Avaliando aspectos de tolerância a falhas em protocolos de roteamento para redes de sensores sem fio*

Daniel F. Macedo[1], **Luiz H. A. Correia**[1,2], **Aldri L. dos Santos**[1,3], **Antonio A. F. Loureiro**[1], **José Marcos S. Nogueira**[1†]

[1]Dep. de Ciência da Computação
Univ. Federal de Minas Gerais
Belo Horizonte-MG, Brasil

[2]Dep. de Ciência da Computação
Univ. Federal de Lavras
Lavras-MG, Brasil

[3]Dep. de Computação
Univ. Federal do Ceará
Fortaleza-CE, Brasil

{damacedo,lcorreia,aldri,loureiro,jmarcos}@dcc.ufmg.br

Abstract. *Fault tolerance is an essential requirement in the design of protocols and applications for Wireless Sensor Networks (WSNs) since communication and hardware failures are frequent. In this paper we studied the resilience of routing protocols for continuous data dissemination WSNs in face of faults. The main causes of silent failure are presented, including some security attacks. Those failures are classified according to extension and persistence, and such classification is used to evaluate routing protocols for continuous data dissemination networks. Results show that failures under a large region of the network are the most damaging. The paper also shows how routing protocols may save energy by temporarily turning off disconnected nodes.*

Resumo. *Tolerância a falhas é um requisito essencial para o projeto de protocolos e aplicações para Redes de sensores sem fio (RSSF), pois falhas de hardware e comunicação são frequentes. Neste trabalho estudamos o comportamento de protocolos de roteamento para redes de disseminação contínua de dados perante a ocorrência de falhas. Apresentamos os principais agentes causadores de falhas silenciosas, incluindo ataques de segurança. Classificamos estas falhas quanto a extensão e persistência, e utilizamos esta classificação para avaliar, via simulação, protocolos de roteamento para redes de disseminação contínua de dados. Verificamos que falhas em grandes regiões da rede são o tipo mais prejudicial, e mostramos como protocolos de roteamento podem economizar energia desligando temporariamente nós isolados da rede.*

1. Introdução

Redes de Sensores Sem Fio (RSSF) são uma subclasse das tradicionais redes ad hoc sem fio. As RSSF são formadas por elementos de rede chamados de nós sensores, que são dispositivos compactos compostos de sensores, processador, rádio, memória e bateria [1]. Esses nós enviam os dados coletados para um Ponto de Acesso (PA), responsável por repassar os dados ao usuário final. Ao contrário das redes ad hoc tradicionais, em geral não é possível trocar ou recarregar a fonte de energia dos nós devido à sua grande quantidade ou às dificuldades impostas pelo ambiente. Assim, o consumo de energia é um fator crítico em RSSF, o que tem motivado o desenvolvimento de protocolos específicos, que tomam decisões procurando otimizar o consumo de energia.

Tolerância a falhas é um dos componentes que constituem um sistema confiável (*dependable*) [2], que é um dos grandes desafios da computação atual [3]. Nas RSSF, falhas são frequentes, e ocorrem em virtude de eventos como a destruição de nós, degradação da qualidade do enlace, entre outros. Visto que essas redes podem ser empregadas em ambientes hostis, como áreas de desastre, os nós podem ser destruídos a qualquer momento, seja por deslizamento, queda de árvores ou prédios, enchentes ou outros agentes naturais. Falhas também ocorrem na comunicação, devido a interferências ocorridas por modificações no clima ou na movimentação de objetos no espaço sensoriado, que bloqueiam o sinal transmitido, bem como agentes maliciosos, que têm como objetivo degradar o serviço da rede. Por se tratar de um ambiente altamente propenso a

*O presente trabalho foi realizado com apoio do CNPq, uma entidade do Governo Brasileiro voltada ao desenvolvimento científico e tecnológico. Processo 55.2111/2002-3.

†Em período sabático nas universidades de Evry e UPMC/Paris6/Lip6, França.

falhas, e considerando o alto grau de interação entre os elementos (os nós em RSSF operam de forma colaborativa), o software dos nós está sujeito a erros devido ao mal funcionamento de outros nós da rede. Assim, protocolos e aplicações em RSSF devem ser desenvolvidos considerando a ocorrência frequente de falhas.

Em protocolos de roteamento as falhas se manifestam como rotas interrompidas e ocorrem devido a erros na comunicação ou defeitos no hardware. Ao detectar um falha na rota, o protocolo de roteamento deve identificar uma rota operacional, permitindo assim que o tráfego entre dois nós seja restaurado. Falhas no roteamento em RSSF são mais graves que em redes ad hoc. Em geral nas RSSF todos os dados são endereçados para um ponto de acesso. Assim, uma rota danificada pode afetar um grande número de fluxos de dados. Em uma rede ad hoc, entretanto, um nó pode enviar dados para qualquer outro nó da rede. Assim, caso um nó falhe, apenas as rotas que dependiam deste nó estarão falhas.

Em RSSF o fluxo de dados segue um padrão, isto é, os dados são processados localmente e enviados para o PA. Este envio pode ser periódico ou esporádico, caracterizando dois tipos de fluxo de dados [4]. Nas *redes dirigidas a eventos*, o envio de dados é ocasional, ocorrendo somente quando verificada uma determinada condição (chamada de evento). Redes dirigidas a evento são utilizadas na localização de animais silvestres, detecção de intrusão, monitoramento de queimadas, entre outros. Nas *redes de disseminação contínua*, os nós enviam mensagens em intervalos regulares para um PA, relatando as leituras atuais dos seus sensores. Em tais redes é possível montar um "mapa" do estado atual da região monitorada, sendo este utilizado para análise de variações espaciais e temporais de eventos. Aplicações dessas redes incluem estudos ambientais, sistemas de tráfego inteligente, monitoração de plantas industriais, entre outros.

Devido às diferenças nas características de tráfego, os protocolos de roteamento são implementados para satisfazer a apenas uma classe de rede. Redes de disseminação contínua de dados tendem a utilizar protocolos pró-ativos, pois a todo momento os nós enviam dados para um PA. Em redes dirigidas a eventos os protocolos são reativos, assim as rotas são construídas somente na ocorrência de um evento de interesse, e apenas na região onde o evento ocorreu. Como os eventos tendem a ser raros e isolados em uma região, a reconstrução periódica de rotas não é recomendável devido ao alto custo de energia associado. O mesmo ocorre com os métodos de tolerância a falhas. Em redes de disseminação contínua, mecanismos pró-ativos se justificam pelo grande volume de dados, enquanto que em redes baseadas em eventos os mecanismos de tolerância a falhas tendem a operar somente quando houver um fluxo de dados.

