

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA  
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

**GABRIEL OLIVEIRA REIS**

**ESTRATÉGIAS PARA AVALIAÇÃO DE CONSUMO DE ENERGIA EM  
REDES DE SENSORES SEM FIO**

**TRABALHO DE CONCLUSÃO DE CURSO**

**PONTA GROSSA  
2015**

**GABRIEL OLIVEIRA REIS**

**ESTRATÉGIAS PARA AVALIAÇÃO DE CONSUMO DE ENERGIA EM  
REDES DE SENSORES SEM FIO**

Trabalho de Conclusão de Curso  
apresentado como requisito parcial à  
obtenção do título de Bacharel em Ciência  
da Computação, do Departamento  
Acadêmico de Informática, da Universidade  
Tecnológica Federal do Paraná.

Orientadora: Prof<sup>ª</sup>. Dr<sup>ª</sup>. Tânia Lúcia Monteiro

**PONTA GROSSA  
2015**



Ministério da Educação  
**Universidade Tecnológica Federal do Paraná**  
Câmpus Ponta Grossa  
Departamento Acadêmico de Informática  
Bacharelado em Ciência da Computação



---

## TERMO DE APROVAÇÃO

### ESTRATÉGIAS PARA AVALIAÇÃO DE CONSUMO DE ENERGIA EM REDES DE SENSORES SEM FIO

por

GABRIEL OLIVEIRA REIS

Este Trabalho de Conclusão de Curso foi apresentado em 09 de junho de 2015 como requisito parcial para a obtenção do título de Bacharel em Ciência da Computação. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

---

Tânia Lúcia Monteiro  
Prof<sup>a</sup>. Orientadora

---

Gleifer Vaz Alves  
Membro titular

---

Lourival Aparecido de Góis  
Membro titular

- O Termo de Aprovação assinado encontra-se na Coordenação do Curso –

## **AGRADECIMENTOS**

Estar sozinho é a dor mais forte. O que eu seria se não tivesse alguém para me alimentar, vestir roupas e me educar? Eu agradeço a toda a minha família por tanto tempo de zelo e dedicação comigo, não tem como eu pagá-los.

Agradeço a minha mãe sempre presente e prestativa, minha maior motivação. E aos meus avós, que mesmo já não estando nesse mundo, me auxiliaram com aconchego, educação e amor.

Agradeço também a minha orientadora pela disposição e a todos da minha universidade, professores e amigos, que me apoiaram durante esses anos letivos.

## RESUMO

REIS, Gabriel O. **Estratégias para Avaliação de Consumo em Redes de Sensores Sem Fio**. 2015. 59 páginas. Trabalho de Conclusão de Curso Bacharelado em Ciência da Computação - Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2015.

A utilização de algoritmos eficientes de roteamento para aumento do tempo de vida de uma rede de sensores sem fio é indispensável durante uma aplicação da mesma. Conforme o tempo passa, nós sensores vão se desgastando e a degeneração da rede se torna iminente, impedindo a coleta de dados por um período de tempo maior. As estratégias empregadas por um algoritmo que gera a árvore de menor caminho para uma rede foram usadas como ponto de partida para criação de dois novos algoritmos que buscam aumentar o tempo de vida da rede, no intuito de descobrir se as técnicas presentes no algoritmo são viáveis para manter a rede operante por um tempo considerável. Ao se comparar diversas topologias levando em consideração um algoritmo criado para aumento de tempo de vida da rede (*Distance-Based*), concluiu-se que ao menos um dos algoritmos propostos se destaca por exercer esta nova função em determinadas topologias, além de abrir possibilidades para diversos outros estudos futuros.

**Palavras-chave:** Redes de sensores sem fio. Bellman-Ford Modificado. Tempo de vida da rede. *Distance-Based*

## ABSTRACT

REIS, Gabriel O.. **Strategies for Consume Evaluation in Wireless Sensor Networks**. 2015. 59 pages. Trabalho de Conclusão de Curso Bacharelado em Ciência da Computação - Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2015.

The algorithms' applications for increase lifetime on Wireless Sensor Networks are necessary for their own applications. As time passes by, sensor networks will be wearing and network's degeneration becomes predictable. The strategies in an algorithm that generate the shortest path tree for topology creation were used as startup for two algorithms' development in order for discover if the strategies are suitable for lifetime applications in wireless sensor networks. When comparing different topologies, based on the performance of an algorithm designed to increase the network lifetime (*Distance-Based*), it was concluded that at least one of the proposed algorithms stands for exercising the new function in certain topologies, and opens many possibilities for other future studies.

**Keywords:** Wireless sensor networks. Modified Bellman-Ford. lifetime network. Distance-Based.

## LISTA DE FIGURAS

|  |    |
|--|----|
| Figura 1 - Posicionamento dos nós Sensores .....                             | 9  |
| Figura 2 - Divisão de Camadas de RSSF .....                                  | 16 |
| Figura 3 - Hardware de um nó Sensor .....                                    | 17 |
| Figura 4 - Divisão Referente ao Consumo de Energia gasta .....               | 19 |
| Figura 5 - Topologia inicial de uma RSSF .....                               | 23 |
| Figura 6 - Estrutura montada para Roteamento .....                           | 23 |
| Figura 7 - Topologia Grid e distribuição de cargas .....                     | 29 |
| Figura 8 - Distâncias dos nós pelo <i>Distance-Based</i> .....               | 29 |
| Figura 9 - Resultado da aplicação do <i>Distance-Based</i> com $a = 3$ ..... | 31 |
| Figura 10 - Transmissão de informação pelo Alg 2 - MBFCPA .....              | 46 |
| Figura 11 - <i>Distance-Based</i> Alterado com $a = 3$ .....                 | 50 |
| Figura 12 - Número de informações obtidas pelos algoritmos .....             | 51 |
| Figura 13 - Alteração de custos dos enlaces na rede de 30 nós .....          | 53 |
| Figura 14 - Alteração de custos dos enlaces na rede de 50 nós .....          | 53 |

## LISTA DE SIGLAS

|        |   |
|--------|---|
| ADC    | Analog-to-Digital Conversor                 |
| Bat    | Bateria                                     |
| C      | Custo                                       |
| CP     | Caminho Percorrido                          |
| d      | Distância                                   |
| DBF    | Distributed Bellman-Ford                    |
| MBF    | Modified Bellman-Ford                       |
| MBFCPA | Modified Bellman-Ford com Pais Alternativos |
| MBFSPA | Modified Bellman-Ford sem Pais Alternativos |
| NP     | Nó Pai                                      |
| OP     | Operação                                    |
| PA     | Pais Alternativos                           |
| PACP   | Pais Alternativos no Caminho Percorrido     |
| PAP    | Pais Alternativos Possíveis                 |
| RF     | Rádio Frequência                            |
| RSSF   | Redes de Sensores Sem Fio                   |
| SPT    | Shortest Path Tree                          |
| T      | Carga                                       |
| TCP    | Transmission Control Protocol               |
| W      | Peso  |



## SUMÁRIO

|  |    |
|--|----|
| <b>1. INTRODUÇÃO</b> .....                                   | 9  |
| 1.1 OBJETIVOS .....  | 12 |
| 1.2 ORGANIZAÇÃO .....  | 13 |
| <b>2. REFERENCIAL TEÓRICO</b> .....                          | 15 |
| 2.1 DIVISÃO DE CAMADAS DE RSSF .....                         | 15 |
| 2.2 LIMITAÇÕES DE HARDWARE .....                             | 17 |
| 2.3 PROTOCOLOS PARA CONSUMO DE ENERGIA .....                 | 20 |
| <b>3. TRABALHOS RELACIONADOS</b> .....                       | 22 |
| 3.1 ALGORITMO MODIFIED BELLMAN-FORD .....                    | 22 |
| 3.1.1 Detalhes da Execução do algoritmo MBF .....            | 25 |
| 3.2 ALGORITMO <i>DISTANCE-BASED</i> .....                    | 27 |
| 3.2.1 Escopo de Funcionamento do <i>Distance-Based</i> ..... | 28 |
| <b>4. PREMISSAS</b> .....                                    | 32 |
| 4.1 TOPOLOGIAS USADAS .....                                  | 32 |
| 4.2 CONSIDERAÇÕES SOBRE ENERGIA E CONTROLE DE CUSTOS. ....   | 33 |
| <b>5. DESCRIÇÃO DOS ALGORITMOS</b> .....                     | 36 |
| 5.1 ALG1 - MBFSPA .....                                      | 37 |
| 5.2 ALG 2 - MBFCPA .....                                     | 40 |
| 5.3 <i>DISTANCE-BASED</i> ALTERADO .....                     | 48 |
| <b>6. TESTES</b> .....                                       | 51 |
| <b>7. CONCLUSÃO</b> .....                                    | 55 |
| 7.1 TRABALHOS FUTUROS .....                                  | 56 |
| <b>REFERÊNCIAS</b> .....                                     | 57 |

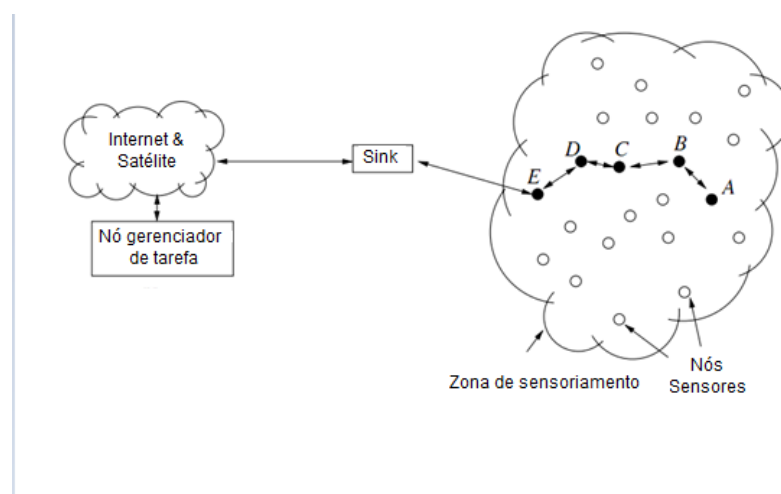
## 1. INTRODUÇÃO

Redes de sensores sem fio (RSSF) possuem uma grande quantidade de aplicações em nosso meio, a ponto de estarem se tornando parte do nosso cotidiano, uma vez que diversas aplicações estão sendo desenvolvidas principalmente na área de monitoramento (ZHOU et al., 2001).

Os dispositivos de uma RSSF são chamados de nó sensor, eles possuem pequeno porte, baixo processamento de dados e são comumente utilizados para monitorar algum evento externo, semelhante a um sensor comum. A função principal destes nós é coletar dados de um ambiente para algum estudo (HOWITT; WANG, 2004).

A posição onde se deve colocar cada um dos sensores não precisa ser pré-determinada, permitindo que terrenos de difícil acesso ou desnivelados possam ser monitorados.

Os nós são posicionados em uma região chamada de zona de sensoriamento. Cada um deles tem a capacidade de coletar dados e enviá-los para nós *sink* conforme mostrado na Figura 1. Estes nós “*sink*” possuem maior capacidade de processamento em relação aos demais nós e servem como interseção da origem (dados coletados pela zona de sensoriamento) e destino (aplicação do usuário que utilizará as informações para algum propósito).



**Figura 1 - Posicionamento dos nós sensores**  
 Fonte: Adaptado de AKYILDIZ; VURAN (2010)

O limitado alcance da comunicação entre os nós faz com que estes devam se comunicar entre si, transmitindo a informação obtida em uma coleta, até chegar a um nó com raio de alcance ao *sink*.

Em geral, na utilização de RSSF o controle do consumo de energia de cada um dos nós é um dos atributos fundamentais durante uma aplicação, devido aos recursos limitados dos nós sensores. Portanto, diferentemente de redes tradicionais que buscam melhorar desempenho de transmissão e diminuição de atraso de informação, em RSSF foca-se em conservar energia (HUSSAIN; ISLAM, 2007).

