

**BRUNO VITOR ALMEIDA PINHEIRO**

**SIMULAÇÃO DE ALGORITMO AUTO ORGANIZADO  
PARA PROTOCOLO MAC EM REDES DE SENSORES  
SEM FIO**

**LAVRAS - MG**

**2013**

**BRUNO VITOR ALMEIDA PINHEIRO**

**SIMULAÇÃO DE ALGORITMO AUTO ORGANIZADO PARA  
PROTOCOLO MAC EM REDES DE SENSORES SEM FIO**

Monografia de Graduação apresentada  
ao Departamento de Ciência da Computação da Universidade Federal de Lavras  
como parte das exigências do curso de  
Sistemas de Informação para obtenção  
do título de Bacharel em Sistemas de Informação.

Orientador

Prof. Dr. Tales Heimfarth

**LAVRAS - MG**

**2013**

**BRUNO VITOR ALMEIDA PINHEIRO**

**SIMULAÇÃO DE ALGORITMO AUTO ORGANIZADO PARA  
PROTOCOLO MAC EM REDES DE SENSORES SEM FIO**

Monografia de Graduação apresentada  
ao Departamento de Ciência da Computação da Universidade Federal de Lavras  
como parte das exigências do curso de  
Sistemas de Informação para obtenção  
do título de Bacharel em Sistemas de Informação.

*Aprovada em Abril de 2013*

Prof. Dr. Luiz Henrique Andrade Correia - UFLA

Prof. Dr. João Carlos Giacomini - UFLA

Prof. Dr. Tales Heimfarth

Orientador

**LAVRAS - MG**

**2013**

## **AGRADECIMENTOS**

À Universidade Federal de Lavras (UFLA) e ao Departamento de Ciência da Computação (DCC), pela oportunidade concedida para realização deste trabalho.

Aos professores do Departamento, pelos ensinamentos transmitidos e harmoniosa convivência.

Ao professor Dr. Tales Heimfarth pela orientação, paciência, amizade, dedicação e seus ensinamentos que foram de grande relevância para a realização deste trabalho.

Aos professores Dr. João Carlos Giacomini e Dr. Luiz Henrique Andrade Correia, pela amizade, companheirismo e ensinamentos, que foram de grande valia para a realização deste trabalho e por se disporem a participar como membros na banca examinadora.

Aos amigos graduandos, em especial a Alex Guimarães e Jesimar Arantes, por todo o apoio e incentivo durante a realização do trabalho.

## SUMÁRIO

<b>Resumo .....</b>	<b>iv</b>
<b>1      Introdução .....</b>	<b>1</b>
<b>1.1    Motivação .....</b>	<b>1</b>
<b>1.2    Objetivos .....</b>	<b>2</b>
<b>1.3    Organização do Trabalho .....</b>	<b>2</b>
<b>2      Redes de Sensores Sem Fio .....</b>	<b>4</b>
<b>2.1    Características de RSSF.....</b>	<b>6</b>
<b>2.2    Protocolos em RSSF .....</b>	<b>8</b>
<b>2.2.1   Camada física (<i>physical layer</i>) .....</b>	<b>8</b>
<b>2.2.2   Camada de ligação de dados (<i>Data Link Layer</i>) .....</b>	<b>9</b>
<b>2.2.3   Camada de rede (<i>Network layer</i>) .....</b>	<b>10</b>
<b>2.2.4   Camada de transporte (<i>Transport Layer</i>).....</b>	<b>11</b>
<b>2.2.5   Camada de aplicação (<i>Application Layer</i>) .....</b>	<b>11</b>
<b>3      Controle de acesso ao meio .....</b>	<b>13</b>
<b>3.1    Padrões de comunicação em RSSFs .....</b>	<b>13</b>
<b>3.2    Classificação de Protocolos de Controle de Acesso ao Meio.....</b>	<b>14</b>
<b>3.2.1   Protocolos de atribuição fixa .....</b>	<b>14</b>
<b>3.2.2   Protocolos de atribuição por demanda .....</b>	<b>15</b>
<b>3.2.3   Protocolos de acesso aleatório .....</b>	<b>16</b>
<b>4      Auto Organização em RSSF .....</b>	<b>17</b>
<b>4.1    Protocolos MAC auto organizáveis .....</b>	<b>17</b>
<b>4.1.1   WiseMAC .....</b>	<b>18</b>
<b>4.1.2   S-MAC .....</b>	<b>19</b>
<b>4.1.3   T-MAC .....</b>	<b>22</b>
<b>4.1.4   D-MAC .....</b>	<b>23</b>
<b>5      Metodologia.....</b>	<b>25</b>
<b>6      Configurável Sensor MAC ou CS-MAC .....</b>	<b>26</b>
<b>7      Algoritmos .....</b>	<b>29</b>
<b>7.1    Sincronização dos nós .....</b>	<b>29</b>
<b>7.2    Funcionamento do protocolo S-MAC.....</b>	<b>30</b>
<b>7.3    Cálculo da distância ao centro da rede .....</b>	<b>31</b>
<b>7.4    Construção do círculo .....</b>	<b>33</b>
<b>7.5    Roteamento dos pacotes .....</b>	<b>35</b>
<b>8      Resultados Obtidos .....</b>	<b>37</b>
<b>8.1    Conformação da Rede e Testes .....</b>	<b>37</b>
<b>8.2    Parâmetros utilizados .....</b>	<b>38</b>
<b>8.3    Análise de desempenho .....</b>	<b>39</b>

<b>9</b>	<b>Conclusão .....</b>	<b>44</b>
----------	------------------------	-----------

## LISTA DE FIGURAS

Figura 1	Exemplo da distribuição dos nós de uma RSSF em uma região de interesse. ....	5
Figura 2	Pilha de protocolos em redes de sensores sem fio. ....	9
Figura 3	Funcionamento do WiseMAC (EL-HOIYDI; DECOTIGNIE, 2004). ....	18
Figura 4	Funcionamento do <i>duty cycle</i> no protocolo S-MAC. ....	20
Figura 5	Exemplo de sincronização de ciclos de atividades com protocolo S-MAC ( <i>clusters</i> A e B e nós de borda). ....	21
Figura 6	<i>Duty cycle</i> adaptativo no protocolo T-MAC. ....	23
Figura 7	<i>Duty cycle</i> sincronizado em níveis no protocolo D-MAC. ....	24
Figura 8	Cadeia de nós selecionados para sincronização. ....	27
Figura 9	Cronogramas de atividade sincronizados sequencialmente. ....	28
Figura 10	Rede utilizada nas simulações. ....	38
Figura 11	Comparação entre os tempos totais de transmissão de pacotes em cada protocolo. ....	40
Figura 12	Casos onde não há ganho de desempenho com o protocolo CS-MAC. ....	41
Figura 13	Comparação entre os tempos de transmissão médio dos protocolos. ....	42
Figura 14	Comparação entre os números de saltos necessários para completar as transmissões em cada protocolo. ....	42
Figura 15	Comparação do número médio de saltos em transmissões. ....	43

## RESUMO

Redes de sensores sem fio (RSSF) são redes compostas por um conjunto de nós sensores que trabalham de forma cooperativa para a realização de tarefas e monitoramento de uma região de interesse. Através do uso de protocolos auto-organizados a rede se mantém em funcionamento sem a necessidade de interferência de agentes externos. Este trabalho propõe um protocolo CS-MAC (*Configurable Sensor-MAC*), que pode ser considerado como uma extensão do protocolo S-MAC (*Sensor-MAC*), apresentado em Ye, Heidemann e Estrin (2002), onde os nós são sincronizados de acordo com agendas que determinam seus ciclos de sono/atividade (*duty cycles*). O CS-MAC adiciona ao S-MAC mecanismos que possibilitam a determinação de agendas de maneira arbitrária por outras camadas da pilha de protocolos. Utilizando esses mecanismos foi desenvolvida uma aplicação capaz de construir uma via de transmissão rápida (*backbone*) de pacotes com o objetivo de reduzir a latência introduzida pela utilização de *duty cycles*. Por meio de simulações, constatou-se que o CS-MAC com um *duty cycle* de 10%, foi possível reduzir a latência média em 25% em relação ao S-MAC, chegando a 58% no melhor caso.

Palavras-chave: Protocolo de controle de acesso ao meio; Auto-organização; Redes de sensores sem fio; RSSF; MAC.



## ABSTRACT

Wireless sensor networks (WSN) are networks composed of a set of sensor nodes working cooperatively to accomplish tasks and monitoring a region of interest. Through the use of self-organizing protocols, the network can operate without interference of external agents. This paper proposes CS-MAC (Configurable Sensor MAC), which can be considered as an extension of the S-MAC (Sensor MAC) protocol, found in Ye, Heidemann e Estrin (2002), where the nodes are synchronized according to schedules that determine sleep/wakeup cycles (duty cycles). The CS-MAC adds to S-MAC mechanisms to enable determination of arbitrary schedules by other layers of the protocol stack. Using these mechanisms, an application able to build a route of rapid transmission (backbone) of packets in order to reduce the latency introduced by the use of duty cycles was developed. Through simulations, it was found that using a 10% duty cycle, the CS-MAC could reduce the average latency in 25% compared to S-MAC, reaching 58% at best.

**Keywords:** Medium access control protocol; Self-organization; Wireless sensor networks; WSN; MAC.

# 1 Introdução

Redes de sensores sem fio (RSSF) tem sido cada vez mais utilizadas em variados tipos de aplicações em diferentes tipos de ambientes, como no monitoramento de florestas para controle de incêndios, controle de irrigação em grandes plantações, controle de ventilação e condicionamento de ar em grandes edifícios, vigilância em áreas de interesse militar, automação industrial, etc. A cada dia surgem novas tecnologias em sensores, rádio transmissores e outros equipamentos utilizados na construção dos nós sensores que compõem esse tipo de rede, o que tem, cada vez mais, reduzido o custo para produção e, conseqüentemente, para a aquisição desses equipamentos.

Os nós sensores utilizados para a construção dessas redes são, em geral, alimentados por baterias, o que implica em sérias restrições ao seu tempo de vida útil (AKYILDIZ *et al.*, 2002b). Considerando o fato de que a maioria das aplicações para esse tipo de rede é feita para utilização em ambientes que dificultam ou impossibilitam a substituição dessas baterias, torna-se necessário ou mais vantajoso substituir os próprios nós por novos. Embora o custo da substituição seja cada vez mais reduzido pelos avanços em tecnologia ainda é mais interessante fazer com que o tempo de vida da rede seja estendido através de mecanismos que possam reduzir o consumo de energia para se evitar uma alta frequência na necessidade de substituição dos nós sensores. Isso é possível através da utilização de protocolos eficientes em consumo de energia.