Neste trabalho estudamos o comportamento de protocolos de roteamento para redes de disseminação contínua de dados, tendo em vista falhas de comunicação e de *hardware* onde o nó não envia dados durante a ocorrência da falha (*falhas silenciosas*). Identificamos os principais agentes causadores de falhas, e caracterizamos as falhas de acordo com a sua extensão e persistência. Esta caracterização é utilizada para extrair características comuns das falhas, e facilitar a avaliação dos protocolos.

O texto está organizado da seguinte forma. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 apresenta uma visão geral dos protocolos avaliados, descrevendo o seu funcionamento e algoritmos de tolerância a falhas. A Seção 4 expõe os agentes causadores de falhas silenciosas em RSSF. A Seção 5 define uma classificação para estas falhas, que é utilizada na avaliação dos protocolos, apresentada na Seção 6. Por fim, a Seção 7 apresenta as conclusões e trabalhos futuros.

2. Trabalhos Relacionados

Avizienis et al. apresentam uma taxonomia de falhas, que engloba questões de segurança [5]. Esta taxonomia considera os desafios e as questões atualmente encontrados nos sistemas atuais. Hollick et al. apresentam os desafios correntes e futuros na tolerância a falhas em RSSF, redes ad hoc e redes celulares, e listam as características necessárias para o desenvolvimento de protocolos confiáveis para estas redes [6]. Koushanfar et al. apresentam uma visão geral de tolerância a falhas em RSSF considerando elementos de hardware, e por fim resumem as técnicas correntes de detecção de falhas bizantinas em sensores, utilizando métodos de correlação [7].

Os primeiros protocolos propostos para RSSF [8, 9] se preocupam com falhas devido ao esgotamento da bateria, e propõem mecanismos para aumentar a vida do nó e distribuir a energia gasta. Estes protocolos não tratam da falha de nós, um evento frequente em RSSF. Outros protocolos se preocupam com falhas na comunicação ocasionadas por quebra de nós, amenizando este problema por uso de múltiplas rotas na transmissão de dados [10, 11]. Visto que cópias dos dados são enviados por múltiplos caminhos, os dados possuem uma maior probabilidade de serem recebidos pelo PA, ao custo de um maior consumo de energia. Um estudo do aumento da probabilidade de recepção de um pacote de acordo com o grau de similaridade das rotas é apresentado por Ganesan et al. [12]. O estudo demonstrou que rotas parcialmente disjuntas são tão eficazes quanto rotas totalmente disjuntas, e possuem um menor custo para serem estabelecidas.

O envio de múltiplas cópias de dados tem um custo elevado, assim é desejável manter apenas uma rota de boa qualidade. De Couto et al. sugerem uma modificação no protocolo de roteamento ad hoc DSR (*Dynamic Source Routing*) para que este considere a qualidade do enlace de uma rota [13], amenizando falhas de comunicação. O protocolo DSR, ao propagar uma requisição de estabelecimento de rota, também propaga um valor acumulado que indica a qualidade do sinal na rota. O nó utiliza sempre a rota com a maior qualidade, evitando enlaces com comunicação intermitente. Alec Woo et al. [14] propuseram um mecanismo para protocolos de roteamento em RSSF que realiza o processamento localmente. Ambas as soluções, entretanto, contemplam um conjunto restrito das falhas em RSSF.

Dado a ocorrência de uma rota falha, é necessário identificar uma rota alternativa. Vieira et al. identificaram duas soluções para a falha de nós devido ao esgotamento de energia [15]. Na primeira, chamada de *Smart-Sink*, o PA notifica aos nós que a rota deve ser modificada, pois um dos nós que compõem a rota possui baixa energia. Na segunda, chamada de *lista de padrastos*, um nó possui uma lista de rotas sobressalentes ao PA. No caso de falha da rota padrão, uma das rotas sobressalentes é utilizada. Ambas as abordagens funcionam apenas para falhas locais, e necessitam de um mecanismo de detecção de falhas, que não é tratado no artigo. Khanna et al. apresentam um mecanismo de rotas alternativas para o protocolo SPIN [16]. Este trabalho apresenta apenas resultados analíticos, não existindo uma definição clara do modelo de falhas tratado. Neste artigo, por outro lado, definimos o modelo de falhas considerado e realizamos uma análise extensiva por simulação dos protocolos avaliados.

Uma outra maneira de tornar um sistema robusto é a prevenção de falhas. Esta abordagem é pouco considerada em RSSF, pois os dispositivos são de baixo custo e possuem capacidades restritas, dificultando o uso de mecanismos para diagnóstico de um nó. Mecanismos indiretos de diagnóstico do nó, como a análise dos valores encontrados nos sensores, podem indicar futuras falhas [17].

3. Os Protocolos Avaliados

Avaliamos o comportamento de três protocolos de roteamento existentes na literatura, TinyOS Beaconing, EAD e PROC, considerando a ocorrência de falhas [11, 18, 19]. Escolhemos esses protocolos por serem desenvolvidos para redes de disseminação contínua de dados e apresentarem diferentes níveis de tolerância a falhas. O TinyOS Beaconing é um protocolo simples, sendo o protocolo padrão da plataforma Mica Motes [20]. O protocolo EAD é um protocolo de roteamento que tem como objetivo a economia de energia. O EAD permite observar como políticas de economia de energia se comportam frente a aspectos de tolerância a falhas. O terceiro protocolo, PROC, é um protocolo que possui mecanismos internos de tolerância a falhas, mostrando em seus resultados os benefícios de prover mecanismos de tolerância a falhas mais elaborados.

O protocolo TinyOS Beaconing recria periodicamente uma árvore de roteamento de menor caminho, com raiz no PA. Para criar essa árvore, o PA envia periodicamente uma mensagem *beacon* para a rede, determinando a distância em saltos dos nós até ele. A seleção das rotas também leva em conta a confiabilidade do enlace, calculada pelo número de pacotes corretamente enviados pelos nós vizinhos. Para reduzir o número de retransmissões, apenas nós com enlaces confiáveis são usados no roteamento. O TinyOS Beaconing utiliza o mecanismo de recriação

periódica das rotas como estratégia de tolerância a falhas.

O protocolo de roteamento EAD (*Energy-Aware Distributed routing*), proposto por Boukerche et al., cria uma árvore de roteamento com o objetivo de maximizar o número de nós folha [18]. Os nós folha não roteiam dados, e portanto podem manter o seu rádio desligado por períodos prolongados de tempo, economizando energia. O EAD atrasa a transmissão de dados em um intervalo proporcional à energia residual, que diminui significativamente a quantidade de colisões. Como ocorre no TinyOS Beaconing, o protocolo depende da reconstrução periódica de rotas para identificar rotas falhas. No EAD, entretanto, o fluxo de dados é concentrado em um número reduzido de nós, logo a ocorrência de falhas nestes nós é mais grave que falhas em nós folha.

