

Escalonamento em Redes de Sensores sem Fio com Canais com Maior Largura de Banda

Rodrigo C. Tavares¹, Marcos A. M. Vieira¹, Luiz F. M. Vieira¹, Bhaskar Krishnamachari²

¹Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)

{rodrigoc, mmvieira, lfvieira}@dcc.ufmg.br

²Ming Hsieh Department of Electrical Engineering
University of Southern California (USC) – Los Angeles, US

bkrishna@usc.edu

Abstract. *Many applications in Wireless Sensor Networks (WSNs) require collecting massive data in a coordinated approach. To do this, a many-to-one pattern (convergecast) communication in tree-based WSN is used. However, traffic near the sink node usually becomes the network bottleneck. Thus, we propose a extension 802.15.4 standard to enable channels with a wider bandwidth. We explore how fast the information can be collected in a WSN organized as a tree, with the radios operating with these wider bandwidth channels. Then, we proposed and implemented FWB (Funneling Wider Bandwidth), an algorithm to minimize schedule length in networks with radios that operate with the proposed extension. We proved the algorithm is optimal in the number of time slots. In this manner, the average throughput is higher and the number of time slots is lesser. The simulation results show that these benefits were achieved. This new approach could be adapted and used with other relevant emerging standards such as WirelessHART, ISA 100.11a and IEEE 802.15.4e TSCH.*

Resumo. *Muitas aplicações em Redes de Sensores sem Fio (RSSFs) necessitam coletar dados massivos em um esquema coordenado. Para fazer isso, um padrão de comunicação muitos-para-um (coleta) baseado em árvore em uma RSSF é usado. Entretanto, o tráfego próximo ao nó sorvedouro se torna o gargalo da rede. Assim, é proposto uma extensão do padrão 802.15.4 para permitir canais com uma maior largura de banda. Explora-se o quão rápido a informação pode ser coletada em uma RSSF organizada como uma árvore, com os rádios operando com canais com essa maior largura de banda. Então, é proposto e implementado o FWB (Funneling Wider Bandwidth), um algoritmo para minimizar o tamanho do escalonamento em redes com rádios que operam com a extensão proposta. Prova-se que o algoritmo é ótimo em relação ao número de intervalos de tempo. Dessa maneira, a vazão média é mais alta e o número de intervalos é menor. Os resultados de simulação mostram que esses benefícios foram alcançados. Essa nova abordagem pode ser adaptada e usada com outros padrões emergentes sem fio relevantes, como o WirelessHART, ISA 100.11a e o IEEE 802.15.4e TSCH.*

1. Introdução

Redes de Sensores Sem Fio (RSSFs) são redes com grande número de micro sensores compactos com capacidade de comunicação sem fio, chamados de nós sensores

[Ruiz et al. 2004]. O objetivo destas redes é sensoriar o meio e coletar dados. Há várias situações onde é necessário transferir dados em uma aplicação de rede de sensores, tais como monitoração de vulcões, oceanos, desertos e florestas. Como algumas dessas aplicações cobrem uma grande área, utiliza-se comunicação sem fio.

Muitas aplicações de redes de sensores coletam dados massivos de forma coordenada, tais como monitoramento de oceanos, florestas, e vulcões [Werner-Allen et al. 2006]. Esse paradigma de comunicação muitos para um é um dos mais comuns, dado que ele atende às necessidades de várias aplicações.

A largura de banda em RSSFs é limitada. A banda ISM não licenciada 2.4 GHz é usada por vários dispositivos, padrões e aplicações. O padrão IEEE 802.15.4 define camadas físicas (PHY) à não licenciada banda de 2.4 GHz. Um total de 16 canais estão disponíveis nessa banda, numerados de 11 a 26, cada um com uma largura de banda de 2 MHz e uma separação de canal de 5 MHz. Assim, as plataformas IEEE 802.15.4, tais como MICAz e Telos, tem rádios operando na máxima largura de banda teórica de 250 Kbps [Vieira et al. 2003], dado a restrição de largura de banda de 2 MHz.

Portanto, protocolos tais como Time Division Multiple Access (TDMA) mitigam esses problemas, dado que eles podem eliminar colisões e retransmissões e fornecer uma garantia de tempo de conclusão. Dado isso, uma das principais metas é minimizar o número de intervalos de tempo para coletar dados de todos os nós da rede. Uma forma de se fazer isso é maximizar o número de transmissões concorrentes, o que permite maximizar o reuso de intervalos de tempo. Em [Incel et al. 2012], os autores propõem heurísticas para minimizar o tamanho de escalonamento e provam que se a interferência é eliminada, o escalonamento é ótimo. Entretanto, eles usam como modelo enlaces com somente uma largura de banda. O que acontece se os enlaces tiverem uma maior largura de banda? Isso é explorado nesse trabalho e mostra-se que com essa abordagem o número de intervalos de tempo para coletar dados de todos os nós da rede é reduzido.

As principais contribuições desse artigo são as seguintes. Primeiro, é proposta uma extensão do padrão 802.15.4 para permitir canais com uma maior largura de banda. Segundo, é proposto um algoritmo para minimizar o tamanho do escalonamento em redes com rádios que operam com essas maiores larguras de banda. Terceiro, é provado a otimalidade do algoritmo. Finalmente, são apresentados resultados de simulação que confirmam os ganhos na redução do número total de intervalos de tempo.

Na seção 2 é apresentada a extensão proposta. O problema é modelado na seção 3. Na seção 4 é discutido o escalonamento TDMA em coletas. Na seção 5 é detalhado o algoritmo. Na seção 6 é fornecida a análise de complexidade do overhead de mensagens e a prova de otimalidade do algoritmo. Na seção 7 são mostrados os resultados dessa simulação e comparação aos prévios protocolos. Na seção 8 é sumarizado e comparado os trabalhos relacionados. Finalmente, na seção 9 é apresentado os pensamentos finais e conclusões.