## 1.1 Motivação

Em redes de sensores, o maior consumo de energia se dá na recepção e sobretudo na transmissão dos dados (HALKES; DAM; LANGENDOEN, 2005). Sendo assim é de suma importância garantir que a comunicação entre os nós sen-

sores seja eficiente, evitando problemas que possam resultar na necessidade de retransmissão de pacotes (como no caso da colisão de pacotes por exemplo), bem como a recepção desnecessária de pacotes pelos nós da rede (*overhearing*) ou mesmo a permanência dos receptores em atividade quando não há pacotes sendo enviados ao nó (*idle listening*). Os responsáveis por evitar ou minimizar a ocorrência desses tipos de problemas são os protocolos de controle de acesso ao meio (*MAC-Medium Access Control Protocols*). Esses protocolos determinam, dentre outras coisas, o momento em que cada nó pode começar ou não uma transmissão e os períodos em que o nó deve permanecer desligado ou ligado (ciclo de sono).

## 1.2 Objetivos

Esse trabalho tem por objetivo investigar mecanismos que promovam redução do consumo de energia em redes de sensores e aumentem seu tempo de vida útil. Ao final, será proposto um algoritmo que seja capaz de amenizar problemas causados por esses mecanismos, como o aumento na latência em transmissões de pacotes.

## 1.3 Organização do Trabalho

Nas seções seguintes serão apresentados a estrutura básica de uma rede de sensores, uma visão geral sobre as camadas que compõem a arquitetura dos nós.

O foco da investigação são os protocolos da camada de controle de acesso ao meio (em inglês *Medium Access Control* - MAC). Serão apresentados exemplos de protocolos existentes para essa camada incluindo uma classificação e exemplos de cada um deles. Também será apresentado um conceito de auto-organização e sua utilização nesses protocolos. O algoritmo proposto nesse trabalho deverá utilizar técnicas de auto-organização para promover uma organização na rede que

possibilite o aumento da velocidade de comunicação entre nós selecionados. Ao final serão apresentados os resultados de simulações que demonstram que houve redução na latência transmissão de pacotes entre pontos distantes ao fazer com que esses passem necessariamente pelos nós selecionados no passo anterior.

## 2 Redes de Sensores Sem Fio

Redes de sensores sem fio (RSSF) são redes compostas por um grande número de nós sensores densamente distribuídos dentro de um fenômeno ou próximo a ele, cujas posições não precisam ser predeterminadas e que trabalham de forma cooperativa para a realização de tarefas. Isso possibilita a distribuição aleatória em regiões inacessíveis, porém também torna necessária a criação de protocolos e algoritmos com capacidade de auto organização (AKYILDIZ *et al.*, 2002a).

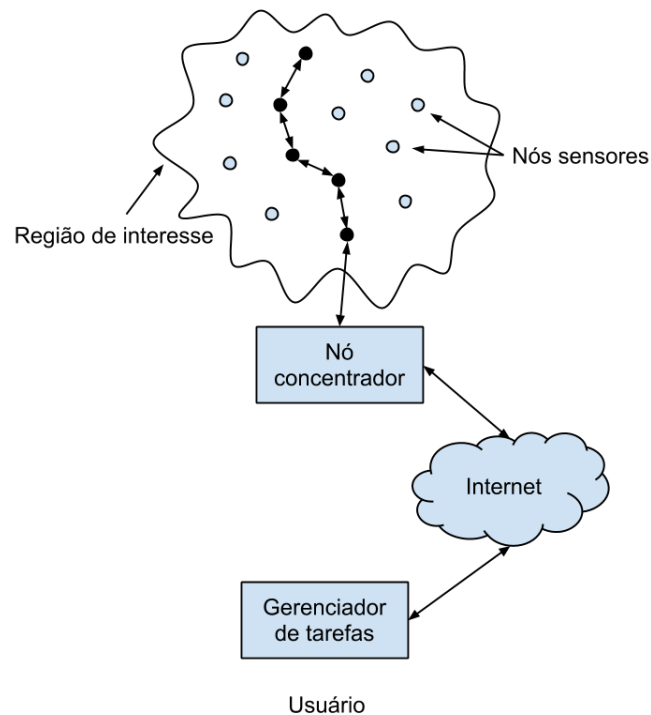
Römer e Mattern (2004) definem RSSF como redes *ad hoc* de larga escala, *multi-hop*, não particionada composta por nós sensores, em sua maioria estáticos, largamente homogêneos, pequenos e com recursos limitados, randomicamente distribuídos em uma área de interesse.

A Figura 1 mostra um exemplo da disposição de uma rede desse tipo. Nesse exemplo os nós da rede de sensores ficam distribuídos em uma região de interesse (*sensor field*). Cada um dos nós (*sensor nodes*) coleta dados do ambiente e pode roteá-los até o nó *sink* (nó que reúne as informações da rede) e este pode comunicar-se com o gerenciador de tarefas por meio da internet.

Karl e Willig (2005) apresentam uma definição de redes *ad hoc* como redes configuradas para, literalmente, atender a um propósito específico e a necessidades específicas de comunicação. Karl e Willig (2005) apresentam também uma definição para *Mobile Ad Hoc Networks* (MANET), uma rede *ad hoc* associada à comunicação sem fio (*wireless*) e à mobilidade dos nós participantes.

Listadas abaixo estão algumas diferenças significativas entre MANET e RSSF:

- O número de nós sensores em RSSF pode ser de uma ordem de magnitude muito acima da encontrada em redes MANET (AKYILDIZ *et al.*, 2002a).



**Figura 1:** Exemplo da distribuição dos nós de uma RSSF em uma região de interesse.

- MANET são utilizadas em aplicações como comunicação por voz ou acesso a infraestrutura remota necessitando assim de equipamentos robustos o suficiente para suportar esse tipo de aplicações ao contrário de RSSFs que são compostas por equipamentos bem mais simples e limitados (KARL; WILLIG, 2005).
- RSSF são capazes de comportar variados cenários de aplicação, devido ao grande número de combinações possíveis de tecnologias de comunicação, computação e sensores. Por exemplo conseguindo trabalhar em diversas e variáveis densidades, o que irá requerer o uso de diferentes protocolos ou

protocolos adaptativos. Tal flexibilidade existe também em MANET, porém em um nível bastante reduzido (KARL; WILLIG, 2005).

- A densidade das RSSF é bem maior que a encontrada nas MANET (AKYILDIZ *et al.*, 2002a).
- Nós sensores de uma RSSF são suscetíveis a erros e a topologia da rede pode mudar com a morte ou adição de novos nós (AKYILDIZ *et al.*, 2002a).
- Em RSSF predomina o paradigma de comunicação *broadcast*, ao contrário do que predomina em redes MANET que é a comunicação *point-to-point*.
- Nós de uma RSSF podem não possuir uma identificação global devido ao grande número de nós sensores, pois isso causaria um grande *overhead*.

## 2.1 Características de RSSF

Há inúmeros problemas encontrados nesse tipo de redes e um número igualmente grande de soluções propostas para cada um desses problemas. De maneira geral, nós de uma rede de sensores sem fio são alimentados por baterias e, conseqüentemente, possuem um tempo de vida útil consideravelmente limitado, tendo em vista que, na maioria das utilizações desse tipo de rede não é possível fazer a substituição das baterias (AKYILDIZ *et al.*, 2002b). Tal asserção baseia-se no fato que esse tipo de solução, dadas as suas limitações, só seria empregada em casos onde outras soluções mais simples ou eficientes não pudessem ser utilizadas, como no monitoramento de áreas de difícil acesso, por exemplo, o interior de florestas, áreas de desastre, regiões com condições extremas ou contaminadas, etc.

Algumas das principais características que uma RSSF deve apresentar são:

- **Tolerância a falhas:** Alguns nós da rede podem deixar de funcionar devido a vários fatores como o fim de suas baterias ou dano físico. Tolerância a fa-

lhas é a capacidade de manter a rede de sensores funcionando sem interrupções devido a falhas em nós sensores (SRISATHAPORNPHAT; JAIKAE; SHEN, 2001; HOBLOS G.ANDSTAROSWIECKI, 2000).

- **Escalabilidade:** O número de nós distribuídos para o estudo de um fenômeno pode atingir valores muito elevados (ordem de centenas de milhares ou mais). Além disso pode existir um número muito grande de nós em uma área muito pequena (alta densidade). Devem ser desenvolvidos mecanismos e protocolos capazes de trabalhar com esse número de nós.
- **Tempo de vida:** A maioria dos cenários de utilização de RSSF limitam ou impossibilitam a substituição de baterias nos nós. Ainda assim, espera-se que uma rede de sensores permaneça em funcionamento por um determinado período de tempo necessário à realização de uma missão ou o maior tempo possível. Portanto, torna-se evidente a necessidade de fazer com que a rede opere de modo eficiente em relação ao consumo de energia (KARL; WILLIG, 2005).
- **Programabilidade:** Os nós de uma rede de sensores devem processar informações, mas além disso precisam ser capazes de se adaptar a mudanças em suas tarefas. Eles devem ser programáveis e sua programação deve ser passível de alterações mesmo quando em operação para atender a novas tarefas que venham a se tornar importantes (KARL; WILLIG, 2005).
- **Manutenibilidade:** Com o passar do tempo podem ocorrer mudanças tanto no ambiente como na rede de sensores. Essas mudanças podem ser causadas por vários fatores, como falha de nós ou surgimento de novas tarefas. A rede deve adaptar-se a essas mudanças e monitorar o seu próprio estado, fazendo as alterações nos parâmetros operacionais quando necessário, por exemplo



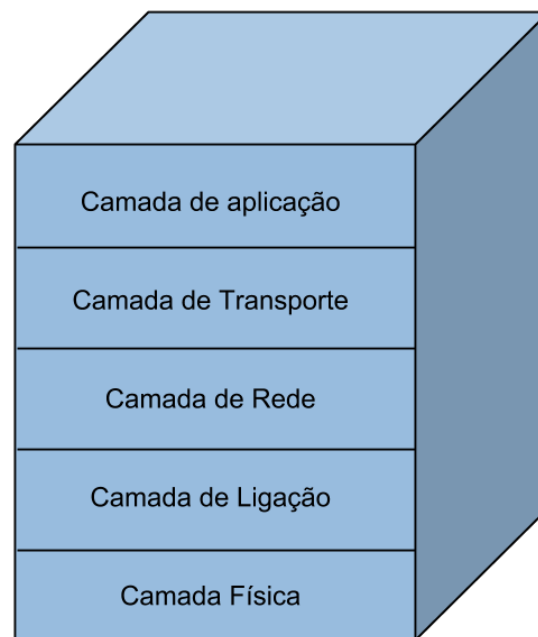
reduzindo a qualidade do serviço quando o nível de energia estiver baixo. Ela poderia interagir com mecanismos externos de manutenção para garantir a continuidade de sua operação a um nível mínimo de qualidade necessário por uma maior período de tempo (MAINWARING *et al.*, 2002).

## 2.2 Protocolos em RSSF

Os protocolos utilizados em redes de sensores sem fio são responsáveis por prover comunicação eficiente em termos de energia e roteamento no meio sem fio, integração entre dados e protocolos de rede e promover o esforço cooperativo dos nós sensores (AKYILDIZ *et al.*, 2002a). Os protocolos são organizados como se dispostos em uma pilha onde cada camada desempenha um papel específico, necessário ao funcionamento da rede de sensores. Uma descrição de cada uma dessas camadas é apresentada em Akyildiz *et al.* (2002a) e descrita resumidamente a seguir. A Figura 2 é uma representação da pilha de protocolos.

