica da rede, se é estática ou dinâmica;
- com relação às ações nos dados, se realiza a entrega (colecção das informações, disseminação das informações, híbrida) ou processamento (agregação dos dados, sinal colaborativo e processamento de Informação - CSIP, Orientado a evento, baseado em agentes móveis);
- com relação ao alcance efetivo dos protocolos;
- de acordo com a arquitetura do sistema, se é *flat* ou hierárquica;
- com relação as abordagens de roteamento, sendo inundação, *Unicast* e *Multicast*;
- se os cenários são proativos (inicializa na fonte, continuamente), reativos (fenômeno / dirigido a evento, consultas dirigidas) ou híbridos;

- de acordo com os objetivos de eficiência de energia híbrido, minimizando o consumo de energia para cada pacote individual enviado pela rede, minimizando o consumo total da rede e balanceando o consumo de energia da rede.

Baseados nestas classificações, comparações de alguns protocolos da camada de rede existentes que pretendem oferecer eficiência em energia pode ser fornecido. Estes protocolos são designados para RSSFs ou foram originalmente designados para redes sem fio ou *ad-hoc*, mas são escaláveis para RSSFs.

### 2.6.2.3 Eficiência de energia nos protocolos de entrega de dados

Uma das responsabilidades críticas dos protocolos da camada de rede é para entrega dos dados entre a fonte e o destino. Em RSSFs, protocolos de entrega dos dados devem ter eficiência de energia no momento de enviar os dados Estes podem ser classificados em grupos distintos de acordo com vários critérios, desde que a finalidade de entrega de fluxo, abordagem de roteamento e objetivos de eficiência de energia.

Em resumo, novas fontes de energia podem ter impactos significativos sobre o desenvolvimento de RSSFs, incluindo:

- melhorar a eficiência dos sistemas de energia;
- assegurar longevidade contra a interrupção de energia;
- expansão futura para escolha de energia;
- promoção da produção de energia e utilização que respeitem a saúde e valores ambientais;

Diante de algumas propostas de projetos para as RSSFs e observando as características pertinentes do meio, no próximo capítulo será apresentado os protocolos que foram utilizados de base para o desenvolvimento do protocolo NSINKS.

# Capítulo 3

## PROTOCOLO NSINKS

Dependendo dos requisitos de uma RSSFs, muitas restrições e limitações devem ser observadas ao se desenvolver um protocolo de roteamento para as RSSFs, pois, conforme mencionado existem fatores que afetam o comportamento e o funcionamento destas redes, como as particularidades dos componentes dos sensores, as especificidades da aplicação, o ambiente ou o fenômeno a ser monitorado. Restrições estas que alteram a forma como estes protocolos devem ser implementados para obter uma melhor produtividade da rede e conseqüentemente um maior tempo de vida útil dos nós sensores.

Apesar das RSSFs serem um tipo de rede *ad-hoc*, estas redes tendem a serem específicas e requerem alterações na sua arquitetura e nos protocolos utilizados, devido as características dos nós sensores e Diante deste cenário, é proposto neste trabalho um protocolo de roteamento para as RSSFs baseado no protocolo DESRO (*Dynamic Energy Saving Routing*) (2)), que foi originado dos protocolos (*Dynamic Source Routing*) (14) e ESDSR (*Energy Saving Dynamic Source Routing*) (15).

O NSINKS foi baseado no protocolo DESRO, adicionando a este o uso de mais de um nó sorvedouro na rede, buscando melhorar o consumo de energia dos nós sensores e conseqüentemente o tempo de vida útil da rede.

### 3.1 DSR (*Dynamic Source Routing*)

O DSR (*Dynamic Source Routing*) (14) é um algoritmo de roteamento reativo para redes *ad-hoc* no qual a origem determina por qual caminho o pacote deve passar para chegar ao seu destino, tendo como característica um melhor desempenho em ambientes onde a velocidade de mobilidade dos nós é baixa. Diferente dos protocolos existentes em redes *ad-hoc*, o DSR não atualiza as informações sobre rotas freqüentemente e tem como objetivo oferecer vantagens em relação aos modelos tradicionais de roteamento

para redes cabeadas, mas possuindo algumas desvantagens em situações de grande mobilidade e um volume grande de dados na rede.

### 3.1.1 Funcionamento do DSR

Segundo (14), o DSR é uma técnica de roteamento onde a origem determina por qual caminho o pacote irá passar até chegar ao seu destino, sendo que as informações da rota são inseridas no cabeçalho de cada pacote enviado, determinando qual será o próximo nó a receber o pacote. Diferente dos demais protocolos para redes *ad-hoc*, o DSR não realiza frequentemente os anúncios de rota, sendo realizado quando necessário consultas em *cache* ou através do protocolo de descoberta de rotas.

No protocolo DSR, ao se enviar um pacote para outro nó, o nó origem deve construir uma rota no cabeçalho do pacote, informando o caminho pelo qual o pacote deve percorrer de forma ordenada, até alcançar o nó destino. O pacote é enviado pelo enlace sem fio até alcançar o nó que estiver mais próximo e que esteja identificado no cabeçalho do pacote. O nó destino ao receber o pacote verifica se este pacote é para ele mesmo e caso não seja, o nó simplesmente retransmite para o próximo nó identificado na rota. No DSR quando há mais de uma rota para o mesmo destino, a rota com o menor número de saltos é sempre escolhida.

Os nós armazenam em *cache* as rotas que já foram aprendidas, as quais são consultadas antes do nó enviar um pacote, e caso o nó não tenha a rota para enviar o pacote, o nó origem usa o protocolo de descobertas de rotas para determinar a rota a ser seguida.

#### 3.1.1.1 Descoberta de Rotas

O protocolo de descobertas de rotas permite aos nós descobrirem de forma dinâmica qual rota deve ser utilizada para ir para outro nó na rede, mesmo o nó não estando em seu alcance, pois este mecanismo permite o uso de caminhos formados por nós intermediários.

O funcionamento da descoberta de rotas começa com um nó enviando um *route request* para todos os nós vizinhos na rede. O *route request* identifica o nó origem e o nó destinatário da rota a ser descoberta. O nó destinatário ao ser atingido, ou algum nó intermediário que já tenha em sua *cache* uma rota para o destino desejado, envia um *route reply*.

De acordo com a Figura 3.1, o nó A quer encontrar uma rota para o nó D, então inicia o procedimento de descoberta de rota, enviando os endereços da origem e do destino para os nós vizinhos. Quando B recebe o *route request* ele verifica se existe

uma rota em *cache* para o destinatário, como não existe, ele adiciona o seu endereço e envia para todos os seus vizinhos. Portanto, C seguirá o mesmo procedimento de B. Quando D recebe o *route request*, ele percebe que o endereço de destino é o dele e então devolve a rota aprendida para A através do *route reply*. Em adição aos endereços de origem e destino, cada *route request* possui um *route record*, no qual registra todos os nós pelos quais o pacote passou durante o período de descoberta de rotas. Cada *route request* também contém um *request id* único, caracterizado por um número em seqüência. Assim, para detectar o recebimento duplicado de requisições de rota, os nós atualizam suas listas de pares (*initiator address, request id*) a cada requisição de rota recebida.

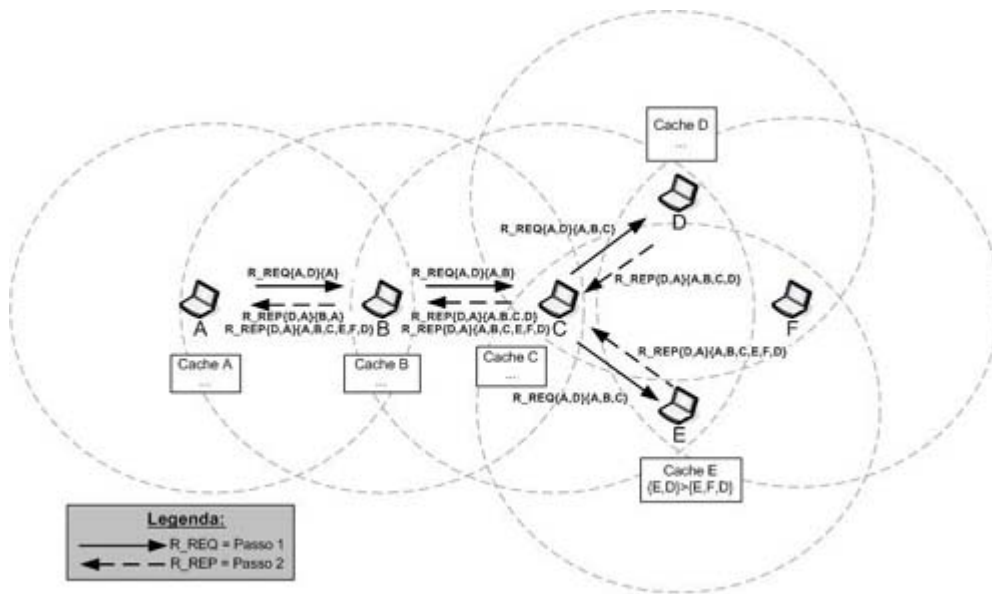


Figura 3.1. Descoberta de Rotas no protocolo DSR

(14)

Quando um nó recebe um *route request*, ele segue os seguintes passos:

- se o par (*initiator address, request id*), para esta requisição de rota, for encontrado na lista de entradas com valor de *request id* mais recente, então o nó descarta o *route request* e não o processa novamente. Desta forma o nó não processará a mesma descoberta de rota mais de uma vez, até que o tempo de vida do par expire.
- se o endereço do nó já estiver listado no *route record*, então o *route request* é descartado e não será processado novamente. Assim, há uma prevenção contra a

ocorrência de *loops*, de modo que se o endereço do nó está no registro de rota, significa que o *route request* já passou por ele uma vez e retornou.

