

Comunicação de Dados baseada no Receptor para Redes de Sensores Sem Fio

Max do Val Machado¹ Raquel A. F. Mini² Antonio A. F. Loureiro¹

¹Departamento de Ciência da Computação
Universidade Federal de Minas Gerais
{maxm, loureiro}@dcc.ufmg.br

²Departamento de Ciência da Computação
Pontifícia Universidade Católica de Minas Gerais
raquelmini@pucminas.br

Abstract

This work proposes the Receiver-based Medium Access Control (Rb-MAC) protocol to be used in cross-layer designs for the receiver-based communication. These models are appropriated for Wireless Sensor Networks deal with energy constraints and dynamic topology. Simulation results reveal that using Rb-MAC, the evaluated receiver-based model increases the delivery ratio, reduces the number of transmissions and energy consumption when compared with a sender-based model. Moreover, Rb-MAC allows a dynamic duty cycle and maintains the delivery ratio for reduced duty cycle values.

Keywords: Wireless sensor networks, data communication, receiver-based, MAC, routing, duty cycle

Resumo

Este trabalho apresenta o protocolo Receiver-based Medium Access Control (Rb-MAC) para ser utilizado em projetos integrados para a comunicação baseada no receptor. Esses modelos são apropriados para as Redes de Sensores Sem Fio por lidarem com restrições de energia e topologia dinâmica. Resultados de simulação mostram que usando o Rb-MAC, o modelo receptor avaliado aumenta a taxa de entrega, reduz o número de transmissões e o consumo de energia quando comparado com um modelo baseado no emissor. Além disso, o Rb-MAC permite um ciclo de trabalho dinâmico e mantém a taxa de entrega para valores reduzidos de ciclo de trabalho.

Palavras-chave: Redes de sensores sem fio, comunicação de dados, receptor, MAC, roteamento, ciclo de trabalho

1. INTRODUÇÃO

O principal desafio para transformar as Redes de Sensores Sem Fio (RSSFs) [1] na tecnologia de sensoramento do futuro é projetar soluções eficientes em termos de energia. Outro desafio crucial é a topologia dinâmica dessas redes que geralmente é consequência dos nós adormecerem para economizar energia. Nesse contexto, a comunicação de dados vem recebendo atenção especial dos pesquisadores porque, normalmente, corresponde à tarefa com o maior custo de energia nas RSSFs e trata das mudanças topológicas. As camadas de rede e de acesso ao meio (MAC) são responsáveis pela comunicação em RSSFs.

A camada de rede é responsável pela escolha sistemática do próximo nó durante o processo de roteamento, o que leva à definição de rotas. Essa escolha, chamada de decisão de propagação, é baseada no emissor ou no receptor. Na primeira, quando um nó recebe um pacote não destinado a ele, o nó verifica se ele foi o escolhido para continuar o roteamento. Se sim, o nó escolhe um subconjunto de vizinhos para continuar o roteamento e, depois, propaga o pacote. Na outra abordagem, quando um nó recebe um pacote não destinado a ele, o nó decide localmente se deve ou não propagar o pacote.

A camada MAC é responsável por entregar o pacote para o próximo nó e, por isso, ela gerencia o uso do canal para evitar/tratar colisões e controla o ciclo de trabalho dos nós para tratar da comunicação quando o próximo nó estiver adormecido. O ciclo de trabalho é a porcentagem de tempo em que um nó permanece acordado durante um

ciclo de vida. O tempo de vida dos nós pode ser dividido em ciclos de vida sendo que cada um desses ciclos é composto por períodos de tempo em que o nó permanece acordado e por períodos em que ele dorme. O término de um ciclo de vida implica no início do próximo. Assim, quando o ciclo de trabalho de um nó é, por exemplo, 10%, o nó fica acordado durante 10% do tempo de cada ciclo de vida. A tarefa de controle do ciclo de trabalho é crucial em RSSFs, pois a melhor forma de economizar energia nessas redes é desativar os nós sempre que possível [3]. Dada a importância da energia em RSSFs, o principal desafio da camada MAC nessas redes é determinar o escalonamento de dormir/acordar dos nós para que o roteamento não seja prejudicado quando o próximo nó estiver adormecido.

Uma característica dos protocolos propostos para a camada MAC das RSSFs é que eles são focados no roteamento baseado no emissor. Isso significa que quando o protocolo MAC recebe um pacote a ser propagado, o próximo destino do pacote foi escolhido pela camada de rede e cabe à MAC entregar o pacote quando o escolhido estiver acordado. A maioria dos protocolos MAC faz com que nós vizinhos compartilhem os seus respectivos escalonamentos de dormir/acordar para que quando um nó enviar um pacote, ele saiba quando o vizinho escolhido estará acordado. A limitação dessa técnica é que o ambiente das RSSFs é normalmente dinâmico e os nós possuem restrições de hardware que levam a desatualização dessas informações. Normalmente, o compartilhamento é atualizado periodicamente através da troca de pacotes de controle (custo de energia). Além disso, o custo-benefício das informações compartilhadas é minimizado pelo fato das RSSFs normalmente possuírem longos períodos de inatividade em que nenhum dado é roteado e as informações sobre vizinhos são atualizadas. Alguns protocolos não compartilham tais informações e para um nó emissor descobrir quando o nó escolhido estará acordado, ele envia vários pacotes de controle. A limitação dessa técnica é o custo de energia com o envio desses pacotes. Dadas as limitações das técnicas existentes, novas soluções para a comunicação quando o próximo nó estiver adormecido devem ser investigadas.

Os modelos de comunicação baseados no receptor podem ser investigados como outra solução para o desafio proposto. O roteamento baseado no receptor é mais robusto às falhas no próximo nó que o baseado no emissor, pois qualquer vizinho que recebe o pacote pode se escolher como o próximo nó. No baseado no emissor, apenas o vizinho escolhido pode ser o próximo nó. O fato da decisão de propagação ser tomada pelo nó receptor faz com que a informação sobre vizinhos seja normalmente desnecessária, o que elimina o seu custo de atualização. Outro ponto relevante é que no roteamento baseado no receptor, o número de opções para o próximo nó é maior

que o do baseado no emissor e, por isso, a estratégia de retransmissão para descobrir vizinhos acordados pode ser eficiente em termos de energia. Além disso, como o objetivo da camada MAC é entregar pacotes para o próximo nó, um protocolo MAC para a comunicação baseada no receptor não pode se limitar a entregar pacotes para um vizinho qualquer e, sim, ele deve aguardar a decisão de propagação dos nós receptores para garantir que um vizinho acordado será o próximo nó.