### 2.2.1 Camada física (*physical layer*)

A camada física é responsável por selecionar a frequência, geração de frequência portadora (*carrier frequency generation*), detecção de sinais, modulação e encriptação de dados. O custo, tanto em termos de consumo de energia quanto em complexidade de implementação, da comunicação sem fio a longas distâncias é muito elevado. Em redes de sensores sem fio, a minimização do consumo de energia tem importância significativa na propagação e nos efeitos que causam desvanecimento do sinal. Em geral, a potência necessária para transmitir um sinal a uma distância  $d$  é diretamente proporcional a  $d^n$ , sendo  $2 \leq n < 4$ .  $n$  é próximo de 4 para antenas muito baixas e canais muito próximas do chão (POTTIE; KAISER, 2000).



**Figura 2:** Pilha de protocolos em redes de sensores sem fio.

### **2.2.2 Camada de ligação de dados (*Data Link Layer*)**

Camada de ligação de dados é a camada responsável por multiplexação de fluxos de dados (*data streams*), detecção de quadros de dados (*data frames*), acesso ao meio (*medium access*) e controle de erros (*error control*).

#### **Protocolos de controle de acesso ao meio**

Os protocolos de controle de acesso ao meio (*Medium Access Control Protocols*) são o foco desse trabalho e atuam nessa camada. Eles coordenam o acesso ao meio físico por nós ativos. Esses protocolos são de grande importância, uma vez que o canal de comunicação sem fio (*wireless*) é inerentemente propenso a

erros, e precisam tratar de problemas específicos desse tipo de rede, tais como: o problema do terminal escondido (*hidden-terminal*), problema do terminal exposto (*exposed-terminal*) e efeitos do desvanecimento de sinal (*signal fading effects*) (KUMAR; RAGHAVAN; DENG, 2006). Esses protocolos serão abordados mais detalhadamente na seção 3.

### 2.2.3 Camada de rede (*Network layer*)

Técnicas convencionais de roteamento não são aplicáveis às características específicas desse tipo de redes (abordadas na seção 2.1). Segundo Gürses (2005), os protocolos dessa camada são desenvolvidos tendo em vista os seguintes aspectos:

- **Não há um identificador global**, devido a existência de poucos dados de eventos e um grande número de nós.
- **Eficiência de energia** é necessária para aumentar o tempo de vida da rede.
- **Descoberta de rotas auto organizável** para lidar com mudanças na topologia da rede.
- **Modelo de distribuição de dados**, segundo Tilak, Abu-ghazaleh e Heinzelman (2002), pode ser dos tipos:

**contínuo** - sensores entregam os dados continuamente;

**dirigido a eventos** - sensores entregam dados ao detectarem um evento de interesse;

**dirigido a consultas** - os nós entregam dados a partir de uma solicitação iniciada pelo observador;

- **Processamento em rede** por meio da utilização de “funções de agregação” é um requisito para comunicação eficiente em termos de energia;
- **Comunicação local inter sensor** é um requisito para aplicações, bem como para o processamento em rede;
- **Qualidade de serviço** (largura de banda, quantidade de erros e latência) deve atender aos requisitos das aplicações;

#### 2.2.4 Camada de transporte (*Transport Layer*)

Muitas aplicações de redes de sensores sem fio requerem mecanismos de controle de congestionamento para regular a quantidade de tráfego injetada na rede com o objetivo de evitar perda de pacotes e para atribuir confiabilidade à entrega de pacotes e eventos (RAHMAN; El Saddik; GUEAIEB, 2008). Em várias aplicações há nós responsáveis por capturar informações críticas e enviá-las a um nó concentrador (*sink node*) (RAHMAN; El Saddik; GUEAIEB, 2008). Os protocolos dessa camada são desenvolvidos para atender a essas restrições e garantir que os pacotes sejam enviados e alcancem seus destinos corretamente.

#### 2.2.5 Camada de aplicação (*Application Layer*)

Segundo Akyildiz *et al.* (2002a), embora haja muitas aplicações propostas para redes de sensores, potenciais protocolos para a camada de aplicação permanecem em uma região altamente inexplorada. Esses protocolos possibilitam a interação entre a camada de aplicação e as camadas inferiores. Há três protocolos mais comuns utilizados nessa camada.

- **Protocolo de gerenciamento de sensores:** protocolos de gerenciamento na camada de aplicação tornam o *hardware* e o *software* das camadas inferiores transparentes para aplicações de gerenciamento de redes de sensores.
- **Protocolo de atribuição de tarefas e anúncio de dados (*data advertisement*):** utilizado para atribuição de tarefas aos nós e também para comunicação de detecção de eventos pelos nós.
- **Protocolo de requisição (*sensor query*) e disseminação de dados:** utilizados para fazer consultas ou requisições de dados aos sensores e devolução das respectivas respostas.

Em redes de sensores sem fio não existe uma hierarquia rígida nessa pilha de protocolos. Os protocolos de cada camada podem interagir diretamente com protocolos de outras camadas sem o envolvimento das camadas intermediárias.

Nas seções seguintes serão apresentados mais detalhes sobre a porção mais significativa da camada de ligação, a subcamada responsável pelo controle de acesso ao meio (MAC), foco principal deste trabalho. Serão apresentados protocolos existentes para essa camada, além de uma classificação desses protocolos em relação à atribuição de recursos aos nós.

### 3 Controle de acesso ao meio

Protocolos de controle de acesso ao meio (MAC) são uma subcamada da DLL (*data link layer*) que é a primeira camada de protocolo acima da camada física. Eles são responsáveis por prover endereçamento e acesso ao canal de comunicação. Segundo Karl e Willig (2005), a tarefa principal de qualquer protocolo MAC é regular o acesso de um número de nós a um meio de comunicação compartilhado, de tal modo que os requisitos de desempenho de cada aplicação sejam satisfeitos.

Segundo Ye, Heidemann e Estrin (2002), para desenvolver protocolos para redes de sensores, é preciso considerar como atributos principais a serem alcançados: eficiência no consumo de energia para aumentar a vida útil da rede, dado às dificuldades de substituição das baterias; escalabilidade para adaptação a mudanças no tamanho, densidade e topologia da rede, uma vez que alguns nós da rede irão “morrer” com o passar do tempo, novos nós podem ser inseridos na rede mais tarde ou ainda alguns nós podem se mover para locais diferentes. Outros atributos como justiça na distribuição de recursos (por exemplo o meio de comunicação), latência, e ritmo de transferência de dados que são tidos como principais em outros tipos de redes sem fio são aqui tidos como secundários.

#### 3.1 Padrões de comunicação em RSSFs

Segundo Demirkol, Ersoy e Alagoz (2006), os tipos de padrões de comunicação em RSSFs determinam o comportamento do tráfego de dados na redes de sensores que deve ser tratado pelos protocolos da camada MAC.

Kulkarni (2004) define três tipos de comunicação em redes de sensores sem fio: *broadcast*, *convergecast*, *local gossip*. O padrão *broadcast* é geralmente utilizados por estações base (*sink nodes*) para transmitir informações para todos

os nós da rede. Pacotes enviados em *broadcast* podem conter requisições por informações dos sensores, atualizações do programa para os nós sensores, pacotes de controle para todo o sistema, etc. *Local gossip* ocorre quando um nó sensor detecta um evento e o comunica a cada um de seus vizinhos localmente (dentro do alcance de seu radio transmissor). Após um grupo de nós detectar um evento, é necessário comunicá-lo ao centro de informações, esse padrão de comunicação é chamado *convergecast* (quando um grupo de sensores se comunica com um sensor específico).

### 3.2 Classificação de Protocolos de Controle de Acesso ao Meio

A cada dia surgem novas propostas de protocolos MAC que utilizam diferentes técnicas para alcançar melhores resultados em termos de consumo de energia, latência, qualidade de serviço, etc., de acordo com as necessidades de cada aplicação. Chandra *et al.* (2000) classifica os protocolos MAC nas seguintes classes: protocolos de atribuição fixa, protocolos de atribuição por demanda e protocolos de acesso aleatório.

#### 3.2.1 Protocolos de atribuição fixa

Cada nó recebe uma porção do recursos por um período de tempo considerado longo (da ordem de minutos, horas, ou mesmo maior) e pode utilizá-los de maneira exclusiva naquele período de tempo, ou seja, somente o nó ao qual um recurso foi atribuído pode acessar esse recurso no espaço de tempo que foi para ele reservado.

A atribuição desses recursos não é feita de maneira permanente devido à necessidade de adaptação a possíveis alterações na topologia da rede (quando ocorre a morte, inserção ou movimentação de nós na rede, por exemplo). Esse

tipo de atribuição feito por períodos consideravelmente longos pode causar efeitos negativos à característica de escalabilidade da rede.

Como exemplos desse tipo de protocolo tem-se o *Time Divided Multiple Access* (TDMA) onde o tempo de uso dos recursos é dividido em *slots* que são atribuídos aos nós da rede (KULKARNI, 2004), *Frequency Division Multiple Access* (FDMA) onde a banda de frequência é dividida em vários sub-canais (faixas de frequência) que são atribuídos aos nós da rede (ARMS *et al.*, 2004).

### 3.2.2 Protocolos de atribuição por demanda

Nesse tipo de protocolo, ao contrário dos protocolos de atribuição fixa, os recursos são distribuídos entre os nós por um período de tempo relativamente curto (da ordem de dezenas de milissegundos), geralmente o tempo necessário para o envio de um conjunto de pacotes de dados. Caso um nó necessite utilizar algum recurso, este deve fazer uma solicitação a um nó central que seja responsável por coordenar o uso de tal recurso, que por sua vez retorna a confirmação ou não da alocação do recurso e, em caso positivo, uma descrição do recurso alocado. Após essa negociação, o nó solicitante poderá utilizar o recurso de maneira exclusiva. A comunicação para requisição de um recurso é, em geral, feita através de protocolos de acesso aleatório em canais dedicados. Uma outra possibilidade, é deixar a estação central organizar os nós associados a ela.

Uma restrição no uso desse tipo de organização é que a estação central precisa estar ligada todo o tempo e, portanto, não pode ter restrições de energia. Esse tipo de protocolos pode ser ainda dividido em duas outras subcategorias: protocolos centralizados (onde exista apenas uma estação central) e protocolos distribuídos (onde exista várias estações centrais, cada uma responsável pela alocação dos recursos em uma determinada região da rede).



Como exemplo temos o *Predictive Demand Assignment Multiple Access* (PRDAMA), que aloca recursos de largura de banda através de estimativa da variação positiva do tráfego da internet para predeterminar a necessidade de recursos das estações (JIANG; LI; LEUNG, 2002).

### 3.2.3 Protocolos de acesso aleatório

Em protocolos desse tipo não há coordenação para o uso de recursos. Cada nó que necessite de um recurso (como o meio de comunicação, por exemplo), deve verificar se o recurso está disponível ou tentar acessá-lo diretamente podendo considerar algum parâmetro aleatório, por exemplo, após aguardar um tempo aleatório ou relacionado com o tempo de chegada de pacotes randômicos. Caso o recurso esteja indisponível ou inacessível repete-se o procedimento de acordo com novos valores aleatórios.