- se o nó atual é o destinatário da requisição, então ele terá que iniciar o procedimento de resposta de rota para o remetente. Logo, ele verifica se há alguma rota em *cache* para o remetente. Caso seja encontrada, esta rota será usada, caso contrário, a rota recebida é simplesmente invertida. Assim, é retornada a cópia da rota utilizada pelo destinatário para o remetente.
- no último caso o nó adiciona o seu endereço na requisição de rota e o repassa mais adiante por difusão.

### 3.1.1.2 Manutenção de Rotas

O processo de manutenção de rotas é utilizado para manter atual todas as rotas determinadas nas *caches* dos nós que fazem parte da rede, sendo que um nó ao repassar um pacote, ele aguarda um reconhecimento do nó que foi encaminhado o pacote. Caso ocorra algum problema e o nó não receba a confirmação do reconhecimento, o nó envia para o nó remetente um pacote de erro de rota, conhecido como *route error*. Neste pacote estará contido o endereço do nó sensor que identificou a falha na rota e o endereço do nó que não confirmou o reconhecimento.

Assim, o remetente remove todas as rotas que utilizavam este enlace "quebrado". O reconhecimento realizado pelos nós pode ser dependente da camada de enlace, através da manutenção baseada totalmente no reconhecimento de pacotes em cada nó da rota. Por outro lado, para fazer o reconhecimento sem utilizar a camada de enlace, os nós ficam em modo promíscuo, escutando o meio e observando se o próximo nó irá repassar o pacote. Este método é conhecido como reconhecimento passivo, já que não há uma interação entre a troca de pacotes pelos nós envolvidos. Portanto, é importante citar que ambos os nós envolvidos em um reconhecimento passivo devem ter, idealmente, o mesmo alcance de transmissão.

## 3.2 ESDSR (*Energy Saving Dynamic Source Routing*)

O protocolo de roteamento ESDSR (*Energy Saving Dynamic Source Routing*) (15) é um protocolo de roteamento baseado no protocolo DSR, mas que implementa uma maneira diferente ao determinar a rota a ser seguida pelo pacote. O protocolo ESDSR tem a preocupação em diminuir o consumo de energia ao realizar a transmissão de

dados, analisando a potência necessária ao transmitir dados para o nó destino, evitando desta forma gasto de energia excessivo, diferentemente do protocolo DSR que define sua tabela de rotas através do caminho mais curto entre o nó origem e o nó destino.

Segundo (15), o objetivo do ESDSR é determinar a expectativa mínima de vida estimada da rota, avaliando os níveis de energia de cada nó sensor através das equações (3.1) e (3.2):

$$C(R, t) = \max(R_j(t)) \quad (3.1)$$

$$R_j(t) = \min\left(\frac{E_i}{P_{ti}}\right) \quad (3.2)$$

onde,  $R_j(t)$  é a energia mínima para potência de transmissão pelo caminho descoberto,  $E_i$  é a energia remanescente no nó sobre o caminho descoberto e  $P_{ti}$  que é a potência de transmissão sobre o caminho descoberto. Conforme mostrado na Figura 3.2, onde o nó 1 é o nó fonte e o nó 5 é o destino, o nó 1 inicializa a descoberta de rota, enviando uma mensagem de *broadcast* para todos os seus vizinhos. Como os nós estão dentro do intervalo de alcance do nó fonte, eles recebem o pacote enviado pelo nó 1. Mas, como os nós 2, 3 e 4 não são o destino eles reenviam o pacote para seus vizinhos, neste caso, chegando ao seu destino o nó 5.

O nó 5 ao receber o pacote, envia através do caminho inverso; neste caso, como o pacote chegou por dois caminhos, o nó 5 encaminha o pacote de resposta através dos dois caminhos, sendo (5 - 3 - 2 - 1) e (5 - 4 - 1) para o nó fonte e adiciona sua expectativa de vida, os nós intermediários ao receber o pacote, verificam sua expectativa de vida e reenviam o pacote até chegar ao nó fonte. As tabelas dos nós intermediários são atualizadas caso sua expectativa de vida seja maior ao que já está armazenada.

Os nós ao receberem o pacote enviado, verificam qual foi a potência enviada e qual potência foi recebida para recalcularem o poder de transmissão para os próximos saltos, sendo que a potência mínima é dada pela equação (3.3):

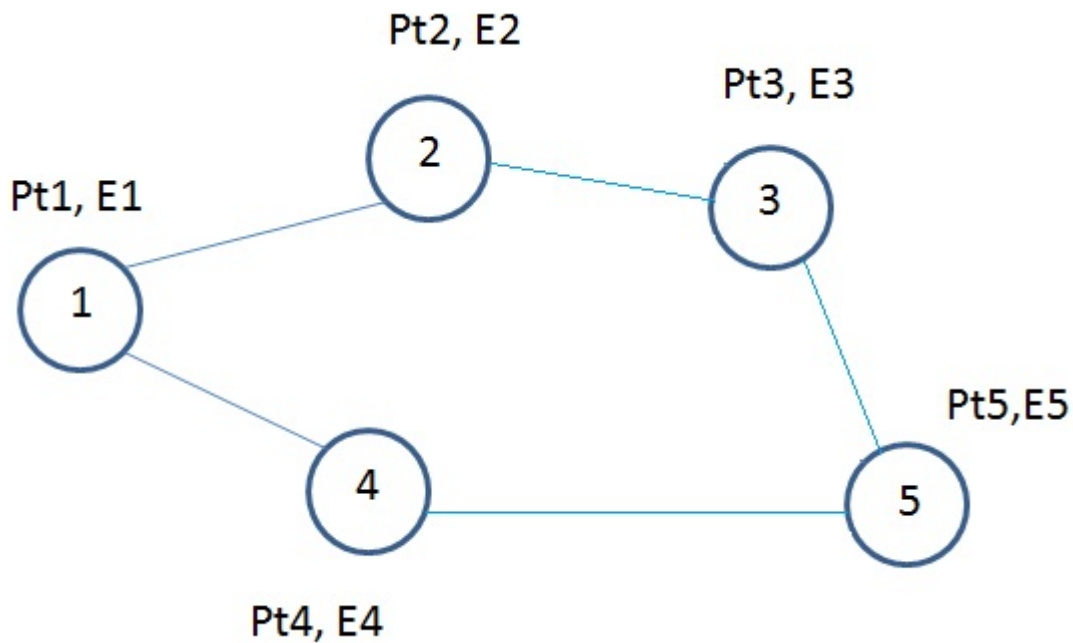
$$P(\min) = P_{tx} - P_{recv} - P_{threshold} \quad (3.3)$$

sendo,  $P_{min}$  a potência mínima,  $P_{tx}$  a potência de transmissão,  $P_{recv}$  a potência de recebimento e  $P_{threshold}$  um limite de segurança. Outra variável que é utilizada devido aos problemas de instabilidade dos *links* é  $P_{margin}$ , que é a margem de segurança. Sendo desta forma a potência mínima é calculada pela equação (3.4).

$$P(\min) = P_{tx} - P_{recv} - P_{threshold} + P_{margin} \quad (3.4)$$

A potência de transmissão é calculado e armazenado em uma tabela de potências.

Cada nó mantém uma tabela de potência para cada nó alvo. A tabela de potência é atualizada cada vez que um nó recebe um pacote de um vizinho. Quando um nó tem algum pacote para enviar, o nó pesquisa em sua tabela de potências para encontrar a potência de transmissão requerido para um nó específico, a informação não sendo detectada na tabela de potências, o nó envia com a potência padrão de transmissão definido.



**Figura 3.2.** Descobrimto de Rotas com o protocolo ESDSR

### 3.3 DESRO

O protocolo DESRO, proposto por (2), contempla a maioria das características dos protocolos de roteamento das redes *ad-hoc*, com a característica de monitoramento de rotas em redes com apenas um nó sorvedouro.

Dentre as características dos protocolos ESDSR e DESRO, o DESRO se destaca por realizar o monitoramento de rotas utilizando temporizadores e/ou contadores de pacotes para selecionar dinamicamente a rota que tiver uma expectativa de vida maior (mais energia), diferentemente do ESDSR que determina, em um dado momento, a rota com maior expectativa de vida, e utiliza esta mesma rota até ocorrer a partição da rede.



### 3.3.1 Funcionamento do DESRO

O protocolo DESRO, ao realizar o descobrimento de rota, determina a rota para um destino que tiver a maior expectativa de vida. Após a descoberta da rota, ele utiliza um temporizador e/ou contador de pacotes que envia uma mensagem de *Route Management* quando o temporizador chegar a zero ou o contador de pacotes atingir um valor determinado. Essa mensagem *Route Management* percorrerá cada nó na rede até alcançar o destino, determinando uma nova rota para o destino e atualiza novamente o temporizador e/ou contador de pacotes.