Este trabalho apresenta o *Receiver-based Medium Access Control* (Rb-MAC), um protocolo para projetos integrados de comunicação baseada no receptor. Um projeto integrado consiste em uma camada ser capaz de violar a arquitetura da pilha de protocolos para acessar informações disponíveis em outra camada a fim de explorar vantagens e evitar pontos fracos. O Rb-MAC elimina o compartilhamento de informações sobre vizinhos e aborda o desafio proposto através de um mecanismo de retransmissão. Esse mecanismo é baseado no escalonamento de dormir/acordar dos nós, no fato de qualquer vizinho pode ser o próximo nó e na importância do nó emissor confirmar que um vizinho decidiu ser o próximo nó. Resultados de simulação revelam que com o Rb-MAC, o modelo receptor avaliado aumenta a taxa de entrega (porcentagem de pacotes entregues aos nós destinos), e reduz o número de transmissões e o consumo de energia quando comparado com um modelo baseado no emissor. O Rb-MAC também garante uma taxa de entrega elevada até mesmo quando os nós são configurados com valores reduzidos de ciclos de trabalho. Contudo, a latência e o número de colisões do modelo emissor avaliado são menores que essas métricas no modelo receptor avaliado. Para um cenário específico em que uma mesma rota é utilizada mais vezes, o Rb-MAC possui uma técnica de ciclo de trabalho dinâmico que proporciona uma latência inferior a do outro modelo. Este trabalho também efetua a verificação formal de algumas propriedades do Rb-MAC.

O restante deste trabalho é dividido da seguinte forma. A seção 2 mostra os trabalhos relacionados. A seção 3 apresenta o protocolo Rb-MAC. A seção 4 mostra os resultados de simulação. A seção 5 descreve as conclusões e as direções futuras deste trabalho.

2. TRABALHOS RELACIONADOS

Esta seção mostra alguns protocolos de roteamento e MAC. Um exemplo de protocolo de roteamento baseado no receptor é o *Trajectory and Energy-based Dissemination* (TEDD) [2] e um baseado no emissor é o *Trajectory Based Forwarding* (TBF) [7]. Ambos utilizam o conceito de “roteamento em curva” que consiste em inserir uma equação de curva no pacote e fazer com que os nós localizados próximos à curva propaguem o pacote. O TEDD e

o TBF são utilizados nos modelos de comunicação avaliados neste trabalho.

A maioria dos protocolos MAC para RSSFs tais como o Sensor MAC (S-MAC) [9] e o Battery Aware MAC [5] compartilha informações de vizinhos, ao contrário do Rb-MAC. Como parte deste trabalho, o S-MAC é utilizado em um modelo de comunicação baseado no emissor. No S-MAC, os nós vizinhos efetuam uma sincronização para que eles acordem e adormeçam ao mesmo tempo. Para evitar colisões, o S-MAC utiliza o mecanismo *Request To Send/Clear To Send/Data/Acknowledge* (RTS/CTS/DATA/ACK) que é restrito à comunicação em que o nó emissor conhece o próximo nó – roteamento baseado no emissor. O S-MAC pode ser usado na comunicação baseada no receptor, contudo, nesse caso, ele evita colisões apenas escutando o meio.

Alguns protocolos não compartilham informações de vizinho [6, 8] e para um nó emissor descobrir quando o nó escolhido estará acordado, ele envia vários pacotes de RTS. Quando o vizinho escolhido acordar e receber um RTS, o esquema CTS/DATA/ACK tradicional é utilizado. Esses protocolos se diferenciam entre si pela forma como enviam o pacote de RTS e se diferenciam do Rb-MAC por serem focados na comunicação baseada no emissor. Esses protocolos não foram avaliados neste trabalho, pois como mostrado por seus autores, normalmente (incluindo o cenário simulado neste trabalho), as técnicas de compartilhamento consomem menos energia. Em [10], tem-se um projeto integrado baseado no receptor em que o esquema de RTS/CTS considera a decisão de propagação do nó receptor para enviar pacotes de CTS. Nesse caso, o RTS é destinado a qualquer vizinho e o primeiro nó que enviar um CTS será o próximo nó. Apesar do nó emissor confirmar que um vizinho tomou a decisão de propagação, o intervalo dos pacotes de RTS não considera o escalonamento de dormir/acordar dos nós o que causa muitas transmissões de RTS. Esse protocolo não foi avaliado neste trabalho devido ao seu custo de energia com transmissões.

3. O PROTOCOLO RECEIVER-BASED MEDIUM ACCESS CONTROL (RB-MAC)

Esta seção apresenta o Rb-MAC, um protocolo para projetos integrados de comunicação baseada do receptor. A seção 3.1 apresenta o funcionamento básico do Rb-MAC, onde o acesso ao meio é aleatório e baseado na técnica *Carrier Sense Multiple Access* (CSMA) p-persistente e os nós dormem/acordam de forma aleatória. A seção 3.2 mostra o mecanismo de retransmissão do Rb-MAC. A seção 3.3 mostra um projeto integrado em que a camada de rede atualiza dinamicamente o ciclo de trabalho dos nós sensores. A seção 3.4 efetua a verificação formal de

algumas propriedades do Rb-MAC.

3.1. FUNCIONAMENTO BÁSICO

O funcionamento básico do Rb-MAC é descrito na máquina de estados finitos da figura 1, onde as transições são eventos gerados por outras camadas ou pelo término de temporizadores do Rb-MAC. Quando um evento acontece e uma transição é realizada, o Rb-MAC muda o seu estado corrente ou efetua uma ação. A mudança de estado e as ações só acontecem devido às transições, contudo, existem transições sem um deles. Quando um nó se torna ativo, ele vai para o estado *escutar* e permite que a camada física escute o meio.

A fase de recepção do Rb-MAC começa quando a camada física identifica que um nó vizinho vai começar uma transmissão. Nesse caso, a camada física gera uma requisição de receber e o Rb-MAC vai para o estado *receber*, onde o nó receberá o pacote. O Rb-MAC permanece nesse estado até que ele receba uma requisição de escutar que será gerada pela camada física quando o meio estiver livre. Quando isso acontece, o Rb-MAC processa o pacote recebido e retorna ao estado *escutar*. No processamento do pacote, se ele foi recebido com sucesso, o Rb-MAC o envia para a camada de rede, caso contrário, uma colisão/falha aconteceu e o Rb-MAC descarta o pacote.