2. Proposta de extensão do padrão 802.15.4

Figura 1 apresenta a extensão proposta ao padrão 802.15.4. O eixo x representa o espectro de frequência. O eixo y mostra a largura de banda. Cada caixa indica um canal. A linha 2 MHz é o atual padrão IEEE 802.15.4.

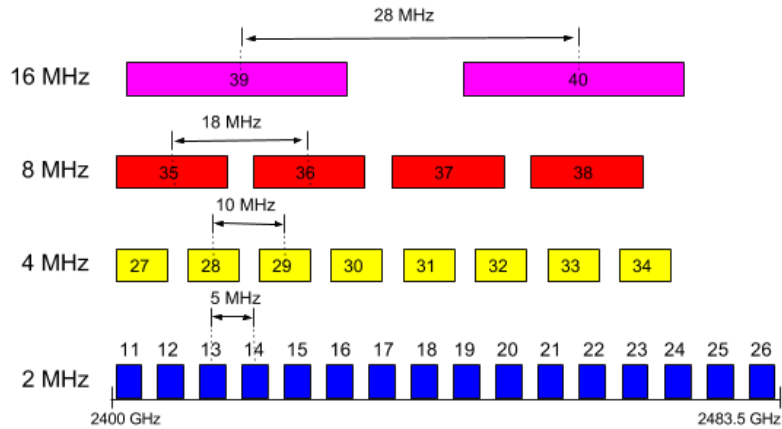


Figura 1. Extensão com canais com larguras de banda de 2, 4, 8 e 16 MHz.

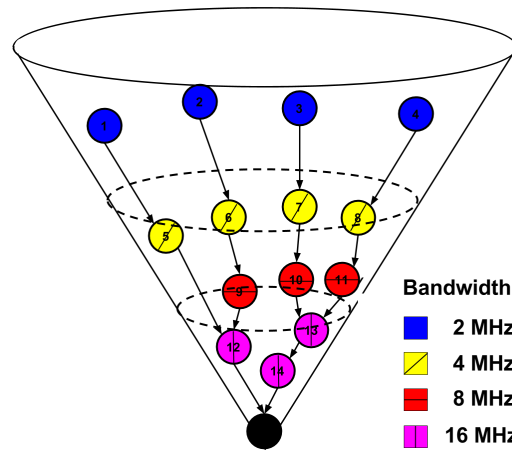


Figura 2. Extensão com canais com maior largura de banda.

Com essa extensão, é possível mitigar problemas gerados pelo efeito de funil [Wan et al. 2005], o que faz com que o tráfego próximo ao sorvedouro se torne o gargalo da rede, como ilustrado na Figura 2. Por atribuir maior largura de banda aos nós próximos ao funil, é possível aumentar a vazão e, portanto, mitigar o gargalo. Além disso, essa extensão permite que os enlaces com maior largura de banda continuem a funcionar normalmente com os enlaces de 2 MHz. A largura de banda dos nós da RSSF é determinada baseada no número de descendentes do nó, o qual é calculado pelo Algoritmo 1.

Na próxima seção, assume-se o modelo do protocolo e, posteriormente, fornece-se um algoritmo (Algoritmo 1) para alocar a largura de banda de cada nó.

3. Modelagem

A RSSF é modelada como um grafo $G = (V, E)$ o qual consiste de um conjunto de nós sensores V que periodicamente geram dados e um conjunto de arestas E representando os enlaces sem fio. Assume-se que os enlaces são bidirecionais. Cada aresta $e = (u, v)$ consiste de nós u e v . Se $e \in E(G)$, então é dito que u é adjacente a v . Os vértices de G serão denotados por $V(G)$ e as arestas por $E(G)$. O nó $s \in V$ denota o nó sorvedouro. Todos os nós exceto s são fontes, os quais geram pacotes e os transmitem acima de uma

árvore de roteamento para s . Esse grafo forma uma árvore em G . A árvore geradora em G com raiz s é denotada por $T = (V, E_T)$, onde $E_T \in E$ representa as arestas da árvore. Um modelo simples é assumido no qual a interferência é evitada usando diferentes canais de transmissão nos enlaces. A estrutura da rede é homogênea. Assim, todo nó sensor tem um transceptor half-duplex que pode ser configurado para operar com maior largura de banda. A conectividade da rede é fixada acima do tempo e o enlace sem fio é simétrico. Considera-se um protocolo TDMA onde o escalonamento de quadros de tempo é dividido dentro de intervalos. A duração de um intervalo de tempo permite transmitir exatamente um pacote de dados mais um intervalo de tempo de guarda para evitar colisões e sincronizar erros. Por simplicidade, usa-se um modelo de protocolo baseado em grafo. Nesse modelo de protocolo, assume-se que o alcance de interferência de um nó é igual ao seu alcance de transmissão, isto é, dois enlaces não podem ser escalonados simultaneamente se o receptor de ao menos um enlace está dentro do alcance de transmissão de outro enlace.

Estuda-se coleta raw-data no contexto de coleta de dados periódica, onde cada nó tem somente um pacote para enviar e o tamanho de cada pacote é constante. Tal como o trabalho de [Incel et al. 2012], pretende-se escalonar as arestas E_T de T usando o menor número de intervalos de tempo, porém usando largura de banda variável e respeitando a seguinte restrição:

Restrição de adjacência. Duas arestas $(i, j) \in E_T$ e $(k, l) \in E_T$ não podem ser escalonadas no mesmo intervalo de tempo se elas são adjacentes entre elas, isto é, se $\{i, j\} \cap \{k, l\} \neq \emptyset$. Essa restrição se deve ao transceptor half-duplex de cada nó o que os impede de transmitir e receber simultaneamente.