O envio de um *Route Management* pode ser disparado por uma das seguintes razões (2):

**temporizador:** assim que uma rota é descoberta, é zerado o temporizador. O temporizador chegando à zero, novamente é necessário a atualização para verificar qual a rota com maior expectativa de energia;

**contador de pacotes:** assim que uma rota é descoberta, o contador de pacotes é zerado e após atingir um valor determinado, é disparada uma mensagem de *Route Management* para descobrir a rota que tem melhor expectativa de energia. Para cada pacote enviado, o contador de pacotes é incrementado;

**temporizador e contador de pacotes:** o funcionamento é o mesmo determinado para o temporizador e para o contador de pacotes, sendo que a diferença é devido ao evento que ocorrer primeiro. A mensagem de *Route Management* será gerada quando o temporizador chegar a zero ou quando o contador de pacotes alcançar o valor determinado, o que ocorrer primeiro.

O processo de determinar qual valor de contador será utilizado não é automático no protocolo DESRO e deve ser definido no momento de implantação da rede. O processo não foi implementado de forma automática para não consumir mais energia na fase de processamento. Uma abordagem interessante é determinar o valor do contador que será utilizado de acordo com a especificidade da rede de sensores. Quando o volume de dados gerados for grande, o contador de pacotes pode ser priorizado para realizar o gerenciamento de rotas devido ao número elevado de pacotes gerados na rede; e quando a rede não é muito ativa, a abordagem pelo temporizador pode ser mais eficiente, pois realiza o gerenciamento de rota conforme o tempo de funcionamento da rede. É representado abaixo o pseudocódigo do DESRO.

LOOP (evento)

Se evento = "enviar novo pacote de dados"então

Se houver rota para o sorvedouro então  
    Enviar Pacote;  
Se gerenciamento de rotas é feito por contagem de pacotes então  
    Soma mais um ao contador;  
    Se contador alcançou o limite pré-estabelecido então  
        Zerar contador;  
        Cria mensagem de gerenciamento de rotas (Route Management) e a en-  
via;  
    Fim se  
Fim se  
Senão  
    Envia uma requisição de rota (Route Request);  
    Aguarda rota colocando o pacote no buffer por um tempo pré-determinado;  
Fim se  
Fim se  
Se evento = "temporizador zerou" então  
    Reinicia temporizador;  
    Cria mensagem de gerenciamento (Route Management) e envia;  
Fim se  
Se evento = "chegada de pacote de dados" então  
    Se nó não é o sorvedouro Repassa pacote;  
Fim se  
Se evento = "chegada de mensagem de requisição de rota" então  
    Se nó é o sorvedouro então  
        Envia uma mensagem de resposta de rota (Route Reply);  
    Senão  
        Repassa mensagem;  
    Fim se  
Fim se  
Se evento = "chegada de mensagem de resposta de rota" então

```
Adiciona/Atualiza rota na tabela de rotas;
Cria/Atualiza temporizadore e/ou contador de pacotes;
Se houver pacote de dados esperando por rota então
    Envia pacotes de dados;
Senão é o destino da mensagem então
    Calcula energia mínima estimada;
    Repassa Mensagem;
Fim se
Fim se

Se evento = "chegada de mensagem de gerenciamento"então
    Se não é o destino então
        Repassa mensagem;
    Senão
        Envia uma mensagem de resposta (Route Replay) ao originador;
Fim se

FIM LOOP
```

### 3.4 NSINKS

O protocolo de roteamento NSINKS é um protocolo originado do protocolo DESRO (2), com a característica de realizar o roteamento de pacotes em redes onde exista mais de um nó sorvedouro. O NSINKS contempla as características desejáveis para uma rede ad-hoc, tendo como objetivo um melhor balanceamento de energia da rede. A existência de mais de um nó sorvedouro na rede pode melhorar o balanceamento de consumo de energia da rede, mas também pode acarretar em um aumento de processamento devido à adição de mais nós sorvedouros.

O protocolo NSINKS foi criado para solucionar o problema que ocorre quando existe apenas um nó sorvedouro na rede, pois os nós sensores que são vizinhos do nó sorvedouro tendem a ser sobrecarregados devido à proximidade com o nó sorvedouro, ocorrendo um consumo excessivo de energia por estes nós, pois além de realizar o monitoramento, realizam a tarefa de retransmitir as mensagens vindas de outros nós sensores que têm como destino o nó sorvedouro, permitindo a ruptura da rede devido

ao esgotamento destes nós sensores.

Com o intuito de aprimorar o funcionamento do protocolo DESRO, o protocolo NSINKS tem a característica de trabalhar com vários nós sorvedouros, diminuindo a sobrecarga dos nós vizinhos aos nós sorvedouros. A quantidade de nós sorvedouros pode variar de acordo com o tamanho da rede, possibilitando adicionar mais nós sorvedouros na rede para realizar um melhor balanceamento da rede e aumentar o tempo de vida da rede.

NSINKS foi desenvolvido tendo como base todas as características dos protocolos que foram à base do protocolo DESRO, como o protocolo DSR e o protocolo ESDSR.

Outra funcionalidade alterada no NSINKS foi a escolha dinâmica do temporizador para o gerenciamento de rotas, determinando se a mensagem de Route Management será enviada após o temporizador zerar, quando o contador de pacotes atingir um valor determinado (valor definido no início do funcionamento da rede) ou quando ocorrer a perda de algum nó sensor. A necessidade de realizar o gerenciamento de rotas ao perder algum nó sensor, evita a perda de pacotes, pois se o nó sensor enviar um pacote que tinha como nó intermediário o nó que foi removido da rede, o gerenciamento de rotas permite que o pacote seja encaminhado para outro nó na rede, evitando a perda do pacote.

O NSINKS foi projetado para RSSFs com nós sensores estáticos, podendo ser extensível para RSSFs com nós móveis.

### 3.4.1 Características do protocolo NSINKS

O NSINKS sendo um protocolo originado de outros protocolos tem como características:

**descobrimto de rotas:** ao iniciar o funcionamento da rede, os nós sorvedouros enviam um pacote em *broadcast* para toda a rede para criar um caminho entre os nós sensores e os nós sorvedouros. Ao iniciar a rede com mais de um nó sorvedouro, os nós sorvedouros enviam os pacotes para os seus vizinhos para estabelecer uma rota entre eles, sendo cada mensagem de descobrimto de rotas enviadas pelos nós sorvedouros tendo a mesma identificação (ID único). O motivo dos nós sorvedouros enviar uma mensagem com a mesma identificação é para realizar o balanceamento da rede, pois o nó sensor ao receber um pacote com o mesmo ID realiza o descarte do pacote, permanecendo com o pacote da rota que recebeu primeiro;

**controle da potência de transmissão:** o nó sensor, antes de realizar a transmissão

de um pacote para o nó vizinho, determina a potência a ser usada para a transmissão do pacote, conforme distância dada entre os nós sensores;

**monitoramento da rota:** o monitoramento da rota vai ocorrer conforme o funcionamento da rede. Foram definidos três variáveis, sendo: contador de pacotes, tempo de funcionamento da rede (temporizador) e se algum nó for removido da rede.

A escolha de qual variável utilizar é definida de forma dinâmica, ou seja, conforme o funcionamento da rede:

- **contador de pacotes** - no início do funcionamento da rede é definido a quantidade de pacotes enviados pela rede, que ao atingir este valor, o contador de pacotes é zerado e é gerado um pacote de gerenciamento de rotas.
- **temporizador** - é definido um tempo de funcionamento da rede e ao ser atingido este tempo é gerado um pacote de gerenciamento de rotas.
- **remoção de um nó sensor** - caso ocorra à perda de um nó sensor, é gerado um pacote de gerenciamento de rotas, para atualizar as rotas dos nós que utilizam aquele nó sensor removido para encaminhar um pacote para o nó sorvedouro, diminuindo a quantidade de pacotes perdidos.

Para avaliação das alterações, foram realizadas simulações comparando o protocolo NSINKS com o protocolo DESRO para validar se as propostas de alterações do protocolo NSINKS foram satisfatórias.

## Capítulo 4

# Análise e Simulações com o NSINKS

Para efeito de validação do protocolo proposto, foram realizadas algumas simulações utilizando o simulador Sinalgo (*Simulator for Network Algorithms*), simulador de algoritmos de rede que tem como característica simular e testar algoritmos de roteamento. Diferentemente da maioria dos outros simuladores de rede, que passam a maior parte do tempo simulando as diferentes camadas, o Sinalgo foca sua verificação sobre os algoritmos de rede e abstrai as camadas subjacentes.

### 4.1 Simulador Sinalgo

O Sinalgo (*Simulator for Network Algorithms*) é um simulador de algoritmos de rede, cujos principais objetivos são testes e validação dos algoritmos. Ele provê suporte a grandes redes, podendo realizar simulações com até  $10^5$  nós sensores em tempo aceitável. O simulador Sinalgo possui um enfoque voltado para a verificação sobre os algoritmos de roteamento e abstrai as camadas subjacentes.

Segue algumas características do Sinalgo:

- prototipação rápida para os algoritmos de rede, desenvolvidos em JAVA;
- é possível estender suas características para atender a diversos cenários;
- permite plugins ajustáveis;
- alto desempenho - executa simulações com 100000 nós em tempo aceitável;

- suporte a 2D e 3D;
- simulação Assíncrona e Síncrona;
- visualização Personalizável do gráfico da rede;
- independente de plataforma.