A fase de transmissão do Rb-MAC é baseada na abordagem CSMA p-persistente e começa quando o protocolo está no estado *escutar* e a camada de rede possui um pacote a ser enviado. Nesse caso, a camada de rede gera uma requisição de aguardar para que Rb-MAC vá para o estado *aguardar*, como ilustrado na figura 1. Quando tal transição acontece, o Rb-MAC realiza a ação de aguardar na qual ele permite que a camada física comece a transmitir o pacote com probabilidade p . Com probabilidade $q = 1 - p$, ele inicializa o temporizador aguardar. Se a camada física começar a transmitir, ela gera uma requisição de transmitir para que o Rb-MAC vá para o estado *transmitir*. Quando a transmissão termina, a camada física gera uma requisição de escutar para que o Rb-MAC retorne ao estado *escutar*. Por outro lado, se o temporizador aguardar foi inicializado, o seu término significa uma requisição de aguardar e essa faz com que o Rb-MAC execute a ação de aguardar novamente. O Rb-MAC permanece no estado *aguardar* até que ele receba uma requisição de transmissão ou uma de receber. Enquanto o Rb-MAC estiver no estado *aguardar*, a camada física pode identificar que algum vizinho vai começar uma transmissão. Nesse caso, essa camada gera uma requisição de receber para que o Rb-MAC interrompa o seu temporizador aguardar e vá para o estado *receber*. Quando a recepção termina, a camada física gera uma requisição de escutar e essa faz com que o Rb-MAC retorne ao estado de *escutar*. Como existe um pacote para ser enviado, o Rb-MAC inicializa o seu temporizador aguardar para gerar uma re-

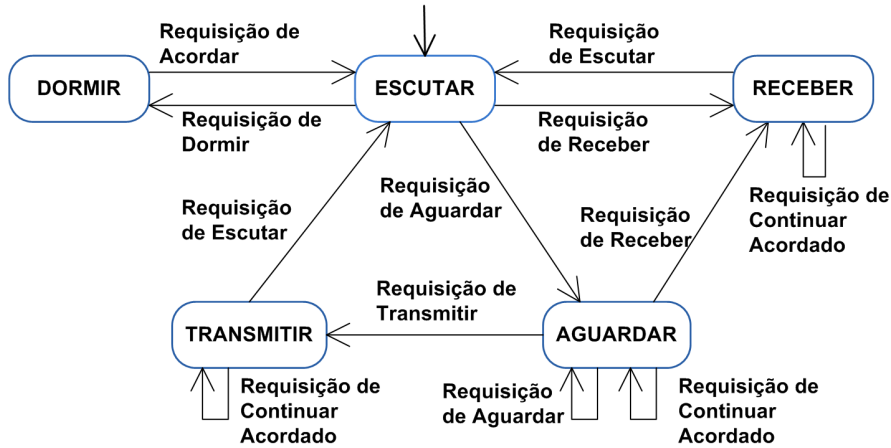


Figura 1. Funcionamento básico do Rb-MAC

quisição de aguardar. Um ponto importante desta fase é que quando o Rb-MAC estiver no estado *transmitir*, o nó não é capaz de receber pacotes, pois o seu rádio é normalmente *half-duplex*. Se dois nós vizinhos começarem a enviar pacotes ao mesmo tempo, as colisões não são detectadas/tratadas por esses nós; isso pode ser feito apenas nos nós que receberem o pacote.

O Rb-MAC também permite que os nós adormeçam para economizar energia. Como mostrado na figura 1, quando um nó estiver no estado *escutar* e receber uma requisição de dormir, ele vai para o estado *dormir*. O Rb-MAC permanece nesse estado até receber uma requisição de acordar para retornar ao estado *escutar*. O procedimento para a geração das requisições de dormir e acordar é cíclico e divide o tempo de vida dos nós em ciclos que são compostos por um período de tempo em que o nó permanece acordado seguido por outro em que ele dorme. O término de um ciclo implica no início do próximo. A duração dos períodos acordado A e dormindo S (do inglês, *sleeping*) são variáveis aleatórias com distribuição uniforme como mostrado nas equações 1 e 2, respectivamente. Os valores mínimo e máximo dessas variáveis são definidos nas equações 3 até 6 em que c é o tempo mínimo de duração de um ciclo, α é o fator de desvio e d é o ciclo de trabalho desejado (do inglês, *duty cycle*). O parâmetro c é um limite inferior para cada período. Quando um nó acorda, é interessante que ele permaneça acordado por um tempo mínimo em que o nó pode participar de tarefas da rede. De forma similar, se um nó dormir por um tempo mínimo após a realização de uma tarefa, o tempo de vida de sua bateria tende a ser maior. Segundo [5], isso é uma propriedade das baterias. O fator de desvio α determina o valor máximo das variáveis A e S . O parâmetro d é uma constante que indica o valor de ciclo de trabalho desejado pelo projetista da rede.

O processo de geração de requisições de

dormir/acordar é descrito a seguir. Quando um nó se torna ativo, ele liga o seu rádio e inicializa o temporizador acordado para expirar após A unidades de tempo. Quando ele expira e o Rb-MAC está no estado *escutar*, esse término significa uma requisição de dormir e essa faz com que o Rb-MAC desligue o rádio, inicialize o temporizador dormir para expirar após S unidades de tempo e vá para o estado *dormir*. O término do temporizador dormir significa uma requisição de acordar e essa faz com que o Rb-MAC ligue o rádio, inicialize o temporizador acordado para expirar após A unidades de tempo e retorne ao estado *escutar*. Por outro lado, quando o temporizador acordado expirar e o Rb-MAC não estiver no estado *escutar*, isso significa uma requisição de continuar acordado e essa faz com que o Rb-MAC inicialize novamente o temporizador acordado para expirar após mais algumas unidades extras de tempo e continue no estado corrente. Esse processo é repetido sempre que o temporizador acordado expirar e o Rb-MAC não estiver no estado *escutar*. Todo o tempo extra acumulado é descontado na duração dos próximos períodos de dormir e acordar. Até que todo o tempo acordado extra seja compensado, a duração deles será S_{max} e A_{min} , respectivamente.

$$A \sim U(A_{min}, A_{max}) \quad (1) \quad S_{min} = (1 - d) \times c \quad (4)$$

$$S \sim U(S_{min}, S_{max}) \quad (2) \quad A_{max} = \alpha \times A_{min} \quad (5)$$

$$A_{min} = d \times c \quad (3) \quad S_{max} = \alpha \times S_{min} \quad (6)$$

O Rb-MAC também permite que as camadas superiores gerem requisições de dormir/acordar, pois elas conhecem situações em que o rádio pode ser ligado/desligado. Por exemplo, quando a aplicação utiliza apenas dados noturnos, a camada de aplicação gera uma requisição de acordar no início da noite e outra de dormir no final. As camadas superiores também podem suspender a transição para o estado *dormir*, pois existem situações em que o nó deve ficar acordado por mais tempo. Por exemplo, no roteamento baseado no receptor, um nó não deve adormecer durante um processo de decisão.

3.2. MECANISMO DE RETRANSMISSÃO

A camada MAC das RSSFs é responsável pelo comportamento de dormir dos nós e, por isso, ela deve tratar as desconexões causadas pelos nós adormecidos. No Rb-MAC, essas desconexões são tratadas através de um mecanismo de retransmissão baseado na confirmação salto a salto. O objetivo desse mecanismo é similar ao dos pacotes de confirmação do protocolo IEEE 802.11; contudo, o mecanismo proposto confirma que um vizinho se escolheu como o próximo nó e o protocolo 802.11 confirma apenas que o próximo nó recebeu o pacote. Além disso, diferente do protocolo 802.11, o Rb-MAC não envia pacotes de controle. No mecanismo proposto, quando um nó transmite um pacote, ele permanece acordado por certo tempo para aguardar a propagação do pacote por algum nó vizinho. Após esse tempo, se o pacote não for recebido, o mesmo é retransmitido.