O protocolo PROC (*Proactive ROuting with Coordination*) tem como meta diminuir o consumo de energia e maximizar o tempo de vida da rede [19]. Como o TinyOS Beaconing e o EAD, o PROC cria uma árvore de roteamento, chamada de *backbone*. A construção do *backbone* é regida pela aplicação, que define quais nós são os mais apropriados para rotear dados. O *backbone* é reconstruído periodicamente por um processo iniciado pelo PA. O protocolo possui um mecanismo de reconstrução local de rotas, que utiliza as mensagens de confirmação de recepção da camada de controle de acesso ao meio (MAC) para identificar nós falhos. Ao detectar que o próximo salto na rota não está confirmando a recepção das mensagens, o nó recalcula a sua rota. A falha de nós do *backbone* pode comprometer a aquisição de dados de uma região. Para solucionar este problema, o PROC monitora o próximo salto da sua rota para reconstruí-las dinamicamente.

3.1. Mecanismos de Tolerância a Falhas

Para melhor caracterizar o comportamento dos protocolos avaliados, é importante conhecer como cada protocolo lida com a ocorrência de falhas. Esta seção descreve os mecanismos de tolerância a falhas empregados nos protocolos avaliados.

A recriação periódica de rotas é um mecanismo de tolerância a falhas empregado pelos três protocolos. A cada nova execução do algoritmo de recriação de rotas dos protocolos EAD, PROC e TinyOS Beaconing, as rotas são completamente reconstruídas, utilizando somente os nós ativos. Este processo ocorre da seguinte forma: O PA envia uma mensagem indicando que as rotas devem ser recriadas. Cada nó repassa esta mensagem para os seus vizinhos em *broadcast*, permitindo que os nós identifiquem quais vizinhos estão ativos. Ao utilizarem como rotas os nós vizinhos que repassaram a última mensagem de recriação de rotas enviada pelo PA, os nós evitam o uso de vizinhos falhos.

O intervalo entre cada recriação de rotas deve ser ajustado de acordo com o grau de tolerância a falhas e o consumo de energia desejados. Por ser um processo que requer o envio de mensagens para toda a rede, a recriação de rotas envia um grande número de mensagens. Do ponto de vista de consumo de energia, este processo deve ocorrer com a menor frequência possível, mas do ponto de vista de falhas, quanto mais frequente for a atualização de rotas, mais rápido as falhas serão contornadas.

O PROC permite que o roteamento recupere rotas falhas mais rapidamente do que os outros protocolos, pois utiliza um mecanismo de monitoração da atividade dos sensores equivalente a mensagens de *ping-pong*. Para cada pacote de dados enviado pelo PROC (ping), o receptor deve retornar um quadro de confirmação (pong). Não são enviadas mensagens adicionais, pois o PROC utiliza os quadros de confirmação de recebimento (ACK) do protocolo MAC. Um contador registra quantos ACKs consecutivos foram perdidos. Quando o contador ultrapassa um certo limiar, o nó assume que seu pai está falho, e recalcula as suas rotas. Este mecanismo permite que o nó identifique rotas falhas antes da recriação periódica de rotas, aumentando a resiliência da rede. O uso de ACKs do protocolo de enlace dispensa o envio de uma mensagem adicional, economizando energia. Devido ao pequeno tamanho dos quadros de dados e às restrições de energia, em geral protocolos MAC para RSSF evitam o uso de quadros de controle. Entretanto, estudos de Polastre et al. mostraram que o aumento da latência e do consumo de energia são mínimos para o protocolo B-MAC quando os quadros de confirmação são ativados [21], justificando assim o seu uso para prover tolerância a falhas.

A falta de confirmação ocorre por dois motivos: A mensagem original não foi recebida pelo emissor, ou esta foi recebida mas sofreu erros na transmissão. Assim, é importante determinar

se um ACK não foi recebido porque o destinatário da mensagem está falho ou se houve um erro de transmissão. Utilizamos a probabilidade de que um quadro seja perdido (PER – *Packet Error Rate*)[1] para determinar o número mínimo de quadros consecutivos não recebidos que identificarão a falha de um nó. O número de quadros consecutivos perdidos que definem uma falha de nós (limiar) deve ser tal que a taxa de falsos positivos (erros de retransmissão considerados como falhas de nós) seja próxima de zero:

$$PER(t)^{limiar} = P, P \approx 0 \tag{1}$$

A escolha de um limiar alto faz com que uma grande quantidade de dados precise ser perdida até que a falha seja detectada, mas a certeza da falha aumenta, enquanto um limiar baixo aumenta a quantidade de falsos positivos, mas diminui o tempo necessário para a detecção de falhas de nós. A probabilidade de que ocorram n quadros consecutivos com erro diminui rapidamente quando aumentamos o número de quadros (n). Para n pequeno, conseguimos facilmente distinguir entre erros de transmissão e falhas de nós. Para efeito de comparação, o rádio CC1000, utilizado pelos nós Mica2 [20], possui taxa de erro por bit típica de 10^{-3} [22]. Na seção 6, onde avaliamos o comportamento de protocolos, utilizamos para o PROC um limiar igual a dois quadros, obtendo uma probabilidade de 0.001% de falsos positivos.

4. Falhas em RSSF

Esta seção identifica as principais causas de falhas de comunicação em RSSF. Neste estudo não consideramos falhas de comunicação em decorrência de resultados errados (falhas erráticas) produzidas pelos nós sensores. A corretude das mensagens de roteamento é assegurada por códigos de detecção de erro e mecanismos de verificação formal da especificação, garantindo que o roteamento não apresente falhas erráticas. Assumindo a corretude do protocolo e suas mensagens, podemos nos concentrar apenas em falhas silenciosas, ocasionadas pela não recepção de pacotes ou erros de hardware nos nós sensores. As seguintes falhas foram identificadas:

Fenômenos atmosféricos – Mudanças nas condições atmosféricas alteram a propagação do sinal, causando um aumento na taxa de erros da comunicação, proporcionado pela atenuação do sinal transmitido. Condições do ambiente como umidade, temperatura, entre outros, modificam a qualidade dos enlaces. Como as características do ambiente são dinâmicas, a qualidade da comunicação varia com o tempo.