Para facilitar a extensão das funcionalidades é oferecido um conjunto de pacotes de extensão, chamado modelos, que permitem simular diferentes situações em uma RSSFs, sendo eles:

- **o modelo de Mobilidade:** descreve como o nó se move na simulação síncrona (simulação assíncrona não oferece suporte a nós móveis). A mobilidade é simulada em termos de rodadas. No início de cada rodada, os nós são autorizados a mudar para uma nova posição, onde permanecem durante a duração da rodada;
- **o modelo de Conectividade:** decide quais os nós vizinhos têm conexão e insere ou remove as bordas. Sendo que no modo de simulação síncrona, cada nó atualiza suas conexões em cada rodada (round) e para simulação assíncrona, o Sinalgo não suporta nós móveis;
- **o modelo de Distribuição:** como os nós são colocados inicialmente no campo de implantação;
- **o modelo de Interferência:** cada nó carrega uma instância do modelo de interferência que decide para cada mensagem enviada para este nó, se a mensagem não pode chegar devido à interferência causada por outros pacotes ou outro ruído também;
- **o modelo de Confiabilidade:** decide por cada mensagem, se ele deve chegar ao destino ou não. Este modelo é interessante para simular uma rede com perdas de pacotes, onde algumas mensagens não chegam devido a várias razões;
- **o modelo de Transmissão:** determina o tempo que uma mensagem precisa até ela chegar ao nó destino.

## 4.2 Metodologia

O projeto consistiu de uma pesquisa exploratória envolvendo os conceitos sobre as RSSF, buscando um conhecimento na área, possibilitando analisar e estudar as difer-

entes propostas para protocolos de RSSF que tem como objetivo a eficiência no consumo de energia da rede ao realizar o roteamento e a transmissão de dados.

Foi identificado durante as pesquisas que o protocolo DESRO não foi projetado para funcionar com mais de um nó sorvedouro e não realiza o gerenciamento de rotas de forma dinâmica. Devido a estas deficiências, o protocolo NSINKS é proposto para torná-lo mais eficiente e extensível a redes com mais nós sorvedouros, buscando um melhor gerenciamento de energia.

A necessidade de adicionar mais de um nó sorvedouro na rede melhora o consumo de energia na rede, pois permite diminuir a sobrecarga nos nós sensores vizinhos ao nó sorvedouro, balanceando a carga na rede e diminuindo a possibilidade de partição da rede. As modificações realizadas para funcionar com mais de um nó sorvedouro e gerenciamento de rotas dinâmico no protocolo DESRO originaram o protocolo NSINKS.

Foram realizadas simulações com os protocolos DESRO e NSINKS para avaliar se as melhorias realizadas no protocolo DESRO aumentaram a expectativa de vida da rede.

### 4.3 Detalhes da Implementação

As implementações seguiram as especificações do protocolo DESRO adicionando as melhorias que originaram o protocolo NSINKS. Um ponto a destacar neste trabalho foi à forma como foram realizadas as simulações, pois quando o protocolo DESRO foi proposto, as simulações utilizaram o simulador Castalia e para este trabalho foi utilizado o simulador Sinalgo, sendo que uma característica que torna os simuladores diferentes é o fato de o simulador Sinalgo abstrair as camadas subjacentes. Desta forma, o protocolo DESRO e o protocolo NSINKS foram implementados conforme especificações do simulador Sinalgo.

Alguns pontos que merecem destaque na implementação do protocolo NSINKS são:

- **implementação dos nós** - foram definidas duas classes para os tipos de nós sensores, *class Fonte* e *class Sink* ambos herdam funcionalidades da *class Node*. Ambas tem características distintas, pois os nós **Fonte** têm a particularidade de enviar pacotes de sensoriamento e a funcionalidade de retransmitir os pacotes enviados em direção ao nó sorvedouro (*class Sink*). O nó sorvedouro (*class Sink*) tem como objetivo receber os pacotes enviados pelos nós Fontes (*class Fonte*) e realizar o descobrimento de rotas no início da simulação.



- **pacotes de dados** - os pacotes de dados foram definidos como pacotes de sensoriamento, pacotes de roteamento e pacotes de gerenciamento de rota. Cada pacote é tratado durante a simulação conforme seu tipo. Sendo o pacote de sensoriamento gerado pelo nó **Fonte** ao coletar eventos da área sensoriada, carregando informações como o nó **Fonte** que originou o pacote, o último nó **Fonte** por onde o pacote passou o dado coletado na simulação (para estas simulações, foi realizada a coleta referente à unidade do solo), a energia restante do nó Fonte e o ID do pacote. Os pacotes de roteamento gerados pelo nó sorvedouro carregam as informações referentes ao ID do pacote, ID do nó sorvedouro, a quantidade de *hops* que o pacote já percorreu e a energia acumulada da rota. Os pacotes de gerenciamento de rota carregam informações como: ID do pacote, energia estimada da rota e próximo nó vizinho.
- **cálculo da potência** - o nó sensor, ao enviar um pacote para o próximo nó, calcula a distância entre o nó origem e o nó destino e define a potência utiliza para transmissão de acordo com o quadrado da distância (foi definido que não existe interferência no ambiente). Foi definido que a maior distância ( $D$ ) irá utilizar a potência máxima, variando a potência de acordo com a distância. Na implementação foram consideradas quatro faixas de distância: até  $D/4$ ; de  $D/4$  até  $D/2$ ; de  $D/2$  até  $3D/4$  e de  $3D/4$  até  $D$ . No simulador Sinalgo a potência é definida pela intensidade do sinal, podendo variar de 0 até 1, alterando a forma como foi realizado as simulações do protocolo DESRO realizado por (2), que utilizou o módulo de rádio CC240 disponível na versão do simulador Castalia (versão 2.3b);
- **determinação dos possíveis vizinhos** - na inicialização da rede, cada nó sensor realiza a descoberta dos vizinhos, de acordo com a distância máxima definida como  $r_{max} = 40$  metros;
- **descobrimto de rotas** - os nós sorvedouros, ao serem inseridos na rede, realizam o procedimento de descobrimto de rotas enviando para os seus vizinhos um pacote de descobrimto de rotas utilizando a potência máxima para a transmissão do pacote. O descobrimto é realizado por todos os nós sorvedouros no mesmo instante, sendo que cada nó sorvedouro gera um pacote de descobrimto de rotas com o mesmo ID para todos os nós sorvedouros, desta forma o nó sensor ao receber o pacote de descobrimto de rotas, verifica o pacote é igual ao último recebido e caso seja o pacote é descartado. Desta forma os nós sorvedouros conseguem criar uma rota entre os nós sensores e os nós sorvedouros.

- **gerenciamento de rotas** - o pacote de gerenciamento de rota é realizado pelos nós sensores de acordo com o funcionamento da rede, ou seja, caso algum nó sensor seja removido da rede, se o número de pacotes enviados na rede atingirem um valor definido ou se o tempo da rede determinado para o monitoramento da rota chegar a zero. Estes valores são restaurados após o envio de um pacote de gerenciamento de rotas. O pacote de gerenciamento de rotas é enviado para os nós vizinhos do nó Origem e retransmitido para os nós vizinhos até atingir ao nó sorvedouro. O pacote do gerenciamento de rotas vai acumulando a energia dos nós sensores até atingir o nó sorvedouro. O nó sorvedouro, ao receber os pacotes de gerenciamento de rotas, envia um pacote de resposta contendo o caminho percorrido pelo pacote e a energia total da rota. O nó sensor, ao receber os pacotes enviados pelo nó sorvedouro, vai determinar qual rota será utilizada, baseado na rota que tiver maior energia.
- **sensoriamento de eventos** - o sensoriamento dos eventos é realizado por todos os sensores. O nó sensor ao realizar o sensoriamento envia um pacote para a rota que possui a maior energia estimada.

## 4.4 Simulações

As simulações foram realizadas seguindo os mesmos parâmetros utilizados nas simulações utilizadas por (2), conforme Tabela 4.1.

Foram realizadas 20 simulações em grade utilizando o protocolo NSINKS e 20 simulações em grade com o protocolo DESRO, simulando um sensoriamento da umidade do solo de uma região de 160x360 metros. Para o protocolo DESRO foram utilizados 49 nós sensores realizando o sensoriamento do solo a cada 10 minutos e 01 nó sorvedouro recebendo os dados do sensoriamento. Para o protocolo NSINKS foram utilizados 49 nós sensores realizando o sensoriamento do solo a cada 10 minutos e 02 (dois) nós sorvedouros recebendo os dados do sensoriamento. A área é sensoriada por 01 (um) mês. Para a distribuição em grade foi definido a energia inicial dos nós sensores em 20 Joules. Para o protocolo NSINKS foram realizadas mais 20 simulações na distribuição em grade, sendo 49 nós sensores realizando o sensoriamento do solo a cada 10 minutos, com uma energia inicial de 20 Joules, onde a quantidade de nós sorvedouros variou em 03, 04 e 05, sendo que para cada configuração foram realizadas 05 simulações.

