

## Projeto Integrado para Comunicação de Dados em Redes de Sensores sem Fio

Felipe Domingos da Cunha<sup>2</sup>, Michelle Nery Nascimento<sup>1</sup>, Raquel A. F. Mini<sup>1\*</sup>,  
Antonio A. F. Loureiro<sup>2</sup>

<sup>1</sup>Departamento de Ciência da Computação – Pontifícia Universidade Católica de Minas Gerais  
Belo Horizonte – MG – 31980-110 – Brasil

<sup>2</sup>Departamento de Ciência da Computação – Universidade Federal de Minas Gerais  
Belo Horizonte – MG – 31270-901 – Brasil

micnery@gmail.com, {fdcunha, loureiro}@dcc.ufmg.br, raquelmini@pucminas.br

**Abstract.** *The design of protocols for wireless sensor networks is often based on a layered architecture in which the exchange of information and interactions among layers are restricted. Also, the communication between two layers not adjacent in the protocol stack is not allowed. On the other hand, the cross-layer design approach makes possible a strong interaction between two or more layers of the protocol stack and information sharing during the protocol runtime. In order to improve the quality of the data routing on network, we evaluated the cross-layer techniques in two situations: broadcast and data collection. Simulations results showed that it is possible achieve a good improvement in routing with the use of cross-layer design.*

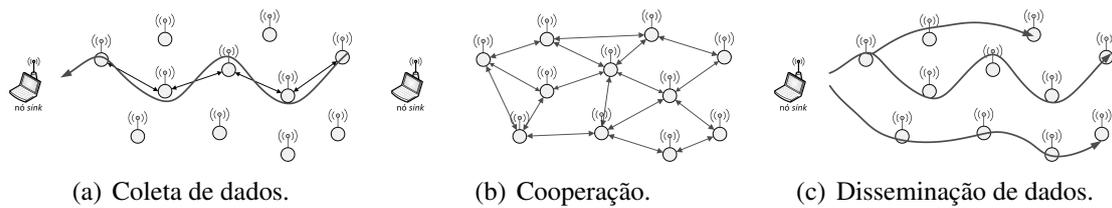
**Resumo.** *O projeto de protocolos para redes de sensores sem fio é muitas vezes baseado em uma arquitetura de camadas na qual a troca de informações e as interações entre as camadas são restritas. Além disso, a comunicação entre duas camadas não adjacentes na pilha de protocolos não é permitida. Por outro lado, uma abordagem integrada de camadas torna possível uma forte interação entre duas ou mais camadas na pilha de protocolos e possibilita também o compartilhamento de informações em tempo de execução. Com o intuito de melhorar a qualidade do roteamento de dados na rede, a técnica de projeto integrado é avaliada neste trabalho em duas situações: disseminação e coleta de dados. Resultados de simulações mostraram que é possível alcançar uma boa melhoria no roteamento com a utilização de um projeto integrado de camadas.*

### 1. Introdução

Rede de Sensores Sem Fio (RSSF) é um tipo especial de redes *ad-hoc*, cujos nós não precisam de uma infraestrutura para se comunicarem [Akyildiz et al. 2002]. Essa rede é composta por milhares de nós sensores autônomos, depositados aleatoriamente em uma determinada área, para coletar dados do ambiente e transmiti-los ao usuário final. Esses nós sensores são equipados com um processador, memória, rádio, sensor e bateria. Entretanto, para diminuição de custos e facilitar a deposição, esses nós possuem tamanho

---

\*Parcialmente financiada pelo Projeto Universal do CNPq número 478175/2008-6 referente ao Edital MCT/CNPq 14/2008.



**Figura 1. Tipos de comunicação de dados em RSSFs [Machado et al. 2005].**

reduzido, o que reduz também a capacidade de seus componentes como: limitações no processamento, baixa capacidade de armazenamento e baterias com duração limitada. Devido a essas características, as RSSFs apresentam vários desafios, sendo a restrição de energia o maior deles [Mini and Loureiro 2009].

Um dos principais objetivos nas RSSFs é transportar as informações sensorizadas pelos nós sensores até o nó de monitoramento (*nó sink*). Este por sua vez tem a responsabilidade de interligar a rede de sensores ao usuário final. Assim, a comunicação de dados torna-se uma tarefa de grande importância nesse tipo de rede. Conforme ilustrado na figura 1 [Machado et al. 2005], a comunicação pode ser dividida em três formas. A primeira denominada Coleta de Dados, ocorre dos sensores para o nó *sink* (figura 1-(a)). Nessa forma de comunicação os nós sensores enviam os dados sensorizados para o nó *sink*. A segunda forma denominada Cooperação (figura 1-(b)), ocorre entre os nós sensores e tem como objetivo a cooperação entre eles. E a terceira forma denominada Disseminação de Dados (figura 1-(c)), ocorre do nó *sink* para os demais nós da rede e tem como objetivo propiciar o envio de informações e requisições para a rede.

A comunicação é uma das fontes de maior consumo de energia em uma RSSF. Segundo [Pottie and Kaiser 2000], a energia dissipada para a realização de uma transmissão está diretamente relacionada à distância em que a mensagem será enviada. Esse cenário sugere a necessidade de projetar protocolos cientes do consumo de energia e que se proponham a reduzir o mesmo, com o objetivo final de aumentar o tempo de vida da rede [Cui et al. 2007]. Essa redução pode ser alcançada com a implementação de técnicas que reduzem as transmissões desnecessárias e que diminuem a distância das transmissões com a utilização de uma comunicação com vários saltos.

Outra maneira eficiente de diminuir o consumo de energia dos nós é reduzir o tempo em que o nó sensor permanece com seu rádio ligado, escutando a rede. Considerando essa premissa, alguns protocolos de controle de acesso ao meio desenvolvidos para esse tipo de rede dividem o modo de operação dos nós sensores em periódicos intervalos de atividade e dormência. Nos períodos de atividade, o nó permanece com seu rádio ligado, escutando ou transmitindo dados na rede. Nos períodos de dormência, o nó permanece com seu rádio desligado com a finalidade de economizar energia. A razão entre o tamanho do período de escuta e o tamanho do ciclo de funcionamento do nó é chamada *duty cycle* [Ye et al. 2002]. Estudos recentes mostram que para uma RSSF alcançar um longo tempo de vida, o *duty cycle* dos nós sensores deve ser muito baixo, algo em torno de 1% [Gu and He 2007, Guo et al. 2009].