A definição do tempo em que um nó aguarda para realizar uma retransmissão e o número máximo de retransmissões é baseada no comportamento de dormir do Rb-MAC que proporciona um limite inferior A_{min} para o tempo em que um nó permanece acordado e outro superior S_{max} para o tempo em que o nó dorme. O limite inferior acordado é utilizado na definição do intervalo de tempo entre as retransmissões e o limite superior dormindo, na definição de quando a última retransmissão será executada. O número máximo de retransmissões é $\lceil \frac{S_{max}}{A_{min}} \rceil$, pois durante S_{max} unidades de tempo, uma retransmissão é executada a cada A_{min} unidades de tempo.

Teorema 1: Seja o intervalo entre retransmissões A_{min} e o número máximo de retransmissões $n = \lceil \frac{S_{max}}{A_{min}} \rceil$, quando um nó envia um pacote e todos os seus vizinhos estão adormecidos, cada vizinho estará acordado em pelo menos uma retransmissão.

Prova: Dado o intervalo e o número máximo de retransmissões, a prova consiste em mostrar que, para cada vizinho, se esse dorme no instante de tempo T_A , acorda em T_B e retorna a dormir em T_C , existe uma retransmissão T_i ($1 \leq i \leq n$) tal que $T_B < T_i < T_C$.

- 1º passo: Sabe-se que:
 - T_0 é a primeira transmissão.
 - $T_A = T_B - S$, pois a variável aleatória S é o intervalo de tempo entre T_A e T_B .
 - $T_A < T_0$, pois o vizinho estava dormindo na transmissão T_0 .
 - $T_0 = T_n - S_{max}$, pois S_{max} é o intervalo entre a primeira e a última retransmissões.
 - $S \leq S_{max}$, pois S_{max} é o maior valor da variável aleatória S .
- 2º passo: Logo, $T_B - S = T_A < T_0 = T_n - S_{max} \Rightarrow T_B < T_n$ e pode-se afirmar que existe pelo menos uma retransmissão T_i tal que $T_B < T_i$.
- 3º passo: Supondo que T_i é a primeira retransmissão tal que $T_B < T_i$. Se a suposição for falsa, $T_{j < i}$ é a primeira retransmissão maior que T_B o que é uma contradição.
- 4º passo: Sabe-se também que:
 - $T_B > T_i - A$, pois T_i é a primeira retransmissão tal que $T_B < T_i$ e se T_B for subtraído de T_i , esse resultado será menor à variável aleatória A .

– $T_B = T_C - A$, pois a variável aleatória A é o intervalo entre T_B e T_C .

- 5º passo: Logo, $T_C - A = T_B > T_i - A \Rightarrow T_i < T_C$.

Um aspecto crucial do mecanismo proposto é o número esperado de retransmissões, pois quando um nó retransmite um pacote diversas vezes, o custo de energia dessa tarefa pode tornar o Rb-MAC inviável para as RSSFs. No esquema proposto, um nó efetua uma retransmissão se ele transmitiu um pacote e não recebeu uma propagação do mesmo. Supondo a existência de vizinhança, os nós vizinhos não propagam um pacote se todos estiverem dormindo no seu envio ou devido às colisões/falhas. Logo, o número esperado de retransmissões depende da probabilidade de colisões/falhas e da dos nós vizinhos estarem dormindo. O evento ocorrência de colisões/falhas pode ser modelado como uma variável aleatória com distribuição uniforme e com probabilidade constante p_{col} . O evento os vizinhos estarem dormindo é uma variável aleatória cuja probabilidade reduz à medida que as retransmissões são executadas. Cada vizinho acorda pelo menos uma vez durante um processo de retransmissão (Teorema 1) e, à medida que as retransmissões são executadas e nenhum vizinho acorda, o final do processo se aproxima assim como a chance de cada vizinho acordar. A esperança da variável aleatória S na retransmissão i corresponde à média aritmética dos valores mínimo e máximo de S em i , como mostrado na equação 7. Nesse caso, como todos os vizinhos dormem A_{min} unidades de tempo por retransmissão, os valores mínimo e máximo de S são reduzidos em A_{min} unidades a cada retransmissão, como mostrado nas equações 8 e 9, respectivamente.

$$E[S_i] = \frac{S_{min}(i) + S_{max}(i)}{2} \quad (7)$$

$$S_{min}(i) = \begin{cases} S_{min} - i * A_{min} & \text{se } S_{min} > i * A_{min} \\ 0 & \text{caso contrário} \end{cases} \quad (8)$$

$$S_{max}(i) = \begin{cases} S_{max} - i * A_{min} & \text{se } S_{max} > i * A_{min} \\ 0 & \text{caso contrário} \end{cases} \quad (9)$$

A equação 10 mostra a probabilidade de um vizinho estar dormindo na retransmissão i , onde $E[A]$ é igual para todas as retransmissões. A equação 11 mostra a probabilidade de um vizinho não propagar com a retransmissão i . Um vizinho não propaga o pacote se ele estiver dormindo ou uma colisão/falha acontecer, eventos independentes. A probabilidade da retransmissão i acontecer é o complemento de “todos os nn vizinhos não propagarem com a retransmissão i ”, como mostrado na equação 12. Contudo, a retransmissão i acontece somente quando todas as $i - 1$ retransmissões anteriores acontecerem e nenhuma propagação for recebida pelo nó que efetua as retransmissões. A função de distribuição de probabilidade das retransmissões é mostrada na equação 13. A esperança da

variável aleatória R representando o número de retransmissões é o somatório de cada retransmissão multiplicada por sua probabilidade, como mostrado na equação 14.

$$probDormir(i) = \frac{E[S_i]}{E[S_i] + E[A]} \quad (10)$$

$$probNaoTx(i) = probDormir(i) + (1 - probDormir(i)) * p_{col} \quad (11)$$

$$prob(i, nn) = 1 - probNaoTx(i)^{nn} \quad (12)$$

$$fdp(i, nn) = prob(i, nn) \prod_{k=0}^{i-1} (1 - prob(k, nn)) \quad (13)$$

$$E[R] = \sum_{i=0}^{\lceil \frac{S_{max}}{A_{min}} \rceil} i * fdp(i, nn) \quad (14)$$

A principal vantagem do mecanismo proposto é eliminar as desconexões causadas por nós adormecidos o que eleva a taxa de pacotes entregues. Essa vantagem é mais interessante para os valores reduzidos de ciclo de trabalho em que os nós dormem mais tempo e o número de desconexões é maior. A sua desvantagem é o número de retransmissões e a latência, pois um pacote pode ser retransmitido mais de uma vez e cada retransmissão representa um atraso. Contudo, o custo de energia das retransmissões pode ser minimizado pelo ganho de energia obtido com a redução do ciclo de trabalho. Outro ponto importante é que o mecanismo proposto não garante que todos os pacotes serão entregues aos seus destinos devido às colisões/falhas. Contudo, ele reduz o efeito das mesmas. Quando uma colisão/falha acontece, nenhum vizinho propaga o pacote, causando uma retransmissão.