4. Escalonamento TDMA de coletas

O problema de minimizar o tamanho de escalonamento para coleta raw-data, considerando o grafo de interferência, foi provado ser um problema NP completo por uma redução do problema NP difícil *Partition Problem* [Choi et al. 2009]. O limite inferior ótimo $\max(2n_k - 1, N)$ é alcançado pelo algoritmo de [Song et al. 2007]. Entretanto, esse ótimo é para rádios com largura de banda fixa. O que acontece caso a largura de banda de alguns enlaces da topologia seja dobrada? A melhoria alcançada é mostrada na Figura 3. O tamanho do escalonamento alcançado é reduzido para 4, dado que os enlaces no topo usam o dobro da largura de banda dos enlaces inferiores, o que faz com que menos intervalos sejam necessários. Os pacotes enviados por cada intervalo são mostrados na tabela da Figura 4. Dado que é considerado a coleta periódica raw-data, cada uma das arestas em E_T é escalonada somente uma vez dentro de cada quadro, e esse escalonamento é repetido acima de múltiplos quadros. Assim, um pipeline é estabelecido depois de um certo quadro, e em seguida o sorvedouro continua a receber pacotes de todos os nós fontes com enlaces de maior largura de banda transmitindo mais pacotes por intervalo. Os enlaces entre o nó 1 e o sorvedouro, e entre o nó 4 e o sorvedouro tem o dobro de largura de banda, o que lhes permite enviar dois pacotes em só um dado intervalo. No quadro 2, o sorvedouro recebe 2 pacotes nos intervalos 2 e 4, dado que ele tem enlaces com maior largura de banda. Assim, no intervalo 2, ele recebe 2 pacotes que estavam no buffer do nó 1, um de cada vez. O mesmo ocorre no intervalo 4.

Por outro lado, quando a largura de banda é três vezes maior, o número de intervalos necessários se reduz ainda mais, como mostrado na Figura 5. Os pacotes enviados

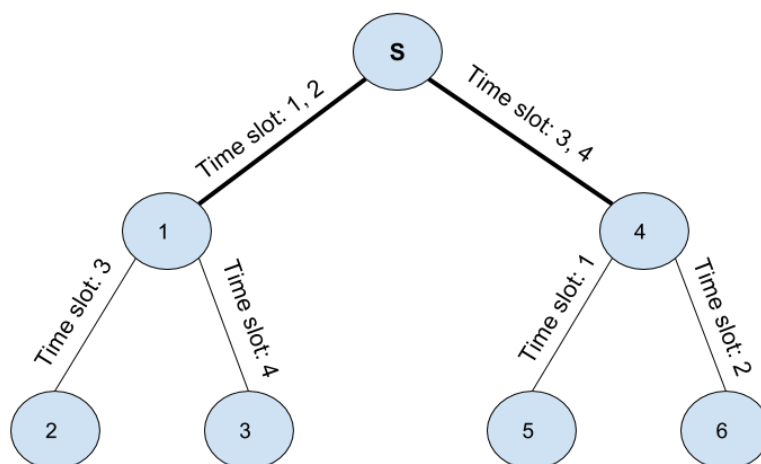


Figura 3. Coleta raw-data em enlaces com o dobro de largura de banda usando o algoritmo proposto e com todos os enlaces que causam interferência removidos. O número nas arestas indica o intervalo de tempo para a transmissão. O tamanho de escalonamento é 4.

Nós Receptores	Quadro 1				Quadro 2			
	SLOT 1	SLOT 2	SLOT 3	SLOT 4	SLOT 1	SLOT 2	SLOT 3	SLOT 4
SINK	1 (Link 4 MHz)		2 (Link 4 MHz)		1 (Link 4 MHz)	3, 4 (Link 4 MHz)	2 (Link 4 MHz)	5, 6 (Link 4 MHz)
1			3 (Link 2 MHz)	4 (Link 2 MHz)			3 (Link 2 MHz)	4 (Link 2 MHz)
4	5 (Link 2 MHz)	6 (Link 2 MHz)			5 (Link 2 MHz)	6 (Link 2 MHz)		

Figura 4. Pacotes enviados em cada intervalo para enlaces com largura de banda de 4 MHz.

por cada intervalo são mostrados na tabela da Figura 6. A partir do quadro 2, o sorvedouro recebe 3 pacotes nos intervalos 1 e 2, dado que possui uma maior largura de banda. Assim, no intervalo 1, ele recebe 3 pacotes que estavam no buffer do nó 1, um por vez. O mesmo ocorre no intervalo 2.

Porém, na topologia da Figura 5, uma maior largura de banda não traria nenhum ganho, de modo que uma maior largura de banda seria um desperdício. O ganho ao aumentar a largura de banda é limitado pelo número de descendentes de cada nó.

5. Descrição do Algoritmo

O objetivo do algoritmo é permitir uma rápida coleta de dados aproveitando dos enlaces com largura de banda variáveis. Para fazer isso, os enlaces que têm muitos descendentes e, portanto, constituem o gargalo da rede, são configurados com uma largura de banda maior. Assim, nós que não são folhas têm uma maior largura de banda alocada. Já para os enlaces das folhas são atribuídos a banda de 2 MHz. Assim, o algoritmo antes de atribuir intervalos no escalonamento, calcula o número de descendentes de cada nó. Com base no número de descendentes, uma largura de banda maior é atribuída aos nós.