Diferentemente das redes tradicionais que são projetadas para dar suporte a uma grande variedade de aplicações, as RSSFs normalmente são projetadas considerando uma aplicação específica [Karl and Willig 2005]. Contudo, os protocolos existentes na lit-

eratura não consideram as especificidades encontradas nesse tipo de rede. Assim, novos protocolos devem ser projetados considerando essas peculiaridades, como também os requisitos da aplicação a ser monitorada [Puccinelli and Haenggi 2005]. Entretanto, para que o projeto de um novo protocolo considere essas características e requisitos, é necessário que os demais protocolos na rede trabalhem de forma cooperativa, trocando informações entre si, com o objetivo de maximizar o desempenho da rede. Para isso, surge o Projeto Integrado de Camadas, um paradigma para a concepção de protocolos que exploram a rica interação entre os protocolos das camadas adjacentes ou não [Kawadia and Kumar 2005].

As camadas responsáveis pela comunicação em RSSFs são as camadas de rede e MAC (*medium access control*). Por isso, neste trabalho são apresentados dois estudos de casos que avaliam a viabilidade de um projeto integrado entre essas camadas, com a finalidade de melhorar a comunicação de dados entre os nós sensores. Ambos os estudos de caso propõem adaptações ao tradicional protocolo S-MAC [Ye et al. 2002]. Na camada de rede, os protocolos utilizados são escolhidos de acordo com a forma de comunicação a ser trabalhada: Disseminação ou Coleta de Dados. O primeiro estudo de caso aplica o conceito de projeto integrado para melhorar a comunicação na disseminação de dados. Nesse estudo, é proposto o protocolo SMAC-RC (*Sensor MAC for Routing on a Curve*) para a camada MAC que trabalha em cooperação com a camada de rede. O objetivo desse protocolo é reduzir a latência e aumentar a confiabilidade na disseminação de dados. O segundo estudo aplica o projeto integrado objetivando melhorias na coleta de dados. Assim, é proposto o protocolo SMAC-DDC (*Sensor MAC with Dynamic Duty Cycle*) para a camada MAC, que trabalha conjuntamente com a camada de rede. A principal característica desse protocolo é ajustar o *duty cycle* dos nós sensores em função do tráfego na rede. Em ambos os estudos de caso, foi possível alcançar melhorias significativas de desempenho na rede com a utilização do projeto integrado de camadas.

O presente trabalho encontra-se dividido da seguinte forma: a seção 2 apresenta alguns trabalhos que utilizam o paradigma de projeto integrado de camadas no desenvolvimento de melhorias para protocolos de RSSFs. A seção 3 apresenta o funcionamento do protocolo SMAC-RC e como a comunicação entre camadas foi realizada. A seção 4 apresenta o funcionamento do protocolo SMAC-DDC e como a adaptação do *duty cycle* foi realizada. A seção 5 apresenta os resultados das simulações realizadas em cenários com e sem os projetos integrados propostos neste trabalho. A seção 6 apresenta as conclusões e direções futuras para este trabalho.

## 2. Trabalhos Relacionados

A comunicação baseada na arquitetura de camadas tipicamente separa as tarefas por camadas, com a clara definição das funcionalidades e serviços que cada uma delas deverá prover. A interação entre elas na pilha de protocolos ocorre somente através de interfaces pré-definidas que as conectam somente quando são vizinhas. Por ser uma interação restrita, o fluxo entre camadas que não são adjacentes não é permitido. Em contrapartida, outro tipo de abordagem para essa comunicação é o conceito de projeto integrado de camadas. Essa abordagem atenta para uma rica exploração da interação entre as camadas, adjacentes ou não, com o objetivo de aumentar o desempenho geral da rede [Jurdak 2007, Srivastava and Motani 2005, Shakkottai et al. 2003, Haapola et al. 2005]. Nesse contexto, o projeto de novos protocolos que fazem uso dessa técnica objetivando melho-

rar o desempenho e propiciar o gerenciamento de alguns recursos em mais de uma camada ao mesmo tempo. A principal característica de um projeto integrado é, por intermédio da violação da arquitetura de camadas, alcançar melhorias no desempenho dos protocolos e por consequência melhorias no funcionamento da rede.

O uso do paradigma de projeto integrado de camadas em RSSFs tem se tornado cada vez mais comum, principalmente pelo fato dessas redes serem de propósito bem específico. Outro fator que acirra o uso do projeto integrado em RSSFs é a possibilidade de melhorar o gerenciamento do consumo de energia, visto que, energia é um recurso altamente escasso e necessita ser poupado. Nos seguintes trabalhos [Van Hoesel et al. 2004, Sichitiu 2004, Souto et al. 2007, Haapola et al. 2005] são apresentadas soluções eficientes no que tange a aplicação do paradigma projeto integrado de camadas para alcançar melhorias no projeto de novos protocolos em RSSF.

Em [Van Hoesel et al. 2004], os autores apresentam um projeto integrado entre as camadas de rede e enlace, utilizando os protocolos *EYES MAC* e o *EYES Source Routing (ESR)*. Neste projeto, o protocolo MAC monitora as mudanças de topologia, falhas no nó e na comunicação e também o *duty cycle* dos nós. A interação entre os protocolos ocorre no momento em que o protocolo de roteamento necessita conhecer a topologia da rede, com a finalidade de entregar os pacotes aos seus devidos destinatários. Assim, acessando a tabela de vizinhos, o protocolo obtém o conhecimento da topologia local, tornando o roteamento mais eficiente. Essa informação é obtida pela camada MAC, que possui uma lista dos nós vizinhos ativos, aptos a participar do roteamento. Essa abordagem mostrou-se eficiente, principalmente em ambientes nos quais os nós são móveis. Diferentemente dos protocolos propostos neste trabalho, o projeto integrado *EYES MAC* baseia sua decisão apenas em informações locais de cada nó, não considerando demais parâmetros que possam influenciar no roteamento.