Um exemplo desse tipo de protocolo é o ALOHA, que permite que os nós enviem pacotes no momento em que existam pacotes a serem enviados. Em seguida eles devem aguardar a confirmação do recebimento dos pacotes e, em caso positivo enviar o pacote seguinte (caso haja) ou, no caso contrário, reenviar o pacote (BACCELLI; BŁASZCZYSZYN; MÜHLETHALER, 2006).

## 4 Auto Organização em RSSF

Auto organização é um processo pelo qual estrutura e funcionalidade (padrão) em nível global de um sistema emergem unicamente de numerosas interações entre componentes de baixo nível de um sistema sem qualquer controle externo ou centralizado. Os componentes do sistema interagem em um contexto local por meio de comunicação direta ou observações ambientais sem referência a um padrão global (DRESSLER, 2008).

### 4.1 Protocolos MAC auto organizáveis

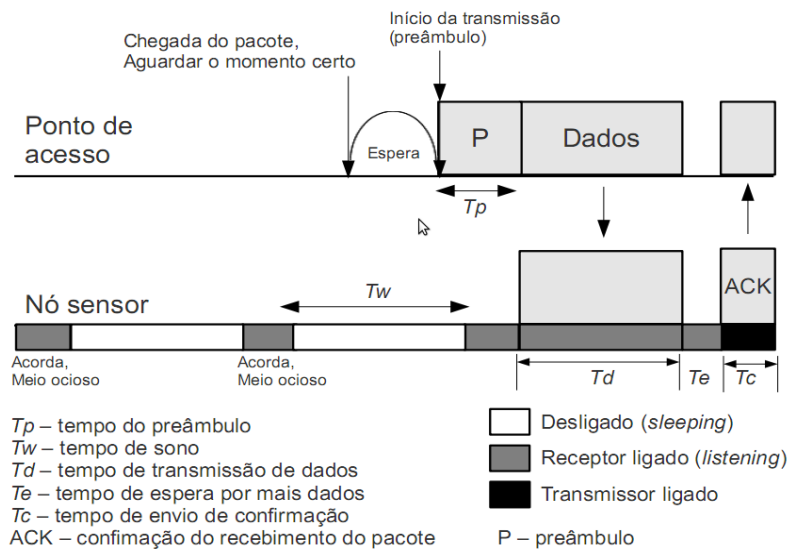
A utilização de auto organização em redes de sensores sem fio possibilita ganhos em certos atributos dessas redes, como escalabilidade, consumo de energia, tolerância a falhas, etc. Em protocolos de acesso ao meio, auto organização é utilizada em mecanismos como: atribuição de recursos aos nós (DRESSLER, 2008), determinação de ciclos de sono<sup>1</sup> (HALKES; DAM; LANGENDOEN, 2005), roteamento de pacotes, etc. Existem muitos exemplos de protocolos de controle de acesso ao meio que utilizam esse tipo de organização para diversos fins, abaixo estão apresentados alguns desses protocolos e o modo como a auto organização é empregada neles.

---

<sup>1</sup>Com o objetivo de aumentar o tempo de vida útil das RSSF são utilizados ciclos de sono (*sleep-wakeup cycle* ou *duty cycle*) onde cada nó é ligado apenas por um curto período de tempo (*wakeup*) e então desligado novamente (*sleep*). A utilização desses ciclos pode aumentar a vida útil das RSSF, devido à redução do consumo de energia durante o período de sono *sleep*, de poucos dias para vários meses ou mesmo anos dependendo da razão entre o tempo em que o nó permanece ligado e o tempo em que permanece desligado.

### 4.1.1 WiseMAC

WiseMAC é baseado em técnicas de amostragem de preâmbulo<sup>2</sup>. Através da verificação periódica do meio em busca de atividades de comunicação (EL-HOIYDI; DECOTIGNIE, 2004). Tal verificação consiste em uma varredura de curta duração do canal de rádio. Cada nó verifica o meio com um mesmo período constante  $T_w$  (Figura 3). Se o meio está ocupado, o nó sensor continua a ouvi-lo até receber um quadro de dados ou até o meio ficar desocupado novamente. Antes de cada quadro de dados ser enviado, o ponto de acesso (*access point*) envia um preâmbulo com a mesma duração do período de verificação para garantir que o receptor esteja acordado no momento em que o quadro de dados for enviado.



**Figura 3:** Funcionamento do WiseMAC (EL-HOIYDI; DECOTIGNIE, 2004).

<sup>2</sup>Preâmbulo é um fluxo de dados utilizado para comunicar a intenção de transmitir. Tem duração variável de acordo com o período de sono, pois é preciso garantir que o destinatário receba esse preâmbulo e identifique-se como receptor dos pacotes de dados que serão enviados em seguida.

Essa técnica promove um baixo consumo de energia quando o canal está desocupado, porém tem como desvantagem o fato de que os preâmbulos utilizados podem causar limitações à velocidade de transmissão entre pontos distantes e maior consumo de energia devido a sobrecarga nos receptores e ao fato de todos os nós permanecerem ouvindo essa transmissão (*overhearing*). Um ganho considerável é conseguido ao fazer com que os pontos de acesso conheçam os ciclos de verificação dos outros nós pois ele poderá iniciar transmissões somente no momento em que os outros nós estarão ouvindo.

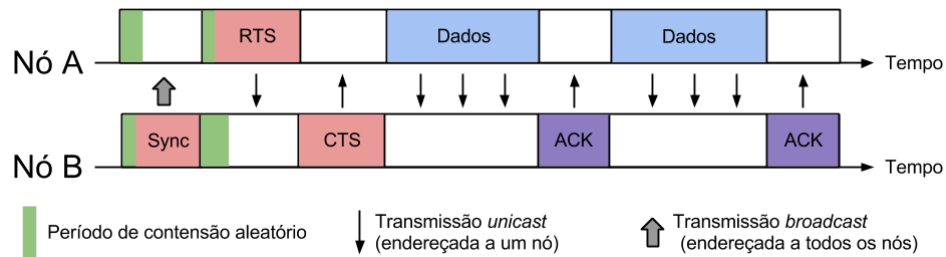
#### 4.1.2 S-MAC

Sensor-MAC ou S-MAC foi desenvolvido para ter primordialmente baixo consumo de energia, além disso conseguiu-se também boa escalabilidade e baixo número de colisões (YE; HEIDEMANN; ESTRIN, 2004). Para conseguir esse resultados foram tratados os seguintes problemas identificados pelo autor como responsáveis pelo consumo excessivo de energia na rede: colisões de pacotes, que causam a retransmissão desses pacotes e consequente consumo extra de energia além de aumentar a latência na rede; *overhearing*, que acontece quando os nós recebem pacotes que não são destinados a eles; excesso (*overhead*) de pacotes de controle; e *idle listening*, quando um nó permanece em modo de recepção aguardando por pacotes que não são enviados.

Para alcançar esses resultados foi adotado um padrão de ciclos de sono com períodos longos de inatividade (período de sono) e períodos de atividade (*duty cycle*) curtos. Auto organização está presente na sincronização dos períodos de atividade/sono dos nós em uma vizinhança, fazendo com que os nós adotem um mesmo cronograma de funcionamento.

No momento em que cada nó é ligado, ele permanecerá ouvindo o meio com o objetivo de verificar a existências de nós vizinhos que já possuam agendas determinadas. Para que seja possível essa detecção, no início de seu *duty cycle*, cada nó verifica o meio e, caso esteja desocupado, envia um pacote de sincronização (*sync packet*) contendo informações da sua agenda. Os receptores desse pacote podem então adotar essa agenda e passam a possuir o mesmo *duty cycle* do nó transmissor.

Para evitar colisões no envio desses pacotes, antes de verificar a disponibilidade do meio para envio do *sync packet* (Figura 4), cada nó aguarda por um período de tempo aleatório. Esse período de espera é chamado período de contenção.



**Figura 4:** Funcionamento do *duty cycle* no protocolo S-MAC.

A segunda parte do *duty cycle* é reservada para transmissão e recepção de dois outros pacotes de controle que determinam se um nó pode enviar ou receber pacotes de dados. O primeiro desses pacotes é o RTS (*Request To Send*). Um nó que deseja transmitir dados deve aguardar um novo período de contenção e então verificar novamente a disponibilidade do meio (Figura 4). Caso o meio esteja disponível, ele enviará ao nó para o qual deseja transmitir um pacote RTS. Após receber um pacote RTS o nó receptor poderá enviar um pacote CTS (*Clear To Send*) para notificar o transmissor que pode receber o pacote (Figura 4). O nó

não enviará o CTS nos casos onde já tenha ouvido um RTS ou CTS de um de seus vizinhos ou caso na sua vizinhança o meio esteja ocupado.

Após receber um CTS o nó transmissor aguarda o final do período de tempo reservado à transmissão dos pacotes de controle e então inicia a sua transmissão (Figura 4). Ao final da transmissão, o nó transmissor aguardará que seu receptor envie um pacote de confirmação de recebimento chamado ACK (*Acknowledgement*). Após receber esse pacote, ele poderá continuar enviando pacotes de dados da mesma maneira. Caso o nó transmissor não receba um ACK, ele tentará reenviá-lo. Caso o nó não obtenha sucesso no envio de um pacote após algumas tentativas, o pacote poderá ser descartado.

Todo esse mecanismo possibilita que eles possam se comunicar com eficiência e se manterem sincronizados apesar do maior tempo de inatividade. Além disso, elimina a necessidade do uso de preâmbulos pois tem-se uma garantia de que os nós aos quais os dados se destinam estarão acordados no mesmo momento (ou em um momento muito próximo) em que aquele que deseja transmitir.



**Figura 5:** Exemplo de sincronização de ciclos de atividades com protocolo S-MAC (*clusters A e B e nós de borda*).

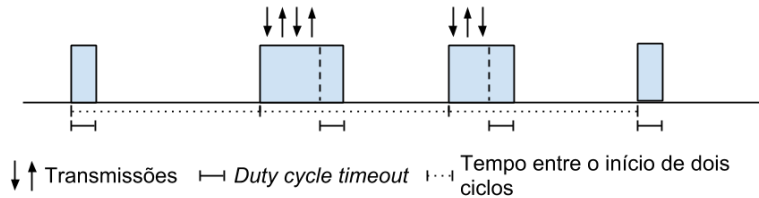
Como desvantagem desse protocolo pode-se destacar que a sincronização dos ciclos de sono se dá entre um certo número de nós de uma região da rede e os nós que ficam no limiar entre duas regiões devem sincronizar-se com ambas as regiões (Figura 5). Isso faz com que o consumo de energia desses nós seja maior, reduzindo-se assim o seu tempo de vida útil.

#### 4.1.3 T-MAC

Dam e Langendoen (2003) apresentam o protocolo T-MAC. O T-MAC é um protocolo de controle de acesso ao meio baseado em contensão e, assim como o S-MAC, também utiliza *duty cycles* e sincronização de nós. O que o diferencia do S-MAC é que o *duty cycle* dos nós é adaptado de acordo com o tráfego da rede de sensores. O fim do período de atividades é determinado por um *timeout*. Esse *timeout* é configurado para que seja suficiente para a realização de um curto período de contenção e uma troca de pacotes RTS e CTS.