Fontes móveis de interferência – Aparelhos que operam em faixas de frequência próximas às utilizadas em RSSF, ou mesmo veículos, animais e pessoas, podem gerar interferência na comunicação. Por operarem em frequências na faixa ISM (*Industrial, Scientifical and Medical*), que não requer licença para operação, RSSF estão expostas a interferência de outros aparelhos que operam nesta faixa de frequência. Para baratear o custo dos nós sensores, o rádio geralmente emprega um único canal, e possui modulação fixa. Estas limitações impedem o uso de mecanismos como seleção de canais com menor interferência, saltos de frequência ou troca dinâmica do mecanismo de modulação [23].

Desastres naturais – Nós sensores podem ser depositados ao ar livre ou em regiões de desastre, estando assim expostos a deslizamentos, terremotos e enchentes. Desastres naturais podem ocasionar a destruição dos nós ou a inutilização de componentes do hardware dos nós sensores. Ao contrário das falhas decorrentes das condições atmosféricas, nós falhos devido a desastres naturais permanecem inoperantes.

Quebra acidental – Nós sensores podem ser destruídos acidentalmente, por exemplo devido ao pisoteamento por animais ou à queda de árvores sobre nós sensores. Em geral, os nós sensores estarão espalhados a alguns metros de distância uns dos outros, assim apenas um nó falha por vez.

Bloqueio do processador – Nos sistemas embutidos são utilizados escalonadores de multitarefa cooperativa, ou sistemas de *run to completion*, assim um software defeituoso pode bloquear o processador por tempo indeterminado. Para evitar tais situações, são utilizados temporizadores

[1] $PER(t) = 1 - (1 - BER)^t$, onde t é o tamanho do quadro enviado e BER é a taxa de erro por bit.

chamados de *watchdogs*[2]. Desta forma, o nó estará bloqueado por um tempo finito, e em seguida retornará à operação normal.

Falhas maliciosas – RSSF estão expostas a falhas maliciosas, como decorrentes de ataques de segurança. Nestas falhas, um nó malicioso ou uma entidade externa provocam erros na rede. Este trabalho não aborda técnicas de prevenção a ataques. Entretanto, é possível utilizar mecanismos simples de tolerância a falhas para identificar regiões ou nós sobre ataque e evitá-las [24]. Neste trabalho abordamos apenas mecanismos para contornar alguns ataques de negação de serviço (*ataques de interferência*, de *colisão* e de *sinkhole*). Estes ataques se comportam como falhas silenciosas, que são o escopo do nosso trabalho.

Esgotamento da bateria – O esgotamento da bateria dos nós sensores pode gerar uma falha de comunicação. O emprego de nós em áreas de difícil acesso ou o número de nós empregados pode tornar inviável o recarregamento de baterias. Por possuírem hardware limitado, os nós sensores atuais não permitem a aferição confiável do nível corrente de energia, desta forma o nó sensor não tem como notificar aos nós vizinhos a iminência da sua falha.

Falhas por esgotamento da bateria não são consideradas devido à dificuldade de se modelar tais falhas. O esgotamento da energia em geral ocorrerá simultaneamente para uma grande quantidade de nós, uma vez que os protocolos tendem a balancear o consumo entre os nós para maximizar o tempo de vida da rede [8]. Nós ativos assumirão as tarefas dos nós indisponíveis, aumentando a sua carga e consequentemente o consumo de energia. Com isso, a rede irá rapidamente deteriorar, em um efeito em cascata. Para se manter conectada, a rede utilizará mecanismos de auto-configuração para ajustar o alcance do rádio dos nós restantes para que estes possam alcançar outros nós ainda ativos.

5. Agrupamento das Falhas

Esta seção apresenta um agrupamento das falhas descritas na seção 4 de acordo com as suas características. Esse agrupamento tem como objetivo facilitar o estudo de falhas em RSSF, sendo resumido na Tabela 1. As falhas são caracterizadas quanto a sua persistência e extensão:

Persistência – Indica se o nó retornará a operar corretamente após um período de tempo falho (*falhas transientes*), ou se a falha é *permanente* [5]. Do ponto de vista do roteamento, falhas transientes são aquelas em que a falha pode ser contabilizada em minutos, enquanto falhas permanentes podem ser contabilizadas em horas. Consideramos que situações de falha devido às mudanças no clima, por exemplo, serão caracterizadas como permanentes.

Extensão – Indica o número de nós afetados. As falhas podem ser *isoladas*, no caso da falha de um único nó, ou *agrupadas*, onde um conjunto de nós falha. O último aumenta a gravidade da falha, uma vez que grande parte da vizinhança de um nó se tornará falha, diminuindo o número de vizinhos que poderão rotear dados. A Figura 1 exemplifica esta classificação (setas representam as rotas dos nós).

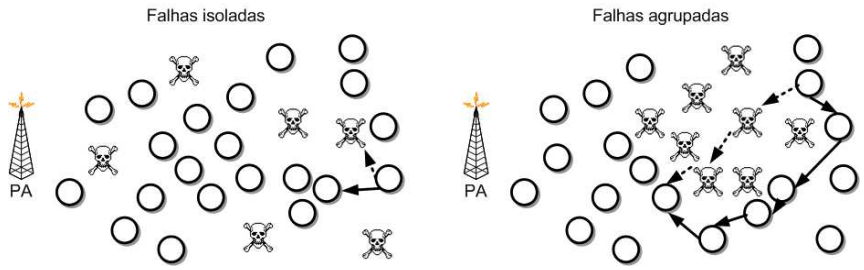

Figura 1: Exemplo de falhas de nós, classificadas quanto a extensão.

As falhas maliciosas decorrentes de ataques de colisão e de *sinkhole* variam de duração de acordo com a intenção do atacante, podendo ser breves para evitar detecção, ou prolongadas, aumentando o estrago causado. Assim, classificamos essas falhas tanto como transientes quanto

[2]O *watchdog* é um temporizador a ser reiniciado periodicamente pela aplicação, ou o processador é reiniciado.

como permanentes. Ataques de interferência, por outro lado, serão permanentes, pois estes são mais efetivos se empregados por um período prolongado, mesmo ocorrendo a detecção do ataque.

Causa da falha	**Persistência**	**Extensão**
Fenômenos atmosféricos	permanente	agrupadas
Fontes móveis de interferência	transiente	isolada
Desastres naturais	permanente	agrupadas
Quebra acidental	permanente	isolada
Bloqueio do processador	transiente	isolada
Ataques de interferência	permanente	agrupadas
Ataques de colisão	ambos	isolada
Ataques de *sinkhole*	ambos	isolada

Tabela 1: Caracterização das falhas de acordo com a sua causa.