Outra proposta de trabalho que utiliza o conceito de projeto integrado é encontrada em [Sichitiu 2004]. Nesse trabalho, é apresentado um esquema de agendamento para os nós sensores, no qual a agenda de cada nó é criada de maneira distribuída, de acordo com o fluxo de dados na rede. Portanto, os nós sensores ficarão acordados somente quando for o momento de estabelecimento das rotas e permanecerão em modo dormência no restante do tempo. Essa abordagem faz uso de um rico compartilhamento de informações entre as camadas MAC e de rede. Além disso, é uma técnica projetada para um cenário com tráfego de dados periódicos, de forma que os agendamentos dos nós sensores serão efetuados objetivando favorecer ao máximo a entrega de dados ao nó *sink*.

Em [Souto et al. 2007], é proposto um trabalho que discute arquiteturas de projeto integrado existentes para RSSF e propõe uma nova arquitetura de rede, denominada *Arquitetura de Rede Orientada a Aspecto (Aspect-Oriented Network Architecture)*. Ao invés da comunicação entre camadas ser estritamente entre camadas adjacentes, essa arquitetura possibilita a comunicação direta entre quaisquer protocolos com a introdução de um novo mecanismo de abstração, o aspecto. Esse mecanismo permite à arquitetura de rede proposta o compartilhamento de informações e, ao mesmo tempo, garante a modularização de interesses que estão dispersos em diferentes camadas como a energia, segurança, o gerenciamento de falhas e outras características relacionadas com parâmetros de QoS.

Outra proposta de trabalho que emprega a técnica de projeto integrado de camadas

em RSSF pode ser encontrada em [Haapola et al. 2005]. Nesse trabalho, o consumo de energia é modelado através de informações de três camadas: rede, enlace e física. Nesse modelo de energia, são consideradas informações de topologia, esquema de agendamento do protocolo MAC e parâmetros de energia consumida pelo rádio transceptor. Assim, os autores apresentam uma análise do consumo de energia em ambientes com único salto e vários saltos. Pode-se concluir que, quando o modelo mais se aproxima da realidade, a comunicação com único salto pode consumir menos energia que a comunicação com vários saltos. Entretanto, essa análise é efetuada em uma topologia de rede linear, o que indica a necessidade de uma avaliação que abranja uma maior diversidade de cenários.

### 3. Projeto Integrado na Disseminação de Dados

Nesta seção, será apresentado o primeiro estudo de caso que propõe a utilização do projeto integrado na disseminação de dados em RSSFs. Nesse estudo é proposto o protocolo *Sensor MAC for Routing on a Curve* (SMAC-RC), que é desenvolvido para a camada MAC, baseado no protocolo S-MAC [Ye et al. 2002, Ye et al. 2004]. O protocolo SMAC-RC trabalha em cooperação com a camada de rede, a qual utiliza como protocolo de roteamento o *Trajectory Based Forwarding* (TBF) [Niculescu and Nath 2003]. O SMAC-RC utiliza o paradigma do projeto integrado para compartilhar informações entre os protocolos da camada MAC que utilizam tabela de vizinhos e protocolos de roteamento do tipo baseado no transmissor.

O TBF é um protocolo projetado para disseminar dados na rede com o auxílio de uma equação de curva. Assim, além dos dados, cada pacote conterá uma equação de curva que será utilizada pelos nós na decisão de qual será o próximo nó a propagar o pacote. Essa escolha é baseada na equação da curva e em uma tabela de vizinhos. Desde modo, ao receber um pacote o nó verifica se o pacote é de atualização da tabela de vizinhos ou é um pacote de dados. Sendo pacote de dados, o nó deve decidir qual será o próximo salto. Portanto o nó consulta sua tabela de vizinhos e verifica qual nó, dentre os seus vizinhos, está mais próximo do ponto de referência obtido pela curva. Na figura 2 o nó  $N_0$  deve eleger o próximo nó que receberá o pacote e, após a consulta a tabela de vizinhos, o nó escolhido é o nó  $N_1$ , por estar mais próximo do ponto de referência. Então o pacote é encaminhado para o nó  $N_1$ .

Quando o paradigma do projeto integrado é aplicado, o protocolo TBF trabalha de maneira diferenciada. A primeira mudança é na construção da tabela de vizinhos. No projeto integrado, o protocolo TBF não construirá essa tabela, pois, essa informação será obtida da camada MAC por meio de um compartilhamento de informações com protocolo SMAC-RC. Essa mudança economizará recursos, pois não serão necessárias as mensagens de construção e nem de atualização do protocolo TBF. A segunda mudança ocorre no momento da escolha do próximo nó que irá propagar o pacote. Neste caso o TBF não escolherá o próximo nó pelo critério de escolha do nó mais próximo ao ponto de referência, e sim, será construída uma lista com todos os nós candidatos a fazer a propagação. Dessa forma, o nó irá escolher, dentre seus vizinhos, os nós mais próximos ao ponto de referência, de acordo com um limiar pré-definido. Essa lista é encaminhada para o protocolo da camada MAC que decidirá qual será o nó a propagar a informação. Nesse projeto integrado, o protocolo SMAC-RC tem como critério de escolha o nó que estará acordado num tempo mais próximo dentre os candidatos contidos na lista.

No exemplo da figura 2, os nós  $N_0$ ,  $N_1$ ,  $N_2$  e  $N_3$  são os envolvidos no processo de disseminação e os números apresentados representam o tempo restante para que os mesmos acordem. Neste caso, é o nó  $N_0$  que deve eleger o próximo nó a propagar o pacote. De posse desta tabela de vizinhos proveniente de uma interação com a camada MAC, o protocolo TBF constrói a lista de prováveis nós a propagar este pacote. Na figura 2, os nós candidatos seriam  $N_1$ ,  $N_2$  e  $N_3$ . Após essa etapa, o TBF entrega a camada MAC essa lista com os nós candidatos. Em seguida o SMAC-RC, baseado nesta lista, escolhe o nó que acordará primeiro. Pelo funcionamento do protocolo SMAC-RC, essa informação é obtida através de uma consulta a tabela de agendamentos que o protocolo possui. Então, o nó escolhido pelo protocolo SMAC-RC para receber o pacote é o nó  $N_2$ , pois este é o nó que estará acordado primeiro.