Ao final desse período, caso o nó não tenha detectado nenhuma atividade no meio ele terminará seu *duty cycle* e voltará a dormir. Caso o nó receba pacotes ou detecte o envio de pacotes entre seus vizinhos, ele irá iniciar um novo *timeout* após o final desta comunicação (Figura 6).

Um problema encontrado nesse protocolo é que um nó que deseja transmitir um pacote, mas perde a contenção para um vizinho não comum, permanecerá em espera até o final da transmissão desse vizinho. Ao final dessa transmissão, ele iniciará um novo *timeout* e enviará um CTS, mas não receberá a resposta do nó destino, uma vez que ele não terá detectado a atividade do nó que venceu a contensão anterior.



**Figura 6:** *Duty cycle* adaptativo no protocolo T-MAC.

#### 4.1.4 D-MAC

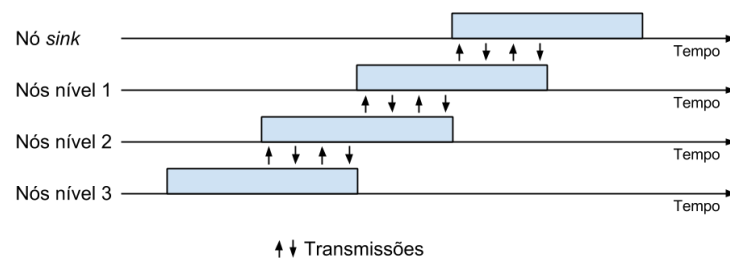
O protocolo D-MAC, apresentado em Lu, Krishnamachari e Raghavendra (2004), também utiliza *duty cycles* para reduzir o consumo de energia dos nós e aumentar o tempo de vida da rede. O que o diferencia dos outros protocolos apresentados é o modo como é feita a sincronização dos nós.

Para o funcionamento desse protocolo é preciso que haja um nó concentrador (*sync node*) conhecido. Assumindo-se que a maior parte da comunicação dentro da rede será realizada em direção a esse nó concentrador, os nós são sincronizados a partir dele de forma que os nós que estejam a um número  $n$  de saltos do nó concentrador iniciam seu *duty cycle* um pouco depois dos nós que estão a uma distância em saltos igual a  $n + 1$  (Figura 7).

Essa sincronização permite que transmissões de pacotes em direção ao nó concentrador partindo de qualquer ponto da rede tenham sua latência bastante reduzida. Cada pacote recebido por um nó do nível  $n$  pode ser enviado para um nó do nível  $n - 1$  quase imediatamente.

O problema com esse protocolo é que ele só promove redução da latência de transmissão no sentido do gradiente em direção ao nó concentrador. Caso haja transmissões seguindo a direção contrária (como pacotes de requisição de dados), não haverá redução na latência dessas transmissões.





**Figura 7:** *Duty cycle* sincronizado em níveis no protocolo D-MAC.

## 5 Metodologia

Este trabalho pode ser classificado como pesquisa aplicada por tratar com problemas reais. Espera-se como resultado um protocolo que reduza o tempo de transmissão de informação em redes de sensores sem fio sem redução significativa da vida útil da rede.

É também pesquisa exploratória, pois parte da solução proposta advém da utilização de algoritmos existente para solução de alguns dos problemas (auto organização na formação de estruturas dentro da rede). Possui também características que permitem defini-la como experimental, pois serão utilizadas simulações para verificação e avaliação do protocolo proposto.

Além disso, se caracteriza ainda como pesquisa quantitativa pela utilização de testes, simulações cujos dados quantitativos resultantes serão avaliados com o objetivo de determinar a eficiência do protocolo proposto nesse trabalho.

## 6 *Configurable Sensor MAC* ou CS-MAC

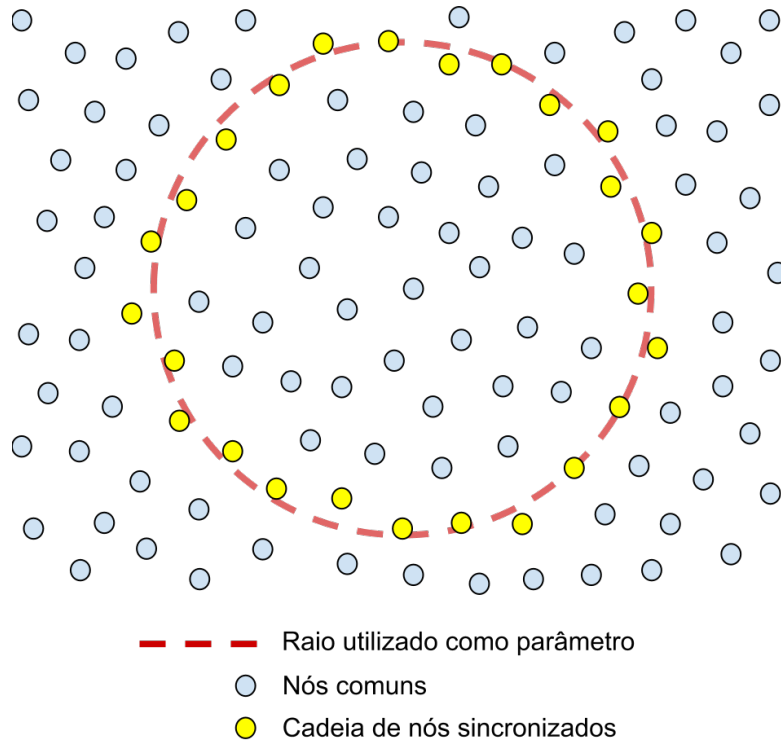
O algoritmo desenvolvido nesse trabalho foi batizado de *Configurable Sensor MAC* ou *Sensor-MAC* configurável pois tem como base o algoritmo S-MAC descrito em Ye, Heidemann e Estrin (2004). Conforme apresentado na seção 4.1.2, o protocolo S-MAC faz com que cada nó em uma vizinhança siga um mesmo cronograma que determina seus períodos de atividade e inatividade. Com isso, cada nó que deseja transmitir pacotes poderá fazê-lo no momento em que iniciar seu ciclo de atividades sem a necessidade do uso de longos preâmbulos pois o nó receptor iniciará seu ciclo de atividades no mesmo momento.

O protocolo CS-MAC possui os mesmos mecanismos de funcionamento do protocolo S-MAC, podendo ser então considerado uma extensão desse protocolo, pois adiciona a ele mecanismos que permitem que seu funcionamento e configurações sejam alterados por outras camadas da pilha de protocolo, no caso deste trabalho, pela camada de aplicação (Seção 2.2.5).

Basicamente, os mecanismos de configuração implementados no protocolo CS-MAC permitem que outra camada possa determinar que o nó siga um cronograma de funcionamento arbitrário especificado por ela com qualquer finalidade. Para o teste dos mecanismos implementados no protocolo CS-MAC, foram implementados na camada de aplicação algoritmos capazes de construir uma “via de acesso rápido” (*backbone*), por onde os pacotes poderiam trafegar mais rapidamente.

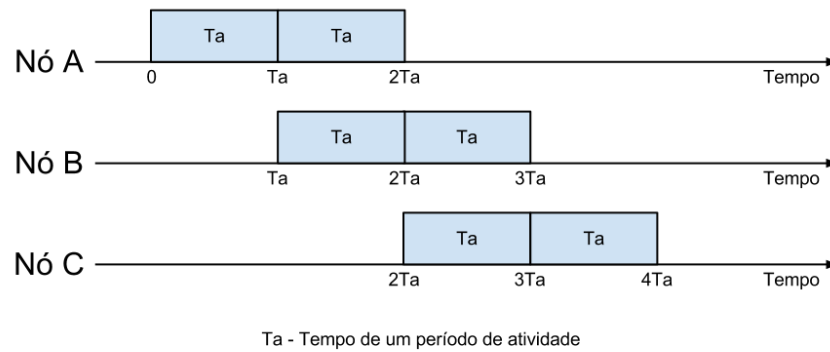
De forma simplificada, os algoritmos implementados são capazes de determinar que uma cadeia de nós situados a uma determinada distância do centro da rede (Figura 8) de sensores seja sincronizada de uma maneira tal que, dados os nós A, B e C pertencentes a estrutura, o tempo de transmissão de um pacote de A para C passando por B seria de aproximadamente  $2 \cdot T_A$  ( $1 \cdot T_A$  = transmissão de A

para B,  $1 \cdot T_A = \text{transmissão B para C}$ ) sendo  $T_A$  a duração do período de atividade de um nó.



**Figura 8:** Cadeia de nós selecionados para sincronização.

Deve-se notar que não há intervalo entre as duas transmissões. Para que isso seja possível foi determinado que cada nó siga dois cronogramas de atividades distintos sendo que o primeiro cronograma determina um período de atividade no instante  $X$  e o segundo determina um período de atividade no instante  $X + T_A$ . Além disso, o primeiro ciclo de atividade de cada nó que participa da cadeia descrita acima deve coincidir com o segundo ciclo de atividades de seu antecessor. A Figura 9 abaixo mostra como estão sincronizados os nós do exemplo anterior.



**Figura 9:** Cronogramas de atividade sincronizados sequencialmente.

Os capítulos seguintes apresentarão uma descrição mais pormenorizada dos algoritmos utilizados para produzir a organização descrita. Ao final serão também apresentados e discutidos os resultados de simulações realizadas para avaliação dos impactos dessa organização no tempo de transmissão de pacotes entre pontos distantes na rede de sensores.

## 7 Algoritmos

Esse capítulo apresentará os algoritmos utilizados para sincronização dos nós da rede, construção da cadeia de nós que funcionará como via de transmissão rápida de pacotes e demais algoritmos utilizados. A seção seguinte apresenta mais detalhes acerca do funcionamento do protocolo S-MAC (protocolo selecionado para a sincronização dos nós) e razões para a escolha do mesmo, bem como das alterações propostas para a construção do protocolo CS-MAC.

### 7.1 Sincronização dos nós

Como o objetivo do trabalho era o desenvolvimento de um protocolo de comunicação capaz de reduzir o tempo de transmissão de pacotes entre nós distantes na rede e ao mesmo tempo manter o consumo de energia no menor patamar possível, o protocolo S-MAC, citado anteriormente (Seção 4.1.2) e encontrado em Ye, Heidemann e Estrin (2004), foi selecionado como base para a construção do novo algoritmo. Esse algoritmo tem como foco o consumo reduzido de energia, além de possuir mecanismos eficientes para redução de perda de pacotes e solução de outros problemas inerentes às RSSFs. Porém, a latência entre nós distantes pode ser muito grande devido ao tamanho dos períodos de dormência dos nós em relação aos curtos períodos de atividade.