6. Avaliação por Simulação

Avaliamos o desempenho dos três protocolos utilizando o simulador NS-2 [25]. Simulamos uma rede homogênea, composta por nós sensores com configuração próxima aos nós sensores da família Mica 2, rodando o sistema operacional TinyOS [20]. Escolhemos esta plataforma devido à sua grande aceitação na comunidade científica e ao elevado número de redes em operação que a utilizam. Simulamos uma aplicação com características de tráfego similares à rede empregada na ilha de Great Duck para estudos do ecosistema e comportamento de aves [17]. Nesta rede, cada nó sensor envia mensagens de dados de 36 bytes a cada 70s. Estas mensagens são enviadas para o PA, que disponibiliza os dados para análise.

O protocolo de acesso ao meio empregado é uma versão modificada da implementação do IEEE 802.11, que emula o comportamento do protocolo B-MAC [21]. Limitamos a banda em 12kbps, como ocorre no TinyOS, e ajustamos os parâmetros do rádio de acordo com as especificações do rádio CC1000 [22], utilizado nos nós Mica2. Definimos o intervalo de recriação dos protocolos EAD e TREE (uma versão do TinyOS Beaconing sem o cálculo de confiabilidade do canal) em 120s, enquanto no protocolo PROC utilizamos 180s. Estes valores, determinados empiricamente, foram utilizados por apresentarem os melhores valores para cada protocolo avaliado [19].

A rede considerada é formada por 150 nós inseridos aleatoriamente em uma área quadrada, com 70m de lado. O PA se encontra no canto da rede em todos os cenários de simulação, maximizando o número de saltos. A rede funciona sem falhas durante 1500s, para que os nós cheguem ao seu estado estacionário (verificado empiricamente). Neste momento ocorre uma falha, e a simulação continua por mais 1500s, permitindo que o protocolo de roteamento se recupere da falha. Nos cenários em que ocorrem falhas isoladas, os nós que irão falhar são escolhidos aleatoriamente. Para falhas agrupadas, é definido um ponto central, e todos os nós que estão a uma distância máxima r_{max} do ponto irão falhar.

As métricas avaliadas são: *taxa de entrega média fim a fim* (porcentagem do total de pacotes recebidos corretamente pelo PA); *latência média*; *distância média em saltos até o PA*; *vazão*; e *consumo médio de energia*. O consumo médio é calculado somente para os nós que estão ativos ao final da simulação. A análise dos resultados enfatiza a energia média consumida, taxa de entrega média e vazão média, pois estas são as métricas mais importantes quando consideramos tolerância a falhas em RSSF. Todos os resultados são obtidos pela média de 33 simulações, e apresentados com intervalo de confiança de 95%.

6.1. Falhas Transientes e Isoladas

As falhas transientes foram avaliadas sobre três dimensões: intervalo de recriação de rotas, tempo da falha e número de nós falhos. A frequência da atualização de rotas influirá no grau de tolerância a falhas do protocolo, pois define o tempo de reação a falhas em protocolos como o EAD e TREE, que dependem da reconstrução de rotas para normalizar o funcionamento da rede. Neste cenário 20 nós falharam durante 120s, e o tempo de ciclo variou de 120 a 300s. Como era

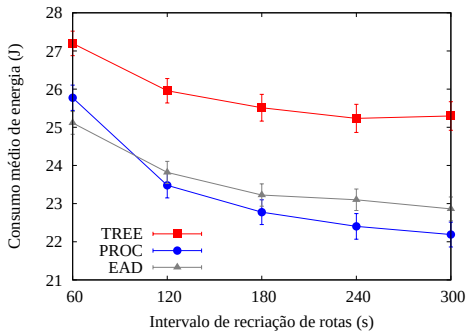

Figura 2: Energia consumida variando o intervalo de recriação de rotas.

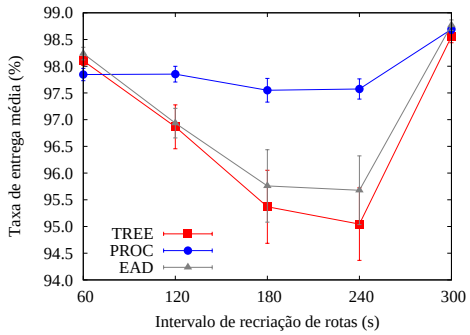

Figura 3: Taxa de entrega média variando o intervalo de recriação de rotas.

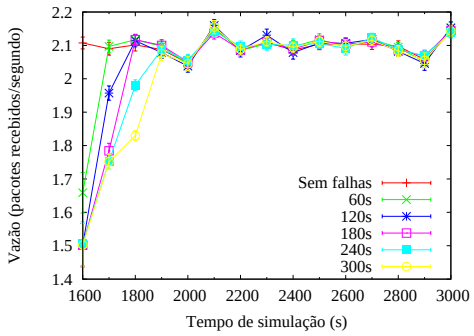

Figura 4: Vazão do PROC variando o tempo de falha.

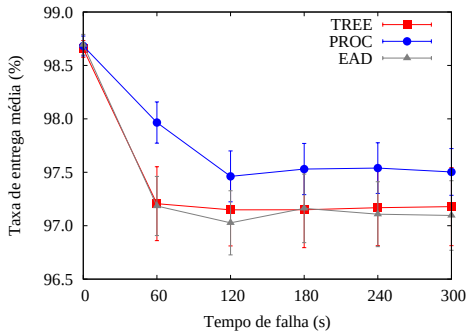

Figura 5: Taxa de entrega variando o tempo de falha.

esperado, os protocolos tendem a consumir mais energia com atualizações de rotas mais frequentes (Figura 2). A taxa de entrega, mostrada na Figura 3, diminui ao aumentarmos o intervalo de recriação de rotas. No PROC a redução é menor, devido ao uso de ACKs para identificar falhas, pois as falhas são corrigidas mais rapidamente. Verificamos um aumento na taxa de entrega para intervalos de 300s, que se deve a uma diminuição na carga da rede, diminuindo o número de pacotes perdidos devido a falta de espaço na fila. Quanto à latência média, todos os protocolos mostraram uma diminuição nesta métrica com o aumento do tempo de recriação de rotas, que ocorre principalmente devido à menor carga imposta à rede.