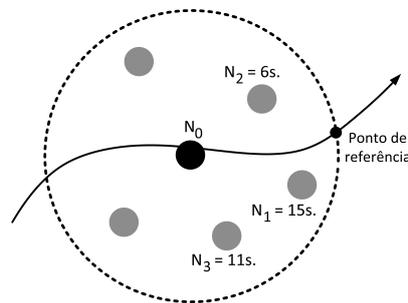


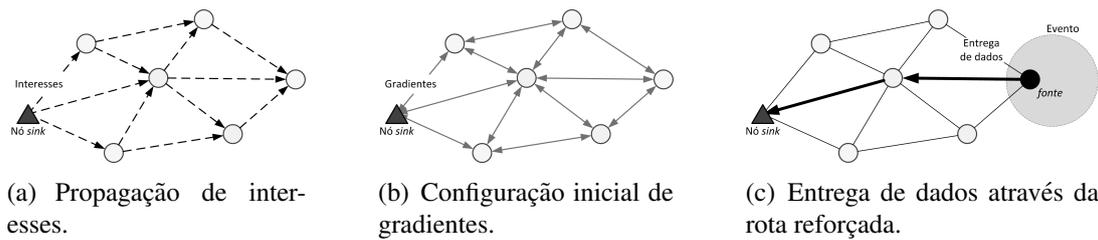
Figura 2. Escolha dos nós que participarão da lista de roteamento.

#### 4. Projeto Integrado na Coleta de Dados

Nesta seção é apresentado o projeto integrado para coleta de dados em RSSFs. Com esse objetivo é proposto o protocolo *Sensor MAC with Dynamic Duty Cycle* (SMAC-DDC). Esse protocolo é projetado para camada MAC, baseado no funcionamento do protocolo S-MAC [Ye et al. 2002, Ye et al. 2004]. O objeto principal desse protocolo é ajustar a duração do *duty cycle* dos nós sensores que pertencem a rota entre o nó fonte e o nó *sink* através da informação compartilhada pela camada de rede. Assim, a arquitetura restrita de camadas é violada e o compartilhamento de informações entre a camada MAC e a camada de rede é possibilitado.

Nesse projeto integrado optou-se por utilizar o protocolo Directed Diffusion (DD) [Intanagonwiwat et al. 2000] na camada de rede por ser um protocolo usado para cenários do tipo *event-driven*, nos quais os sensores somente enviam dados quando um evento ocorre. Este é um protocolo de roteamento centrado em dados em que os nós sensores têm conhecimento do tipo de dados transmitido. Além disso, o DD cria a rota entre o nó fonte e o nó *sink* permitindo a inserção de pontos de agregação ao longo da rota. A criação da rota, na versão tradicional do DD, é dividida em 3 fases: (a) propagação de interesses, (b) configuração inicial de gradientes e (c) entrega de dados através da rota reforçada (Figure 3). O protocolo DD apresenta três versões: *Two phase pull*, *One phase pull* e o *Push* [Heidemann and Silva 2003]. Neste trabalho utilizou-se uma versão reativa desse protocolo chamada de *Push*, que elimina a fase de propagação de interesses. Nessa versão as rotas entre o nó fonte e o nó *sink* serão criadas somente no momento em que o primeiro evento é detectado na rede.

No protocolo S-MAC, os nós possuem o *duty cycle* fixo durante toda a operação da rede. Por isso, o *throughput* numa coleta de dados pode ser afetado em cenários em



**Figura 3. Funcionamento do protocolo DD [Intanagonwiwat et al. 2000].**

que a frequência de eventos na rede é variável. Isso ocorre devido ao fato desse protocolo permitir somente uma única transmissão *unicasting* por ciclo de trabalho, e com um baixo *duty cycle*, o tempo de dormência é grande, o que acarreta o aumento da latência e redução da quantidade de dados entregues ao nó *sink*.

Em contrapartida, o protocolo SMAC-DDC reduz a latência dos pacotes que chegam ao nó *sink* através do ajuste do *duty cycle* para os nós que pertencem à rota entre o nó fonte e o nó *sink*. Então, as camadas MAC e de rede compartilham informações entre elas, de modo que o protocolo de roteamento informa para a camada MAC quando o nó pertence ou não a rota. No protocolo DD em sua versão *Push*, uma rota é criada quando o nó *sink* recebe o primeiro pacote do evento. Assim, como resposta um pacote do tipo reforço positivo é encaminhado, com o intuito de definir a rota para entrega de dados. Este pacote informa para cada nó se ele pertence ou não a rota. Quando o pacote for do tipo reforço positivo, o nó que o recebeu pertence à rota e, quando o pacote for do tipo reforço negativo, o nó não pertence à rota. Assim, nesse instante de recebimento do pacote de reforço, informações são compartilhadas entre as camadas e o *duty cycle* do nó é ajustado.

O algoritmo 1 é executado pelo SMAC-DDC para ajustar o *duty cycle* dos nós quando um pacote é recebido. De acordo com o algoritmo, quando um nó recebe um novo pacote, o protocolo SMAC-DCC acessa o cabeçalho do protocolo DD para ler as informações de tipo de pacote. Assim, quando o pacote é do tipo reforço positivo, o nó aumenta o seu *duty cycle* e programa uma nova sincronização de sua nova agenda com seus vizinhos. Neste momento, todos os nós que receberem um reforço positivo irão alterar os seus *duty cycle*, formando assim uma rota que entrega os pacotes de dados mais rápido. Esta rota permanece ativa durante todo o tempo em que o evento é sensoriado e, nesse período, os demais nós da rede que não fazem parte da rota permanecerão com seu valor de *duty cycle* inicial.

Quando um evento é finalizado ou outro vizinho entrega um pacote com menor latência, o nó *sink* envia um pacote de reforço negativo para a rota atual. Esse reforço negativo é propagado vizinho-a-vizinho, removendo os gradientes e desfazendo o caminho existente. Quando o nó recebe um reforço negativo, ele ajusta o seu *duty cycle* para o valor inicial. Uma nova sincronização com seus vizinhos será necessária para informar o seu novo agendamento. Assim, o nó operará com baixo *duty cycle* pois não estará roteando nenhum pacote.