Por exemplo, em uma árvore de grau 3, se um nó tem 3 descendentes e se há apenas uma largura de banda, o nó tem que alocar 3 intervalos de tempo para enviar

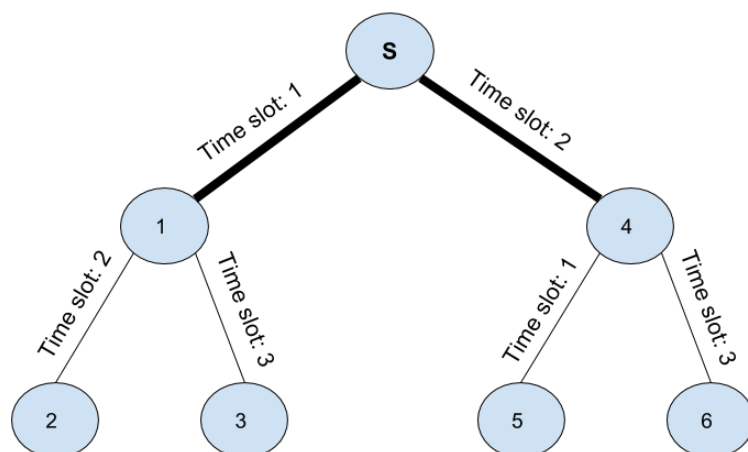


Figura 5. Coleta raw-data em enlaces com o triplo de largura de banda usando o algoritmo proposto e com todos os enlaces que causam interferência removidos. O número nas arestas indica o intervalo de tempo para a transmissão. O tamanho de escalonamento é 3.

Nós Receptores	Quadro 1			Quadro 2		
	SLOT 1	SLOT 2	SLOT 3	SLOT 1	SLOT 2	SLOT 3
SINK	1 (Link 8 MHz)			1 3 4 (Link 8 MHz)	2 5 6 (Link 8 MHz)	
1		3 (Link 2 MHz)	4 (Link 2 MHz)		3 (Link 2 MHz)	4 (Link 2 MHz)
4	5 (Link 2 MHz)		6 (Link 2 MHz)	5 (Link 2 MHz)		6 (Link 2 MHz)

Figura 6. Pacotes enviados em cada intervalo para enlaces com largura de banda de 8 MHz.

dados de cada um de seus filhos mais um para seus próprios dados. No entanto, uma vez que eles são o gargalo na rede, aumentando a largura de banda de seus enlaces, aumenta a capacidade do canal, o que diminui o número de intervalos de tempo necessários como mostrado na Figura 7.

O cálculo do número de descendentes é feito da seguinte maneira (Algoritmo 1). Depois de receber uma solicitação do nó do sorvedouro, os nós encaminham a solicitação em modo broadcast. Quando esta mensagem atinge os nós folhas, elas enviam uma mensagem de resposta aos pais. Os pais, portanto, atualizam um campo (count) que determina o número de descendentes desse enlace e o número total de descendentes a partir das mensagens de resposta. Após isso, envia uma mensagem para seu pai que contém o número total de descendentes. Com base nas mensagens recebidas, o nó pai calcula o número total de descendentes e o envia para o seu pai. Esta operação é executada até que as mensagens atinjam o nó sorvedouro. Assim, os nós conhecem o número de descendentes que eles têm de cada enlace.

Se um nó já tiver x descendentes, então ele precisa de $x + 1$ de largura de banda (para sua mensagem) para economizar no número de intervalos. Além disso, se um nó tiver x descendentes, então seria um desperdício de largura de banda alocar y largura de

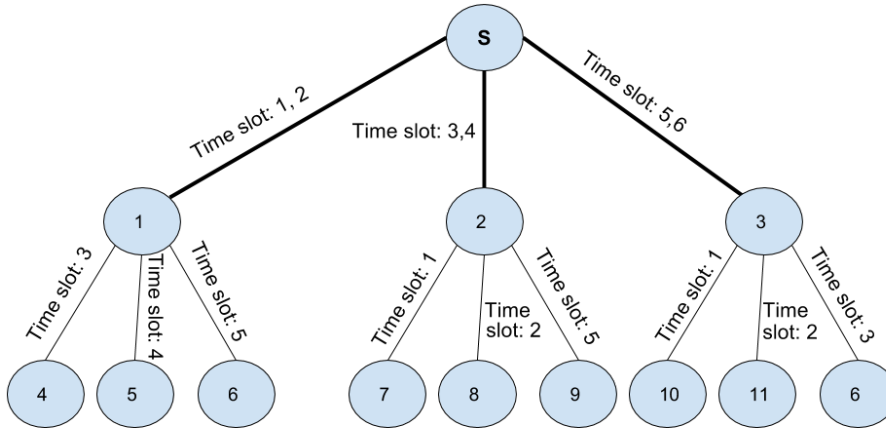


Figura 7. Árvore com 12 nós fontes

banda com $y > x + 1$. O Algoritmo 2 usa isso para alocação de intervalos de tempo ótima.

O Algoritmo 2 apresenta o escalonamento. Em cada iteração, uma aresta e é escolhida na ordem da busca em largura (BFS) começando a partir de qualquer nó e lhe é atribuída uma largura de banda com base no número de descendentes do nó. Depois, são atribuídos intervalos mínimos de nTS que são diferentes de todas as arestas adjacentes. O nTS é calculado com base no número de descendentes da aresta e na árvore e pelo *fator*. O *fator* é calculado com base no ganho da capacidade do canal.

6. Análise de Complexidade

Nesta seção, assumimos um modelo de comunicação síncrono com passagem de mensagem ponto a ponto. Neste modelo, todos os nós começam a executar o algoritmo simultaneamente e o tempo é dividido em rodadas síncronas, ou seja, quando uma mensagem é enviada do nó v para seu vizinho u no intervalo de tempo t , ela deve chegar a u antes do intervalo de tempo $t + 1$.