Como para muitas aplicações o reabastecimento energético dos sensores é impossível, o tempo de vida de uma RSSF depende do tempo de vida da bateria de seus nós. Além disso, os nós são expostos a frequentes falhas, decorrentes da inacessibilidade do local onde eles são instalados, trazendo um grande desafio para manutenção da topologia (CHANG; TASSIULAS, 2000).

Como o objetivo principal de uma RSSF é monitorar um fenômeno de interesse de forma eficiente, necessita-se acompanhar o consumo de energia dos nós, garantindo o maior tempo possível de funcionamento da rede sem haver perda de qualidade da informação transmitida. Por isso, diversas formas de roteamento em redes comuns acabam não sendo viáveis, pois torna-se relevante considerar o consumo de energia no roteamento de RSSF (AKYILDIS; VURAN, 2010).

Uma *Shortest Path Tree* (SPT) representa uma técnica importante a ser utilizada em ambientes de RSSF, uma vez que calcula a menor rota necessária para que um dado coletado, em um vértice qualquer, chegue até o destino previsto em uma rede. Como mostrado na Figura 1 a zona de sensoriamento contém diversos nós sensores, e cada um deles precisa saber com quem se comunicar dentre todos os vizinhos que estão ao seu alcance numa transmissão de dados. A SPT serve exatamente para que a escolha do vizinho de cada nó seja a melhor possível.

Existem diversos algoritmos para calcular a SPT de uma RSSF, dentre eles o *Modified Bellman-Ford* (MBF) (LACHOWSKI, 2014), uma versão adaptada do já conceituado *Distributed Bellman-Ford* (DBF) (AWERBUCH et al., 1994).

Ambos constroem uma SPT bastante similar, porém o DBF sofre com sobrecarregamento (*overhead*) de mensagens na rede, o que atrapalha a comunicação dos nós. Isto ocorre porque durante o processo de construção da SPT, o algoritmo constantemente vai encontrando rotas com custos menores para os nós. Sempre quando isto acontece, deve-se comunicar toda a rede para que os outros

nós saibam o novo valor de custo para se chegar até este nó com custo atualizado. No MBF nem sempre quando um caminho de menor custo é encontrado haverá este processo de comunicação, pois existe uma verificação para saber o quanto o novo caminho encontrado é melhor em relação ao caminho atual. Graças a este processo de verificação, trocas não tão eficientes para a rede são evitadas, o que diminui o *overhead* da rede (LACHOWSKI, 2014).

O MBF atua apenas na criação da SPT, porém o desafio da manutenção da topologia é no mínimo tão importante quanto este, pois após o início de uma aplicação, alguns nós gastarão mais energia do que outros. Por exemplo, ao se considerar um nó próximo ao *sink*: durante boa parte do seu tempo de vida ele estará operando para transportar a informação que chega até ele, já que em uma RSSF deve-se mandar a informação coletada sempre para o mesmo nó. O nó *sink*, forma um funil (concentração de tráfego) na rede e, assim, nós próximos do *sink* estarão sempre sendo requisitados.

Caso um nó fique sem energia, a comunicação de outros nós com este nó fica impossibilitada, e qualquer tipo de informação na rede que seja mandada para ele será perdida, tornando o desempenho da rede comprometido. A fim de evitar tais situações algumas medidas precisam ser tomadas para sua sobrevivência ou para, pelo menos, evitar que a rede perca estas informações. Tais providências serão aqui denominadas de “manutenção de topologia”.

O MBF possui também algumas outras características não presentes no DBF como a presença de nós alternativos para cada nó, possibilitando que um nó ao mandar uma informação para um nó sem energia, a informação possa ser mandada para um destes nós alternativos, que nada mais são do que os outros vizinhos do nó em questão, evitando que a informação se perca. Ou seja, enquanto o nó possuir pelo menos um vizinho que possua energia, a informação carregada por ele poderá ainda ser transmitida.

Aproveitando-se deste cenário, foram criados dois algoritmos Alg1 – MBFSPA (*Modified Bellman-Ford Sem Pais Alternativos*) e Alg2 – MBFCPA (*Modified Bellman-Ford Com Pais Alternativos*). Em ambos os algoritmos propostos, uma informação é gerada na rede e deve ser transportada pela rede até que a informação chegue ao destino (nó *sink*). O caminho utilizado e os nós selecionados para transportar a informação serão determinados pelo MBF.

Ou seja, a partir de uma topologia qualquer de rede, aplica-se o MBF para determinar a SPT da mesma. A partir da SPT e da informação de quem é o nó pai de cada nó, aplica-se os dois algoritmos propostos, onde o caminho montado pela SPT será utilizado para transportar uma informação coletada.

O algoritmo Alg1 – MBFSPA direciona a informação integralmente pelo caminho gerado pela SPT, enquanto que o Alg2 - MBFCPA utiliza os pais alternativos dos nós, quando necessário, numa transmissão.

Para comparar o desempenho destes algoritmos, com relação ao tempo de vida da topologia, encontrou-se na literatura um algoritmo criado especialmente para esta função, o *Distance-Based* (CHANG; CHANG, 2008). O algoritmo aproxima nós sensores que recebem muitas informações da rede e afasta os que recebem poucas. Como a distância de envio influencia no gasto de energia de um sensor, os nós mais afastados gastarão mais energia em cada transmissão, equilibrando o consumo gasto pela rede no geral. Além deste, criou-se uma versão alterada do *Distance-Based* onde as distâncias dos nós são menores ainda que as encontradas no *Distance-Based* original, diminuindo ainda mais o consumo.

A escolha do *Distance-Based* se deve à facilidade de implementação do algoritmo e por seu desempenho ser considerado mediano (ISHMANOV et al., 2011). A escolha de um algoritmo com eficiência muito alta ou baixa para manutenção de topologia poderia não representar o real desempenho dos algoritmos propostos neste trabalho. Caso estes algoritmos apresentem desempenho superior a um algoritmo considerado ruim para manutenção de topologia, o resultado poderia não ser considerado como satisfatório, e vice-versa.

## 1.1 OBJETIVOS

O objetivo principal deste trabalho de conclusão de curso é utilizar as características do algoritmo MBF em dois algoritmos separados (Alg1 – MBFSPA e Alg2 – MBFCPA), a fim de avaliar se os mesmos se mostram viáveis com relação a manutenção de topologia.

Objetivos específicos:

- Verificar se a utilização dos nós alternativos considerados pelo Alg2 - MBFCPA garantem maior sobrevivência da rede durante uma aplicação em relação ao Alg1 - MBFSPA;
- Encontrar formatos de topologias que melhores se encaixam no Alg1 – MBFSPA e Alg2 - MBFCPA através das análises de desempenho;

Entende-se por manutenção de topologia, estratégias adotadas pelos algoritmos a fim de manter que a rede fique operante pelo maior tempo possível, como por exemplo, diminuir a distância entre os nós para que o gasto energético de uma transmissão seja menor ou a utilização de um nó alternativo para mandar uma informação quando um determinado nó não é capaz de recebê-la.

Os algoritmos devem fazer uma informação coletada trafegar pela RSSF até o *sink*, através da SPT montada pelo MBF. Para efeito de comparação, o processo de manutenção de topologia empregado pelo *Distance-Based* e sua versão alterada serão utilizadas nos mesmos cenários de simulação do Alg1 – MBFSPA e do Alg2 - MBFCPA.

Serão considerados cenários de rede com quantidade de nós variando entre 25 e 100 nós, com cada nó possuindo uma densidade máxima de até 6 vizinhos. Qualquer nó da zona de sensoriamento está sujeito a receber uma informação e transmiti-la até o *sink*. O critério adotado para determinar o desempenho de cada algoritmo será o número de informações recebidas pelo *sink*, enquanto a rede se mantém operante.

## 1.2 ORGANIZAÇÃO

O trabalho está organizado da seguinte forma: a seção 2 explica as características dos nós sensores, suas limitações e consumo de energia. Na seção 3 são apresentados os trabalhos relacionados que tratam os algoritmos MBF e *Distance-Based*, um algoritmo simples de manutenção de topologia. A seção 4 descreve as premissas utilizadas pelas simulações e a seção 5 descreve os algoritmos propostos para o desenvolvimento deste trabalho. A seção 6 exhibe os

testes realizados. A seção 7 é a conclusão deste trabalho e as proposições de trabalhos futuros.

## 2. REFERENCIAL TEÓRICO

Neste capítulo serão abordados os fundamentos teóricos sobre RSSF necessários para o entendimento do trabalho. Primeiramente mostrar-se-á a divisão de camadas de uma RSSF para melhor entendimento de suas funções e capacidades.

Na sequência são detalhadas as características dos nós sensores, bem como cada um de seus componentes e o consumo de energia por eles apresentado. Por fim, são explicadas algumas das estratégias adotadas por protocolos de consumo de energia, descrevendo algumas situações onde suas utilizações se tornam necessárias.

### 2.1 DIVISÃO DE CAMADAS DE RSSF

Segundo Akyildiz e Vuran no livro *Wireless Sensor Networks* - 2010, a pilha de protocolos utilizada pelos nós sensores consiste das camadas física, enlace de dados, rede, transporte e de aplicação. Além destas, há os planos de sincronização, localização, gerenciamento de topologia, controle de energia, gestão de mobilidade e de gerenciamento de tarefas, como exibido na Figura 2.

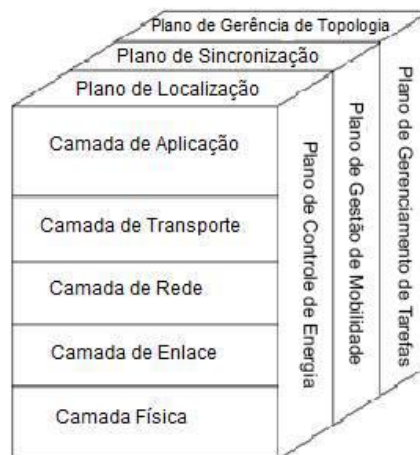
A camada física trabalha com modulação, transmissão e técnicas de recepção de dados. A camada de enlace se responsabiliza em garantir uma comunicação satisfatória através de técnicas de controle de erros e gerência de acesso ao canal pelo protocolo de *Media Access Control* (MAC) para minimizar colisões, podendo-se construir diferentes tipos de programas na camada de aplicação conforme as necessidades impostas durante a utilização dos sensores. A camada de rede encaminha dados recebidos para a camada de transporte, que ajuda a manter o fluxo de dados, controlando e organizando os pacotes recebidos.

O plano de controle de energia gerencia como um nó sensor utiliza sua energia e o plano de gestão de mobilidade detecta e registra o movimento dos nós sensores, de modo a ter sempre um caminho até o usuário, conseguindo manter um controle de seus vizinhos e equilibrar a demanda energética entre eles. Já o gerenciamento de tarefas planeja as tarefas de sensoriamento dentro de uma região



específica. Nem todos os nós sensores de uma região são necessários para monitorar um evento ao mesmo tempo: alguns deles executam mais requisições do que outros, dependendo do seu nível de energia e de suas posições em relação ao fenômeno monitorado. (CHANG; TASSIULAS, 2000).

Uma RSSF é mais eficiente se os nós sensores colaborem uns com os outros, prolongando o tempo de vida da rede (AKYILDIZ et al., 2001).



**Figura 2 - Divisão de camadas de RSSF**  
**Fonte: Adaptado de AKYILDIZ; VURAN (2010)**

Ao contrário dos protocolos como TCP, o endereçamento em uma RSSF não é realizado com endereços globais para cada dispositivo (nós sensores), pois para redes de sensores a viabilidade de uma rota no exato momento em que haverá uma transmissão de dados é o ponto chave para se indicar o destino do pacote de dados (AKYILDIZ; VURAN, 2010).

O desenvolvimento de novos protocolos para a camada de transporte também é necessário pelas restrições de hardware do nó sensor. Como eles são limitados em termos de processamento, armazenamento e consumo de energia, os protocolos da camada de transporte devem explorar as capacidades de colaboração destes nós sensores até a informação chegar ao nó *sink* (AKYILDIZ; VURAN, 2010).