O *duty cycle* dinâmico também reduz o consumo de energia na rede, pois os nós iniciam sua operação com um baixo *duty cycle* e somente aumenta esse valor quando o nó pertence à uma rota entre o nó fonte e o nó *sink*. Por exemplo, para comparar os

---

**Algorithm 1: SMAC-DDC – Recebendo um pacote**


---

```

input: Pacote recebido

//Protocolo SMAC-DDC acessa no pacote informações do
cabeçalho da camada de rede.
if tipoPacote = REFORÇO POSITIVO then
    //o nó pertence à rota
    pertençaRota = true;
    dutyCycle = 2d;
    sincronizaAgenda();
else if tipoPacote = REFORÇO NEGATIVO then
    //o nó não pertence à rota
    pertençaRota = false;
    dutyCycle = d/2;
    sincronizaAgenda();
end

```

---

protocolos S-MAC e SMAC-DDC, definiu-se o parâmetro  $d$ . Esse parâmetro configura o *duty cycle* de cada nó, e enquanto o protocolo S-MAC inicia a rede com *duty cycle* configurado com valor  $d$ , o protocolo SMAC-DDC começa com o *duty cycle* configurado para  $d/2$ . Desse modo, quando um evento é detectado o *duty cycle* é alterado para  $2d$ .

## 5. Resultados de Simulações

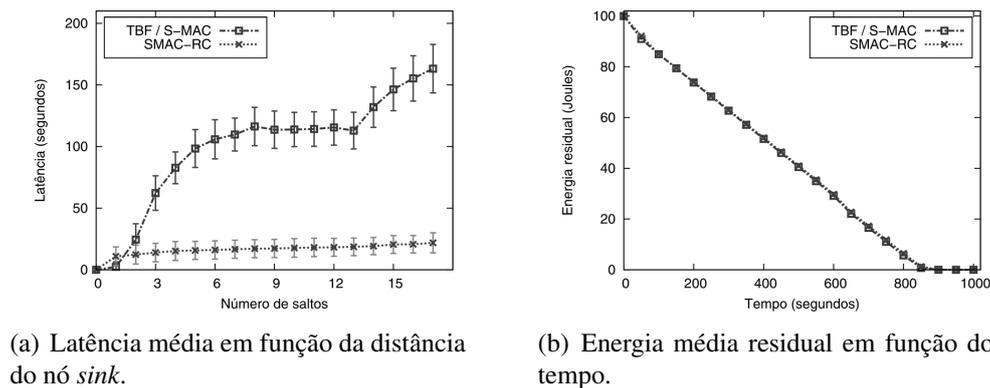
Esta seção apresenta os resultados de simulações para os dois estudos de caso. Na seção 5.1, são apresentados os resultados para o estudo de caso SMAC-RC. Na seção 5.2, são apresentados os resultados de simulação para o estudo de caso SMAC-DDC. Todas as simulações foram feitas no simulador Network Simulator 2.32 [ns2 2002]. Além disso, foi considerado que os nós sensores são depositados aleatoriamente, formando uma topologia plana em um campo de sensoriamento de  $50 \times 50 m^2$  para o estudo de caso SMAC-RC e  $100 \times 100 m^2$  para o estudo de caso SMAC-DDC. O nó *sink* está localizado na posição fixa  $(0, 0)$ . Em todas as simulações executadas, o consumo de energia foi configurado de acordo com o nó sensor modelo Mica2 [Crossbow 2008]. Todos os resultados de simulação correspondem a média aritmética de  $r$  simulações, em que  $r$  foi escolhido tal que o desvio padrão máximo observado seja inferior a 5 % da média.

### 5.1. Disseminação de Dados

O objetivo dessa seção é comparar o desempenho do projeto integrado proposto (SMAC-RC) com uma abordagem sem a utilização do projeto, no qual o TBF é executado com o S-MAC original. Assim, durante 1000 s, as mensagens são uniformemente disseminadas na rede. Essas disseminações são efetuadas sobre cinco trajetórias, e a rota dos pacotes na rede é determinada por essas trajetórias que são geradas pelo protocolo TBF. As simulações consideram uma rede de sensores com 500 nós, estáticos e homogêneos. Cada nó possui uma energia inicial de 100 J, um raio de comunicação de 5 m e *duty cycle* de 30%. As disseminações são iniciadas na rede a partir do instante 100 s, devido a fase de descoberta de vizinhos do protocolo S-MAC e SMAC-RC. Como métricas para avaliação foram escolhidas a latência em função da quantidade de saltos que o pacote efetuou a cobertura das disseminações e o consumo de energia.

A figura 4-(a) apresenta os resultados comparativos para a latência em função da quantidade de saltos para ambas as configurações. Pode-se observar que o SMAC-RC apresenta um resultado 90% menor que a abordagem TBF/S-MAC. Isso ocorre, pois, por utilizar constantes ciclos de dormiência e escuta o protocolo S-MAC só encaminha os pacotes em seu período de escuta. Então, se o próximo nó escolhido pelo TBF para rotear estiver em seu período de dormiência, o S-MAC aguardará que o nó retorne ao período de escuta para encaminhar o pacote. Por esse motivo, percebe-se o aumento da latência à medida que a distância ao *sink* aumenta. No protocolo SMAC-RC, este atraso no envio das mensagens é reduzido, quando não, extinto, uma vez que a camada MAC recebe do TBF uma lista com os prováveis nós a rotear o pacote, e então seleciona aquele que acordará primeiro.

O consumo de energia também foi avaliado para ambos os protocolos na figura 4-(b). Entretanto esse consumo não sofreu nenhuma diferença significativa e os resultados foram bastante semelhantes. Isso ocorre, pois, a abordagem proposta não faz nenhuma alteração no valor do *duty cycle* dos nós sensores. Sendo assim, os protocolos trabalham com mesmo valor para o *duty cycle*, explicando o fato do consumo de energia ser semelhante.



(a) Latência média em função da distância do nó *sink*.