No protocolo S-MAC, cada nó possui pelo menos uma agenda que determina os momentos em que o rádio-transmissor do nó será ligado (quando o nó poderá enviar e receber pacotes) ou desligado. Essa agenda é compartilhada entre nós vizinhos, podendo haver regiões que possuam agendas distintas chamadas *clusters* (Figura 5). Para que seja possível a comunicação entre essas regiões distintas, os nós que estejam no limiar dessas regiões compartilham ambas as agendas, dessa forma podendo enviar e receber pacotes para outros nós de ambas as regiões.

O mecanismo de criação e disseminação de agendas no protocolo S-MAC funciona de maneira auto-organizada, não necessitando de interferência de agente externo. O novo protocolo desenvolvido neste trabalho, assim chamado CS-MAC, propõe então uma extensão ao protocolo S-MAC de modo a disponibilizar mecanismos que permitam a configuração de agendas para os nós de maneira arbitrária por outras camadas da pilha de protocolos. Foi desenvolvido também neste trabalho um algoritmo para a camada de aplicação que irá determinar, através desse mecanismo de configuração, que certos nós sigam agendas específicas que farão com que possuam ciclos de atividade com intervalos muito pequenos entre si (Figura 9), de forma a construir uma cadeia de nós que acordem sequencialmente onde, conseqüentemente, a transmissão de pacotes por múltiplos nós ocorrerá de maneira muito mais rápida.

Para possibilitar um melhor entendimento do protocolo S-MAC e do protocolo proposto, a seção seguinte irá apresentar mais detalhes do funcionamento de ambos.

## **7.2 Funcionamento do protocolo S-MAC**

A partir do momento em que os nós que utilizam o protocolo S-MAC são ligados, eles realizam as seguintes operações:

- “Ouvir” um ciclo inteiro esperando receber uma agenda que possa seguir.
- Caso não receba uma agenda, fará um teste utilizando um valor aleatório para determinar se criará sua própria agenda ou continuará a “ouvir” o meio por mais um ciclo.
- Tendo criado ou recebido uma agenda, irá configurar-se para acordar e dormir de acordo com essa agenda.

A partir do momento em que um nó adquire uma agenda seus ciclos de atividade e sono passarão a ser executados repetidamente. Os pacotes de controle e o funcionamento do *duty cycle* estão descritos mais detalhadamente na seção 4.1.2. Informações complementares e mais detalhadas sobre o funcionamento do protocolo S-MAC podem ser encontrados em Ye, Heidemann e Estrin (2004).

Além dos mecanismos providos pelo protocolo S-MAC, o novo algoritmo adiciona uma *API* (*Application Program Interface*) para que outras camadas da pilha de protocolo possam determinar novas agendas para o nó. Por meio desta, diferentes tipos de configurações para a sincronização entre os nós podem ser determinadas, por exemplo, pela camada de aplicação.

Neste trabalho, o algoritmo desenvolvido utiliza agentes que percorrem a rede em busca de nós que cumpram determinados requisitos, de modo a formarem uma cadeia circular (Figura 8). Para cada um dos nós selecionados é determinado pelos agentes que eles devem seguir duas novas agendas uma compartilhada com seu antecessor outra com seu sucessor na estrutura criada (Figura 9) e essas agendas tem um intervalo muito pequeno entre si, suficiente para uma transmissão de pacotes entre os nós. Como resultado do trabalho dos agentes, os nós da cadeia circular estarão sincronizados sequencialmente. Transmissões de pacotes entre nós distantes na rede deverão percorrer essa estrutura visando reduzir o tempo total de transmissão do pacote do nó de origem ao nó destino. As seções seguintes discutirão os algoritmos utilizados para configuração da rede e disseminação dos agentes.

### 7.3 Cálculo da distância ao centro da rede

Para se determinar a distância de cada nó ao centro da rede é necessário conhecer a sua posição. A maneira mais simples é a utilização de nós que pos-



suam sistema de GPS (*Global Positioning System* ou Sistema de Posicionamento Global). Tal sistema permite determinar com precisão a posição de cada nó. Bastaria então que os agentes conhecessem as coordenadas de um ponto no centro da rede para que pudessem determinar se um nó satisfaz ou não aos parâmetros que determinam o raio da estrutura circular a ser construída.

Para a utilização de sistema de posicionamento global (GPS) é necessário que cada nó possua *hardware* específico para tal, o que aumenta o seu custo e consumo de energia. Como alternativa ao uso desse sistema existem vários algoritmos de localização propostos que se utilizam de cálculos para determinar a posição de cada nó em relação à rede como um todo.

A abordagem da maioria dos algoritmos de localização existentes consiste de duas fases: estimação de distâncias (ou ângulos) e combinação de distâncias (ou ângulos). Esse cálculo pode ser realizado através de técnicas como o Indicador de Força de Sinal Recebido (*Received Signal Strength Indicator*), estimação de Ângulo de Chegada (*Angle-of-Arrival estimation*), Trilateração Hiperbólica (*Hyperbolic trilateration*), dentre outras (PAL, 2010).

Um maior aprofundamento foge ao escopo deste trabalho, maiores informações do estado da arte e uma análise detalhada de técnicas de localização em RSSF podem ser encontradas em Pal (2010).

As seções seguintes apresentarão os algoritmos utilizados para a construção da via de transmissão rápida de pacotes (*backbone*). Para a implementação dos algoritmos propostos neste trabalho assume-se que os nós conheçam suas posições e de todo os outros nós da rede de sensores.

## 7.4 Construção do círculo

Dado que a distância de cada nó em relação ao centro da rede pode ser obtida (conforme apresentado na seção 7.3), foram utilizados três agentes para a construção da estrutura circular:

- **Root finder:** responsável por percorrer a rede em busca de algum nó que cumpra o requisito de distância do centro da rede (raio do círculo formado). O nó selecionado por esse agente será o nó raiz na estrutura a ser formada. Após encontrar o nó raiz, esse agente irá criar o próximo agente.

---

**Algoritmo 1:** Algoritmo utilizado para encontrar o nó raiz e determinar a criação do agente *Circle Builder*.

---

**Entrada:** Posição central da rede, Posição do nó corrente, Raio do backbone, Variabilidade

mínimo = Raio do backbone - Variabilidade;

máximo = Raio do backbone + Variabilidade;

**se** *mínimo*  $\leq$  *distância do nó ao centro*  $\leq$  *máximo* **então**

    (nó corrente);

    circleBuilder = ();

    circleBuilder.configurarAgendasDeBackbone (nó corrente);

    destino = encontreVizinhoMaisProximoDoRaio ();

    encaminhe (circleBuilder, destino);

**senão**

    destino = encontreVizinhoMaisProximoDoRaio ();

    encaminhe (rootFinder, destino);

---

- **Circle builder:** responsável por criar as agendas para cada nó (incluindo o nó raiz), determinar qual vai ser o próximo nó que fará parte da estrutura. Ele

selecionará vários nós (conforme os mesmos parâmetros do nó anterior) até alcançar algum que já faça parte da estrutura (fechando o círculo), quando então ele criará o próximo agente.

---

**Algoritmo 2:** Algoritmo utilizado para criar o círculo e determinar a criação do agente *Circle Revoker*.

---

**Entrada:** Agente CircleBuilder

circleBuilder.configurarAgendasDeBackbone (nó corrente);

**se** nó corrente *NÃO* pertence ao backbone **então**

```

    guardarAntecessor (circleBuilder.noQueEnviou ());
    destino = encontreVizinhoMaisProximoDoRaio ();
    encaminhe (circleBuilder, destino);

```

**senão**

```

    destino = antecessorCorrente;
    guardarAntecessor (circleBuilder.noQueEnviou ());
    se destino NÃO é nulo então
        circleRevoker = criarAgenteCircleRevoker ();
        encaminhe (circleRevoker, destino);

```

---

- **Circle revoker:** caso o último nó encontrado pelo agente *Circle builder* não coincida com o nó raiz encontrado pelo agente *Root finder*, esse agente será responsável por remover as agendas extras dos nós que foram selecionado, mas não fazem parte do círculo formado (“pontas soltas”), fazendo com que eles voltem a seguir apenas suas agendas originais.

---

**Algoritmo 3:** Algoritmo utilizado para eliminar os nós deverão deixar de ser parte do *backbone*.

---

**Entrada:** Agente CircleRevoker

circleRevoker.removeAgendasDeBackbone (nó corrente);

destino = antecessorCorrente;

**se destino NÃO é nulo então**

    encaminhe (circleRevoker, destino);

---

## 7.5 Roteamento dos pacotes

Para que a configuração circular formada dentro da rede pelos algoritmos descritos nas seções anteriores produza efeitos sobre o tempo das transmissões de pacotes dentro da rede, é preciso rotear os pacotes de forma que, caso seja possível, eles sejam encaminhados através da estrutura formada até um ponto próximo do nó destino, quando o pacote deixará a estrutura e será encaminhado ao seu destino através dos mecanismos normais.

Existem diversos algoritmos propostos para roteamento de pacotes em RSSF. Al-karaki e Kamal (2004) apresenta o estado da arte desses protocolos. Como neste trabalho assume-se que os nós conheçam suas posições e posições de todos os outros nós, algoritmos de roteamento geográficos seriam a opção mais adequada devido à maior simplicidade e eficiência dos mesmos (PAL, 2010).

A implementação de um algoritmo de roteamento de pacotes foge ao escopo desse trabalho. Portanto, com o objetivo de avaliar o desempenho do algoritmo proposto, o roteamento foi feito pelo gerenciador da simulação de forma a simular um algoritmo de roteamento geográfico que seguisse os seguintes critérios:

- Caso haja um nó vizinho ao nó transmissor que pertencente à estrutura formada e que esteja mais próximo do nó de destino do pacote do que o transmissor, esse nó será selecionado
- Caso o critério anterior não seja cumprido, será selecionado como receptor, aleatoriamente, um dos três nós vizinhos mais próximos do nó de destino

Utilizando esses critérios foi possível simular um algoritmo de roteamento capaz de fazer com que os pacotes sejam transmitidos através da estrutura circular formada, possibilitando assim a avaliação de seu desempenho. O algoritmo utilizado é apresentado abaixo em pseudo-código:

---

**Algoritmo 4:** Algoritmo utilizado para roteamento de pacotes.

---

*se existe algum vizinho que é parte do backbone então*

*se esse vizinho está mais próximo do destino então*

        | **retorna** [*esse vizinho*]

*candidatos = encontreOsTresVizinhosMaisProximos ();*

*vizinhoSelecionado = selecioneAleatoriamente (candidatos);*

**retorna** [*vizinhoSelecionado*]

---

## 8 Resultados Obtidos

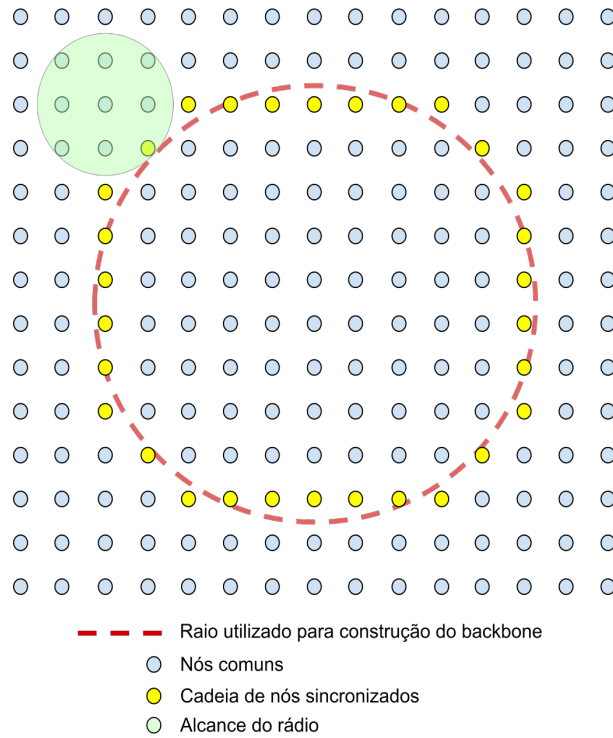
Nesta seção serão apresentados e discutidos os resultados dos testes de desempenho do algoritmo proposto nesse trabalho, assim chamado CS-MAC, ao utilizar-se uma estrutura circular – construída utilizando-se os algoritmos descritos na seção 7.4 – como via de transmissão rápida de pacotes. Os dados obtidos nessas execuções foram comparados aos obtidos na utilização do algoritmo S-MAC encontrado em Ye, Heidemann e Estrin (2004) e apresentado anteriormente (seção 4.1.2).