## 2.2 LIMITAÇÕES DE HARDWARE

Um nó sensor é geralmente composto de quatro componentes básicos: unidades de detecção, de processamento, de transmissão e de energia, conforme ilustrado na Figura 3. Além disso, componentes adicionais podem ser integrados dentro do nó sensor dependendo da aplicação.

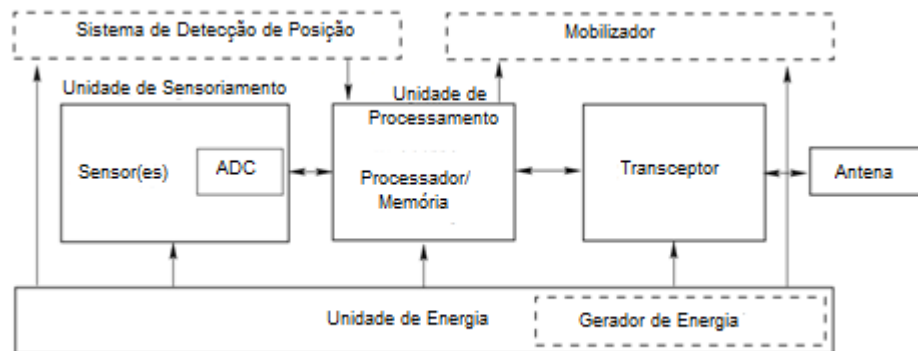


Figura 3 – Hardware de um nó sensor  
Fonte: Adaptado de AKYILDIZ et al. (2001)

- **Unidade de sensoriamento:** ou de detecção, é o componente principal dos sensores sem fios, responsável por captar informações sobre o mundo físico. Cada sensor é responsável pela coleta de informações de um determinado tipo como temperatura, umidade e luz. O conversor A/D (ADC) é utilizado para converter estes dados recebidos em valores digitais para o sensor;
- **Unidade de processamento:** controla e gerencia o nó sensor, sendo constituída de uma memória integrada utilizada para executar os procedimentos que permitem que o nó sensor realize suas operações;
- **Transceptor:** responsável pela comunicação entre quaisquer dois nós de sensores sem fio, onde são implementados os procedimentos necessários para converter os bits recebidos em rádio frequência (RF). Essencialmente, a RSSF está ligada à rede através desta unidade;
- **Unidade de energia:** basicamente composto por uma bateria, porém outras fontes de energia também poder ser usadas. Cada componente

do nó é alimentado através desta unidade com uma capacidade limitada, exigindo eficientes meios de consumo de energia do dispositivo;

- **Sistema de detecção de posição:** a maioria das aplicações de RSSF necessitam de conhecimento da localização física de um nó. Este sistema pode ser constituído por um módulo de GPS ou algum algoritmo responsável por encontrar a localização do sensor;

- **Mobilizador:** utilizado quando se deseja mover nós sensores para realizar tarefas específicas. O suporte à mobilidade necessita de recursos energéticos e deve ser fornecido de forma eficiente, operando junto à unidade de detecção e o processador para controle da movimentação dos nós;

- **Gerador de energia:** dispositivo extra de energia usado em aplicações mais robustas. Para aplicações ao ar livre, as células solares são utilizadas como gerador de energia, como proposto em Roundy et al., 2003.

O tamanho de um nó sensor normalmente se compara a uma caixa de fósforos (INTANAGONWIWAT et al., 2000), podendo em certas aplicações ser menor do que um centímetro cúbico (POTTIE; KAISER, 2000). Além do tamanho, existem também algumas outras limitações rigorosas para nós sensores como: operar em alta densidade; ter baixo custo de produção e ser dispensável; ser autônomo e operar sem intervenção humana e ser adaptável para o ambiente (KAHN et al., 1999).

Todas estas restrições se devem a seu baixo processamento e alcance de transmissão, das taxas de falhas provocadas devido a geografia do terreno e da necessidade de utilização em locais de pouco acesso, o que inviabiliza a troca dos sensores. Neste trabalho, as estratégias de mobilidade e uso de bateria extra não são exploradas devido aos custos extras gerados pela sua imposição aos nós sensores.

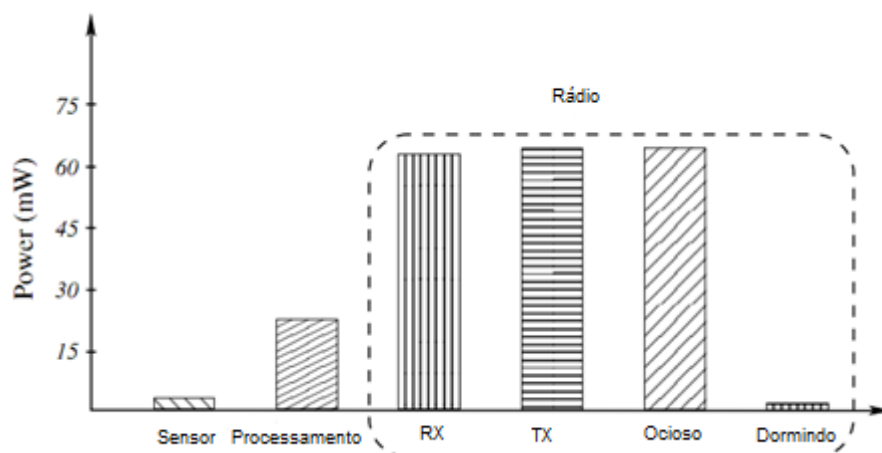
### 2.2.1 Consumo de Energia

O consumo de energia é um ponto chave em RSSF. Para muitas aplicações, a rede é montada em locais inacessíveis e com isto, o tempo de vida da rede, que é

o tempo para o qual a rede permanece operacional, também se torna limitado (PANTAZIS et al., 2013).

Um nó sensor é responsável por receber os dados enviados por seus vizinhos (RX) e transmiti-los para um dos seus vizinhos de acordo com as decisões de encaminhamento (TX). A principal tarefa de um nó sensor em um campo de sensores é detectar eventos, realizar o processamento de dados e, em seguida, transmitir os dados (AKYILDIZ et al., 2001). O consumo de energia pode, portanto, ser dividido em três domínios: a detecção, o processamento de dados e a comunicação, que são realizadas pelos sensores, pela CPU, e pelo rádio, respectivamente.

A divisão referente ao consumo de energia de um nó sensor é mostrada na Figura 4. Pode-se observar que, entre estes três domínios, um nó sensor gasta maior parte de sua energia durante a comunicação.



**Figura 4 – Divisão referente ao consumo de energia gasta**  
**Fonte: Adaptado de AKYILDIZ; VURAN (2010)**

O gasto energético de um sensor quando ele está ocioso é equivalente ao as operações de Rx ( recepção) e Tx (transmissão). Este estado corresponde ao momento onde o rádio do sensor está ligado para receber e transmitir informações do ambiente, porém nenhum de seus vizinhos se comunica com ele, gastando energia inutilmente.

Em contrapartida, quando o rádio do nó sensor é colocado para dormir, a energia gasta por ele é mínima. Dessa forma, utilizar este estado de dormência durante os momentos onde ocorreria ociosidade do rádio é uma das práticas mais

recorrentes e indispensáveis para uma RSSF, sendo muito comuns desde os primeiros protocolos de roteamento que surgiram (SALEH et al., 2013).

### 2.3 PROTOCOLOS PARA CONSUMO DE ENERGIA

Os protocolos voltados à economia de energia em uma RSSF buscam, através de determinadas métricas, diminuir o consumo energético dos nós sensores a fim de garantir que a rede fique operante pelo maior tempo possível (SALEH et al., 2013).

Inicialmente, pensava-se que por ser possível se identificar os componentes que mais gastavam energia, a medida preventiva para criação de um bom protocolo seria apenas cuidar do consumo de cada nó individualmente. Porém, isto não é uma verdade absoluta. Por exemplo, um protocolo de RSSF ao monitorar eventos do mundo real: após calcular a SPT de uma rede, deixa os nós operarem sem uma intervenção futura, e apenas toma algumas medidas de precaução como colocá-los em dormência, quando há inatividade. Esta prática poderá apresentar graves problemas durante uma aplicação.

Como os eventos são estocásticos, alguns nós poderão captar mais informações do ambiente do que outros, tornando o caminho por onde as informações são enviadas, mais utilizados do que outros caminhos e, por consequência, estes nós pertencentes a este trajeto gastarão mais energia, causando esgotamento de recursos em uma parte da rede (ANASTASI et al., 2008).

Ou seja, deve-se considerar além das características físicas dos sensores, que a rede se mantenha conectada e funcional, garantindo que o consumo de energia seja equilibrado em toda sua extensão.

Sobre o tempo de vida de uma rede, não existe uma definição padrão formal, devido ao grande número de aplicações onde as RSSF podem ser empregadas. Alguns autores consideram o tempo de vida de uma rede como o tempo total até a morte do primeiro nó desta rede. Outros consideram o tempo de vida até uma fração da rede ficar sem recursos. Outros ainda consideram até o momento onde a rede não consegue mais realizar as operações que fora proposta, independentemente de quantos nós tenham ficado sem energia (PANTAZIS et al., 2013).

Como evidenciado em Pantazis et al. (2013), há diversas categorias de redes como, por exemplo: redes onde alguns nós são mais potentes em termos de alcance de transmissão e capacidade da bateria, do que os outros, e servem como um destino local para envio das informações para os nós próximos a ele. Além disso, há redes com múltiplos *sinks* a fim de expandir os caminhos possíveis para se enviar uma informação.

A diversidade de modelos de topologias existentes cria a possibilidade para inserção de protocolos diversificados para atender esta demanda. Porém, a grande maioria destas imposições acarreta em aumento do custo para gerir a rede.

### 3. TRABALHOS RELACIONADOS

Neste capítulo serão apresentados os algoritmos que possuem relação com o desenvolvimento do trabalho proposto. Estes algoritmos são: *Modified Bellman-Ford* (MBF) e *Distance-Based*.

O MBF foi desenvolvido para criação de rotas de menor custo, sendo uma adaptação de um algoritmo chamado *Distributed Bellman-Ford* (DBF). Já o *Distance-Based* é um algoritmo que utiliza uma metodologia fixa para determinar os custos de transmissão, que trata da distância entre cada um dos nós em relação ao *sink*: quanto mais distante do *sink*, maior a distância entre um nó e outro e quanto mais próximo, menor a distância entre os nós.

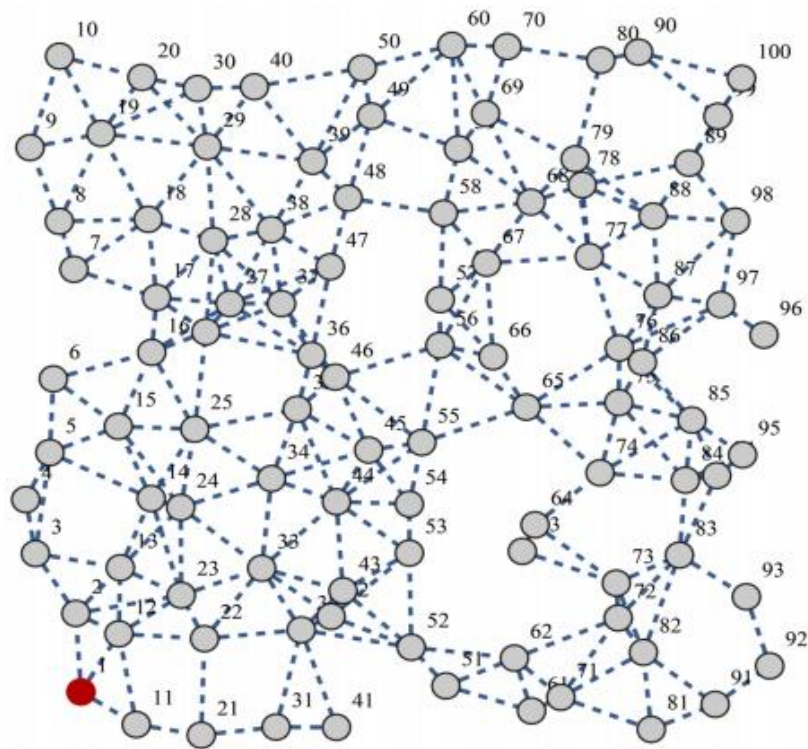
#### 3.1 ALGORITMO MODIFIED BELLMAN-FORD

Diferentemente de uma rede *Ad hoc*, em RSSF tem-se um nó para onde as informações convergem e devem ser enviadas: o *sink*. O caminho utilizado por um nó até o *sink* é montado por algoritmos de roteamento utilizando-se métricas específicas (HUSSAIN; ISLAM, 2007).