Em seguida avaliamos como o tempo de falha afeta o desempenho dos protocolos. Para todos os protocolos, verificamos que existe uma queda na vazão durante o tempo de falha, que é recuperada quase completamente após um intervalo de recriação de rotas (Figura 4). Isto ocorre porque os protocolos utilizam como mecanismo de tolerância a falhas a recriação completa das rotas em intervalos regulares de tempo. Novamente, o PROC apresentou uma taxa de entrega média levemente superior (em torno de 0.5%), devido ao uso de ACKs para detectar falhas, como mostrado na Figura 5. O intervalo de recriação de rotas mostrou-se adequado, uma vez que os protocolos EAD e TREE, mesmo sem possuírem mecanismos ativos de detecção de falha como o PROC, obtiveram bons resultados. Como anteriormente, a quantidade de energia consumida por nó diminuiu, uma vez que estes tiveram que rotear menos dados. Dentre os protocolos avaliados, o PROC consumiu menos energia, consumindo em torno de 22 a 23J, enquanto o EAD e TREE consumiram em torno de 4% a 14% mais energia que o PROC, respectivamente.

Finalmente, variamos o número de nós falhos. Simulamos falhas de 25 a 100 nós, com incrementos de 25 nós. Identificamos que todos os protocolos possuem um comportamento similar quando variamos o número de nós falhos. Os protocolos apresentam um declínio de vazão, que é restaurada em até 200s após a falha. A queda na vazão é determinada pelo número de nós falhos, exemplificada pelo comportamento do PROC, mostrado na Figura 6. O mecanismo de identificação de falhas empregado pelo PROC permitiu que este protocolo recuperasse mais rapidamente da falha, desta forma aumentando a sua taxa de entrega em relação aos outros protocolos em torno de 0.5%. Como a falha tem um intervalo de tempo pequeno em relação ao tempo total de simulação, o ganho por uma recuperação mais rápida da falha é amenizado no resultado final.

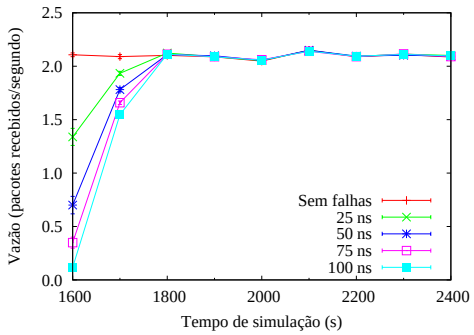

Figura 6: Vazão do PROC variando o número de nós falhos.

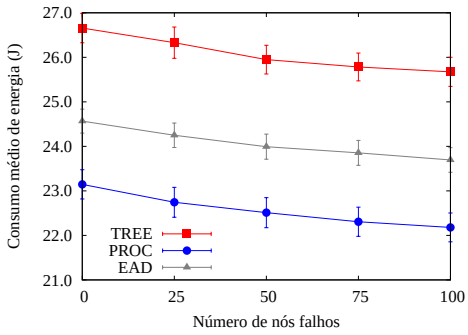

Figura 7: Consumo de energia variando o número de nós falhos.

A latência e o número de saltos médios não sofrem alterações com a falha de nós, entretanto a energia consumida decresce. Isto ocorre devido à diminuição do número de pacotes enviados ao PA, correspondente ao tráfego originado dos nós falhos. A Figura 7 mostra o gráfico do consumo médio de energia dos protocolos. A pequena variação do consumo médio e da taxa de entrega média mostram que o número de nós falhos não influi de forma significativa na rede. Como as falhas são distribuídas pela rede, novas rotas são facilmente encontradas.

6.2. Falhas Permanentes e Isoladas

Neste cenário avaliamos o comportamento dos protocolos na ocorrência de falhas permanentes e isoladas. Realizamos simulações variando o número de nós falhos em 20, 40 e 60 nós. Verificamos que os protocolos se recuperam rapidamente, entretanto a vazão cai após a ocorrência da falha, pois os nós falhos deixam de enviar dados. A desativação dos nós fez com que o número de saltos até o PA aumentasse levemente, em torno de 0.1 saltos para cada 20 nós falhos. A latência média se manteve quase inalterada, mostrando que a diminuição do tráfego de dados compensou o aumento do número de saltos médios. Quanto à taxa de entrega média, verificamos uma queda com o aumento do número de nós falhos, mostrado na Figura 8.

Em comparação com as falhas transientes isoladas, verificamos que as falhas permanentes são mais prejudiciais à rede do que as falhas transientes, pois acarretam uma queda mais significativa no consumo médio de energia e na taxa de entrega média.

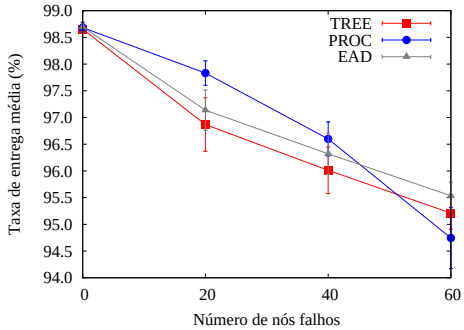

Figura 8: Taxa de entrega média para falhas permanentes isoladas.

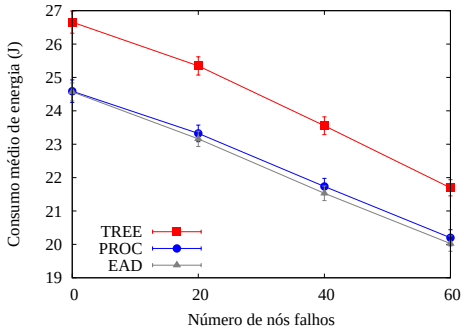

Figura 9: Consumo de energia para falhas permanentes isoladas.

6.3. Falhas Permanentes e Agrupadas

Neste cenário avaliamos o efeito de falhas permanentes agrupadas. O número de nós falhos depende do raio de falha, que variou de 5 a 40m. Os resultados obtidos mostram que este tipo de falha é o mais severo dentre os avaliados, assim realizamos uma análise mais completa.

A taxa de entrega média cai até 9% com o aumento do raio de falha, como mostra a Figura 10. Como nos cenários anteriores, os protocolos se recuperam da falha após a recriação completa das rotas. Neste cenário, entretanto, verificamos uma grande variação da taxa de entrega média, causada por cenários onde ocorrem partições na rede. É sabido que nós próximos ao PA repassam mais dados que os nós distantes, assim falhas nestes nós irão degradar significativamente o serviço

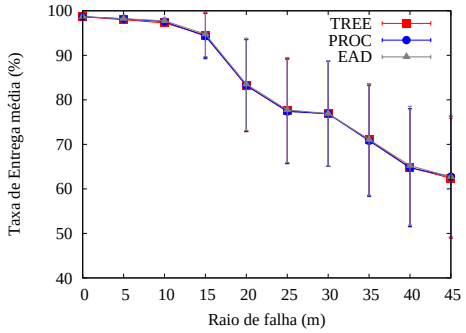

Figura 10: Taxa de entrega para falhas permanentes agrupadas.