3.3. CICLO DE TRABALHO DINÂMICO

Esta seção apresenta um projeto integrado em que o Rb-MAC permite que a camada de rede atualize o ciclo de trabalho dos nós para criar caminhos especiais de roteamento compostos por nós que não dormem (ciclo de trabalho de 100%). O modelo proposto também considera que a camada de rede pode alterar o seu processo de decisão para que os nós do caminho transmitam pacotes sem atrasos. Os protocolos de roteamento baseados no receptor costumam inserir atrasos nesse processo de decisão [2]. Uma vantagem do modelo proposto é que a maioria dos nós não pertence ao caminho e é configurada com valores reduzidos de ciclo de trabalho para economizar energia. Outra vantagem é que o roteamento nos caminhos especiais é efetuado com latência reduzida e a sua limitação consiste em ser válido apenas se a rota é utilizada mais vezes.

O funcionamento básico da técnica proposta é descrito nos algoritmos 1 e 2 que são executados pela camada de rede. Após um nó enviar um pacote, o algoritmo 1 insere o nó no caminho especial. No primeiro passo (Linha 1), a camada de rede verifica se o nó pertence a um caminho. Se falso, a camada de rede viola a arquitetura da pilha

de protocolos e aumenta o ciclo de trabalho (Linha 2) e elimina o atraso da decisão de roteamento para todos os pacotes a serem roteados pelo caminho (Linha 3). Depois, a camada de rede escalona o tempo de retorno para identificar quando o caminho expira (Linha 4). Por outro lado, quando um nó pertence ao caminho (Linha 5), a camada de rede reescalona o tempo de retorno desse nó para que ele permaneça por mais tempo em um caminho (Linha 6). Quando o tempo de retorno expirar, o algoritmo 2 é executado e, assim, a camada de rede retorna os valores originais da decisão (Linha 1) e do ciclo de trabalho (Linha 2).

Algoritmo 1 : Início do Ciclo de Trabalho Dinâmico

- 1: **se** eu não pertencço a um caminho especial de roteamento **então**
- 2: Aumentar o meu ciclo de trabalho no Rb-MAC para 100%
- 3: Eliminar o meu atraso para a propagação de pacotes do caminho
- 4: Escalonar o tempo de retorno
- 5: **senão**
- 6: Reescalonar o tempo de retorno
- 7: **fim se**

Algoritmo 2 : Término do Ciclo de Trabalho Dinâmico

- 1: Retornar o meu atraso original para a propagação de pacotes do caminho
- 2: Retornar o meu ciclo de trabalho original no Rb-MAC

3.4. VERIFICAÇÃO FORMAL

Esta seção efetua a verificação formal de algumas propriedades de um modelo de comunicação baseado no receptor que utiliza o Rb-MAC. O modelo de verificação foi o NuSMV e o conjunto φ de propriedades verificadas é listado abaixo.

- φ_1 : Cada estado s é alcançável a partir do estado inicial s_0 e s_0 é alcançável a partir de cada estado s . Essa propriedade garante a inexistência de estados de erro ou inalcançáveis.
- φ_2 : Se o Rb-MAC estiver no estado *dormir*, as camadas MAC e de rede não recebem nem enviam pacotes. Essa propriedade garante que o rádio está desligado no estado *dormir*.
- φ_3 : Os protocolos MAC e de rede podem receber/enviar pacotes simultaneamente. Essa propriedade, φ_4 e φ_5 garantem (ou limitam) a independência das ações de receber/enviar.
- φ_4 : O Rb-MAC não pode receber e enviar pacotes simultaneamente. Essa propriedade garante a limitação física dos nós sensores em que o rádio é geralmente *half-duplex*.
- φ_5 : A camada de rede pode receber e enviar pacotes simultaneamente, pois a camada MAC armazena os pacotes ainda não processados.

- φ_6 : A camada de rede recebe um pacote se e, somente se, o Rb-MAC o recebeu anteriormente. Essa propriedade e φ_7 garantem a ordem em que um pacote é processado.
- φ_7 : O Rb-MAC envia um pacote se e, somente se, a camada de rede já o enviou.

4. RESULTADOS DE SIMULAÇÃO

Esta seção tem como objetivo comparar modelos de comunicação baseados no emissor e no receptor em cenários de disseminação para RSSFs em que um nó monitor deseja enviar dados para os nós sensores. Esse tipo de comunicação é crucial em RSSFs, pois o nó monitor realiza tarefas específicas tais como alterar o modo de funcionamento dos sensores, ativar/desativá-los, e enviar requisições ou interesses para eles.

4.1. PARÂMETROS DE SIMULAÇÃO

Em todas as simulações, utilizou-se uma rede com 500 nós estáticos, homogêneos e com uma capacidade de energia finita e não renovável. O consumo de energia de cada nó foi baseado no Mica2 e a energia inicial foi suficiente para nenhum nó morrer por falta de energia. O raio de comunicação dos nós foi de 100 m, eles foram depositados de forma aleatória em uma área $1000 \times 1000 \text{ m}^2$ e cada nó conhece a sua localização.

Um nó monitor sem restrição de recursos foi colocado no canto inferior esquerdo da rede e realizou várias disseminações intercaladas uniformemente e destinadas a 25 nós. Em um cenário de simulação (seção 4.2), cada disseminação foi destinada a um subconjunto aleatório de nós. No outro (seção 4.3), um subconjunto aleatório de destinos foi escolhido e utilizado em todas as disseminações. O tempo de simulação foi de 1000 s, contudo, a primeira disseminação foi efetuada apenas com 50 s e a última com 600 s. O atraso inicial é necessário para a inicialização do S-MAC. A última disseminação acontece antes do final da simulação para garantir a execução de todo o roteamento.

Os modelos de comunicação comparados são ilustrados na figura 2 e descritos abaixo. O modelo receptor avaliado combina o TEDD que é um protocolo de roteamento baseado no receptor específico para a disseminação de dados com o Rb-MAC. Este modelo apresenta uma interação entre as duas camadas em que o TEDD é capaz de proibir que um nó adormeça quando esse estiver em decisão de propagação. O modelo emissor avaliado combina o TBF e o S-MAC. O primeiro foi escolhido por ser um protocolo de roteamento baseado no emissor específico para a disseminação de dados que trabalha como o TEDD e, o segundo, por prover informações atualizadas de vizinhos que são exploradas pelo TBF para a escolha do próximo nó através de interações entre as duas camadas.