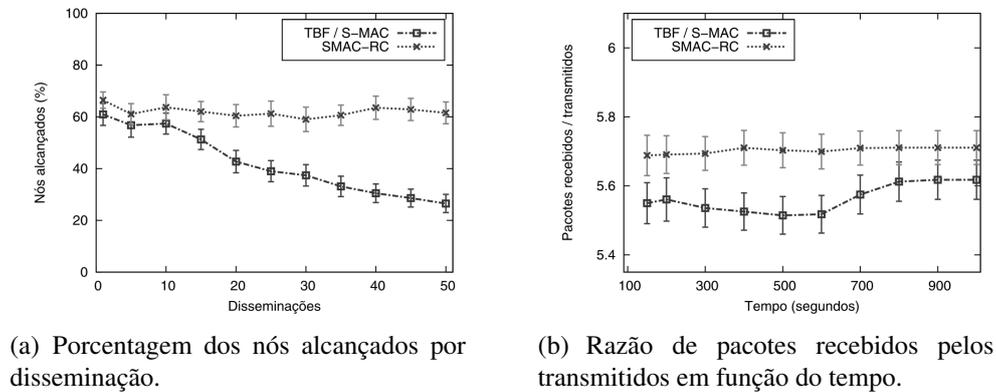
(b) Energia média residual em função do tempo.

**Figura 4. Latência e energia residual na disseminação de dados.**

Na figura 5-(a) são apresentados os resultados da cobertura de cada protocolo. A cobertura da rede é dada pela quantidade de nós que recebem os pacotes disseminados pelo nó *sink*. Caso o objetivo seja difundir uma informação para toda rede, é desejável que esse valor aproxime-se de 100% dos nós. Pode-se observar que a cobertura do protocolo SMAC-RC mostra-se melhor do que o TBF/S-MAC a partir da 15<sup>a</sup> disseminação. Isto ocorre devido ao fato do protocolo TBF, quando executado sem a cooperação com a camada MAC, necessita enviar pacotes para atualização da tabela de vizinhos. Assim, esses pacotes concorrem com os pacotes de dados, pois, para a camada MAC ambos são pacotes de dados, provenientes da camada de rede e recebem o mesmo tratamento. Como a cobertura apresentada neste gráfico corresponde ao percentual de nós que receberam pacotes de dados, o SMAC-RC permite o envio de um maior número de pacotes de dados, enquanto que no TBF/SMAC, uma fração da largura de banda é desperdiçada com envio de pacotes de atualização da tabela pelo TBF.

Finalmente, na figura 5-(b), o gráfico apresenta a relação entre os pacotes recebidos e os pacotes transmitidos na rede. É desejável que essa razão seja a maior possível, pois, isso representa que menos transmissões são necessárias para que um mesmo grupo

de nós receba os pacotes. Pode-se observar que o protocolo SMAC-RC apresenta um desempenho, em geral, melhor que o TBF/S-MAC. Além disso, é possível verificar também uma tendência constante a curva do protocolo SMAC-RC, o que representa num comportamento estável do protocolo frente ao TBF/S-MAC.

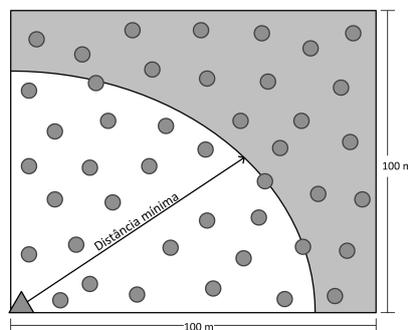


**Figura 5. Cobertura na disseminação de dados.**

## 5.2. Coleta de Dados

Nessa seção serão apresentados os resultados do projeto integrado SMAC-DDC comparado a um cenário sem a interação entre camadas, DD com o S-MAC em sua versão original. Para o cenário de coleta de dados foram considerados 200 nós depositados aleatoriamente, estáticos e homogêneos. Cada nó possui 25 J de energia inicial e um raio de comunicação de 15 m. O tempo de simulação é de 1200 s e, nesse tempo, os eventos acontecem de acordo com dois modelos: modelo de geração de eventos em área restrita e modelo de geração de eventos em área global.

No modelo de geração de eventos em área restrita, a localização dos eventos é aleatoriamente escolhida dentre todos os locais que estão a uma distância mínima do nó *sink*, de forma que somente os pacotes que tiverem percorrido um grande número de saltos serão considerados. Conforme é ilustrado na figura 6, os eventos do modelo de geração em área restrita só podem ocorrer na região acinzentada. Nas simulações, o número de eventos restritos é determinado pela variável  $\sigma$ .



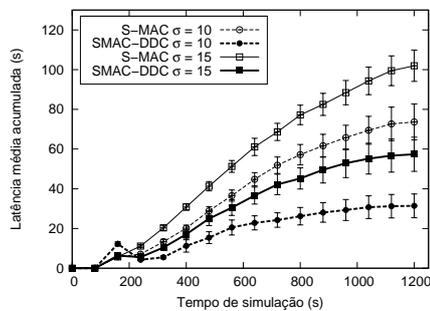
**Figura 6. Distância mínima para ocorrência dos eventos restritos.**

No modelo de geração de eventos em área global, a chegada de eventos à rede é modelada de acordo com uma distribuição de Poisson com  $\lambda$  eventos por segundo. Esses

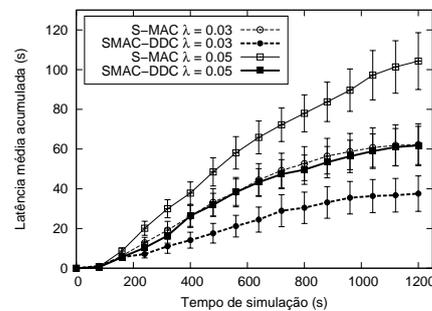
eventos acontecem na rede em regiões aleatórias e são detectados por apenas um único nó sensor. Nas simulações, a coleta de dados só é iniciada após os 70 s devido ao fato de que, até esse instante, os protocolos S-MAC e SMAC-DDC estão executando a fase de descoberta de vizinhos. Como métricas para avaliação, foram escolhidas a latência dos pacotes que chegam ao nó *sink*, quantidade de pacotes entregues ao *sink* e consumo de energia.