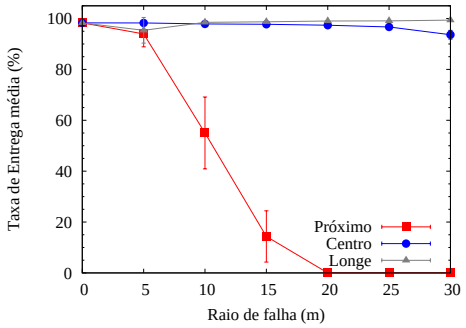

Figura 11: Taxa de entrega para falhas em pontos distintos da rede.

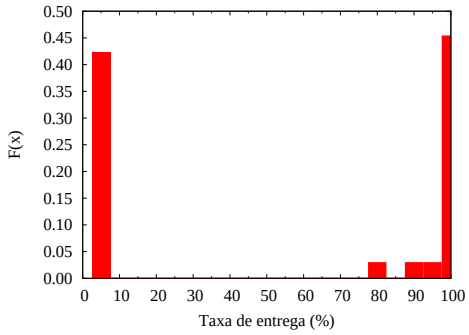

Figura 12: Histograma da taxa de entrega em falhas perto do PA.

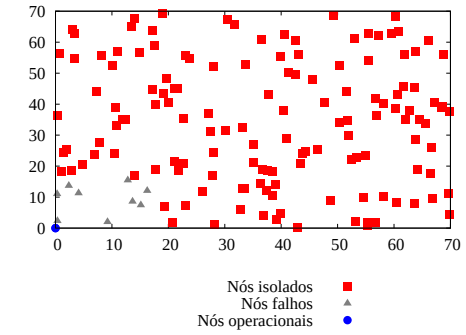

Figura 13: Exemplo de rede particionada.

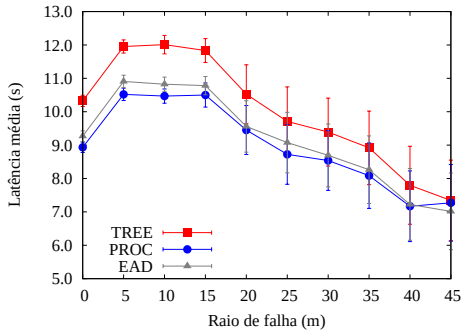

Figura 14: Latência média para falhas permanentes agrupadas.

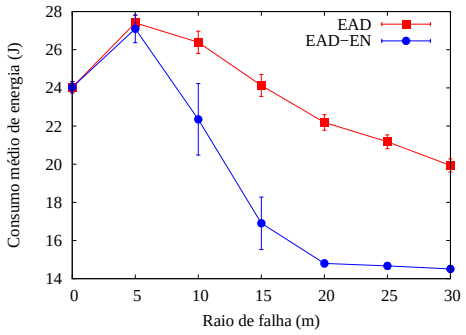

Figura 15: Consumo de energia com mecanismos de desligamento do rádio.

da rede. A Figura 11 exemplifica a taxa de entrega para falhas em pontos distintos da rede. A curva “Próximo” apresenta a taxa de entrega média para falhas que possuem o seu centro em até 17m do PA. A curva “Centro” mostra falhas na região central da rede, enquanto a curva “Longe” apresenta falhas em até 17m do canto oposto ao PA. Os resultados apontam que falhas de nós em pontos não muito próximos do PA pouco afetam a taxa de entrega. Entretanto, falhas próximas ao PA podem degradar substancialmente o desempenho da rede.

Encontramos um intervalo de confiança de até 10% para a taxa de entrega na curva “Próximo” da Figura 11, que verificamos ser ocasionado por partições na rede. Uma análise do histograma da taxa de entrega para a curva “Próximo” (Figura 12), em grupos de 5%, mostra que os resultados estão aglomerados próximo de 5% e de 95%, o que explica o alto intervalo de confiança, e identifica o papel crucial da partição da rede no desempenho dos protocolos neste cenário. As partições ocorrem em cenários como o mostrado na Figura 13, onde a ocorrência de falhas isolou todos os nós operacionais da rede, impedindo a comunicação com o PA. A figura classifica os nós em “Nós falhos” (nós que sofreram falhas), “Nós operacionais” (nós que estabelecem rotas para o PA) e “Nós isolados” (nós que não conseguem estabelecer rotas para o PA). Uma forma para recuperar partições na rede, que necessita de suporte no MAC, é o aumento da potência de transmissão. Com a negociação de uma nova potência de transmissão entre os nós

próximos à região de falhas, a comunicação poderia ser reestabelecida. O roteamento também pode contribuir para amenizar a severidade de uma partição da rede detectando a ocorrência de falhas e adotando medidas de economia de energia.

Verificamos que a latência média (Figura 14) e a distância média em saltos até o PA aumentam quando o raio da falha é pequeno, pois é necessário que as rotas evitem a região falha, para tanto aumentando o caminho percorrido. Para falhas mais extensas, devido a partições na rede, verificamos uma diminuição em ambas as métricas. Como apenas os nós próximos do PA conseguem enviar dados, o número de saltos médios e a latência média diminuem.

A Figura 15 mostra mecanismos de economia de energia que podem ser utilizados na ocorrência de uma partição na rede para diminuir o consumo dos nós. Comparamos o EAD ao EAD-EN, uma versão modificada do EAD, que recusa o envio de mensagens da aplicação caso o nó não tenha recebido mensagens de recriação de rotas em um intervalo de tempo equivalente a dois períodos entre a recriação de rotas (curva "EAD-EN" na figura). Como visto na figura, o uso de técnicas de economia de energia permite uma diminuição substancial no consumo (de 16% a 33% apenas evitando o envio de mensagens), não interferindo em nenhuma outra métrica da rede. Verificamos resultados semelhantes para os outros protocolos avaliados, justificando a implementação destas técnicas em protocolos de roteamento.

7. Conclusões e Trabalhos Futuros

Redes de sensores sem fio são aplicadas em ambientes inóspitos, estando sujeitas a intempéries e variações climáticas, que acarretam em um aumento significativo na frequência e severidade das falhas. Os nós sensores devem se adaptar às condições do ambiente para continuar a prover um serviço dentro dos requisitos de qualidade esperados. Para tanto, é necessário que a rede mantenha rotas dos nós sensores ao ponto de acesso mesmo na presença de falhas e ataques de segurança. Neste artigo caracterizamos os principais agentes causadores de falhas silenciosas nestas redes, e a partir desta caracterização avaliamos o comportamento de protocolos de roteamento na ocorrência de falhas.