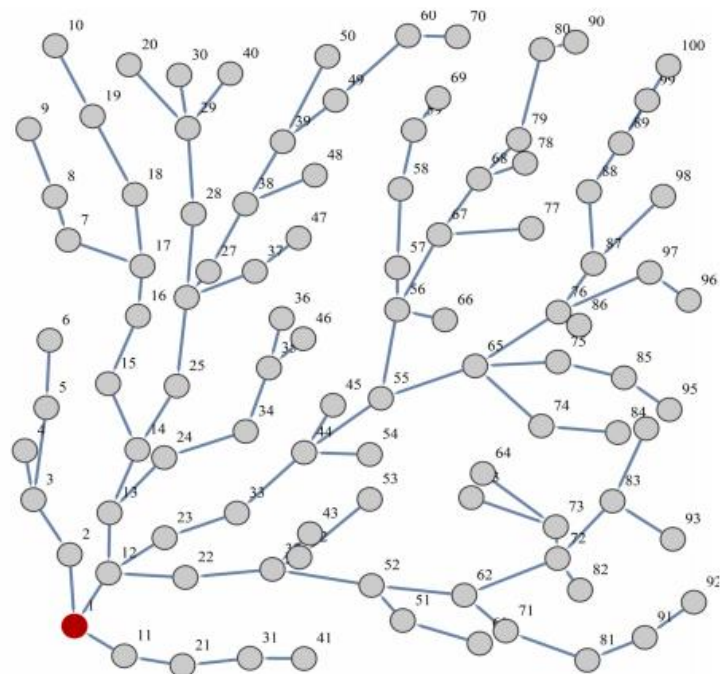
Os caminhos montados do nó até o *sink* formam uma árvore (LACHOWSKI, 2014). A partir de um modelo inicial de topologia construída com 100 nós, apresentado na Figura 5, representa-se na Figura 6 uma SPT montada para roteamento.

A SPT é a árvore montada de melhor desempenho segundo alguma métrica. Por exemplo, se a métrica for a menor distância entre enlaces, a SPT deve ser uma árvore em que para todo nó da rede, o vizinho selecionado seja aquele que possui menor distância em relação a ele.

O nó mais escuro é o nó *sink* enquanto os demais nós são nós sensores comuns. Para determinar a rota de cada nó, utilizam-se métricas como distância do enlace, custo energético e diversas outras (AWERBUCH et al., 1994).



**Figura 5 - Topologia inicial de uma RSSF**  
**Fonte: LACHOWSKI (2014)**



**Figura 6 - Estrutura montada para Roteamento**  
**Fonte: LACHOWSKI (2014)**



O algoritmo DBF trabalha com busca em largura, onde um nó se comunica com cada um de seus filhos e logo a seguir, cada filho, por sua vez, se comunica com seus próprios filhos (netos do primeiro nó). Este processo é realizado por todos os nós do grafo e seu propósito é descobrir qual é o nó pai para cada um destes nós. O nó pai indica com qual nó o sensor atual deverá se comunicar. Toda vez que um nó  $i$  escolhe um novo nó pai  $j$ , deve-se atualizar seu peso e enviar uma mensagem *broadcast* por toda a rede, informando do ocorrido para o resto da rede. Este procedimento é de grande importância para que os outros nós vizinhos de  $i$  em futuras iterações do algoritmo vejam se com este novo valor a comunicação com  $i$  é mais viável do que com seus respectivos nó pai atuais.

Esta busca em largura inicia-se pelo *sink*. Partindo deste ponto, verificará entre todos os vizinhos dele, qual possui a melhor métrica que, neste caso, será quem possui o menor custo de enlace (distância entre os nós sensores). Quem possuir o menor custo terá como nó pai o *sink*, pois como a informação vem dos outros nós em direção a ele, quando chegar a este determinado vizinho, ele então saberá que deve se comunicar com o *sink*.

A comunicação entre um nó e seus filhos será, portanto, visando sempre encontrar a menor distância de um sensor para outro, com a finalidade de formar uma árvore de custo ótimo entre os nós. Sempre que um nó encontrar um vizinho que esteja numa distância menor que o seu nó pai, substitui-se o nó pai atual do nó por este nó encontrado.

Verificou-se que as constantes trocas de nós pai causava um *overhead* na rede devido às mensagens *broadcast* que eram espalhadas pelo algoritmo DBF. A ideia principal do MBF é diminuir o número de trocas e, por consequência, diminuir o *overhead*, sem causar um grande impacto na qualidade da árvore de menor custo gerada. Assim sendo, todo o processo de busca em largura realizado pelo DBF é mantido neste algoritmo, porém há uma verificação para saber quando um nó deve trocar de nó pai (LACHOWSKI, 2014).

Para realizar esta verificação, foi criado um parâmetro para saber quando trocar o nó pai, pois algumas trocas não apresentavam tantos ganhos em relação à diminuição de custos. Por exemplo, se um nó  $i$  possui um enlace com seu nó pai  $j$  de custo 20 e existe um enlace de  $i$  até um nó  $k$ , vizinho de  $i$ , com custo 19, a execução da troca do nó pai do nó  $i$  indo de  $j$  para  $k$  seria realizada, aumentando o *overhead*

da rede sem trazer grandes benefícios, já que o ganho de custo com o novo caminho é pequeno.

Este parâmetro no MBF é chamado de *alfa*. Ele representa o valor mínimo que a diferença entre o caminho atual estabelecido em relação ao novo caminho encontrado. O valor de *alfa* ideal encontrado em testes realizados por Lachowski (2014) foi de 0.1, o que significa que o custo do nó *i* para um nó vizinho *k* qualquer, deve ser pelo menos 10% menor que o custo de *i* a *j*, onde *j* é o atual nó pai de *i*, para que a troca de nó pai de *i* seja efetuada.

Outra alteração entre os algoritmos consiste que o MBF considera uma lista de pais alternativos. Em DBF, ao se escolher um novo nó pai, o antigo era desconsiderado. No algoritmo MBF é armazenado uma lista de pais para cada nó, fazendo com que todo vizinho de *i* seja seu nó pai ou esteja presente na lista de pais alternativos de *i*. Esta lista é utilizada caso a comunicação entre um nó e seu pai não possa ser realizada, utilizando então um dos nós da lista de pais alternativos para a comunicação.

Esta medida busca aumentar o tempo de vida da rede, pois em DBF se um nó *j* não tivesse mais energia e um nó *i* tivesse como pai este nó *j*, a comunicação não seria possível e a informação carregada em *i* seria perdida, assim como qualquer outra informação que chegasse até eles. Em MBF pode-se considerar outros nós vizinhos de *i* que não seja *j* para enviar a informação.

### 3.1.1 Detalhes da Execução do algoritmo MBF

Um pseudocódigo do MBF pode ser visualizado nos algoritmos 1 e 2. Este pseudocódigo foi extraído de Lachowski (2014). As siglas NP, W e PA se referem, respectivamente, ao nó pai, peso e lista de pais alternativos do nó em questão. A denominação de custo se refere ao custo do enlace atual que interliga dois nós adjacentes. Assim, a definição de peso de um nó é o valor gasto partindo do *sink* até chegar ao nó em questão, enquanto a variável custo é apenas o valor que interliga dois nós.

O algoritmo inicializa a partir do nó *sink*, quando se atribui peso de valor zero para este nó. Logo a seguir, uma mensagem *broadcast* é enviada pela rede informando os outros nós deste seu peso. O *sink* não tem um nó pai propriamente

dito na prática, já que ele é o nó raiz da rede, onde todas as informações devem convergir.

Na execução do algoritmo, os nós começam com peso infinito, não possuindo um nó pai para então dar-se início com a busca em largura. Durante este processo, os nós ficarão recebendo mensagens de seus vizinhos e caso um nó  $i$  não possua um nó pai, o nó que se comunicou com  $i$  será seu nó pai. Além disso, o peso para se chegar até  $i$  será dado pela soma do peso para se chegar até este outro nó com o custo de enlace dele com  $i$ . Na sequência uma mensagem *broadcast* é enviada informando aos outros nós do novo peso de  $i$ , conforme mostram as linhas 4, 5 e 6 do algoritmo 2.

Caso o nó  $i$  já tenha um pai, então verificará se a soma do peso deste nó que se comunicou com  $i$  e o custo de enlace entre eles se torna vantajosa para o nó  $i$ . Esta verificação se dá por uma inequação que deve ser satisfeita para um valor de *alfa* estabelecido. Considerando o exemplo dado anteriormente, então se  $W_i = 20$  e  $(W_j + \text{Custo}_{i,j}) = 19$ . Assim, o resultado do passo 8 será de  $(20-19)/20 = 0,05$ . Considerando o *alfa* ideal encontrado de 0,1 a troca de nó pai do vértice  $i$  não ocorrerá, adicionando o nó na lista de pais alternativos de  $i$  pela linha 17 do algoritmo 2.

Se o valor for pelo menos igual a *alfa* então é verificado se o nó não se trata do próprio nó pai de  $i$  que devido a alguma iteração anterior no algoritmo teve seu peso reduzido, devendo apenas atualizar o peso de  $i$ , ou se trata de outro nó. Se for o segundo caso, além dos passos encontrados nas linhas 13 e 14 do algoritmo 2 de atualização do peso de  $i$  e envio de mensagem *broadcast*, deve-se incluir o atual nó pai na lista de pais alternativos e realizar a troca de nó pai, conforme as linhas 10 e 11 respectivamente.

---

**Algoritmo 1** *Modified Bellman-Ford* – Inicialização (através do nó *sink*)

---

1:  $NP_{sink} = sink; W_{sink} = 0$

---

2: mensagem *broadcast*:  $\langle sink, W_{sink} \rangle$

---



---

**Algoritmo 2** *Modified Bellman-Ford* – Execução (outros nós da rede)

---

1:  $NP_i = \text{null}; W_i = \infty$

---

2: **while** nó  $i$  receber mensagem  $\langle j, W_j \rangle$  de  $j$

---

---

```

3: if NPi = null then
4:   NPi = j
5:   Wi = Wj + Custoi,j
6:   mensagem broadcast: <i, Wi>
7:else
8:   if ( Wi – (Wj + Custoi,j) ) / Wi >= alfa then
9:     if NPi ≠ j then
10:      PAi = PAi U { (NPi, Wi) }
11:      NPi = j ;
12:     end if
13:     Wi = (Wj + Custoi,j)
14:     mensagem broadcast: <i, Wi>
15:   end if
16:   if NPi ≠ j then
17:     PAi = PAi U { (j, Wj + Custoi,j) }
18:   end if
19:end if
20:end while

```

---

Como evidenciado no trabalho sobre MBF, este algoritmo obteve resultados melhores do que o DBF. As trocas onde o ganho com o caminho eram relevantes foram todas efetuadas, tornando a SPT do MBF bastante semelhante à do DBF e o *overhead* da rede foi significativamente diminuído (LACHOWSKI, 2014).

### 3.2 ALGORITMO *DISTANCE-BASED*

Como citado anteriormente na seção 2.3, apenas medidas preventivas em cada nó, de forma individual, não são sempre eficazes para garantir que uma RSSF fique operante por um longo tempo. Como mostrado em Pantazis et al. (2012), não existe uma medida preventiva padrão para RSSF que garanta sempre um bom resultado: fatores como heterogeneidade dos nós em relação a potência e a utilização de nós móveis pela rede fazem surgir protocolos específicos para alguns casos.