### 8.1 Conformação da Rede e Testes

Com o objetivo de garantir uma precisão maior dos resultados obtidos nos testes, a rede utilizada foi constituída de nós sensores uniformemente distribuídos em uma região retangular de forma que cada nó alcançasse somente um nó em cada direção, conforme apresentado na Figura 10. Além disso, todos os nós da rede foram forçados a seguir uma única agenda (exceto no caso dos nós pertencentes ao *backbone*). Essa última medida foi tomada com o objetivo de evitar variações nos resultados das execuções decorrentes da diferença de tempo entre os ciclos de atividades dos nós pertencentes a diferentes *clusters*, uma vez que o mecanismo de construção de *clusters* é imprevisível.

As métricas utilizadas na comparação dos algoritmos foram as seguintes:

- **Número de saltos:** número total de saltos de um pacote desde a origem até o destino
- **Latência:** tempo total de transmissão de um pacote desde a origem até o destino



**Figura 10:** Rede utilizada nas simulações.

## 8.2 Parâmetros utilizados

Para a simulação e obtenção dos dados para comparação dos algoritmos, a rede utilizada foi uma rede homogênea composta por 210 nós estacionários, distribuídos uniformemente em uma região retangular plana, como mostra a Figura 10. Foram executadas 100 simulações de envios de pacote para cada protocolo avaliado. Em cada uma das simulações, foi selecionado como nó emissor (criador do pacote) um dos nós da borda da rede. Esse critério foi utilizado com o objetivo de garantir que nós que estivessem dentro do raio do *backbone* não fossem selecionados como emissores, pois as transmissões originadas nessa região dificilmente

passariam pelo backbone e seriam, portanto, de pouca relevância para a análise dos algoritmos. Os nós destino dos pacotes foram selecionados aleatoriamente.

Os 100 pares origem/destino utilizados nas simulações do protocolo CS-MAC foram replicados para as simulações do protocolo S-MAC, com o objetivo de garantir maior precisão e confiabilidade dos resultados obtidos na comparação dos protocolos. Além disso, cada uma das transmissões foi realizada de maneira isolada, isto é, a partir do momento em que um pacote foi criado e enviado, um novo pacote só seria criado após este ter alcançado seu destino, evitando assim interferências<sup>3</sup>. As agendas utilizadas para as simulações de ambos os protocolos eram idênticas e determinavam um *duty cycle* com duração de 1 segundo e 9 segundos de dormência (*duty cycle* de 10%). No caso do *backbone*, o segundo período de atividade<sup>4</sup> (Figura 9) de cada nó era iniciado imediatamente após o término do primeiro, coincidindo com primeiro período de atividades do nó seguinte.

### 8.3 Análise de desempenho

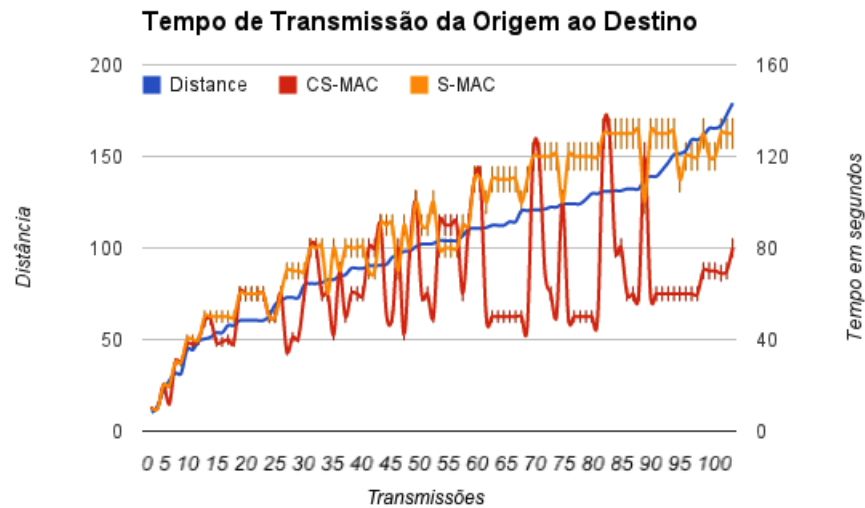
A Figura 11 mostra os resultados obtidos com relação ao tempo de transmissão de pacotes em ambos os protocolos. No eixo horizontal os resultados das simulações foram dispostos ordenadamente de acordo com a distância real (em linha reta) do nó origem até o nó destino. O parâmetro distância foi adicionado para facilitar a visualização dos resultados e os valores para este parâmetro estão representados no eixo vertical esquerdo. No eixo vertical direito estão representados os valores referentes ao tempo total de transmissão de cada pacote desde a origem até o seu destino.

---

<sup>3</sup>Apenas os pacotes da camada de aplicação foram isolados. Os pacotes de controle dos protocolos avaliados e das outras camadas continuaram a ser transmitidos normalmente

<sup>4</sup>Referentes às agendas exclusivas do *backbone*

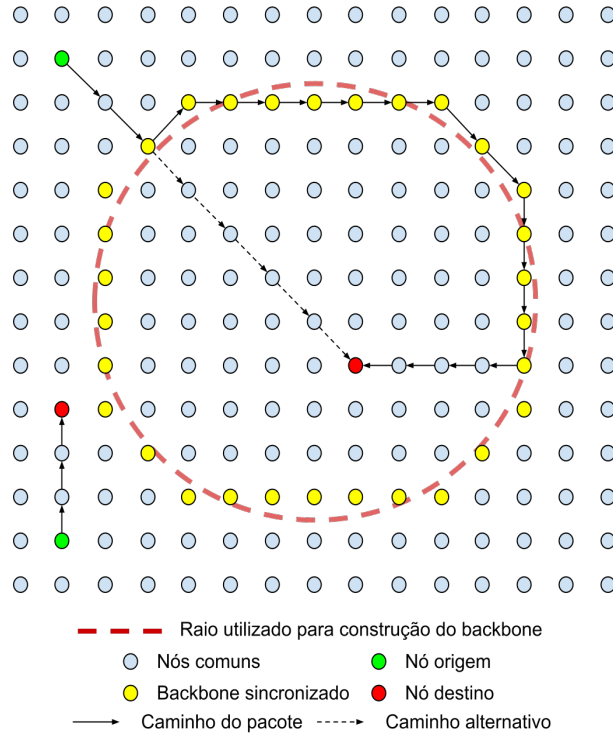




**Figura 11:** Comparação entre os tempos totais de transmissão de pacotes em cada protocolo.

Pode-se verificar que o tempo de transmissão em cada protocolo se diferencia sensivelmente na maior parte dos casos, sobretudo à medida que a distância entre os nós aumenta. Entretanto, em alguns casos onde a distância é muito curta, os valores coincidiram devido ao caminho percorrido pelo pacote não ter incluído os nós da via de transmissão rápida (Figura 12). Pode-se notar também que houve alguns casos onde, mesmo à grande distância, os valores coincidiram. Isso ocorre pois o número de saltos necessários para se alcançar o nó destino passando pelo backbone foi bem maior do que sem a sua utilização, havendo inclusive alguns casos onde o tempo de transmissão via *backbone* foi maior. Esses casos provavelmente aconteceram quando o nó destino se encontrava na região interna ao círculo do *backbone* (Figura 12).

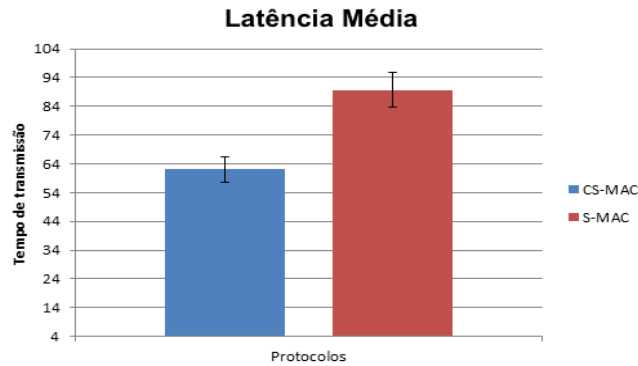
A Figura 14 mostra a comparação entre o número de saltos utilizados para transmissão de cada pacote, do nó origem até o nó destino. Os dados foram orde-



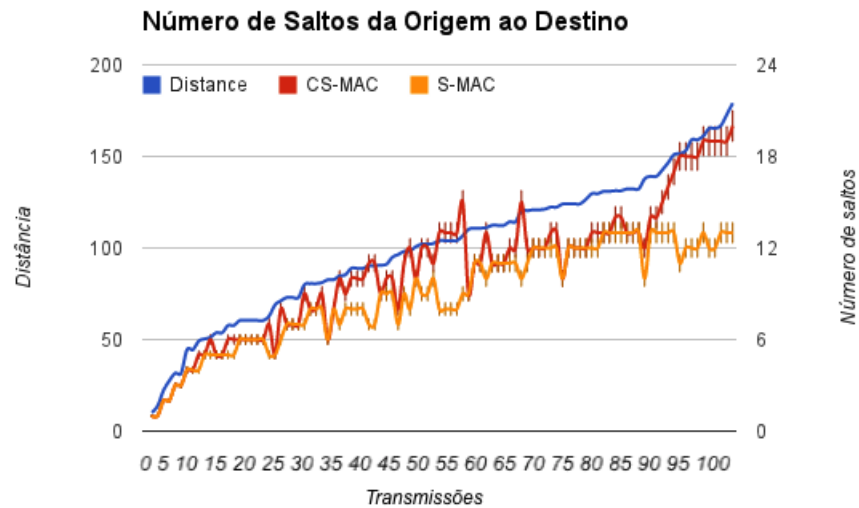
**Figura 12:** Casos onde não há ganho de desempenho com o protocolo CS-MAC.

nados de acordo com a distância real (em linha reta) do nó origem até o nó destino, para facilitar a visualização. Pode-se observar que o número de transmissões aumentou consideravelmente (17,3% em média) ao se utilizar o protocolo CS-MAC. Apesar disso, como observado anteriormente, o ganho de tempo ao se utilizar a via de transmissão rápida foi significativo (25% em média). A Figura 13 mostra uma comparação entre os tempos médios de transmissão de pacotes para cada protocolo.