Verificamos que os protocolos avaliados apresentam mecanismos de auto-estabilização, como a recriação periódica de todas as rotas, que permitem recuperação em situações de falha. Falhas transientes isoladas e permanentes isoladas são facilmente tratadas com mecanismos de recriação de rotas, e apresentam pouco impacto no funcionamento da rede. Falhas permanentes e agrupadas, entretanto, podem causar grandes perdas de dados, pois podem gerar partições na rede. Mecanismos de tolerância a falhas que tratam falhas permanentes agrupadas devem ser mais agressivos perto do PA, pois falhas nesta região podem comprometer o funcionamento de toda a rede. Uma maneira de abrandar o custo destas falhas é o desligamento temporário de nós que não se comunicam com o PA, o que permite economia significativa de energia em situações onde ocorre uma falha prolongada.

Como trabalhos futuros, iremos identificar como os parâmetros de QoS se comportam na presença de falhas, verificando se os protocolos atuais podem ser considerados confiáveis quando aplicados em redes com altas taxas de falhas.

Referências

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A Survey on Sensor Networks. *IEEE Communications*, 40(8):102–114, 2002.

[2] Benjamin Lussier, Raja Chatila, Felix Ingrand, Marc-Olivier Killijian, and David Powell. On fault tolerance and robustness in autonomous systems. In *3rd IARP-IEEE/RAS-EURON Joint Workshop on Technical Challenges for Dependable Robots in Human Environments*, 2004.

[3] The British Computer Society. Grand Challenges in Computing. http://www.nesc.ac.uk/esi/events/Grand_Challenges/, 2004.

[4] Linnyer Beatrys Ruiz, Antonio A. F. Loureiro, and Jose Marcos Nogueira. Functional and information models for the MANNA architecture. In *GRES03 - Colloque Francophone sur la Gestion de Reseaux et de Services*, pages 455–470, February 2003.

[5] Algirdas Avizienis, Jean-Claude Laprie, Brian Randell, and Carl Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Trans. Dependable Secur. Comput.*, 1(1):11–33, 2004.

[6] Matthias Hollick, Ivan Martinovic, Tronje Krop, and Ivica Rimac. A Survey on Dependable Routing in Sensor Networks, Ad hoc Networks, and Cellular Networks. In *Proceedings of the 30th IEEE EUROMICRO Conference 2004*, pages 495–502, Rennes, France, September 2004.

[7] Farinaz Koushanfar, Miodrag Potkonjak, and Alberto Sangiovanni-Vincentelli. Fault tolerance in wireless sensor networks. In *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*. CRC Press, 2004.

[8] Wendi Rabiner Heinzelman and Anantha Chandrakasan and Hari Balakrishnan. Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In *Proceedings of the 33rd Hawaii International Conference on System Sciences*, 2000.

[9] K. Sohrabi and J. Gao and V. Ailawadhi and G. Pottie. Protocols for Self-Organization of a Wireless Sensor Network. *IEEE Personal Communications*, 7(5):16–27, 2000.

[10] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin, John Heidemann, and Fabio Silva. Directed diffusion for wireless sensor networking. *ACM/IEEE Transactions on Networking*, 11(1):2–16, February 2002.

[11] Chris Karlof, Yaping Li, and Joseph Polastre. ARRIVE: Algorithm for robust routing in volatile environments. Technical Report UCB//CSD-03-1233, University of California, Berkeley, CA, March 2003.

[12] Deepak Ganesan and Ramesh Govindan and Scott Shenker and Deborah Estrin. Highly-Resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 5(4):11–25, 2001.

[13] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris. A high-throughput path metric for multi-hop wireless routing. In *Proceedings of the 9th ACM International Conference on Mobile Computing and Networking (MobiCom '03)*, San Diego, California, September 2003.

[14] Alec Woo, Terence Tong, and David Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *Proceedings of the first international conference on Embedded networked sensor systems*, pages 14–27. ACM Press, 2003.

[15] Marcos Augusto M. Vieira, Luis Filipe M. Vieira, Linnyer Beatrys Ruiz, Antonio Alfredo F. Loureiro, Antônio O. Fernandes, José Marcos S. Nogueira, and Diógenes Cecílio da Silva Jr. Como Obter o Mapa de Energia em Redes de Sensores Sem Fio? Uma Abordagem Tolerante a Falhas. In *Anais do 5o. Workshop de Comunicação sem Fio (WCSF)*, pages 183–189, 2003.

[16] Gunjan Khanna, Saurabh Bagchi, and Yu-Sung Wu. Fault tolerant energy aware data dissemination protocol in sensor networks. In *IEEE Dependable Systems and Networks Conference*, June 2004.

[17] R. Szewczyk, J. Polastre, A. Mainwaring, and D. Culler. Lessons from a sensor network expedition. In *Proceedings of the First European Workshop on Sensor Networks (EWSN)*, pages 307–322, Jan 2004.

[18] Azzedine Boukerche, Xiuzhen Cheng, and Joseph Linus. Energy-aware data-centric routing in microsensor networks. In *Proceedings of the 6th international workshop on Modeling analysis and simulation of wireless and mobile systems*, pages 42–49. ACM Press, 2003.

[19] Daniel F. Macedo, Luiz H. A. Correia, Aldri L. dos Santos, Antonio A. Loureiro, and José M. Nogueira. A pro-active routing protocol for continuous data dissemination wireless sensor networks. In *10th IEEE Symposium on Computer and Communications (ISCC)*, Jun 2005.

[20] Philip Levis, Sam Madden, Joseph Polastre, Robert Szewczyk, Kamin Whitehouse, Alec Woo, David Gay, Jason Hill, Matt Welsh, Eric Brewer, and David Culler. TinyOS: An operating system for wireless sensor networks. In W. Weber, J. Rabaey, and E. Aarts, editors, *Ambient Intelligence*. Springer-Verlag, New York, NY, 2004.

[21] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 95–107. ACM Press, 2004.

[22] CC1000. Chipcom corporation. CC1000 low power FSK transceiver. http://www.chipcom.com, 2004.

[23] Bernhard Walke, Norbert Esseling, Jörg Habetha, Andreas Hettich, Arndt Kadelka, Stefan Mangold, Jörg Peetz, and Ulrich Vornefeld. IP over Wireless Mobile ATM - Guaranteed Wireless QoS by HiperLAN/2. *Proceedings of the IEEE*, 89:21–40, Jan 2001.

[24] Anthony D. Wood and John A. Stankovic. Denial of service in sensor networks. *Computer*, 35(10):54–62, 2002.

[25] NS-2 simulator. http://www.isi.edu/nsnam/ns/, January, 2004.