Teorema 1. *O Algoritmo 2 é ótimo em relação ao número de intervalos de tempo.*

Demonstração. Suponha que o Alg. 2 não seja ótimo. Então existe um algoritmo $Alg_{ÓTIMO}$ de atribuição de intervalos de tempo com menor nTS . Dado que $nTS = \text{teto}((\text{cargaDeTrabalho}(\text{nó}))/\text{fator})$ e a carga de trabalho é constante, o **fator** $_{Alg_{ÓTIMO}}$ deve ser maior que o **fator** $_{Alg. 2}$ para produzir um menor nTS . O fator é diretamente proporcional à *larguraDeBandaAlocada*. Logo, $\text{larguraDeBandaAlocada}_{Alg_{ÓTIMO}}$ deve ser maior que $\text{larguraDeBandaAlocada}_{Alg. 2}$. Mas, $\text{larguraDeBandaAlocada}_{Alg. 2}$ já é a maior largura de banda. Logo, não é possível que $\text{larguraDeBandaAlocada}_{Alg_{ÓTIMO}}$ seja maior que $\text{larguraDeBandaAlocada}_{Alg. 2}$. \square

Teorema 2. *O Algoritmo 1 possui complexidade de número de mensagens $O(|V|)$ e complexidade de tempo de $O(\text{profundidade}(\text{árvore}))$. Essa complexidade de tempo e mensagem é assintoticamente ótima.*

Demonstração. Observe que o algoritmo requer que uma topologia de árvore tenha sido construída pela rede antes que a alocação de intervalo de tempo seja iniciada, ou seja, todo

Algoritmo 1 Cálculo do número de descendentes

```
1: → Init
2:   contador[filhos] = 0;
3:   descendentes = 0;
4:   tempo = 0;
5:   maxTempo = limiar;
6:   envie mensagem ao pai;
7: → temporizador expirado
8:   tempo * = 2;
9:   if (tempo > maxTempo) then
10:    pare;
11: → Recebeu mensagem
12:   if (contador[filho] mudou) then
13:     atualiza(contador[filho]);
14:     descendentes =  $\sum$ contador[filho];
15:     envie mensagem ao pai;
16:     reinicie o temporizador;
```

Algoritmo 2 Busca em Largura

```
1: Input :  $T = (V, E_t)$ 
2: while  $E_t \neq \phi$ 
3:   cargaDeTrabalho = numeroDeDescendentes(no) + 1;
4:   if (cargaDeTrabalho  $\geq$  maxLarguraDeBanda)
5:     then larguraDeBandaAlocada = maxLarguraDeBanda;
6:   else
7:     larguraDeBandaAlocada = mínima largura de banda maior que ou igual a
       cargaDeTrabalho;
8:     fator = larguraDeBandaAlocada / LarguraDeBandaOriginal;
9:     nTS = teto((cargaDeTrabalho(nó)) / fator);
10:    for  $i = 1 : nTS$ 
11:      Atribui mínimo intervalo de tempo  $t$  à aresta  $e$  respeitando restrição de adjacência;
12:       $E_t \leftarrow E_t \setminus e$ ;
13: end while
```

nó sabe quem é seu pai na árvore. Para o cálculo do número de descendentes, todo nó deve enviar uma mensagem para o seu pai depois de receber uma mensagem de seus filhos, o que necessita de $\Omega(|V|)$ mensagens. Além disso, uma mensagem enviada por cada nó folha deve alcançar a raiz, a uma distância de $\leq \text{profundidade}(\text{árvore})$, a qual necessita de $\Omega(\text{profundidade}(\text{árvore}))$ intervalos de tempo. Assim, a complexidade da mensagem e do tempo é $O(|V|)$ e $O(\text{profundidade}(\text{árvore}))$, respectivamente. \square

Teorema 3. *O Algoritmo 2 necessita de $\Omega(\text{profundidade}(\text{árvore}))$ intervalos de tempo e tem complexidade de tempo $O(|E| + |V|)$.*

Demonstração. A busca em largura tem complexidade de tempo $O(|V| + |E|)$. Em cada iteração do Algoritmo 2 (linhas 2-13) uma aresta é escolhida na sequência da busca em largura, iniciada a partir de qualquer nó, e lhe é atribuída uma largura de banda. Após a alocação, há um procedimento de broadcast, uma mensagem (com informações de alocação de intervalos de tempo) deve ser enviada para todo nó pelo seu respectivo pai, a qual necessita de $\Omega(|V|)$ mensagens. Além disso, a mensagem enviada pela raiz deve alcançar todos os nós distantes a $\text{profundidade}(\text{árvore})$ saltos, o que necessita de $\Omega(\text{profundidade}(\text{árvore}))$ intervalos de tempo. \square

7. Resultados

Foram feitas simulações com árvores binárias balanceadas e não balanceadas, e árvores aleatórias. Para o primeiro experimento, foram criadas árvores binárias completamente cheias com tamanho n . Para o segundo experimento, árvores binárias não balanceadas de tamanho n foram criadas. Finalmente, na topologia aleatória, n nós foram distribuídos uniformemente de forma aleatória em um plano da dimensão 1000×1000 . Para cada n , foram geradas 35 instâncias para realizar os experimentos.

Para o nosso conhecimento, não existe trabalhos relacionados que utilizam largura de banda variável para diminuir o número de intervalos de tempo para coletar dados em uma RSSF. Por isso, provemos a largura de banda de 2 MHz para comparação, isto é, quando apenas uma largura de banda é usada na rede, que é o atual estado da arte.