Como neste trabalho, o foco se dá em redes homogêneas, um dos fatores que mais afetarão o consumo dos nós será a distância de envio que cada um deles será submetido: dependendo de sua posição na rede, um sensor pode ter um único vizinho e este ser localizado a uma distância longa, causando um custo alto de comunicação entre eles.

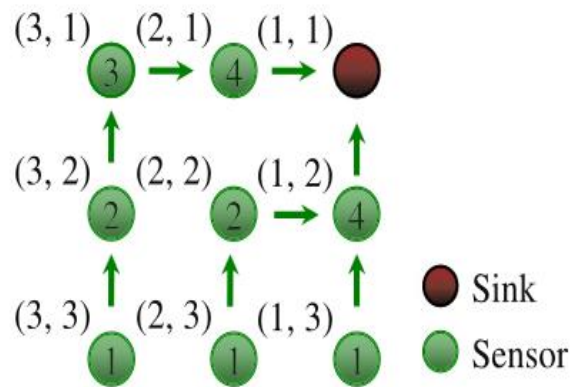
Em Chang, Chang (2008), buscou-se controlar as distâncias em que um nó se encontrava dos outros seguindo uma regra: enlaces entre nós que estão mais afastados do nó *sink* devem possuir distâncias maiores do que enlaces que interligam nós que estão mais próximo ao nó *sink*.

Como toda informação da RSSF converge para o nó *sink*, a probabilidade de que um nó próximo a ele seja utilizado para repassar uma informação é maior do que nós afastados.

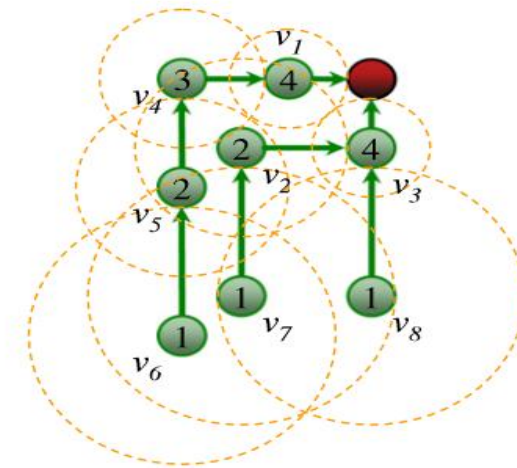
Desta forma, com enlaces de distância menores entre os nós próximos ao *sink* consegue-se diminuir o gasto de energia para cada transmissão realizada, formando um equilíbrio energético com os nós afastados que estariam gastando mais para cada envio. Além disso, a rede também estaria equilibrada em relação a área de abrangência por consumo gasto, pois se todos os nós estivessem próximos, a área de abrangência seria reduzida e se estivessem muito distantes uns dos outros, o consumo para o envio da informação faria a rede ficar inoperante rapidamente.

### 3.2.1 Escopo de Funcionamento do *Distance-Based*

Este algoritmo foi utilizado em Chang; Chang (2008), em redes no formato de Grid, que são redes matriciais onde a posição de cada sensor é representada como coordenadas de um ponto cartesiano. A Figura 7 exibe uma rede em Grid. Os números representados, internamente em cada sensor, representam a carga do nó. A carga de um nó é a quantidade de nós que podem enviar uma informação até ele, incluindo ele mesmo. Quanto maior este número, menor deve ser a distância dele para o próximo nó (nó pai), pois entende-se que ele está próximo ao *sink* e receberá frequentemente informações de outros nós.



**Figura 7: Topologia Grid e distribuição de cargas**  
**Fonte: CHANG; CHANG (2008)**



**Figura 8: Distâncias dos nós pelo *Distance-Based***  
**Fonte: CHANG; CHANG (2008)**

A Figura 8 apresenta o escopo do funcionamento do *Distance-Based*: baseando-se na carga dos nós, pode-se determinar suas distâncias. Ou seja, nós mais distantes (carga menor) do *sink* possuirão maior alcance, enquanto que nós próximos (carga maior) a ele terão alcances restritos. Como forma de padronização, o nó pai será aquele que o nó atual deve se comunicar. Como por exemplo, na Figura 8 o nó  $V_5$  é o nó pai de  $V_6$ , assim como  $V_3$  é o nó pai de  $V_8$  e  $V_2$ .

Seja  $T(i)$  a carga presente no nó  $i$ ,  $T(j)$  a carga presente no nó  $j$ ,  $d_i^j$  a distância de  $i$  até seu nó pai  $j$  e  $d_j^k$  a distância de  $j$  até seu nó pai  $k$ . Para determinar a distância entre um nó  $j$  e seu nó pai  $k$ , utiliza-se a seguinte equação:

$$d_j^k = a \sqrt{\frac{T(i)}{T(j)}} * d_i^j \quad (1)$$

Onde  $a$  é a constante de perda de informação no percurso, com valor variando de 2 a 4 segundo Rappaport (1996, apud CHANG; CHANG, 2008). A variação deste valor ocorre devido aos diferentes canais de comunicação que podem ser empregados numa rede.

Com esta equação, a partir de um valor inicial de distância entre os nós folhas e seus respectivos nós pai, que geralmente equivale a maior distância que o nó sensor consegue transmitir, todos os outros nós da rede terão suas distâncias calculadas ao utilizar como base a distância do respectivo nó com o seu nó filho.

Considerando que o alcance máximo de um nó seja de 30 e baseando-se na distribuição de pais e filhos da Figura 8, tem-se que a distância dos enlaces dos nós folhas  $V_6$ ,  $V_7$  e  $V_8$  valem 30. Como  $a$  varia de 2 a 4, optou-se por utilizar o valor médio de três. A partir disso, o enlace que parte de  $V_5$  para  $V_4$  pode ser calculado ao utilizar a distância entre o nó folha  $V_6$  e seu nó pai  $V_5$ :

$$d_5^4 = \sqrt[3]{\frac{T(V_6)}{T(V_5)}} * d_6^5 = \sqrt[3]{\frac{1}{2}} * 30 = 0,7937 * 30 \cong 23,8$$

De forma semelhante, os enlaces que partem de  $V_4$  para  $V_1$  e de  $V_1$  para o *sink* podem ser calculados utilizando a mesma equação:

$$d_4^1 = \sqrt[3]{\frac{T(V_5)}{T(V_4)}} * d_5^4 = \sqrt[3]{\frac{2}{3}} * 23,8 = 0,87 * 23,8 \cong 20,8$$

$$d_1^{sink} = \sqrt[3]{\frac{T(V_4)}{T(V_1)}} * d_4^1 = \sqrt[3]{\frac{3}{4}} * 20,8 = 0,91 * 20,8 \cong 18,9$$

Partindo da utilização dessa fórmula, a Figura 9 mostra o resultado do algoritmo na rede representada nas Figuras 7 e 8.

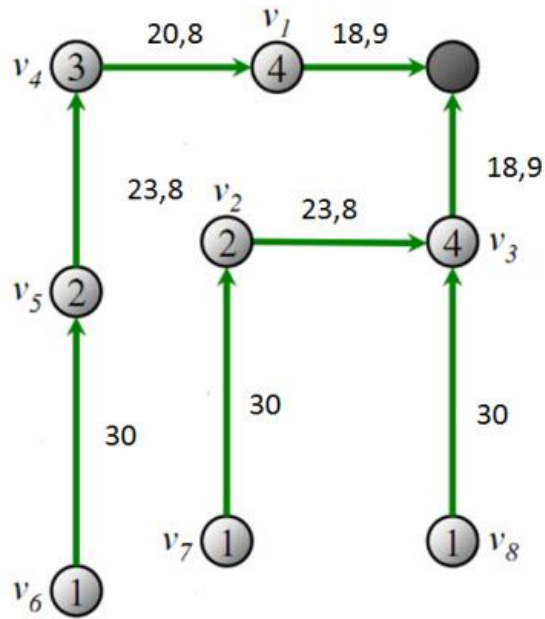


Figura 9: Resultado da aplicação do *Distance-Based* com  $\alpha = 3$

Fonte: Autoria própria

Ishmanov et. al (2011) classificou o *Distance-Based* como um algoritmo de aplicabilidade reduzida, por ser utilizado apenas em redes homogêneas e ter de posicionar os nós em posições pré determinadas, conforme a distância encontrada para cada enlace. A eficiência energética para manter os nós operantes pelo maior tempo possível é mediana, segundo o mesmo autor.

A necessidade de posicionamento dos nós em locais fixos pode acarretar em queda do desempenho esperado para o algoritmo em ambientes irregulares ou habitado por animais, pois os nós ficam sujeitos a serem movimentados durante uma aplicação.



## 4. PREMISSAS

As premissas representam todas as considerações utilizadas para melhor entendimento dos testes realizados. Todos os algoritmos foram criados e testados utilizando a linguagem de programação C. Os gráficos gerados para exemplificar as simulações foram feitas no Excel 2007. As premissas foram separadas em dois grupos: topologias usadas e considerações sobre energia e controle de custos.

### 4.1 TOPOLOGIAS USADAS

Foram consideradas redes homogêneas para as topologias utilizadas, significando que todos os nós possuem energia, capacidade de processamento e potência de transmissão equivalentes (excluindo o nó *sink*). Todos os nós apresentam uma densidade máxima de 6 vizinhos. Há apenas um único nó *sink* na rede.

Não foram utilizadas técnicas como clusterização (PANTAZIS et al., 2013): todos os nós são dispostos nas redes e devem individualmente realizar o trabalho de transmissão dos dados para um único *sink*.

Foram considerados 6 cenários de rede:

- Duas redes de 25 nós cada: na primeira o *sink* possui um único vizinho e na segunda o *sink* possui 5 vizinhos. Para ambas as redes, o número de vizinhos dos outros nós varia de 1 a 5;
- Uma rede de 30 nós onde cada nó possui 3 vizinhos, incluindo o nó *sink*;
- Uma rede de 50 nós onde cada nó possui 4 vizinhos, incluindo o nó *sink*;
- Uma rede de 70 nós onde cada nó possui 6 vizinhos, incluindo o nó *sink*;
- Uma rede de 100 nós onde cada nó possui 2 vizinhos, incluindo o nó *sink*.

A estratégia em se utilizar os diferentes cenários de rede, com quantidades de nós e vizinhos distintos, deveu-se a necessidade de se avaliar o desempenho de cada algoritmo, sua escalabilidade, assim como as topologias em que cada um pudesse melhor se adaptar.

## 4.2 CONSIDERAÇÕES SOBRE ENERGIA E CONTROLE DE CUSTOS.

Todos os nós sensores excluindo o *sink* estão sujeitos a captar algum evento do ambiente. Denomina-se de informação o dado que é coletado por um sensor e deve ser transmitido pela rede até o *sink*. No desenvolvimento deste trabalho, o tamanho desta informação foi padronizado como sendo um pacote de 1Kb.

Inicialmente, todos os nós estão com o seu rádio desligado, apenas monitorando o ambiente. Assim que uma informação é coletada, o nó *i* que a recebeu liga seu rádio e deve transmiti-la para seu nó pai *j*. Este por sua vez liga seu rádio, recebe a informação, muda o modo de operação de recepção para transmissão e a transmite para seu respectivo nó pai *k*.

Desta forma, para os nós pai existem quatro momentos onde há consumo de energia: ao ligar o rádio do sensor, ao receber uma informação, ao efetuar a troca de modo de operação do sensor e ao transmitir a informação para outro nó, enquanto que para o nó que coleta a informação inicial (evento), só há dois: ligar o rádio e enviar a informação. Neste trabalho considerou-se o custo para deixar o rádio desligado como incluso no custo para monitorar o ambiente.