As simulações foram realizadas no modo de simulação síncrona, onde a execução das ações são executadas a cada *round*. A execução é estritamente seqüencial, onde para cada *round* é realizado:

**Tabela 4.1.** Parâmetros utilizados nas simulações

<i>Descrição</i>	<i>Valores</i>
Número Simulações em grade - DESRO	20
Número Simulações em grade - NSINKS	20
Área sensorizada	160 X 360 metros
Quantidade de nós sensores	49 nós sensores
Intervalo de sensoriamento	10 minutos
Energia Inicial dos nós sensores	20 joules

1. O Sinalgo incrementa a variável global *round* em 1.
2. É decrementado o tempo da rede determinado para o monitoramento de rota; verifica se o tempo da simulação atingiu os 30 dias; verifica se foram removidos mais que 50% dos nós sensores da rede e se a rede atingiu o tempo definido para realizar o sensoriamento.
3. O Sinalgo envia os pacotes, seja de gerenciamento ou de sensoriamento.
4. O Sinalgo move os nós de acordo com o modelo de mobilidade definido, sendo para estas simulações não existe mobilidade.
5. O Sinalgo executa o modelo de conectividade para atualizar os vizinhos dos nós sensores.
6. O Sinalgo testa a interferência para todas as mensagens enviadas, se a interferência foi habilitada. Para estas simulações foi definido que não existe interferência.
7. Caso a energia de algum nó sensor fique menor ou igual a zero (energia = 0) o nó sensor é removido da simulação.

Como o Sinalgo realiza as ações conforme o número de *rounds* e o tempo determinado para cada simulação foi de 30 dias, foi definido que cada dia durou 23100 *rounds* fazendo com que cada simulação durasse 693000 *rounds*. O gerenciamento de rotas foi configurado com um temporizador de 115500 *rounds* e o contador de pacotes para 144 transmissões, conforme parâmetros definidos em (2).

Para análise do consumo de energia, foi utilizado um modelo conforme proposto por (2), onde considera apenas o consumo de energia gasta com transmissões de pacotes de dados e de controle da camada de rede, não computando a energia gasta pela camada de enlace, pois o simulador Sinago abstrai esta camada. Conforme (2), a energia gasta

por um pacote de dados de comprimento  $D$  bytes transmitido numa potência  $Pt$  é dada pela equação (4.1):

$$E(D, Pt) = \frac{PtD}{L} \quad (4.1)$$

onde  $L$  é a largura de banda do *link* em bytes por segundo definida em 250 *kpbs* e  $Pt$  a potência de transmissão calculada de acordo com a distância entre o nó origem e o nó destino e  $D$  o comprimento do pacote de dados definido em 9 bytes.

As simulações foram realizadas em um notebook com o processador AMD Athlon(tm) X2 Dual-Core QL-62 2.00 GHz, 4,00 GB de memória RAM (*Random Access Memory*), SO Windows 7 Ultimate - 64 bits. A geração dos dados foi implementada de forma dinâmica, gerando os dados no final de cada simulação, sendo observadas as seguintes variáveis:

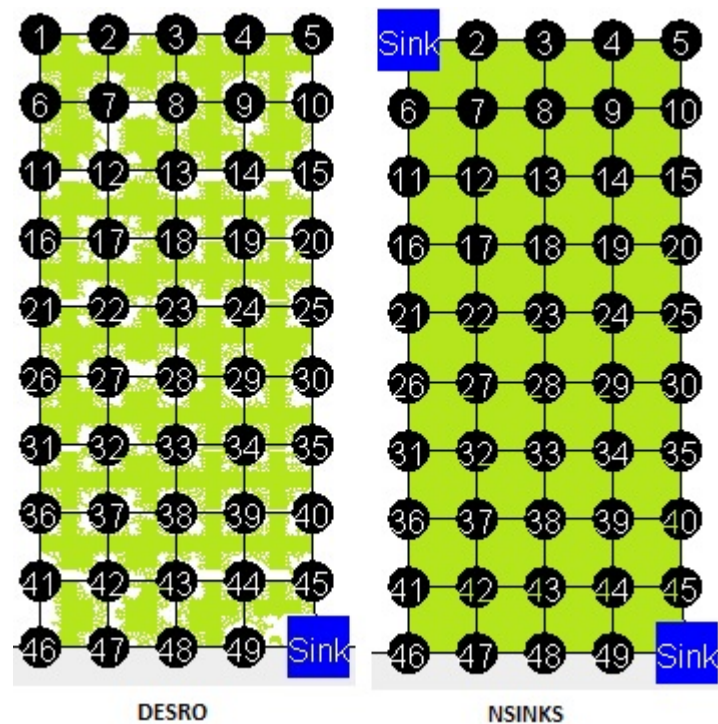
- consumo de energia;
- distribuição de energia;
- taxa de entrega;

## 4.5 Resultados

### 4.5.1 Distribuição em grade

Na distribuição em grade os nós foram distribuídos com uma distância de 40 metros entre si. O nó sorvedouro é representado na Figura 4.1 como *Sink*, sendo os demais nós sensores fontes. Na Figura 4.1 é representado também uma rede com um nó sorvedouro utilizado nas simulações do protocolo DESRO e uma rede com dois nós sorvedouros utilizado para as comparações entre o protocolo NSINKS e o protocolo DESRO, sendo que na área que representa a simulação do protocolo NSINKS, o nó fonte um está na mesma posição que o nó sink superior.

Conforme a disposição dos nós na distribuição em grade, o total de mensagens geradas pela rede para as 20 simulações foi de 5.181.925 para o protocolo NSINKS e 4.214.327 para o protocolo DESRO. A explicação para esta diferença de 967.598 a mais para o protocolo NSINKS é devido à inserção de mais um nó sorvedouro na rede, balanceando o envio de pacotes entre os nós sorvedouros, dividindo a sobrecarga para os nós sensores vizinhos aos dois nós sorvedouros. Com este balanceamento os nós sensores vizinhos aos nós sorvedouros ficaram menos sobrecarregados e esgotaram sua energia de forma mais lenta, evitando a perda de comunicação com o nó sorvedouro e o esgotamento total de sua energia. Os nós sensores não esgotando sua energia,



**Figura 4.1.** Representação do nó sorvedouro

continuam realizando o sensoriamento do solo e com isto gera-se uma quantidade maior de pacotes na rede.

O número de pacotes entregues para o protocolo NSINKS foi de 2.686.625 pacotes para o nó sorvedouro 01 e de 2.451.984 para o nó sorvedouro 02, totalizando 5.138.925 (uma taxa 99,16% de pacotes entregues) pacotes recebidos pelos nós sorvedouros com uma perda de 43.406 (taxa de 0,84% pacotes perdidos) pacotes. Para o protocolo DESRO, foram entregues 3.740.642 (taxa de 88,76% pacotes entregues) pacotes para o nó sorvedouro com uma perda de 473.685 (taxa de 11,24% pacotes perdidos) pacotes.

O NSINKS não só gerou mais pacotes como também entregou mais aos nós sorvedouros. A Figura 4.2 exibe os resultados da média da taxa de entrega dos pacotes nas simulações para cada protocolo. É apresentado a porcentagem da taxa de entrega para as simulações.

O objetivo de inserir mais de um nó sorvedouro na rede ajudou a balancear a carga de pacotes enviados aos nós sensores vizinhos dos nós sorvedouros, evitando que ocorra o esgotamento de energia destes nós sensores e a perda de comunicação com os nós sorvedouros, aumentando a expectativa de vida destes nós sensores. O gerenciamento de rotas permitiu tanto para o protocolo NSINKS como para o protocolo DESRO um melhor roteamento de pacotes.

As Figuras 4.3 e 4.4 ilustram a distribuição média de energia gasta pelos nós

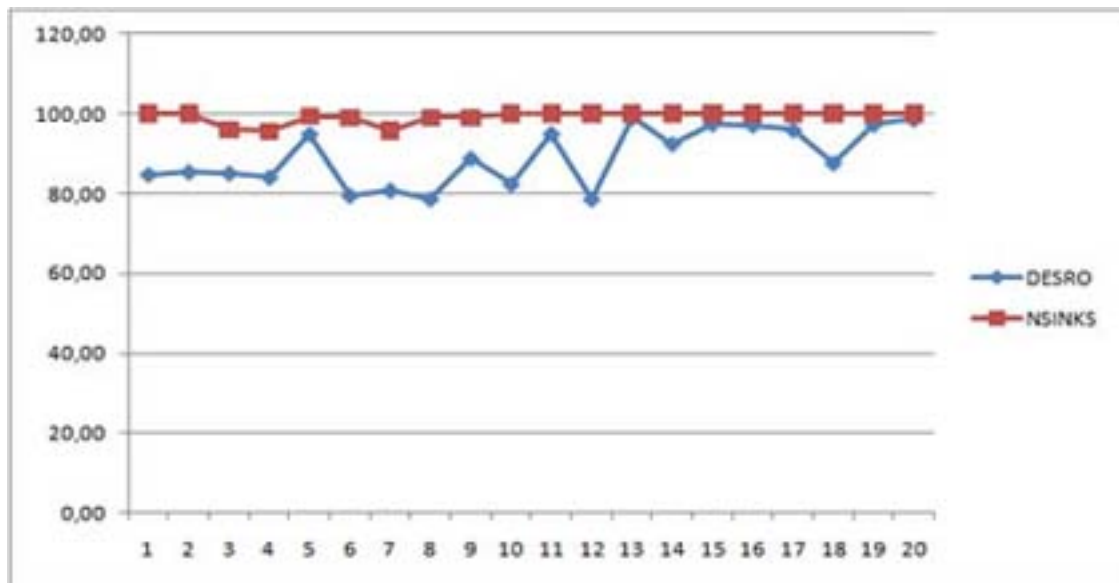


Figura 4.2. Taxa de Entrega dos protocolos

sensores ao utilizar os protocolos NSINKS e DESRO durante as simulações realizadas.