A Figura 8 mostra os resultados para árvores balanceadas. Quando os enlaces entre todos os nós, exceto entre as folhas, têm o dobro de largura de banda, o número de intervalos necessários é reduzido à metade mais um. Por outro lado, quando a largura de banda disponível é duas vezes e três vezes maior, o número de intervalos necessários é reduzido em um terço mais um. Finalmente, quando a largura de banda é duas vezes, três e quatro vezes maior, o número de intervalos necessários é reduzido em um quarto mais um. A largura de banda é alocada de acordo com o número de descendentes. A Figura 9 mostra os resultados para árvores não balanceadas. Os resultados foram semelhantes ao anterior. Uma largura de banda maior irá reduzir o número de intervalos de tempo.

Finalmente, a figura 10 mostra os resultados de experimentos realizados para as instâncias de árvores geradas aleatoriamente com o intervalo de confiança. Os resultados mostram que quanto maior a largura de banda, menor será o número de intervalos necessários. Conforme observado neste resultado, o aumento no número de nós não implica necessariamente em um aumento no número de intervalos de tempo. Conforme discutido no artigo [Incel et al. 2012], a taxa de coleta de dados é muito limitada pela topologia da árvore de roteamento.

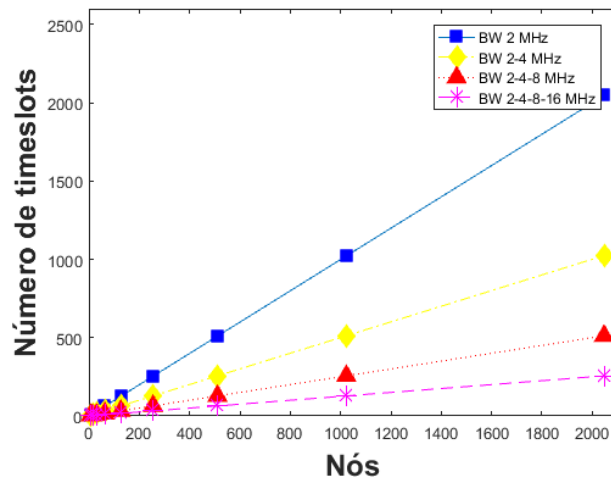


Figura 8. Árvore binária balanceada

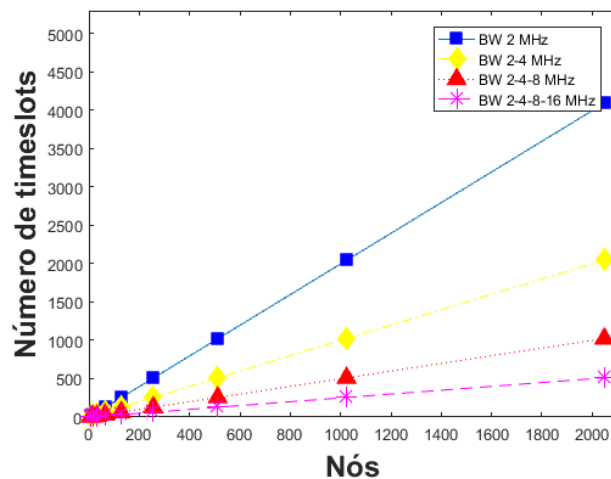


Figura 9. Árvore binária não balanceada

A proposta diminui o intervalo de tempo para realizar a coleta de dados. Em geral, o consumo de energia é dado por tensão * corrente * tempo. Portanto, nosso algoritmo permite reduzir energia deixando o nó menos tempo ligado.

8. Trabalhos relacionados

Muitos protocolos foram propostos na literatura para melhorar a vazão. CodeDrip [Ribeiro Júnior et al. 2017] usa codificação de rede para melhorar a disseminação de dados, mas não propõe alocação de canal. Já o FlushMF [Tavares et al. 2016] usa múltiplas frequências para melhorar a vazão da rede, mas sem considerar maiores larguras de banda. PIP [Raman et al. 2010] e P3 [Doddavenkatappa and Chan 2014] são protocolos de transporte que usam vários canais. Esses protocolos usam um esquema TDMA para sincronizar transmissões acima do pipeline estabelecido. Em [Ahn et al. 2006], os autores implementaram um protocolo MAC CSMA/TDMA híbrido. Para mitigar o efeito de funil, o CSMA/CA puro opera em toda a rede. Já na região de intensidade do funil, o escalona-

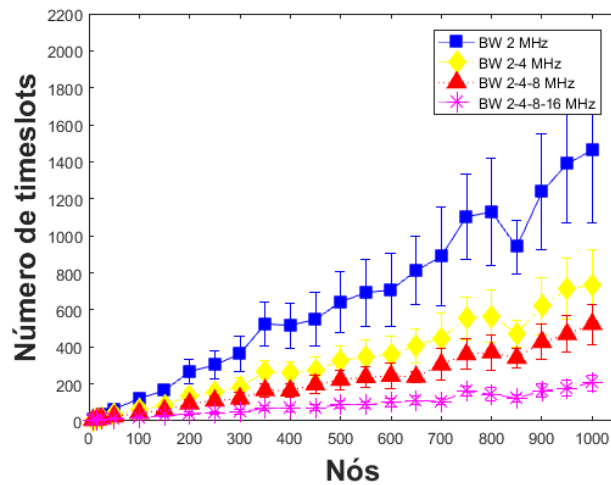


Figura 10. Árvore aleatória

mento TDMA é usado para mitigar esse efeito e melhorar a vazão.