Este trabalho também avalia um modelo de comunicação híbrido que combina o S-MAC e o TEDD. Esse modelo é denominado híbrido por combinar um protocolo MAC que compartilha informações de vizinho e um de roteamento baseado no receptor. O modelo híbrido foi proposto porque os outros apresentam diferentes protocolos MAC e de rede, o que dificulta identificar os ganhos em cada camada. Nesse modelo, o funcionamento básico do S-MAC foi alterado para permitir uma interação entre camadas em que o TEDD é capaz de proibir que um nó adormeça quando estiver em decisão de propagação.

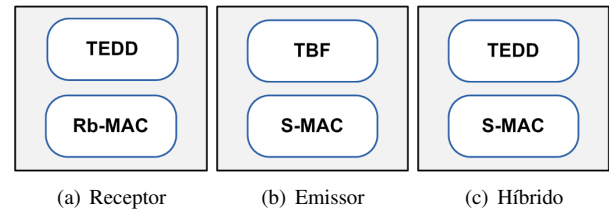


Figura 2. Modelos Avaliados para a Comunicação de Dados

O Rb-MAC e o S-MAC foram utilizados na camada MAC dos modelos avaliados. O ciclo de trabalho deles foi estático e avaliado para diferentes valores. Contudo, no segundo cenário de simulação, o Rb-MAC utilizou a sua técnica de ciclo de trabalho dinâmico onde os nós do caminho especial possuem um ciclo de trabalho de 100% e os demais de 10%. A frequência com que os nós atualizam as informações de vizinhos no S-MAC foi definida da forma mais justa possível, balanceando o número de transmissões e a taxa de pacotes recebidos. Se essa frequência aumentar, maior será o número de transmissões (logo, o consumo de energia) e a taxa de entrega obtidos pelo modelo emissor. Por outro lado, se ela for reduzida, o resultado oposto é verificado. A frequência de atualização utilizada no S-MAC foi de 150 ciclos de vida. No S-MAC, o tempo de vida dos nós é dividido em ciclos de vida e a frequência utilizada significa que a cada 150 ciclos de vida, os nós trocam informações com os seus vizinhos. Além disso, considerou-se que todos os pacotes (de controle ou dados) possuem o mesmo tamanho.

Todos os modelos foram implementados no *Network Simulator 2* e os resultados de simulação correspondem à média aritmética de n simulações, onde n é o menor tamanho de amostra que proporciona o intervalo de confiança desejado. O nível de confiança foi de 95% e o teste- T [4] com 0.05 de significância foi utilizado para afirmar que um protocolo é melhor/pior do que outro para uma dada métrica.

4.2. DISSEMINAÇÃO PARA UM SUBCONJUNTO DINÂMICO DE DESTINOS

Esta seção avalia o desempenho dos modelos propostos em um cenário de disseminação em que o subconjunto

de destinos é alterado a cada disseminação. Esse cenário acontece, por exemplo, quando o nó monitor deseja enviar dados para os sensores que possuem uma propriedade dinâmica tal como ter mais/menos energia e ser responsável por uma tarefa. O objetivo nesse cenário é entregar pacotes e economizar energia. A taxa de entrega é fundamental, pois pacotes podem ser perdidos durante o roteamento. A economia de energia é objetivo de qualquer solução para as RSSFs. Outros objetivos são as reduções de latência e colisões. A latência é crucial em aplicações dependentes do tempo tal como a detecção de incêndio. As colisões devem ser reduzidas, pois elas tendem a reduzir a taxa de entrega e aumentar o consumo de energia e a latência.

O desempenho dos modelos avaliados quando se aumenta o valor estático do ciclo de trabalho é mostrado na figura 3. A figura 3-a apresenta a taxa de entrega em que o modelo receptor avaliado entregou 1,05 e 1,03 vezes mais pacotes que os modelos emissor e híbrido, respectivamente. Esse resultado acontece porque o mecanismo de retransmissão do Rb-MAC elimina as perdas de pacote causadas pelos nós adormecidos e reduz o efeito das colisões. Além disso, ele mantém a taxa de entrega para os menores valores de ciclo de trabalho. As perdas de pacote do modelo receptor são consequência de colisões. No modelo emissor, além das colisões, as perdas foram causadas por nós adormecidos que foram selecionados para o roteamento. O aumento da entrega desse modelo foi porque, no S-MAC, o aumento do ciclo de trabalho implica em mais atualizações das informações de vizinhos e, assim, menos nós adormecidos são escolhidos para o roteamento. No modelo emissor, apesar do esquema RTS/CTS/DATA/ACK, as colisões acontecem, por exemplo, devido aos nós que dormem e acordam em tempos distintos e deturpam o esquema utilizado. No modelo híbrido, que não efetua retransmissões, as perdas são causadas pelas colisões e pelo modelo permitir que um nó efetue transmissões quando não existem vizinhos acordados. O aumento da taxa à medida que o ciclo de trabalho aumenta é porque a possibilidade de um nó efetuar uma transmissão sem vizinhos acordados é menor.

O número de transmissões quando se aumenta o valor estático do ciclo de trabalho é mostrado nas figuras 3-b e 3-c. A diferença entre elas é que a primeira considera as transmissões de pacotes de controle ou de dados e a outra, apenas as de dados. O modelo receptor apresenta o mesmo resultado nas duas figuras, pois o Rb-MAC não transmite pacotes de controle. Nos outros modelos, o S-MAC envia pacotes de controle para atualizar informações de vizinhos. No modelo emissor, ele também envia pacotes RTS/CTS/ACK. Os modelos emissor e híbrido efetuaram respectivamente 4,8 e 2,1 vezes mais transmissões que o receptor devido aos pacotes de controle do S-MAC. Neste ponto, observa-se que para os mo-

delos avaliados, o custo de transmissão inserido pelas retransmissões do Rb-MAC foi menor que o inserido pelos pacotes de controle do S-MAC. O modelo emissor enviou 2,3 vezes mais pacotes que o híbrido devido aos pacotes de controle extras. Aumentando o ciclo de trabalho, o número de transmissões efetuadas pelo modelo receptor reduz conforme o funcionamento básico do mecanismo de retransmissão do Rb-MAC. Por outro lado, o número efetuado pelos outros modelos aumenta, pois, no S-MAC, o quanto maior o ciclo de trabalho, mais frequente são os envios de pacote para atualizar as informações de vizinhos. Considerando apenas os pacotes de dados, como o modelo receptor efetua retransmissões, ele efetuou 1,4 e 1,1 vezes mais transmissões que os modelos emissor e híbrido, respectivamente.