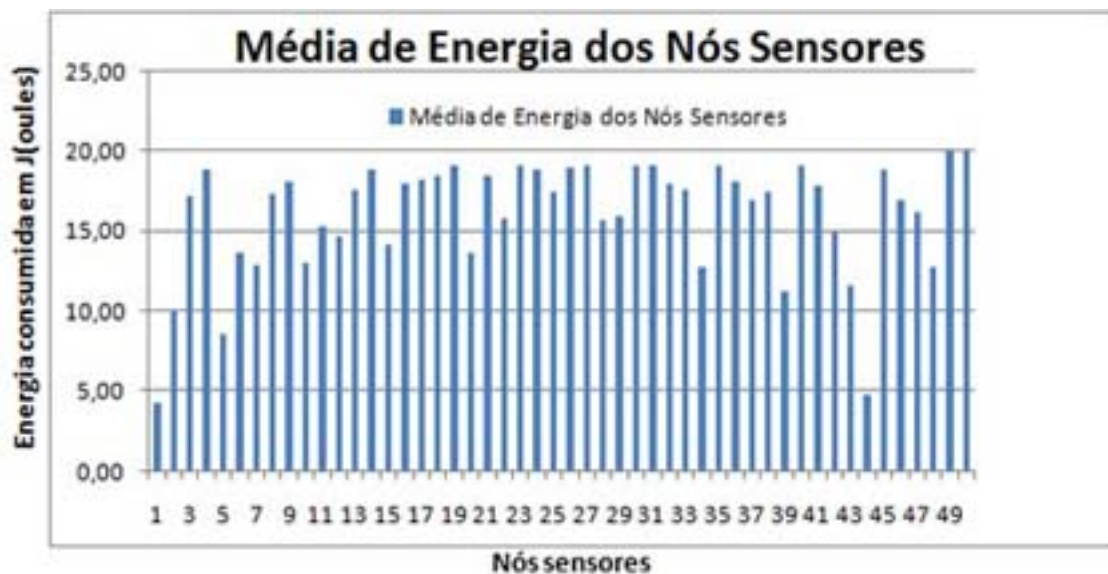


Figura 4.3. Média de Energia para o NSINKS

Pelas Figuras 4.3 e 4.4 podemos observar que os nós sensores que consumiram mais energia durante as simulações foram os nós sensores que estão próximos aos nós sorvedouros, sendo para o protocolo NSINKS temos os sensores de ID: 02 e 44 e para o protocolo DESRO temos os sensores de ID: 44, 45 e 49. Podemos observar pela Figura 4.4 que a média de energia dos sensores vai diminuindo à medida que os sensores estão próximos aos nó sorvedouro, mas devido ao gerenciamento de rotas foi possível

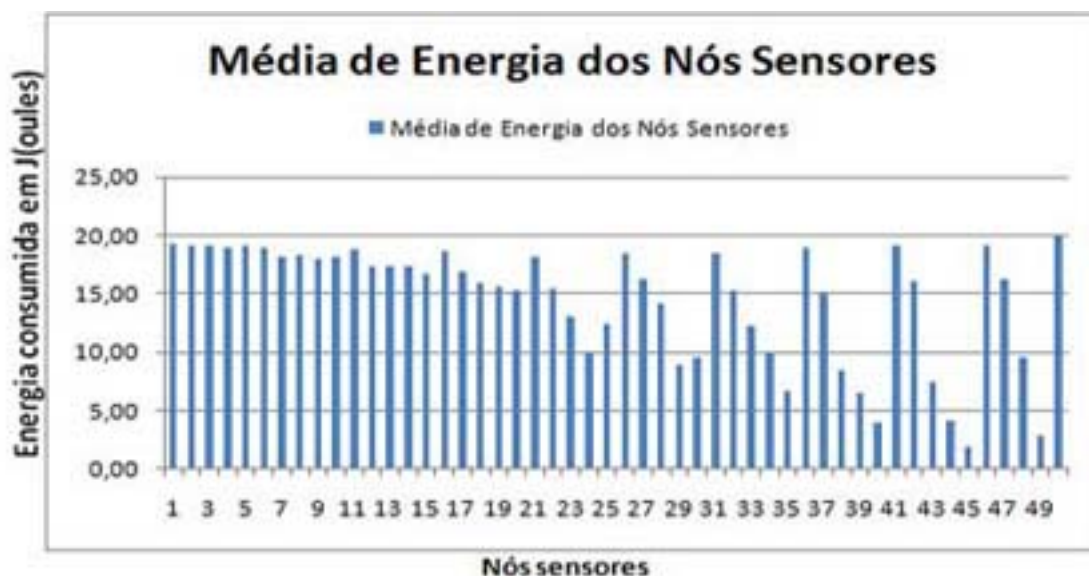


Figura 4.4. Média de Energia para o DESRO

balancear a transmissão do pacote para o nó sorvedouro pelos nós sensores que fazem vizinhança com o nó sorvedouro, desta forma evitando o esgotamento de energia destes nós.

A Figura 4.5 destaca as regiões da rede que consumiram mais energia durante as simulações e simboliza os dados representados pelas Figuras 4.3 e 4.4.

#### 4.5.2 Simulação do protocolo NSINKS com 03, 04 e 05 nós sorvedouros

Na distribuição em grade utilizando o protocolo NSINKS e variando o número de nós sorvedouros em 03, 04 e 05 nos nós foram distribuídos com uma distância de 40 metros entre si. O nó sorvedouro é representado na Figura 4.6 como Sink, sendo os demais nós sensores fontes. Na Figura 4.6 é representado as redes variando o número de nós sorvedouros utilizados nas simulações. O objetivo destas simulações foi verificar se o número de nós sensores para um mesmo conjunto de nós fontes muda o comportamento da rede, seja em número de pacotes de sensoriamento enviados ou em relação ao esgotamento de energia dos nós sensores.

Conforme a disposição dos nós na distribuição em grade, o total de mensagens geradas para o protocolo NSINKS pela rede para as 05 simulações com 03 nós sorvedouros foi de 1.561.028, 1.562.139 para as 05 simulações com 04 nós sorvedouros e 1.560.775 para as 05 simulações com 05 nós sorvedouros. A inserção de mais nós sorvedouros na rede, evitou o esgotamento dos nós sensores vizinhos aos nós sorvedouros,