Utilizando-se como base as informações presentes em Polastre et al. (2004) e a relação entre Ampere e Joule/segundo onde 1 Ampere = 1Joule/Segundo, considerou-se os gastos energéticos de um nó sensor como:

- Transmissão (Tx):  $20 \mu\text{J}/\text{Segundo} = 20\mu\text{A}$
- Recepção (Rx):  $15 \mu\text{J}/\text{Segundo} = 15 \mu\text{A}$
- Ligar o rádio de um nó:  $1 \mu\text{J}/\text{Segundo} = 1 \mu\text{A}$
- Troca de modos (de Rx para Tx):  $15 \mu\text{J}/\text{Segundo} = 15 \mu\text{A}$
- Monitorando ambiente (rádio desligado):  $0,03\mu\text{J}/\text{Segundo} = 0,03 \mu\text{A}$
- Bateria: 2 pilhas AA = 1,5 A

Onde se assume que cada uma destas operações leva o tempo exato de 1 segundo para ser realizada.

O monitoramento do ambiente se refere à coleta de dados realizada pelo nó sensor, ocorrendo continuamente até que uma informação seja coletada e então o rádio seja ligado para transmiti-la.

Como uma forma de melhor esclarecimento dos custos associados aos nós, optou-se para criação de equações, a fim de que se entenda como os custos das operações foram inseridos e utilizados nos algoritmos deste trabalho. Todas as equações abaixo são, portanto, de autoria própria.

O custo do enlace (distância por onde a informação trafega) também interfere no gasto energético. Para este trabalho foi considerado o custo total para uma transmissão ou recepção de uma informação como o resultado da multiplicação do custo unitário da operação pela distância em que a informação é enviada ou recebida:

$$Tx_i^j = 20 * d_i^j \quad (2)$$

$$Rx_i^j = 15 * d_i^j \quad (3)$$

Onde j=nó pai, i=nó filho e  $d_i^j$  = distância entre i e j.

Considerando estas informações e que o nó j também possui um nó pai k. O custo de transmissão de j para k será a soma do custo gasto para acordar o nó j, receber a informação de i, mudar o estado de Rx para Tx e enviar a informação para k:

$$C_j^k = 1 + Rx_i^j + 15 + Tx_j^k \quad (4)$$

Se o nó j é quem captou a informação, este nó não terá os custos de Rx e de troca de modos de Rx para Tx e, assim para este nó, o custo é de:

$$C_j^k = 1 + Tx_j^k \quad (5)$$

Das equações 4 e 5 e considerando-se um inteiro  $X \in \{0,1\}$ . Se o nó em questão foi quem captou a mensagem, então  $X = 0$ , caso contrário  $X = 1$ . Desta forma, a fórmula para cálculo do gasto energético para envio de um pacote de um nó j para k é:

$$C_j^k = 1 + X * (Rx_i^j + 15) + Tx_j^k \quad (6)$$

Assim, pode-se calcular a energia gasta pelos nós em suas operações.

Após a informação chegar ao *sink* verificam-se os nós que não foram percorridos e desconta-se deles a energia gasta por terem ficado em estado de dormência.

Para saber o desempenho de cada algoritmo, considerou-se a quantidade total de mensagens que foram recebidas pelo *sink* ao final de cada algoritmo, sendo este, portanto, a representação do tempo de vida determinado para a rede. A seção 6 apresenta os valores encontrados.

## 5. DESCRIÇÃO DOS ALGORITMOS

Foram avaliados quatro algoritmos neste trabalho Alg1 – MBFSPA, Alg2 – MBFCPA, *Distance-Based* e *Distance-Based* Alterado. Em todos eles há o sorteio de um nó como modo para simular uma informação captada por um nó sensor. Após o sorteio, a informação trafega pelos nós pai de cada nó até chegar ao *sink*. A descrição completa do modo transmissão e os custos gastos pelos nós são mostrados na seção 4.

A SPT montada como resultado do algoritmo MBF nas topologias descritas na seção 4.1 serve como base para utilização em todos os algoritmos deste trabalho.

Os algoritmos Alg1 – MBFSPA e Alg2 – MBFCPA são os algoritmos criados para manutenção de topologia a partir das estratégias empregadas no MBF. A diferença entre os dois algoritmos consiste em que o Alg1 – MBFSPA não utiliza os pais alternativos propostos pelo algoritmo MBF, diferentemente do Alg2 – MBFCPA. Com isto, os resultados de ambos servem para atingir um dos objetivos específicos deste trabalho, onde a diferença entre eles determinará a melhoria encontrada pela utilização de pais alternativos.

O algoritmo *Distance-Based* utiliza a SPT montada pelo MBF nas topologias da seção 4.1 para determinar a carga dos nós e em seguida calcular a distância entre os nós segundo a equação 1. Já o algoritmo *Distance-Based* Alterado trata-se de uma versão criada para testes onde ainda segue-se o padrão de que nós próximos ao *sink* possuem distância menor do que nós mais afastados a ele. Porém, nesta versão, o custo dos nós folhas para seus respectivos nós pai não é o valor de alcance máximo do sensor e sim um valor retirado pela utilização de uma equação conforme descrito na seção 5.3.

Este valor é consideravelmente menor que o encontrado para distância dos nós folhas no *Distance-Based* original e, por consequência, as distâncias dos outros nós também são menores.

A seguir, os algoritmos citados são explicados com exceção do *Distance-Based* original que já foi descrito na seção 3.2.

## 5.1 ALG1 – MBFSPA (MBF SEM PAIS ALTERNATIVOS)

O pseudocódigo deste algoritmo pode ser visualizado no Algoritmo 3. As siglas Bat e OP se referem respectivamente à quantidade de energia restante na bateria do nó e o status da operação do nó sensor. A operação se refere ao estado atual do nó. Este inicia-se pelo estado de dormência (representado por status 0) e é alterado quando recebe ou capta uma informação (representado por status 1).

Existem três procedimentos distintos dentro do Alg1 – MBFSPA, que representam operações realizadas apenas no nó *sink*, denominado procedimento 1, no nó coletor de uma informação, denominado procedimento 2, e dos outros nós, denominado procedimento 3.

O procedimento 1 inicia ao ser atribuído o valor zero para a variável num\_info. Toda mensagem recebida pelo *sink* trata de uma informação coletada pelos nós sensores na rede ou de um sinal enviado para informá-lo que algum nó da rede não possui energia suficiente para continuar encaminhando informações. Desta forma, para toda mensagem recebida pelo *sink*, de um de seus vizinhos, este verifica se a mesma se trata de uma mensagem indicando que um nó na rede não possui energia suficiente. Enquanto tal mensagem não é recebida pelo *sink*, a variável num\_info é incrementada. O valor de num\_info ao final do algoritmo será o número de informações que satisfatoriamente chegaram ao *sink*.

---

### Algoritmo 3 Alg1 - MBFSPA

---

#### Procedimento 1: para nó sink

---

num\_info = 0

---

**para** cada <vizinho\_sink, mensagem> **faça**

---

**if** mensagem = sem\_energia **then**

---

**return** num\_info

---

**end if**

---

    num\_info = num\_info+1

---

    envia: <aplicativo, mensagem>

---

**end para**

---

**end procedimento**

---

O procedimento 2 é responsável pela captura de uma informação realizada por um nó sensor. O nó  $i$  se refere ao nó coletor da informação, enquanto que o nó  $j$  se refere ao nó pai de  $i$ . Como citado anteriormente, todo nó está monitorando o ambiente com o seu rádio desligado como forma de economia de energia e, devido a isto, a primeira instrução dentro deste procedimento é mudar o modo de operação do nó para o valor 1, como forma de representar que o rádio do nó agora está ligado, descontando-se da energia total o valor para ligar o rádio. Conforme mostrado na seção 4.2, este valor equivale a 1 miliAmpere ( $1 \mu A$ ).

Na sequência, o procedimento 2 verificará se o nó possui energia para transmitir a informação para seu nó pai. Se o nó possuir energia, a informação será enviada para o pai e o custo de transmissão será também descontado da bateria do nó sensor. Caso não se tenha energia suficiente para transmissão, uma mensagem de tamanho mínimo é enviada para o nó  $j$ , ao invés da informação coletada. Esta mensagem de tamanho reduzido é um sinal indicando que o nó não possui energia para realizar a transmissão das informações. O custo de envio desta mensagem é muito baixo e é desconsiderado neste procedimento.

---

### Algoritmo 3 Alg1 - MBFSPA

---

#### Procedimento 2: para nó coletor da informação

---

$OP_i = 1$ ;  $Bat_i = Bat_i - 1$

---

**if**  $Bat_i \geq Tx_i^j$  **then**

---

$Bat_i = Bat_i - (Tx_i^j)$

---

envia:  $\langle j, \text{informação} \rangle$

---

**else**

---

envia:  $\langle j, \text{sem\_energia} \rangle$

---

**end if**

---

$OP_i = 0$

---

**end procedimento**

---

O procedimento 3 representa o processo realizados pelos nós que farão a interseção entre o nó coletor da informação com o *sink*, sendo executado após um nó realizar o procedimento 2, transmitindo a mensagem coletada para seu nó pai. O

nó  $j$  representa o nó sensor que recebe a mensagem encaminhada por seu filho  $i$ , e o nó  $k$  é o pai de  $j$ .

Os processos para ligar o sensor e de verificação de energia restante para transmissão também ocorrem neste procedimento. Porém, neste procedimento o custo total é dado pela soma dos valores de recepção, de troca de modos de Rx para Tx e de envio da informação, conforme foi estabelecido na seção 4.2.

Se o nó  $j$  não possuir energia para realizar estas operações, o mesmo processo de envio de uma mensagem de tamanho reduzido descrito no procedimento 2 irá ocorrer. Se tal mensagem for gerada por algum nó, seja no procedimento 2 ou no 3, os nós pai por onde esta mensagem irá trafegar enviarão a mensagem diretamente sem precisar calcular custos de recepção e de transmissão.

Ou seja, quando um nó não possui mais energia suficiente para a retransmissão das informações, o sinal por ele enviado é entregue ao *sink* sem que outros nós gastem muita energia, já que a condição fundamental para continuidade do algoritmo foi quebrada.

---

### Algoritmo 3 Alg1 - MBFSPA

---

**Procedimento 3:** para todos os outros nós

---

**while**  $j \neq \textit{sink}$

---

$OP_j = 1$ ;  $Bat_i = Bat_i - 1$

---

**if** mensagem = sem\_energia **then**

---

$Bat_i = Bat_i - 15$

---

        envia:  $\langle k, \text{mensagem} \rangle$

---

**else**

---

**if**  $Bat_j \geq (Rx_i^j + 15 + Tx_j^k)$  **then**

---

            recebe:  $\langle i, \text{mensagem} \rangle$

---

$Bat_j = Bat_j - (Rx_i^j + 15 + Tx_j^k)$

---

            envia:  $\langle k, \text{mensagem} \rangle$

---

**else**

---

            envia:  $\langle k, \text{sem\_energia} \rangle$

---

**end if**

---

**end if**

---



---

$OP_j = 0$

---

**end procedimento**

---

Tanto no procedimento 2 quanto no procedimento 3, conforme parte do algoritmo 3 abaixo citada, a cada segundo transcorrido, o custo para manter os nós monitorando o ambiente (rádio desligado) é descontado de suas respectivas baterias. O processo é realizado independentemente se o rádio do nó está ligado ( $OP_i = 1$ ) para processar uma informação ou não ( $OP_i = 0$ ). Se o rádio está desligado, desconta-se o valor de 0,03 miliAmperes ( $0,03 \mu A$ ), valor considerado como o custo para monitoramento conforme destacado na seção 4.2.

---

**Algoritmo 3 Alg1 - MBFSPA**

---

**para** cada segundo corrido **faça**

---

**if**  $OP_i < 1$  **then**

---

$Bat_i = Bat_i - 0,030$

---

**end if**

---

**end\_para**

---

O valor de num\_info ao final do algoritmo será o número de informações que satisfatoriamente chegaram ao *sink*.

## 5.2 ALG 2 – MBFCPA (MBF COM PAIS ALTERNATIVOS)

Este algoritmo segue o mesmo princípio de Alg 1 – MBFSPA (algoritmo 3), possuindo também três procedimentos distintos para realizar as operações do nó *sink*, do nó coletor da informação e dos outros nós da rede, porém no Alg2 – MBFCPA considera-se os pais alternativos de cada nó.