O consumo de energia quando se aumenta o valor estático do ciclo de trabalho é apresentado na figura 3-d. À medida que o ciclo de trabalho aumenta, o consumo de energia aumenta, pois os nós sensores permanecem mais tempo acordados. Os modelos emissor e híbrido consomem 1,3 e 1,1 vezes mais energia que o modelo receptor avaliado, pois esses enviam mais pacotes que o último. O modelo emissor avaliado consome mais energia que o híbrido (1,2 vezes), pois o primeiro efetua mais transmissões.

A latência quando se aumenta o valor estático do ciclo de trabalho é apresentada na figura 3-e. A latência inicial do modelo emissor é 5,0 vezes maior que a do modelo receptor e, a partir do ciclo de trabalho de 30%, a do receptor torna-se 3,7 vezes maior que a do emissor. O principal responsável pela latência do modelo receptor é o TEDD que insere atrasos no roteamento para efetuar a decisão de propagação. No modelo emissor, esse papel é do S-MAC, pois ele proíbe que um nó propague um pacote no mesmo ciclo de vida em que esse é recebido. A latência reduz com o aumento do ciclo de trabalho, pois esse aumento provoca a redução do tamanho do ciclo de vida. A latência do modelo híbrido é 4,6 e 2,5 vezes maior que a dos modelos receptor e emissor, pois ele combina o S-MAC e o TEDD. Uma consideração crucial para a comunicação baseada no receptor é que como o Rb-MAC é capaz de manter a taxa de entrega para valores reduzidos de ciclo de trabalho, o projetista da rede deve definir o valor do ciclo de trabalho de acordo com o maior valor aceitável de latência. O quanto maior esse valor, menor o ciclo de trabalho.

O número total de colisões com pacotes de dados quando se aumenta o valor estático do ciclo de trabalho é mostrado na figura 3-f. Todos os modelos foram eficientes em termos de colisões quando esse valor é comparado com o número de transmissões efetuadas por cada modelo. Os modelos receptor e emissor efetuam uma colisão a cada cem pacotes de dados transmitidos e, no modelo híbrido, essa relação é o dobro. O modelo emissor

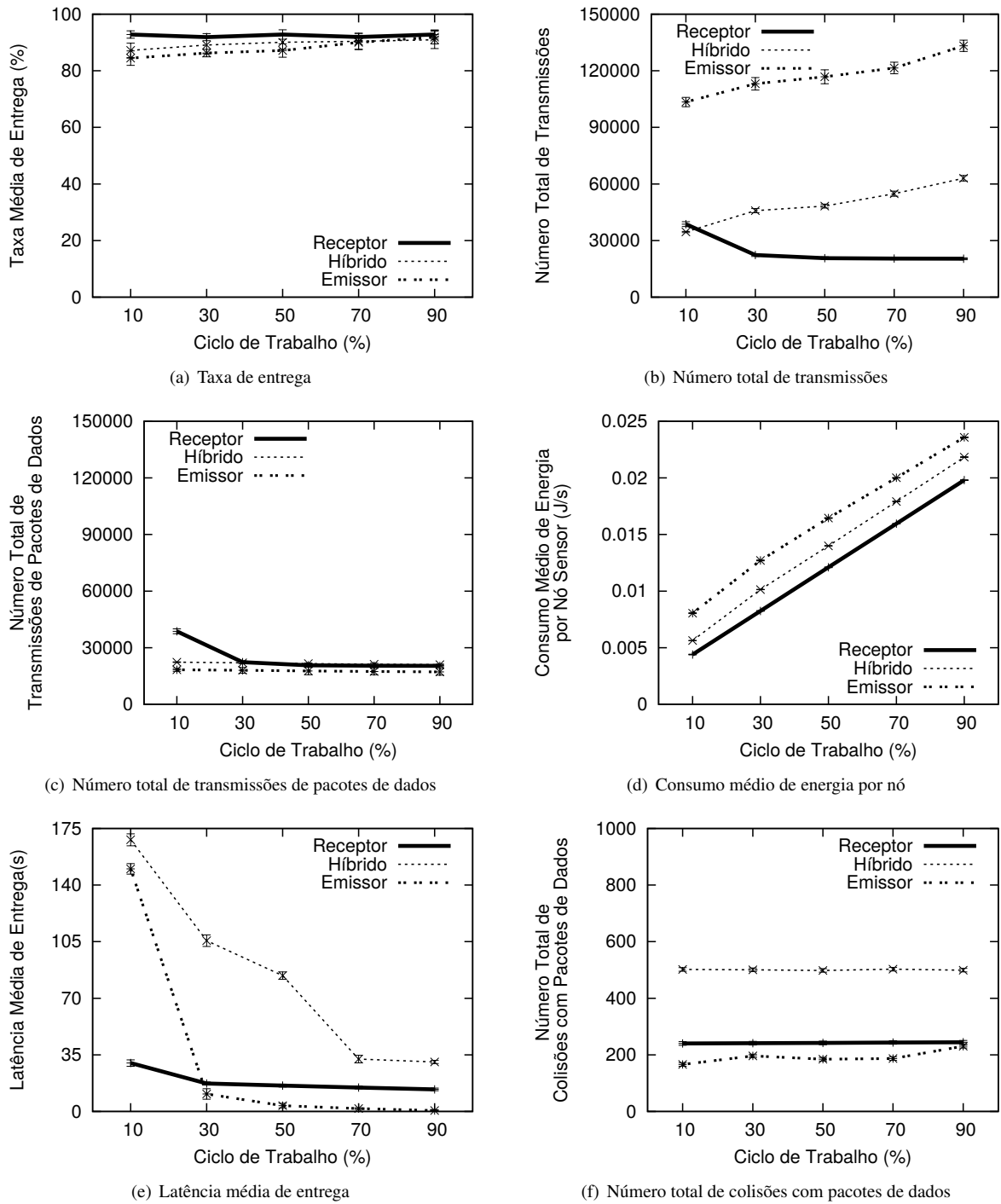


Figura 3. Parâmetros avaliados para o primeiro cenário de simulação

efetuou menos transmissões devido ao uso da técnica de RTS/CTS/DATA/ACK. O modelo receptor reduz as colisões através do esquema CSMA p-persistente. O modelo híbrido permite mais colisões por apenas escutar o meio como forma de reduzir colisões.