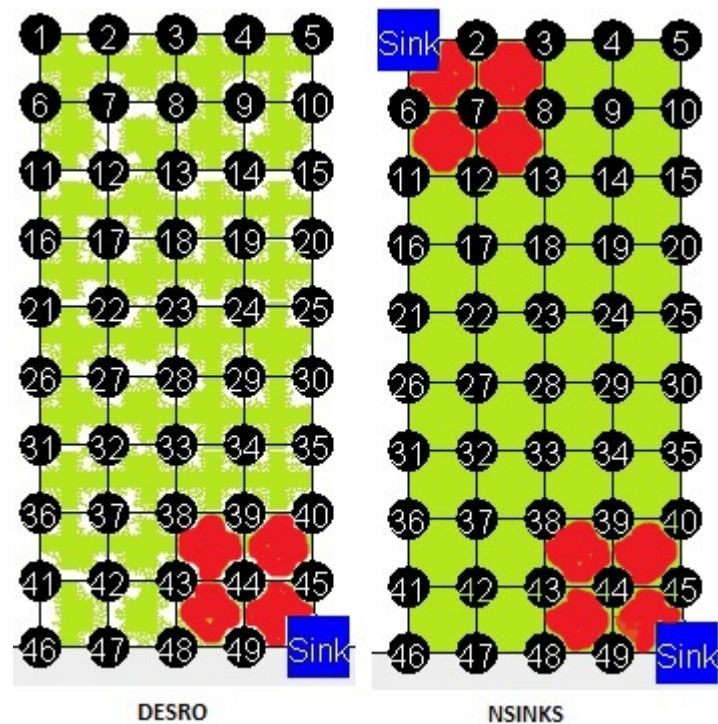


Figura 4.5. Regiões que consumiram mais energia

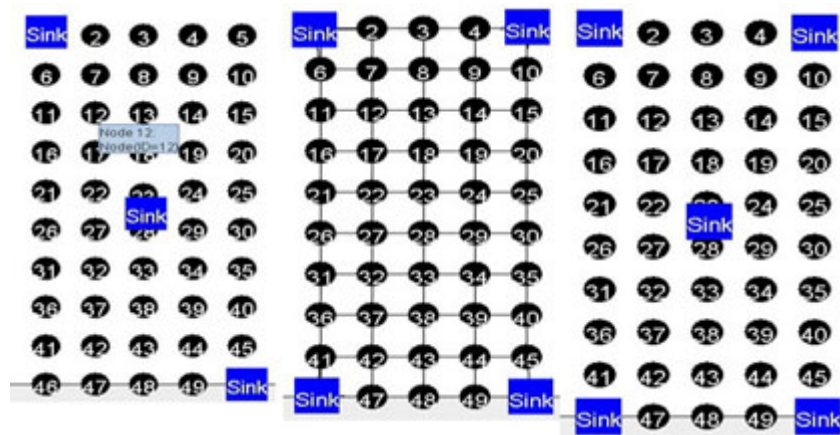


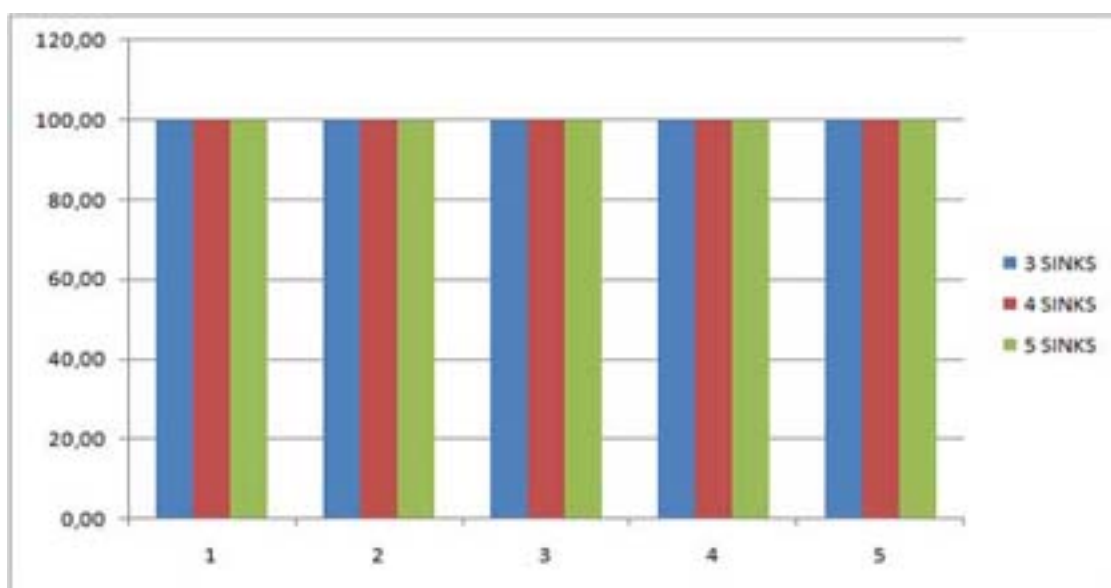
Figura 4.6. Representação do nó sorvedouro e os ambientes de simulação com n nós sorvedouros

realizando o balanceamento de envio de pacotes entre os nós sorvedouros, dividindo a sobrecarga para os nós sensores vizinhos aos nós sorvedouros. Com este balanceamento os nós sensores vizinhos aos nós sorvedouros ficaram menos sobrecarregados e evitaram o esgotamento da sua energia, evitando a perda de comunicação com o nó sorvedouro e o esgotamento total de sua energia. Os nós sensores não esgotando sua energia continuam realizando o sensoriamento do solo gerando uma quantidade maior de pacotes na rede.



A taxa de entrega foi de 100% para as 15 simulações variando apenas a quantidade de nós sorvedouros e a quantidade de pacotes entregues para cada nó sorvedouro. O motivo pelo qual o protocolo NSINKS entregou 100% foi devido à inserção de mais nós sorvedouros na 55 rede e do gerenciamento de rotas que é realizado durante a simulação, evitando utilizar apenas uma rota.

A Figura 4.7 exhibe os resultados da média da taxa de entrega dos pacotes nas simulações para cada variação do número de nós sorvedouro na rede. É representada a porcentagem da taxa de entrega para as simulações.



**Figura 4.7.** Taxa de Entrega conforme a variação do número de nós sorvedouros

O objetivo de inserir mais de um nó sorvedouro na rede ajudou a balancear a carga de pacotes enviados aos nós sensores vizinhos dos nós sorvedouros, evitando que ocorra o esgotamento de energia destes nós sensores e a perda de comunicação com os nós sorvedouros, aumentando a expectativa de vida destes nós sensores. O gerenciamento de rotas permitiu um melhor balanceamento da rede.

As Figuras 4.8 e 4.9 e 4.10 ilustram a distribuição média de energia gasta pelos nós sensores durante as simulações, variando a quantidade de nós sorvedouros.

Pelas Figuras 4.8, 4.9 e 4.10 podemos observar que os nós sensores que consumiram mais energia durante as simulações foram os nós sensores que estão próximos aos nós sorvedouros.

Na Tabela 4.2 podemos observar que com a inserção de mais nós sorvedouros na rede, o gerenciamento de rotas balanceia os pacotes de sensoriamento, distribuindo os pacotes entre os nós sensores. Sendo que à medida que fomos adicionando nós sorve-

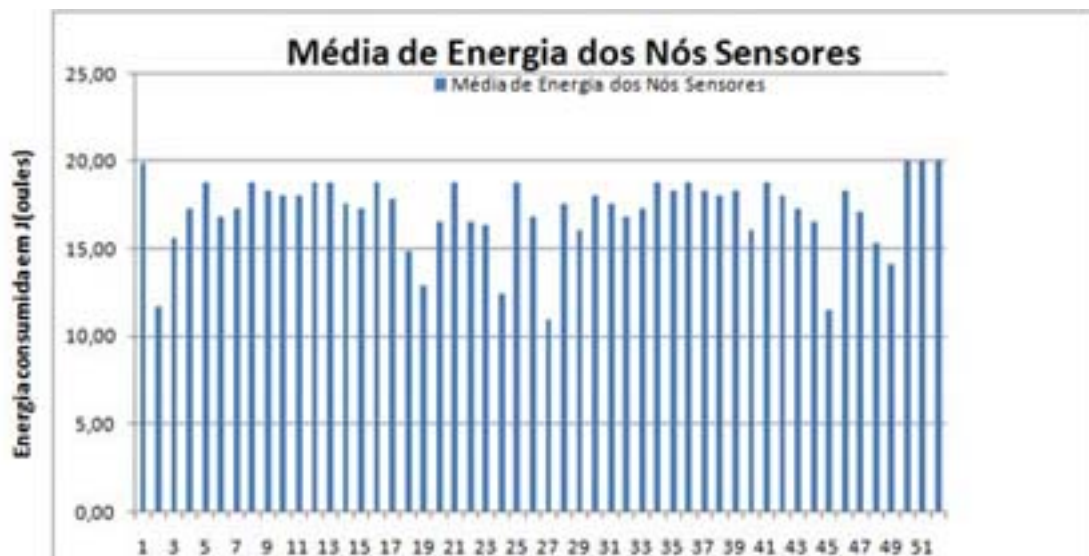


Figura 4.8. Média de Energia para uma rede com três nós sorvedouros

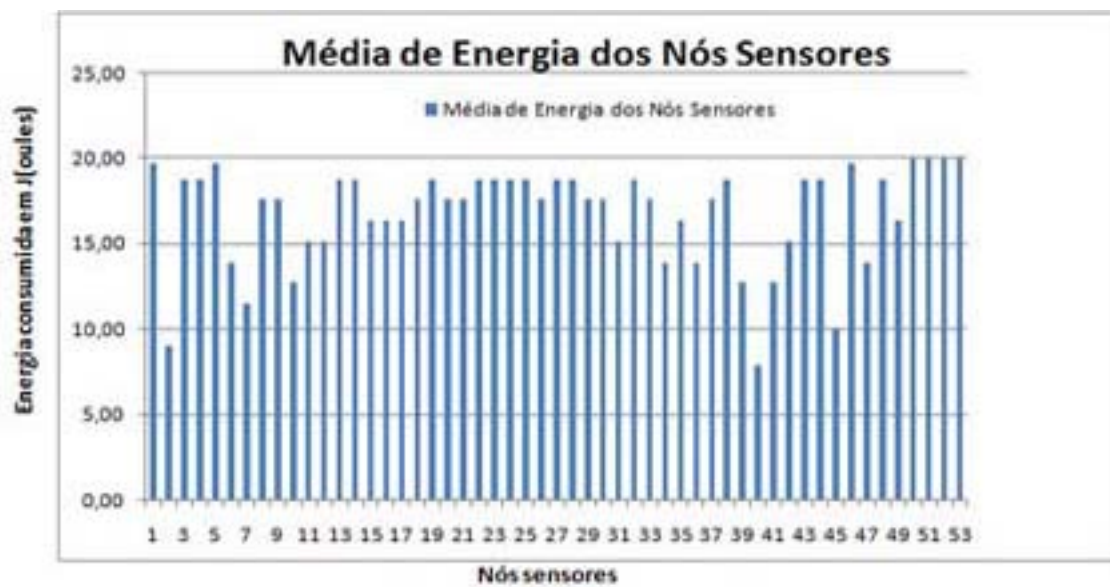
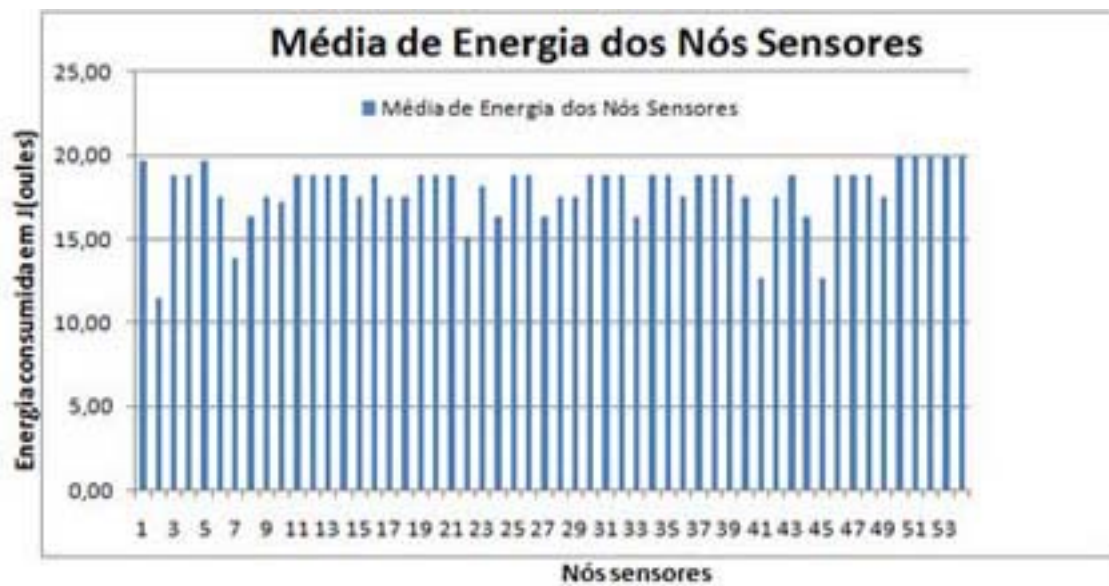