O procedimento 1 possui as mesmas instruções do Alg1 – MBFSPA, porém os outros dois procedimentos sofrem algumas alterações. Como as alterações são

as mesmas para ambos (procedimentos 2 3 ), elas serão citadas de forma única. O nó  $j$  representa o nó atual da rede, enquanto que  $i$  representa o nó filho de  $j$ .

Estas mudanças ocorrem quando o nó atual não tem energia suficiente para enviar uma informação. Mas, ao invés de enviar um sinal para seu nó pai informando de sua situação, conforme ocorre no Alg1 – MBFSPA, este nó deve enviar a informação para seu nó filho  $i$  novamente junto com um sinal para todos os seus vizinhos (incluindo o nó filho que recebeu a mensagem) informando que ele não pode mais ser utilizado para realizar o tráfego de informações. Além disso, o nó sem energia terá seu rádio desligado permanentemente. A partir deste momento, o nó fica impossibilitado de realizar transmissões.

A sigla CP se refere ao conjunto de nós pertencentes ao caminho percorrido pela informação, que deve ser constantemente atualizada pelos nós sensores, se tornando um parâmetro extra que é enviado durante as transmissões da informação. Inicialmente o nó coletor é o único nó pertence a CP. Conforme a informação trafega na rede, os nós por onde esta informação passou são incluídos em CP. Abaixo, mostra-se um pseudocódigo que realiza estas operações:

---

**Algoritmo 4 Alg2 - MBFCPA**

---

**if**  $Bat_j < energia\_necessária$  **then**

---

envia :  $\langle i, mensagem, \{CP - j\} \rangle$

---

mensagem *broadcast*:  $\langle vizinhos\_j, sinal \rangle$

---

$OP_j = 0$

---

**end if**

---

**end procedimento**

---

O termo “energia\_necessária” presente no pseudocódigo se refere ao custo de energia que o sensor deve gastar ou para apenas transmitir a informação (se ele for o nó que coletou a informação) ou a soma para receber a informação, trocar de modo de Rx para Tx e transmitir a informação (se o nó atual não foi quem coletou a informação).

Se o nó sem energia for o nó coletor da informação, então não haverá um nó anterior para se enviar a informação. Quando isto ocorre, o nó deve enviar uma mensagem informando de sua incapacidade para enviar a informação. Esta mensagem é semelhante à mensagem de um nó não ter energia suficiente que ocorre no Alg1 – MBFSPA e será repassada pela rede até chegar ao nó *sink*, com o seu custo sendo desconsiderado. Devido a incapacidade do nó que coletou a informação, o algoritmo se encerra.

Quando o nó sem energia é algum outro nó da rede, então o seu nó filho, que recebeu a informação do nó sem energia, deve consultar os outros vizinhos dele, incluindo os nós por onde a informação já trafegou (CP), a fim de saber se há algum nó com energia suficiente para transmissão, ou seja, se algum nó não enviou uma mensagem indicando incapacidade para transmitir mensagens. Este nó filho que recebeu a mensagem também é chamado de *j* nos pseudocódigos abaixo.

A sigla PAP se refere ao conjunto de pais alternativos possíveis que o nó pode utilizar. Todo nó vizinho que não enviou um sinal de que não pode mais transmitir informação pertencerá a PAP. Se o vizinho do nó *j* pertencer a CP, então este nó será incluído também em um outro conjunto: o PACP.

Este conjunto demarca os pais alternativos do nó que possuem energia restante e que pertencem ao caminho já percorrido pela informação. Um exemplo de um PACP do nó *j* é o seu nó filho, que enviou a informação para ele e, naturalmente, é um vizinho de *j*.

---

#### **Algoritmo 4 Alg2 - MBFCPA**

---

**para** cada vizinho\_ *j* **faça**

---

**if** vizinho\_ *j* não enviou sinal **then**

---

PAP<sub>*j*</sub> = PAP<sub>*j*</sub> U vizinho\_ *j*

---

**if** vizinho\_ *j* ∈ CP **then**

---

PACP<sub>*j*</sub> = PACP<sub>*j*</sub> U vizinho\_ *j*

---

**end if**

---

**end if**

---

**end para**

---

Se após percorrer todos os vizinhos de  $j$ , o único nó que não enviou o sinal de incapacidade para transmitir for o seu próprio nó filho, então o nó  $j$  deve também enviar um sinal para seus vizinhos informando que ele não poderá mais ser utilizado para realizar o tráfego de informações e deve se retirar do conjunto CP dos nós por onde a informação já percorreu.

Caso exista mais de um nó que não enviou o sinal, então exclui-se de PAP os nós por onde a informação já percorreu (PACP). Neste processo de exclusão o nó filho do nó atual também é excluído.

Dos nós restantes, verifica-se qual deles possui o menor peso  $W$ , encontrado pelo algoritmo MBF durante sua execução. A informação será então encaminhada para este vizinho selecionado, que no pseudocódigo é chamado de  $k$ .

Se no processo de exclusão dos nós pertencentes a PACP, não sobrou nenhum nó em PAP, então o nó atual deve enviar a informação para seu nó filho e novamente o processo de verificação dos nós vizinhos para saber quais entre eles ainda não enviou o sinal que representa incapacidade para transmitir será acionado.

---

#### **Algoritmo 4 Alg2 - MBFCPA**

---

**if**  $PAP_j = i$  **then**

---

envia :  $\langle i, \text{mensagem}, \{CP - j\} \rangle$

---

mensagem *broadcast*:  $\langle \text{vizinhos}_j, \text{sinal} \rangle$

---

$OP_j = 0$

---

**else**

---

$PAP_j = PAP_j - PACP_j$

---

**if**  $PAP_j = \emptyset$  **then**

---

envia :  $\langle i, \text{mensagem}, \{CP - j\} \rangle$

---

**else**

---

**para** cada nó  $\in PAP$  **faça**

---

encontre menor  $W$

---

**end para**

---

envia:  $\langle k, \text{mensagem}, \{CP + j\} \rangle$

---

**end if**

---

$OP_j = 0$

---

**end if**

---

Se a informação chegar até o nó que a coletou e este também não possuir um vizinho para enviar a informação, então uma mensagem de sinal de que a informação em questão não pode ser entregue ao *sink* será enviada para seu nó pai e o algoritmo será encerrado.

Este peso  $W$  citado no último pseudocódigo, representa a distância do nó ao *sink* conforme explicado no Algoritmo 2 na seção 3.1.1, ou seja, o novo nó pai de um nó será aquele que possui menor distância para o *sink*.

Em condições normais, os nós por onde a informação trafega não se repetem em uma transmissão, já que o resultado do algoritmo MBF é uma árvore. Porém, quando um nó utiliza um de seus pais alternativos, não se sabe se este nó selecionado já foi percorrido ou não, devido ao critério de escolha adotado.

Por causa disto, os nós por onde a informação já trafegou não podem ser escolhidos para ser o nó pai de um nó, pois caso algum destes nós fosse escolhido, a rede entraria em um ciclo, já que a informação voltaria para um nó onde ela já passou e eventualmente chegaria até este nó novamente.

Há diferença entre:

- o nó não possuir vizinhos com energia restante e;
- os vizinhos com energia restante serem impossibilitados de se tornar nó pai por já terem trafegado a informação.

No primeiro caso, o nó atual está isolado, só podendo encaminhar informações para seu nó filho, se tornando um caminho sem uma saída para a informação, que deve retornar para o nó anterior.

No segundo caso, o nó possui vizinhos com quem se comunicar, mas devido ao caminho percorrido pela informação, eles não podem ser acessados no momento. Porém, em futuras iterações do algoritmo, estes vizinhos podem não ter sido utilizados pela informação, podendo ser utilizados como pai alternativo do nó em questão. Devido a isto, quando ocorre este segundo caso, o nó em questão não envia o sinal de incapacidade para transmitir uma informação

O Alg2 – MBF CPA ao seguir os passos supracitados consegue prolongar o número de mensagens que o *sink* recebe. Diferentemente do Alg1 – MBF SPA onde o primeiro nó sem energia era o suficiente para o algoritmo encerrar-se, no Alg2 –

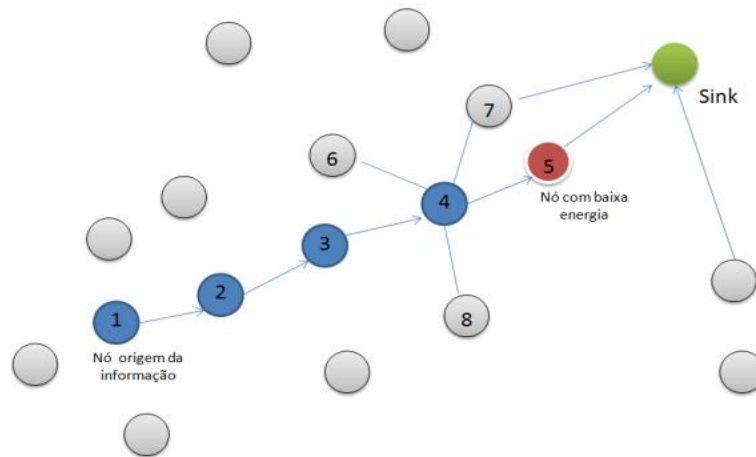
MBFCPA é necessário que nenhum nó por onde a informação trafega tenha um vizinho com energia restante.

A utilização de pais alternativos auxilia que outros nós sejam usados para transmissão. Conforme discutido na seção 2.3, como a informação gerada é estocástica, parte da rede pode ser mais utilizada que outras, causando um esgotamento dos recursos naquela região. Ao utilizar os pais alternativos, os nós não irão mandar uma informação através do caminho estabelecida pelo MBF e desta forma, nós menos utilizados serão acionados.

A Figura 10 mostra um exemplo da utilização dos pais alternativos em uma rede. Nesta figura, a maioria dos enlaces foram omitidos para melhor visualização. O nó 1 representa onde a informação foi captada e os nós subsequentes 2, 3 e 4 indicam por onde a informação trafega. Os enlaces direcionais são utilizados para se indicar os nós pai dos vértices filhos. O nó 5 representa o ponto onde a energia está baixa. Os enlaces não direcionais, como os ligados ao nó 4 representam os candidatos a pais alternativos deste nó. Como todo vizinho que não é o nó pai são os pais alternativos do nó, então o nó 3 também é um pai alternativo de 4. Quando o nó 5 percebe que não tem energia para transmitir a mensagem recebida, ele envia uma mensagem para seus vizinhos indicando que ele não pode mais transmitir mensagens e reencaminha a mensagem recebida para o nó 4 (seu nó filho). Este nó, por sua vez, verifica entre os nós 3, 6, 7 e 8 se algum deles também mandou uma mensagem indicando a incapacidade de transmitir uma mensagem. Considere que o nó 8 já tenha enviado tal mensagem, então este nó é desconsiderado como candidato a pai alternativo do nó 4. Após isto, verificam-se os nós por onde a informação já percorreu. Como o nó 3 é filho do 4, então o nó 3 também não pode ser pai alternativo do 4. Para os nós restantes, 6 e 7, verifica-se o peso  $W$  obtido no algoritmo MBF, ou seja, verifica qual deles possui menor distância para o nó *sink*. O nó escolhido se torna o novo pai de 4. Se todos os nós vizinhos de 4 com energia restante já tivessem sido percorridos, então o nó 4 encaminha a informação para o nó 3, devendo este nó verificar seus respectivos pais alternativos. Se o único nó com energia restante dentre os vizinhos do nó 4 fosse o nó 3, então o nó 4 além de reencaminhar a mensagem para seu nó filho, também enviaria um sinal para seus vizinhos informando de sua incapacidade de transmitir uma informação.

Nenhum nó pode enviar uma informação para um nó que tenha enviado o sinal de incapacidade para transmitir mensagens, com a exceção se a mensagem

for um sinal que uma informação não pode ser entregue ao *sink*. Neste caso, a mensagem é enviada e o custo é desconsiderado.