A latência de todos os pacotes quando o número de eventos é variado, é avaliada e os resultados são apresentados na figura 7. Na figura 7-(a), é apresentada a influência da mudança do número de eventos no modelo de geração de eventos em área restrita e, na figura 7-(b), é mostrada a influência da mudança do número de eventos no modelo de geração de eventos em área global. É possível observar que, em ambos os cenários, quando o número de eventos aumenta, a latência também aumenta. Isso ocorre devido ao fato de que, quanto mais eventos acontecem na rede, maior será a quantidade de pacotes para serem roteados.

É importante notar que nos dois cenários, o valor da latência para o protocolo SMAC-DDC é menor que o protocolo S-MAC para todas as quantidades de eventos gerados. Além disso, nota-se uma tendência constante nas curvas do protocolo SMAC-DDC a partir do instante de 800 s no qual o último evento é detectado. Isso indica que, após esse momento, poucos pacotes chegam ao nó *sink* ou então, a latência dos pacotes que chegam é a mesma do valor médio. Por isso, o valor da latência permanece invariável. Esse comportamento não é verificado nas curvas do protocolo S-MAC, pois se percebe uma tendência crescente nessas curvas indicando que mesmo tendo terminado o último evento alguns pacotes retardatários chegam ao nó *sink*, conforme é ilustrado na figura 7.



(a) Geração de eventos em área restrita.

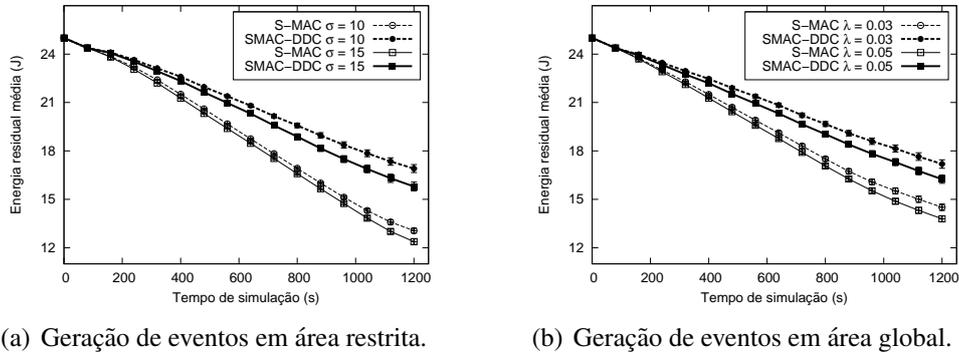


(b) Geração de eventos em área global.

**Figura 7. Latência média acumulada em diferentes modelos de geração de eventos.**

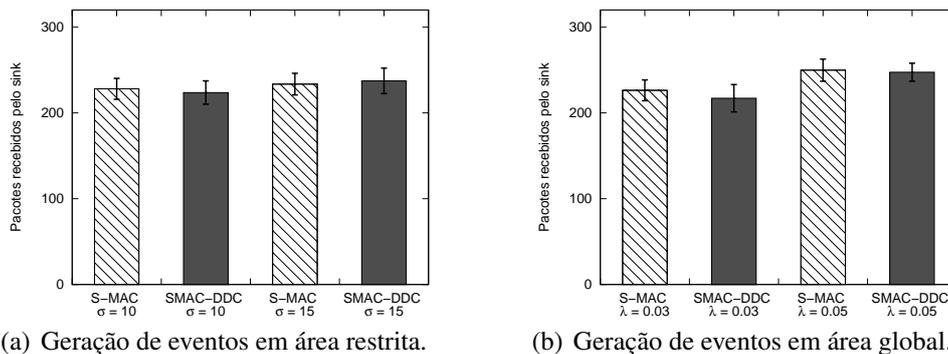
Seguindo a avaliação, são apresentados na figura 8 os resultados para o consumo de energia em ambos os cenários. A figura 8-(a) apresenta os resultados usando o modelo de geração de eventos para área restrita, e a figura 8-(b) apresenta os resultados quando o modelo de geração de eventos para área global é usado. Verifica-se que o consumo do protocolo S-MAC é maior que o consumo do SMAC-DCC. Esse comportamento é esperado uma vez que, com o SMAC-DDC, os nós trabalham com dois valores de *duty cycle*. O valor menor é usado quando não faz parte de nenhuma rota e o maior quando o nó pertence à rota para escoamento de dados em direção ao nó *sink*. Contudo, no cenário de eventos em área restrita, a distância entre o nó que detectou o evento e o nó de monitoramento é sempre maior que a distância mínima pré-definida, explicando o

fato que, apesar de ocorrerem poucos eventos, a rede apresenta um consumo de energia elevado para os dois protocolos.



**Figura 8. Energia residual média em diferentes modelos de geração de eventos.**

Finalmente, a figura 9 apresenta o número de pacotes que chegam ao nó *sink*. A figura 9-(a) mostra os resultados quando o modelo de geração de eventos em área restrita e a figura 9-(b) apresenta os resultados para o modelo de geração de eventos em área global. Nessas figuras, pode-se verificar que ambos os protocolos SMAC-DDC e S-MAC têm, aproximadamente, a mesma taxa de entrega. Aplicando o teste da média zero, não é possível dizer que a diferença das médias é significativamente diferente de zero [Jain 1991]. Assim, pode-se concluir que esses protocolos têm o desempenho similar para taxa de entrega de dados. Entretanto, como os nós só podem realizar uma transmissão *unicasting* por ciclo, a quantidade de dados que chegam ao *sink* é limitada pelo *duty cycle* e, conseqüentemente, o aumento no número de eventos gerados implicou em um pequeno aumento na taxa de entrega.