O impacto negativo desse aumento no número de transmissões é o aumento no consumo de energia pelos nós integrantes do *backbone*. A comparação do número médio de saltos em cada protocolo é apresentada na Figura 15.



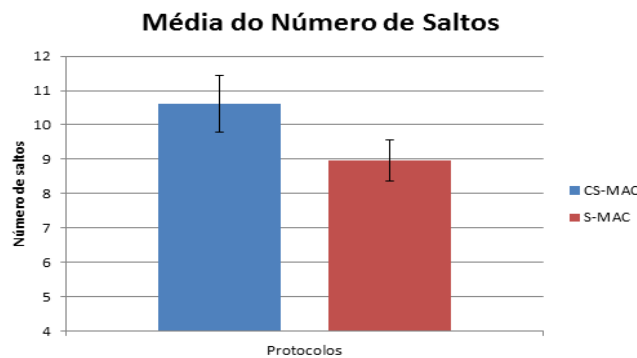
**Figura 13:** Comparação entre os tempos de transmissão médio dos protocolos.



**Figura 14:** Comparação entre os números de saltos necessários para completar as transmissões em cada protocolo.

Nota-se que a variação não é muito grande, mas o impacto no consumo de energia é relevante principalmente porque a maior parte das transmissões são realizadas pelos nós participantes do *backbone*. Entretanto, o ganho de velocidade de transmissão ao se utilizar o *backbone* seria proporcionalmente maior se o ta-

manho do período de inatividade fosse aumentado. Sendo assim, seria possível compensar esse gasto extra de energia ao se aumentar o tempo entre o início de um ciclo de atividades e o início do ciclo seguinte, sem abrir mão da redução no tempo de transmissão médio. Esse aumento no período de inatividade poderia ser ainda maior nas agendas que determinam o *backbone*, uma vez que a velocidade de transmissão não seria alterada significativamente, pois o impacto disso ocorreria apenas antes da primeira transmissão dentro do *backbone*.



**Figura 15:** Comparação do número médio de saltos em transmissões.

## 9 Conclusão

Foi proposto e desenvolvido neste trabalho um protocolo para a camada de controle de acesso ao meio em redes de sensores sem fio denominado CS-MAC, assim chamado por se tratar de uma extensão do protocolo existente S-MAC.

O protocolo CS-MAC foi desenvolvido com o objetivo de promover uma redução na latência em transmissões entre nós distantes na rede de sensores em relação ao protocolo original S-MAC e se mostrou eficiente conforme se pode comprovar a partir da análise dos resultados obtidos através dos experimentos computacionais realizados.

O trabalho demonstrou que o protocolo proposto conseguiu alcançar uma redução significativa na latência em relação ao protocolo S-MAC, satisfazendo as expectativas iniciais.

O CS-MAC apresentou uma redução média da latência em torno de 25% em relação ao S-MAC, chegando a mais de 58% nos casos de melhor desempenho. O protocolo produziu um efeito negativo ao aumentar o número de transmissões realizadas em 17,3% em média, porém demonstrou-se que esse efeito poderia ser amenizado através de alterações nas configurações do protocolo.

Para trabalhos futuros podem ser considerados:

- **Estruturas diferenciadas de *backbone*:** construção de vias rápidas de transmissão com estruturas diferentes para avaliação de desempenho e capacidade de atender a demandas diferenciadas em relação ao fluxo de pacotes.
- **Uso de agendas diferenciadas:** avaliação do uso de agendas com diferentes períodos de atividade e inatividade com o objetivo de avaliar o impacto no consumo de energia.

- **Uso de protocolos diferenciados:** utilização de protocolos diferentes para a sincronização dos nós fora do backbone e avaliação de seus impactos.

## Referências

AKYILDIZ, I. F.; SU, W.; SANKARASUBRAMANIAM, Y.; CAYIRCI, E. A survey on sensor networks. *Communications Magazine, IEEE*, 2002. v. 40, n. 8, p. 102–114, November 2002. Disponível em: <<http://dx.doi.org/10.1109/MCOM-2002.1024422>>.

AKYILDIZ, I. F.; SU, W.; SANKARASUBRAMANIAM, Y.; CAYIRCI, E. Wireless sensor networks: a survey. *Computer Networks*, 2002. v. 38, p. 393–422, 2002. Survey. Disponível em: <<http://impact.asu.edu/~mcn/papers/sensorpapers-wsn>>.

AL-KARAKI, J. N.; KAMAL, A. E. Routing techniques in wireless sensor networks: A survey. *IEEE Wireless Communications*, 2004. v. 11, p. 6–28, 2004.

ARMS, S. W.; TOWNSEND, C. P.; CHURCHILL, D. L.; HAMEL, M. J.; GALBREATH, J. H.; MUNDELL, S. W. *Frequency Agile Wireless Sensor Networks*. 2004.

BACCELLI, F.; BŁASZCZYSZYN, B.; MÜHLETHALER, P. An aloha protocol for multihop mobile wireless networks. *IEEE Transactions on Information Theory*, 2006. v. 52, p. 421–436, 2006.

CHANDRA, A.; GUMMALLA, V.; JOHN, O.; LIMB, O.; INSTITUTE, G.; TECHNOLOGY, O. *2 WIRELESS MEDIUM ACCESS CONTROL PROTOCOLS*. 2000.

DAM, T. van; LANGENDOEN, K. An adaptive energy-efficient mac protocol for wireless sensor networks. In: *Proceedings of the 1st international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2003. (SenSys '03), p. 171–180. ISBN 1-58113-707-9. Disponível em: <<http://doi.acm.org/10.1145/958491.958512>>.

DEMIRKOL, I.; ERSOY, C.; ALAGOZ, F. Mac protocols for wireless sensor networks: a survey. *Communications Magazine, IEEE*, 2006. v. 44, n. 4, p. 115–121, 2006. Disponível em: <[http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=1632658](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1632658)>.

DRESSLER, F. A study of self-organization mechanisms in ad hoc and sensor networks. *Comput. Commun.*, 2008. Butterworth-Heinemann, Newton, MA, USA, v. 31, n. 13, p. 3018–3029, 2008. ISSN 0140-3664.

EL-HOIYDI, A.; DECOTIGNIE, J.-D. Wisemac: an ultra low power mac protocol for the downlink of infrastructure wireless sensor networks. *Computers and Communications, IEEE Symposium on*, 2004. IEEE Computer Society, Los Alamitos, CA, USA, v. 1, p. 244–251, 2004.

GÜRSES, E. *Network Layer in Wireless Sensor Networks*. 3 2005.

HALKES, G.; DAM, T. van; LANGENDOEN, K. Comparing energy-saving MAC protocols for wireless sensor networks. *Special issue of Kluwer MONET on WLAN Optimization at the MAC and Network Levels*, 2005. v. 10, n. 5, p. 783–791, oct 2005. ISSN 1383-469X (Paper) 1572-8153 (Online). Disponível em: <<http://www.st.ewi.tudelft.nl/~koen/papers/mac-comp.pdf>>.

*Optimal design of fault tolerant sensor networks*. 467–472 p.

JIANG, Z.; LI, Y.; LEUNG, V. C. *A Predictive Demand Assignment Multiple Access Protocol for Broadband Satellite Networks Supporting Internet Applications*. 2002.

KARL, H.; WILLIG, A. *Protocols and Architectures for Wireless Sensor Networks*. [S.l.]: John Wiley & Sons, 2005. ISBN 0470095105.

KULKARNI, S. S. Tdma services for sensor networks. In: *Proc. 24th Int'l. Conf. Distrib. Comp. Sys. Wksp.*, Mar. [S.l.: s.n.], 2004. p. pp. 604–09.

KUMAR, S.; RAGHAVAN, V. S.; DENG, J. Medium access control protocols for ad hoc wireless networks: a survey. *Journal*, 2006. v. 4, p. 326–358, 2006.

LU, G.; KRISHNAMACHARI, B.; RAGHAVENDRA, C. An adaptive energy-efficient and low-latency mac for data gathering in wireless sensor networks. In: *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*. [S.l.: s.n.], 2004. p. 224–.

MAINWARING, A.; CULLER, D.; POLASTRE, J.; SZEWCZYK, R.; ANDERSON, J. Wireless sensor networks for habitat monitoring. In: *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. New York, NY, USA: ACM, 2002. p. 88–97. ISBN 1-58113-589-0.

PAL, A. Localization algorithms in wireless sensor networks: Current approaches and future challenges. *Network Protocols and Algorithms: Special Issue on Surveys and tutorials*, 2010. v. 2, n. 1, 2010. Disponível em: <<http://www.macrothink.org/journal/index.php/npa/article/viewFile/279/276>>.



POTTIE, G. J.; KAISER, W. J. Wireless integrated network sensors. *Commun. ACM*, 2000. ACM, New York, NY, USA, v. 43, n. 5, p. 51–58, 2000. ISSN 0001-0782.

RAHMAN, M. A.; El Saddik, A.; GUEAIEB, W. Wireless sensor network transport layer: State of the art. In: MUKHOPADHYAY, S.; HUANG, R. (Ed.). *Sensors: Advancement In Modeling, Design Issues, Fabrication And Practical Applications*. [S.l.]: Springer-Verlag, 2008, (Lecture Notes In Electrical Engineering). (To appear).

RÖMER, K.; MATTERN, F. The design space of wireless sensor networks. *IEEE Wireless Communications*, 2004. v. 11, n. 6, p. 54–61, Dezember 2004.

SRISATHAPORNPHAT, C.; JAIKAE, C.; SHEN, C. chung. Sensor information networking architecture and applications. *IEEE Personal Communications*, 2001. v. 8, p. 52–59, 2001.

TILAK, S.; ABU-GHAZALEH, N. B.; HEINZELMAN, W. A taxonomy of wireless micro-sensor network models. *ACM Mobile Computing and Communications Review*, 2002. v. 6, p. 28–36, 2002.

YE, W.; HEIDEMANN, J.; ESTRIN, D. An energy-efficient mac protocol for wireless sensor networks. In: USC/INFORMATION SCIENCES INSTITUTE. *Proceedings of the IEEE Infocom*. New York, NY, USA: IEEE, 2002. p. 1567–1576. Disponível em: <<http://www.isi.edu/~johnh/PAPERS/Ye02a.html>>.

YE, W.; HEIDEMANN, J.; ESTRIN, D. Medium Access Control With Coordinated Adaptive Sleeping for Wireless Sensor Networks. *IEEE/ACM Transactions on Networking*, 2004. IEEE Press, Piscataway, NJ, USA, v. 12, n. 3, p. 493–506, jun. 2004. ISSN 1063-6692. Disponível em: <<http://dx.doi.org/10.1109/tnet.2004.828953>>.