Portanto, é necessário minimizar o tamanho de escalonamento para aumentar a vazão. Muitos protocolos focados no escalonamento TDMA são propostos tais como [Gandham et al. 2006], [Choi et al. 2009], [Incel et al. 2012] e [Chen et al. 2014]. Em [Gandham et al. 2006] os autores propõem um algoritmo de escalonamento de coleta distribuído (DCSA) que requer no máximo $3N$ intervalos de tempo, onde N representa o número de nós na rede. Eles provaram que o limite inferior do número de intervalos de tempo necessários para completar a coleta em uma topologia de linha é $3N - 3$. Depois disso, eles provaram que se N representa o número de nós na rede e n_k representa o número máximo de nós em um ramo, então o limite inferior no número de intervalos de tempo necessários para o escalonamento de coleta em redes multilinhas é dado por $\max(3n_k - 3, N)$. Além disso, para topologias de árvore, o número de intervalos de tempo necessários é dado por $\max(3n_k - 1, N)$.

Em [Choi et al. 2009], o problema do escalonamento é formulado como o problema Minimum Information Gathering Time Problem (MIGTP) baseado em uma representação baseada em grafos da topologia da rede. Eles desenvolveram algoritmos LPIPE para obter explicitamente o escalonamento de transmissão de cada nó enquanto satisfaz todas as restrições. Além disso, eles provaram que o algoritmo LPIPE minimiza o tempo total de transmissão para toda a rede e que o MIGTP é NP-hard para grafos gerais. Uma heurística para soluções subótimas é proposta para grafos gerais.

Em [Wu et al. 2008] foi proposto um protocolo de roteamento baseado em árvore com múltiplos canais, o Tree-based Multi-Channel Protocol (TMCP). Esse protocolo primeiramente acha o número de canais ortogonais disponíveis. O TMCP particiona então a árvore em várias subárvores e aloca canais diferentes para cada subárvore e, em seguida, encaminha cada fluxo de dados somente ao longo de sua subárvore correspondente. Eles mostram que esse esquema funciona bem com um pequeno número de canais e tem um esquema de transmissão muito simples, sem a necessidade de sincronização nos nós. Nesse trabalho, eles definiram e solucionaram o problema de particionar uma rede de sensores em k árvores de vértices disjuntos e minimizando o maior valor da árvore interna de in-

interferência dentre todas as árvores possíveis, chamado de problema PMIT. Eles provaram que o problema é NP-completo e propuseram um algoritmo guloso para resolvê-lo. Após a atribuição do canal, intervalos de tempo são atribuídos aos nós.

Em [Incel et al. 2008], os autores propõem um método de atribuição de canal chamado RBCA, onde eles atribuem os canais aos receptores estaticamente. Com isso, eles removem muitos enlaces que causam interferência, resultando em um menor tamanho de escalonamento.

Em [Incel et al. 2012], é proposto um algoritmo de escalonamento para coleta, chamado JFTSS, onde o escalonamento de canais é acoplado com o escalonamento de intervalo de tempo. O JFTSS oferece uma solução gulosa para construir um escalonamento maximal, de modo que é dito que um escalonamento é *maximal* se ele atende as restrições de adjacência e interferência, e nenhum outro enlace pode ser escalonado para transmissões concorrentes em qualquer intervalo de tempo e canal sem violar as restrições. Além disso, os autores provaram que se todos os enlaces que causam interferência são eliminados, o tamanho do escalonamento para uma coleta one-shot raw data é limitado inferiormente por $\max(2n_k - 1, N)$. Coleta de dados one-shot refere-se à coleta de dados gerados por algum evento assíncrono, no qual os dados são transmitidos individualmente para o sorvedouro. Além disso, os autores propuseram um algoritmo (LOCAL-TIMESLOTASSIGNMENT) para a atribuição desses intervalos de tempo e provaram que, quando os enlaces de interferência são eliminados, o tamanho do escalonamento alcançado por este algoritmo é o $\max(2n_k - 1, N)$.

Em [Yuan and Hollick 2012], os autores estudaram o problema de escalonamento de coleta em uma árvore com vários canais (TCMC). Eles derivam uma solução ótima baseada em programação inteira para o escalonamento do tamanho mínimo e o tamanho do buffer, assim como o escalonamento do tamanho mínimo e do número do canal. Além disso, eles propõem e implementam 4 heurísticas (4H) para a configuração ótima do número de canais e topologia da árvore de roteamento de uma RSSF.

Em [Chen et al. 2014], os autores desenvolveram um protocolo de escalonamento de tempo de coleta de dados multicanais, MCC. Ele usa múltiplos canais e múltiplos caminhos para mitigar interferências e aumentar a vazão. Ele inclui mecanismos para fazer a conectividade de rede, formação da árvore de roteamento, alocação de canal, sincronização de tempo, escalonamento de tempo e coleta de dados.

Em [Pan and Lee 2016], é definido um problema para achar uma árvore em um grafo e atribuição de canal e intervalo sem interferência, tal que a latência da coleta é minimizada, denominada problema de escalonamento de atraso mínimo (MDS). Eles provaram que o problema é NP-completo e propõem um algoritmo heurístico (HA) para resolvê-lo, que contém três fases. A fase de formação da árvore conecta os nós pela árvore do caminho mais curto com restrição de graus. A fase de atribuição de intervalo atribui intervalos a enlaces para obter uma latência ótima (independentemente de interferências). A fase de atribuição de canal atribui canais de frequência a nós para eliminar interferências entre enlaces. Resultados de simulação e implementação mostram que o protocolo reduz a latência de coleta em RSSFs com múltiplos canais.

A Tabela 1 mostra o sumário dos protocolos de escalonamento propostos na literatura.