**Figura 9. Quantidade de dados recebidos pelo *sink* em diferentes modelos de geração de eventos.**

## 6. Conclusão e Direções Futuras

Este trabalho apresentou dois protocolos para verificar o benefício do projeto integrado de camadas em RSSF. O primeiro aplica o projeto integrado na disseminação de dados e o segundo aplica na coleta de dados. Em ambos os projetos, as camadas envolvidas são a camada MAC e a camada de rede, pois são as camadas responsáveis pela comunicação. A meta principal é alcançar melhorias no roteamento de dados nas duas situações. Resultados de simulações mostraram que o protocolo SMAC-RC oferece uma menor latência e

uma melhor cobertura na rede. Para o protocolo SMAC-DDC, os resultados de simulações mostraram que é possível reduzir a latência e o consumo de energia dos nós sensores, sem comprometer a taxa de entrega a coleta de dados. Como trabalhos futuros planeja-se estudar outras maneiras para construir um projeto integrado, além do compartilhamento de diferentes informações entre as camadas. Pretende-se também construir um novo protocolo MAC com a finalidade de reduzir a quantidade de pacotes de controle enviados e também o consumo de energia advindo de transmissões desnecessárias.

## Referências

- [Akyildiz et al. 2002] Akyildiz, I., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). A survey on sensor networks. *Communications Magazine, IEEE*, 40(8):102–114.
- [Crossbow 2008] Crossbow (2008). Mica2 868, 916 mhz. <http://www.xbow.com/>.
- [Cui et al. 2007] Cui, X., Zhang, X., and Shang, Y. (2007). Energy-saving strategies of wireless sensor networks. In *Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications, 2007 International Symposium on*, pages 178–181.
- [Gu and He 2007] Gu, Y. and He, T. (2007). Data forwarding in extremely low duty-cycle sensor networks with unreliable communication links. In *SenSys '07: Proceedings of the 5th international conference on Embedded networked sensor systems*, pages 321–334, New York, NY, USA. ACM.
- [Guo et al. 2009] Guo, S., Gu, Y., Jiang, B., and He, T. (2009). Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links. In *MobiCom '09: Proceedings of the 15th annual international conference on Mobile computing and networking*, pages 133–144, New York, NY, USA. ACM.
- [Haapola et al. 2005] Haapola, J., Shelby, Z., Pomalaza-Raez, C., and Mahonen, P. (2005). Cross-layer energy analysis of multihop wireless sensor networks. In *Wireless Sensor Networks, 2005. Proceedings of the Second European Workshop on*, pages 33–44.
- [Heidemann and Silva 2003] Heidemann, J. and Silva, F. (2003). Matching data dissemination algorithms to application requirements. In *In Proceedings of the ACM SenSys Conference*, pages 218–229. ACM Press.
- [Intanagonwiwat et al. 2000] Intanagonwiwat, C., Govindan, R., and Estrin, D. (2000). Directed diffusion: a scalable and robust communication paradigm for sensor networks. In *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 56–67, New York, NY, USA. ACM Press.
- [Jain 1991] Jain, R. (1991). *The Art of Computer Systems Performance Analysis: techniques for experimental design, measurement, simulation and modeling*. John Wiley.
- [Jurdak 2007] Jurdak, R. (2007). *Wireless Ad Hoc and Sensor Networks: A Cross-Layer Design Perspective*. Springer-Verlag.
- [Karl and Willig 2005] Karl, H. and Willig, A. (2005). *Protocols and Architectures for Wireless Sensor Networks*. John Wiley & Sons.
- [Kawadia and Kumar 2005] Kawadia, V. and Kumar, P. (2005). A cautionary perspective on cross-layer design. *Wireless Communications, IEEE*, 12(1):3–11.

- [Machado et al. 2005] Machado, M., Goussevskaia, O., Mini, R., Rezende, C., Loureiro, A., Mateus, G., and Nogueira, J. (2005). Data dissemination in autonomic wireless sensor networks. *Selected Areas in Communications, IEEE Journal on*, 23(12):2305–2319.
- [Mini and Loureiro 2009] Mini, R. A. and Loureiro, A. A. (2009). Energy in wireless sensor networks (tutorial). MiNEMA Winter School 2009, March 23-26, Göteborg, Sweden (<http://www.cse.chalmers.se/minema2009/>).
- [Niculescu and Nath 2003] Niculescu, D. and Nath, B. (2003). Trajectory based forwarding and its applications. In *MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 260–272, New York, NY, USA. ACM.
- [ns2 2002] ns2 (2002). The network simulator. [www.isi.edu/nsnam/ns](http://www.isi.edu/nsnam/ns).
- [Pottie and Kaiser 2000] Pottie, G. and Kaiser, W. (2000). Embedding the internet wireless integrated network sensors. *Communications of the ACM*, 43:51–58.
- [Puccinelli and Haenggi 2005] Puccinelli, D. and Haenggi, M. (2005). Wireless sensor networks: applications and challenges of ubiquitous sensing. *Circuits and Systems Magazine, IEEE*, 5(3):19–31.
- [Shakkottai et al. 2003] Shakkottai, S., Rappaport, T., and Karlsson, P. (2003). Cross-layer design for wireless networks. *Communications Magazine, IEEE*, 41(10):74–80.
- [Sichitiu 2004] Sichitiu, M. (2004). Cross-layer scheduling for power efficiency in wireless sensor networks. *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, 3:1740–1750 vol.3.
- [Souto et al. 2007] Souto, E., Aschoff, R., Sadok, D. H., and Kelner, J. (2007). Projeto cross-layer entre as camadas mac, roteamento e aplicação em redes de sensores. In *Anais do XXV Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, volume 1, pages 45–58, Pará, Belém.
- [Srivastava and Motani 2005] Srivastava, V. and Motani, M. (2005). Cross-layer design: a survey and the road ahead. *Communications Magazine, IEEE*, 43(12):112–119.
- [Van Hoesel et al. 2004] Van Hoesel, L., Nieberg, T., Wu, J., and Havinga, P. (Dec. 2004). Prolonging the lifetime of wireless sensor networks by cross-layer interaction. *Wireless Communications, IEEE [see also IEEE Personal Communications]*, 11(6):78–86.
- [Ye et al. 2002] Ye, W., Heidemann, J., and Estrin, D. (2002). An energy-efficient mac protocol for wireless sensor networks. *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, 3:1567–1576 vol.3.
- [Ye et al. 2004] Ye, W., Heidemann, J., and Estrin, D. (2004). Medium access control with coordinated adaptive sleeping for wireless sensor networks. *Networking, IEEE/ACM Transactions on*, 12(3):493–506.