Figura 4.9. Média de Energia para uma rede com quatro nós sorvedouros

douros na rede, o número de pacotes recebidos por cada nó sorvedouro vai diminuindo, comprovando a eficiência do gerenciamento de rotas.



**Figura 4.10.** Média de Energia para uma rede com cinco nós sorvedouros

**Tabela 4.2.** Pacotes Recebidos pelos Nós Sorvedouros

<i>Nó Sorvedouro</i>	<i>Média de pacotes recebidos nas simulações para cada nó sorvedouro</i>
Nó Sorvedouro 1	81839,4
Nó Sorvedouro 2	125177,6
Nó Sorvedouro 3	105188,6
Nó Sorvedouro 1	90365,0
Nó Sorvedouro 2	58330,4
Nó Sorvedouro 3	87974,6
Nó Sorvedouro 4	75757,8
Nó Sorvedouro 1	41538,6
Nó Sorvedouro 2	64382,4
Nó Sorvedouro 3	42456,2
Nó Sorvedouro 4	57633,4
Nó Sorvedouro 5	106144,4

## Capítulo 5

# Conclusões e Trabalhos Futuros

Neste trabalho foi apresentado o protocolo NSINKS, um protocolo que surgiu das alterações realizadas do protocolo DESRO, como a característica de utilizar mais de um nó sorvedouro na rede e realizar o gerenciamento de rotas de forma dinâmica para melhor o tempo de vida da rede. NSINKS contempla as características desejáveis para uma rede *adhoc* (sem o uso de estação base), tendo como objetivo um melhor balanceamento de energia da rede. NSINKS foi projeto para RSSFs com nós sensores estáticos, podendo ser extensível para RSSFs com nós móveis.

Os resultados apresentados demonstraram que o protocolo NSINKS obteve desempenho melhor que o protocolo DESRO em todos os cenários simulados. Isso pode ser explicado, de forma geral, devido à inserção de mais de um nó sorvedouro realizando o balanceamento do envio de pacotes entre os nós sorvedouros, dividindo a sobrecarga para os nós sensores vizinho aos dois nós sorvedouros. Com este balanceamento os nós sensores vizinhos aos nós sorvedouros ficaram menos sobrecarregados e esgotaram sua energia de forma mais lenta, evitando a perda de comunicação com o nó sorvedouro e o esgotamento total de sua energia. Os nós sensores, não esgotando sua energia, continuaram realizando o sensoriamento do solo e com isto gerando pacotes na rede.

Nas simulações realizadas com o protocolo NSINKS variando apenas a quantidade de nós sorvedouros, a taxa de entrega de pacotes foi de 100% para os trinta dias de simulação, pois o gerenciamento de rotas conseguiu balancear os pacotes de sensoriamento para serem entregues entre os diferentes nós sorvedouros disponíveis na rede, evitando o esgotamento de energia por parte dos nós sensores que fazem vizinhança com os nós sorvedouros, permitindo um maior envio de pacotes de sensoriamento. Conforme adicionamos mais nós sorvedouros na rede, o número de pacotes recebidos por cada nó sorvedouro vai diminuindo, comprovando a eficiência do gerenciamento de rotas.

Como um possível trabalho futuro sugere-se a realização de novas simulações

para verificar o comportamento da rede em situações mais extremas, aumentando o tempo de simulação ou reduzindo a carga inicial dos nós sensores. Desta forma poderia ser verificado de forma mais precisa o efeito do uso de vários nós sensores na taxa de entrega de mensagens, em situações onde haverá problemas de falta de energia dos sensores.

Outra sugestão de trabalho futuro é a disposição dos nós sensores de maneira não tão determinísticas. A simulação de cenários como estes se justificam por nem sempre ser possível a definição precisa da localização dos nós sorvedouros. Principalmente em aplicações de monitoramento ambiental ou em catástrofes onde o acesso aos locais de interesse não é possível e os nós sensores são espalhados com auxílio de alguma aeronave, tripulada ou não.

Uma evolução deste tipo de trabalho poderia ser a investigação sobre o número de nós sensores e de nós sorvedouros necessários para se definir uma cobertura em uma região de interesse e a determinação de um tempo de vida mínimo para a rede com tais características. Neste caso, a determinação de uma relação custo-benefício poderia ser analisada em função das características da aplicação alvo. Por exemplo, em uma missão de resgate em uma área atingida por terremoto, qual o número necessário de nós sensores e sorvedouros para detectar sinais de sobreviventes em um período de cinco dias.

# Referências Bibliográficas

- 1 TANENBAUM, A. *Redes de Computadores*. [S.l.]: Campus, 4ª Edição, 2003.
- 2 DANTAS, M. Desro: um protocolo de roteamento com gerenciamento dinâmico de energia para redes de sensores sem fio. *Dissertação - Universidade Federal de Viçosa - UFV*, 2010.
- 3 AKYILDIZ I.; SU, W. S. Y. C. E. A survey on sensor networks. *In A, editor, IEEE Comm. Magazine*, v. 40, p. 102–114, 2002.
- 4 PAPAVALASSILOU S.; ZHU, J. *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*. [S.l.: s.n.], 2005. 15-3 à 15-5 p.
- 5 FENG J.; KOUSHANFAR, F. P. M. Handbook of sensor networks: Compact wireless and wired sensing systems. In: . [S.l.]: CRC Press LLC, 2005. Sessão III - Capítulo 12, p. 12–1 à 12–16.
- 6 WEILIAN S; ERDAL, C. O. B. Handbook of sensor networks: Compact wireless and wired sensing systems. *In: Proceedings of 3rd International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks. S.l.: IEEE*, Sessão IV - Capítulo 16, p. 16–1 à 16–14, 2005.
- 7 LI L.; HALPERN, J. Minimum-energy mobile wireless network, revisited. *IEEE Int. Conf. Conf.- New York*, 2002.
- 8 HEINZELMAN W. R.; CHANDAKASAN, A. B. H. Energy-efficient communication protocol for wireless microsensor networks. *IEEE Proc. Hawa Int. Conf. Syst. Sci.*, Janeiro 2000.
- 9 SOHRABI, K. Protocols for self-organization of a wireless sensor network. *IEEE Personal Commum*, p. 16–27, Outubro 2000.

- 10 HEDETNIEMI S.M.; HEDETNIEMI, T. L. A. A survey of gossiping and broadcasting in communication networks. *Networks - An International Journal*, v. 18, 4<sup>a</sup> Edição, p. 319–349, 1988.
- 11 HEINZELMAN W.R.; CHANDAKASAN, A. B. H. Adaptive protocols for information dissemination in wireless sensor networks. *ACM Mobicom 99-Seattle, W*, p. 174–185, Agosto 1999.
- 12 INTANAGONWIWAT C.; GOVINDAN, R. E. D. Directed diffusion: a scalable and robust communication paradigm for sensor networks. *ACM Mobicom - Boston MA*, Agosto 2000.
- 13 WANG Q.; HASSANEIN, H. Handbook of sensor networks: Compact wireless and wired sensing systems. In: *Proceedings of 3rd International Symposium on Modeling and Op-timization in Mobile, Ad Hoc and Wireless Networks. S.l.: IEEE*, p. 18–1 à 18–22, 2005.
- 14 JOHNSON D.B.; MALTZ, D. Dynamic source routing in ad hoc wireless networks. In: *Imielinski, T.; Korth, H. F. (Ed.). Mobile Computing. [S.l.]: Kluwer Academic Publishers, (The International Series in Engineering and Computer Science, v. 353)*, p. 153–181, 1996.
- 15 TARIQUE M.; TEPE, K. N. M. Energy saving dynamic source routing for ad hoc wireless networks. In: *Proceedings of 3rd International Symposium on Modeling and Op-timization in Mobile, Ad Hoc and Wireless Networks. S.l.: IEEE*, Sessão IV, Capítulo 18, p. 305–310, 2005.