4.3. DISSEMINAÇÃO PARA UM SUBCONJUNTO ESTÁTICO DE DESTINOS

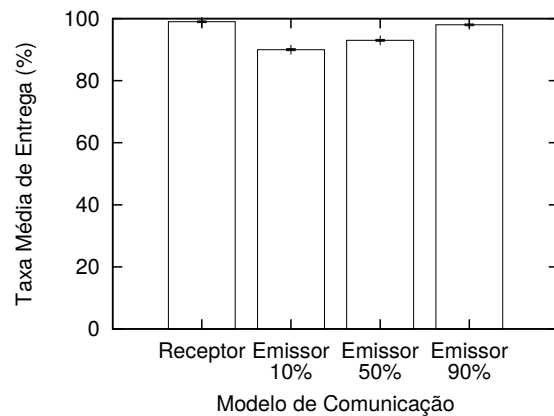
Esta seção avalia o desempenho dos modelos propostos em um cenário de disseminação em que todas as disseminações são destinadas para um mesmo subconjunto de nós definido aleatoriamente antes da simulação. Esse

cenário acontece, por exemplo, quando o nó monitor deseja enviar sistematicamente dados para nós localizados em posições estratégicas ou nós responsáveis por tarefas estáticas. Novamente, o objetivo é entregar pacotes e economizar energia. Outros objetivos relevantes são as reduções de latência e colisões. Além disso, o modelo receptor avaliado utiliza a técnica de ciclo de trabalho dinâmico do Rb-MAC. Os resultados de simulação obtidos pelos modelos emissor e híbrido são similares aos do cenário anterior, pois esses não se aproveitam das rotas serem repetidas. Os resultados do modelo híbrido não são mostrados e o modelo emissor avaliado utiliza os valores estáticos de ciclo de trabalho como 10%, 50% e 90%. Tanto no modelo receptor como no emissor, o protocolo de roteamento utilizou o mesmo conjunto de rotas em todas as disseminações de uma simulação.

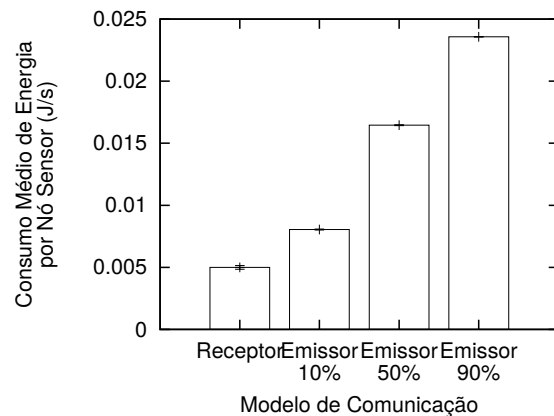
A taxa de entrega, o consumo de energia e a latência dos modelos receptor e emissor são mostrados na figura 4. O modelo receptor avaliado apresentou a maior entrega e os menores valores de consumo de energia e latência. Esse resultado é devido à técnica de ciclo de trabalho dinâmico do Rb-MAC. O modelo receptor entregou 1,1 mais pacotes que o emissor devido às retransmissões do Rb-MAC. O modelo emissor consome 9,6 vezes mais energia que o receptor, pois, nesse modelo, os nós não pertencentes ao caminho de roteamento são configurados com valores reduzidos de ciclo de trabalho. Além disso, o fato do modelo emissor transmitir pacotes de controle aumenta o seu consumo de energia. A latência do modelo emissor é 405,7 vezes maior que a do receptor, pois, nesse modelo, alguns nós ficam acordados e aguardando para rotear. Os números de transmissões e colisões não são mostrados por serem similares aos apresentados na seção anterior.

5. CONCLUSÕES E TRABALHOS FUTUROS

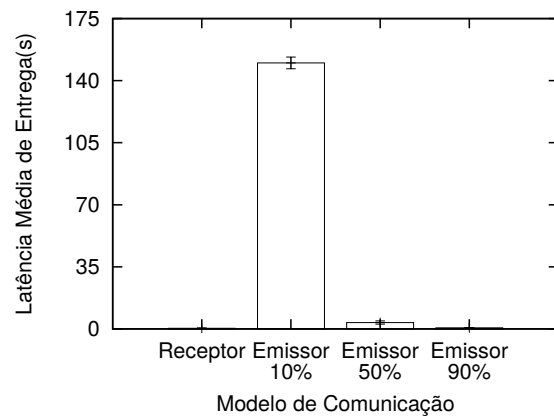
Este trabalho propõe o Rb-MAC, um protocolo para ser utilizado em soluções integradas com o roteamento baseado no receptor. O Rb-MAC explora o funcionamento básico desse tipo de roteamento para salvar recursos e permitir interações. Resultados de simulação mostram que o modelo receptor avaliado com o Rb-MAC aumentou a taxa de entrega, reduziu o número de transmissões e o consumo de energia quando comparado com um modelo baseado no emissor. Contudo, a latência e o número de colisões desse modelo foram menores. Resultados de simulação também revelaram que o Rb-MAC é capaz de manter a taxa de entrega para valores reduzidos de ciclo de trabalho. Além disso, o Rb-MAC possui uma técnica de ciclo de trabalho dinâmico que proporciona ao modelo receptor resultados de latência menores que os do



(a) Taxa de entrega



(b) Consumo médio de energia por nó



(c) Latência média de entrega

Figura 4. Parâmetros avaliados para o segundo cenário de simulação

modelo emissor avaliado, mas tal técnica é aplicada apenas em um cenário específico de disseminação.

A comunicação baseada no receptor é um tópico de pesquisa promissor, pois apresenta taxa de entrega elevada e consumo de energia reduzido. Um tópico de pesquisa que deve ser avaliado é o projeto integrado de

outras funções de rede (por exemplo, fusão de dados) com o modelo de comunicação baseados no receptor. Trabalhos futuros devem definir o ciclo de trabalho em função do maior valor aceitável para a latência.

Referências

- [1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: A survey. *Computer Networks*, 38(4), 02.
- [2] O. Goussevskaia, M. Machado, R. Mini, A. Loureiro, G. Mateus, and J. Nogueira. Data dissemination based on the energy map. *IEEE Communications Magazine*, 43(7), Jul. 05.
- [3] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister. System architecture directions for networked sensors. *SIGPLAN Not.*, 35(11), 00.
- [4] R. Jain. *The Art of Computer Systems Performance Analysis: techniques for experimental design, measurement, simulation and modeling*. 91.
- [5] S. Jayashree, B. S. Manoj, and C. Murthy. On using battery state for medium access control in ad hoc wireless networks. *ACM/IEEE Mobicom*, 04.
- [6] W. Liao and H. Wang. An asynchronous mac protocol for wireless sensor networks. *J. Netw. Comput. Appl.*, 31(4):807–820, 08.
- [7] D. Niculescu and B. Nath. Trajectory-based forwarding and its applications. *ACM/IEEE International conference on Mobile computing and networking*, 03.
- [8] S. Rashwand, J. Misic, V. Misic, S. Biswas, and M. Haque. A novel asynchronous, energy efficient, low transmission delay mac protocol for wireless sensor networks. *29th IEEE ICDCS*, Jun. 09.
- [9] W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Trans. Netw.*, 12(3), 04.
- [10] M. Zorzi and R. Rao. Geographic random forwarding (geraf) for ad hoc and sensor networks: Energy and latency performance. *IEEE Transactions on Mobile Computing*, 2(4), 03.