Tabela 1. Sumário dos protocolos relacionados

Protocolo	Centralizado	Topologia	Maior largura de banda
DCSA[Gandham et al. 2006]	✗	Linha	✗
LPIPE[Choi et al. 2009]	✓	Linha	✗
Greedy[Choi et al. 2009]	✓	Linha	✗
BFS-TIMESLOT ASSIGNMENT[Incel et al. 2012]	✓	Árvore	✗
LOCAL-TIMESLOT ASSIGNMENT[Incel et al. 2012]	✓	Árvore	✗
RBCA[Incel et al. 2012]	✓	Árvore	✗
JFTSS[Incel et al. 2012]	✓	Árvore	✗
TMCP[Wu et al. 2008]	✓	Árvore	✗
4H[Pan and Lee 2016]	✓	Árvore	✗
HA[Yuan and Hollick 2012]	✓	Árvore	✗
FWB	✓	Árvore	✓

9. Conclusões

Este artigo apresentou um algoritmo para escalonamento considerando a extensão do padrão 802.15.4, a qual permite canais com uma maior largura de banda. É proposto e implementado um algoritmo para minimizar o tamanho de escalonamento em redes com rádios que operam com maior largura de banda. A principal ideia é que permitir uma maior largura de banda possibilita transmitir mais dados em um determinado intervalo. Assim, o tamanho de escalonamento é minimizado e, portanto, a vazão é aumentada. Os resultados mostraram que o número de intervalos de tempo necessários para coletar dados de cada nó diminuiu. Essa extensão pode ser adaptada e usada com outros padrões emergentes sem fio relevantes, como o IEEE 802.15.4e TSCH e o WirelessHART.

Referências

- Ahn, G.-S., Hong, S. G., Miluzzo, E., Campbell, A. T., and Cuomo, F. (2006). Funneling-mac: A localized, sink-oriented mac for boosting fidelity in sensor networks. In *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems, SenSys '06*, pages 293–306, New York, NY, USA. ACM.
- Chen, Y., Gomes, P. H., and Krishnamachari, B. (2014). Multi-channel data collection for throughput maximization in wireless sensor networks. In *Mobile Ad Hoc and Sensor Systems (MASS), 2014 IEEE 11th International Conference on*, pages 443–451. IEEE.
- Choi, H., Wang, J., and Hughes, E. A. (2009). Scheduling for information gathering on sensor network. *Wireless Networks*, 15(1):127–140.
- Doddavenkatappa, M. and Chan, M. C. (2014). P3: A practical packet pipeline using synchronous transmissions for wireless sensor networks. In *IPSN-14 Proceedings of the 13th International Symposium on*, pages 203–214. IEEE.
- Gandham, S., Zhang, Y., and Huang, Q. (2006). Distributed minimal time convergecast scheduling in wireless sensor networks. In *Distributed Computing Systems, 2006. ICDCS 2006. 26th IEEE International Conference on*, pages 50–50. IEEE.

- Incel, O. D., Ghosh, A., Krishnamachari, B., and Chintalapudi, K. (2012). Fast data collection in tree-based wireless sensor networks. *IEEE Transactions on Mobile computing*, 11(1):86–99.
- Incel, O. D., Ghosh, A., Krishnamachari, B., and Chintalapudi, K. K. (2008). Multichannel scheduling for fast convergecast in wireless sensor networks.
- Pan, M.-S. and Lee, Y.-H. (2016). Fast convergecast for low-duty-cycled multi-channel wireless sensor networks. *Ad Hoc Networks*, 40:1–14.
- Raman, B., Chebrolu, K., Bijwe, S., and Gabale, V. (2010). Pip: A connection-oriented, multi-hop, multi-channel tdma-based mac for high throughput bulk transfer. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 15–28. ACM.
- Ribeiro Júnior, N. d. S., Tavares, R. C., Vieira, M. A., Vieira, L. F., and Gnawali, O. (2017). Codedrip. *Ad Hoc Netw.*, 54(C):42–52.
- Ruiz, L. B., Correia, L. H. A., Vieira, L. F. M., Macedo, D. F., Nakamura, E. F., Figueiredo, C. M., Vieira, M. A. M., Maia, E. H. B., Câmara, D., Loureiro, A. A., et al. (2004). Architectures for wireless sensor networks. *Proceedings of the 22nd Brazilian Symposium on Computer Networks (SBRC 04)*, pages 167–218.
- Song, W.-Z., Yuan, F., LaHusen, R., and Shirazi, B. (2007). Time-optimum packet scheduling for many-to-one routing in wireless sensor networks. *The International Journal of Parallel, Emergent and Distributed Systems*, 22(5):355–370.
- Tavares, R. C., Vieira, M. A. M., and Vieira, L. F. M. (2016). Flushmf: A transport protocol using multiple frequencies for wireless sensor network. In *Mobile Ad Hoc and Sensor Systems (MASS), 2016 IEEE 13th International Conference on*, pages 192–200. IEEE.
- Vieira, M. A. M., Coelho, C. N., da Silva, D. C., and da Mata, J. M. (2003). Survey on wireless sensor network devices. In *EFTA 2003. 2003 IEEE Conference on Emerging Technologies and Factory Automation. Proceedings*, volume 1, pages 537–544 vol.1.
- Wan, C.-Y., Eisenman, S. B., Campbell, A. T., and Crowcroft, J. (2005). Siphon: Overload traffic management using multi-radio virtual sinks in sensor networks. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems, SenSys '05*, pages 116–129, New York, NY, USA. ACM.
- Werner-Allen, G., Lorincz, K., Ruiz, M., Marcillo, O., Johnson, J., Lees, J., and Welsh, M. (2006). Deploying a wireless sensor network on an active volcano. *IEEE Internet Computing*, 10(2):18–25.
- Wu, Y., Stankovic, J. A., He, T., and Lin, S. (2008). Realistic and efficient multi-channel communications in wireless sensor networks. In *The 27th Conference on Computer Communications. IEEE*. IEEE.
- Yuan, D. and Hollick, M. (2012). Tree-based multi-channel convergecast in wireless sensor networks. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a*, pages 1–9. IEEE.