**Figura 10: Transmissão de informação pelo Alg 2 - MBFCPA**

**Fonte: Autoria própria**

Durante a implementação do Alg2 – MBFCPA verificou-se que determinados nós ainda tinham energia para transmissão, diferentemente dos vizinhos dele que já tinham esgotado suas baterias, deixando-o isolado na rede. Como o nó não tinha com quem se comunicar, ele acabava sendo obrigado a ficar em estado de dormência junto com seus vizinhos. A inserção de algoritmos de reorganização de topologias (AZIZ et al., 2013) resolveria este problema, ao mudar as ligações da rede e, assim, poder inserir o nó com energia restante em um local na rede onde houvesse nós na mesma situação que ele.

O termo energia suficiente sofre uma alteração em relação ao Alg2 - MBFCPA. Dentro do algoritmo, esperar que um nó tenha tão pouca energia a ponto de que na próxima vez que for transmitir a informação, ele acabe morrendo não é a melhor alternativa.

A razão disso se deve ao fato de que os enlaces têm custos variados, ficando difícil determinar uma medida e, principalmente, que quando o nó é colocado para dormir até o fim do algoritmo, ele ainda está gastando um pouco de energia. Considere o seguinte exemplo: um nó  $j$  vizinho do *sink* possui distância de 10 até ele e recebe uma informação de um nó filho  $i$ , sendo que a distância entre  $i$  e  $j$  vale 15. O custo total para repassar a informação de  $j$  para o sink será:

$$Rx_i^j = Rx * 15 = 15 * 15 = 225$$

$$Tx_j^{sink} = Tx * 10 = 20 * 10 = 200$$

$$C_j^{sink} = 1 + X * (Rx_i^j + 15) + Tx_j^{sink} = \\ = 1 + 1 * (225 + 15) + 200 = 1 + (235) + 200 = 436 \mu A$$

Se considerarmos que este nó vizinho do *sink* tem 450  $\mu A$  de energia restante na bateria, e como o custo para esta transmissão não o deixa sem energia ainda, a transmissão seria efetivada e o nó passará a ter sua bateria próxima a zero. Assim, na próxima vez que um nó se comunicar a ele, este nó não terá energia para enviar a informação para o *sink*, enviando um sinal de sua incapacidade para transmitir informações, reencaminhando a mensagem para seu filho e sendo colocado para dormir até o fim do algoritmo.

O custo para reencaminhar a mensagem para seu nó filho acrescido do valor gasto a cada segundo em estado de dormência pode fazer a bateria se extinguir completamente ao passar do tempo. No exemplo dado, o nó j teria 14  $\mu A$  de energia restante, o que não seria suficiente nem para receber a mensagem do nó i.

Por meio de testes, fixou-se que o valor mínimo que uma bateria pode apresentar é de 4000  $\mu A$  para todos os nós, onde se consegue garantir que a energia restante será duradoura o suficiente para que algum nó tenha todos os seus vizinhos com baixa energia e o algoritmo seja finalizado pela impossibilidade de transmissão.

Estas alterações causam grande impacto na complexidade dos nós sensores. Para o Alg1 – MBFSPA, os sensores apenas precisavam saber quem era o nó pai e processar se uma mensagem recebida era uma informação coletada da rede ou um sinal de um nó informando sobre a falta de energia para transmissão.

No Alg2 – MBFCPA, os nós precisarão saber da existência de todos os seus vizinhos, além de ter controle dos nós onde a informação passou e dos nós vizinhos a ele que não possuem energia restante.



### 5.3 DISTANCE-BASED ALTERADO

Como mostrado na seção 3.2.1, os nós folhas no algoritmo *Distance-Based* possuem custo igual ao alcance do nó. Porém este valor de alcance pode ser utilizado da seguinte forma:

$$C_{folha}^{Pai\_folha} = a \sqrt{\frac{T(folha)}{T(Pai\_folha)}} * \max\_alcance \quad (7)$$

Onde  $\max\_alcance$  se refere ao valor de alcance máximo de transmissão do nó sensor. Por causa desta diferença de implementação e sabendo que a carga de um nó pai é sempre maior que a carga do seu nó filho, o valor encontrado de distância pela equação 7 entre um nó folha e seu pai é menor que o encontrado no *Distance-Based* original, pois a fração sempre será menor que 1.

Utilizando como base o mesmo exemplo dado na seção 3.2.1, onde a distância do enlace que parte de  $V_6$  para  $V_5$  presente na Figura 8 era de 30 ao utilizar-se o *Distance-Based* original, pois o nó  $V_6$  é um nó folha e sempre sua distância com seu nó pai corresponde ao valor de alcance máximo de transmissão ( $\max\_alcance$ ).

Nesta nova versão, a distância de  $V_6$  para  $V_5$  é calculada pela equação 7:

$$d_6^5 = \sqrt[3]{\frac{T(V_6)}{T(V_5)}} * \max\_alcance = \sqrt[3]{\frac{1}{2}} * 30 = 0,7937 * 30 \cong 23,8$$

Nota-se, portanto, que a distância de um nó folha e seu pai já se torna consideravelmente menor se comparado ao *Distance-Based* original e ao utilizar-se desta mesma equação para determinar as distâncias dos enlaces dos outros nós, toda a rede apresentará um valor de enlace inferior ao que seria encontrado no *Distance-Based* original, conforme se mostra na Figura 11.

Assim, considerando  $i$  um nó folha, a equação 7 encontra a distância dele para seu nó pai  $j$  e para determinar a distância de  $j$  para seu nó pai  $k$  neste novo modelo, a equação utilizada é:

$$d_j^k = a \sqrt{\frac{T(j)}{T(k)}} * d_i^j \quad (8)$$

Ou seja, nesta versão a carga presente no nó pai  $j$  interfere no custo da distância entre ele e seu filho  $i$ . Assim, se existirem pais de nós folhas com cargas diferentes, a distância entre este nó e o folha será diferente dos demais.

Por exemplo, considerando novamente a Figura 8 e o alcance máximo de transmissão como 30 ( $\text{max\_alcance}$ ), para calcular a distância do nó folha  $V_7$  para o seu nó pai  $V_2$  e de  $V_8$  para  $V_3$  utilizando a equação 7 tem-se que:

$$d_7^2 = \sqrt[3]{\frac{T(V_7)}{T(V_2)}} * \text{max\_alcance} = \sqrt[3]{\frac{1}{2}} * 30 = 0,7937 * 30 \cong 23,8$$

$$d_8^3 = \sqrt[3]{\frac{T(V_8)}{T(V_3)}} * \text{max\_alcance} = \sqrt[3]{\frac{1}{4}} * 30 = 0,63 * 30 \cong 18,9$$

Para determinar a distância de enlace dos outros nós, como por exemplo de  $V_2$  para  $V_3$  e de  $V_3$  até o *sink*, foi utilizada a equação 8:

$$d_2^3 = \sqrt[3]{\frac{T(V_2)}{T(V_3)}} * d_7^2 = \sqrt[3]{\frac{2}{4}} * 23,8 = 0,7937 * 23,8 \cong 18,9$$

$$d_3^{\text{sink}} = \sqrt[3]{\frac{T(V_3)}{T(\text{sink})}} * d_2^3 = \sqrt[3]{\frac{4}{8}} * 18,9 = 0,7937 * 18,9 \cong 15$$

Ao aplicar a equação 8 nos demais enlaces, os valores encontrados para a distância dos nós passam a ser os exibidos pela Figura 11.

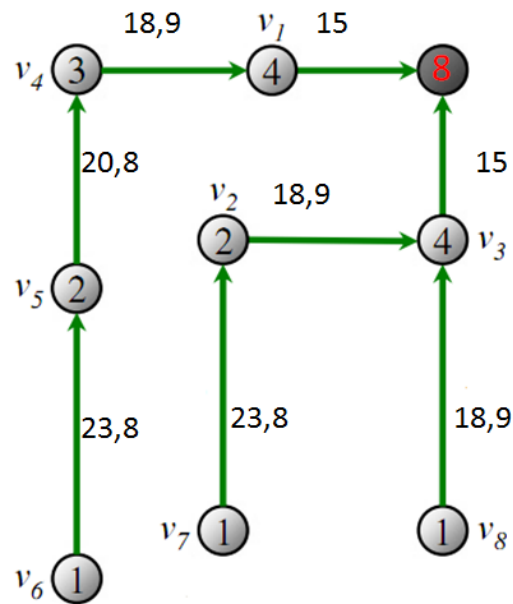


Figura 11: *Distance-Based Alterado* com  $\alpha = 3$

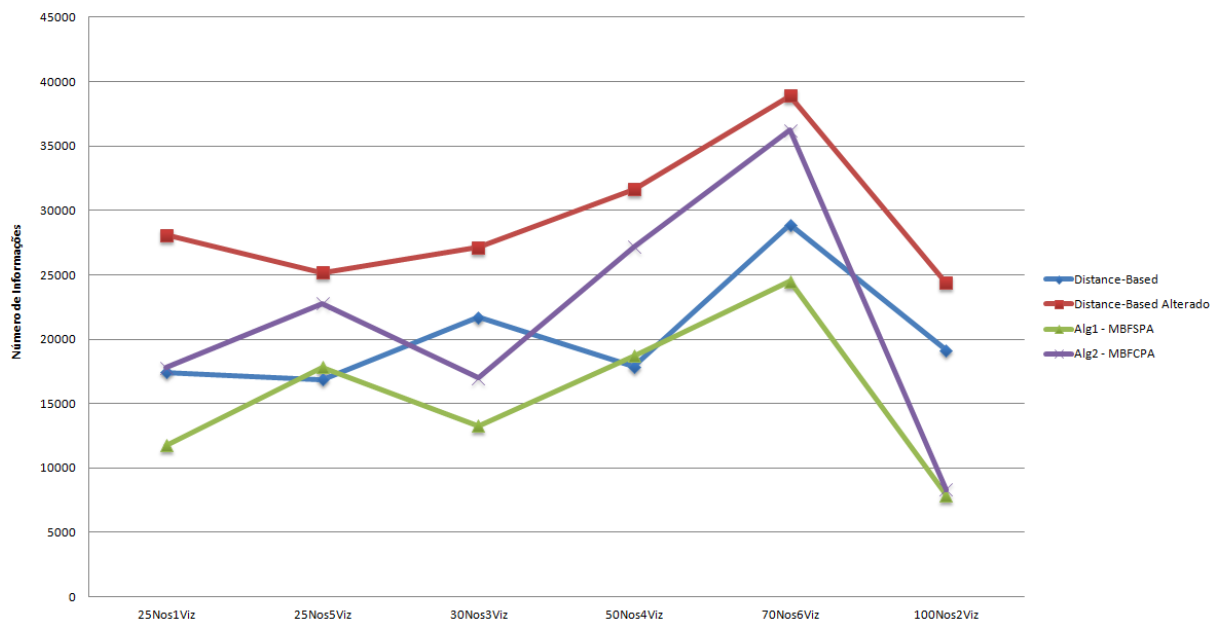
Fonte: Autoria própria

Como se utiliza a carga do nó atual e do nó pai, os enlaces que chegam até o *sink* são calculados utilizando a carga do mesmo que neste caso vale 8.

Este método foi criado apenas para testes neste trabalho. A utilização *Distance-Based Alterado* é útil para tornar as distâncias de enlaces menores que o habitual (o que corresponde à distância dos sensores em uma aplicação) e garantir um tempo maior de vida para a rede. Contudo, há uma diminuição considerável da área abrangida neste algoritmo, pois todas as distâncias entre os nós foram reduzidas. Desta forma, esta versão não deve ser considerada mais eficiente que o *Distance-Based* original sem que testes, que fogem do escopo deste trabalho, sejam realizados.

## 6. TESTES

Considerando as premissas anteriormente citadas, realizou-se os testes dos algoritmos. As duas redes de 25 nós são diferenciadas pelos sufixos de “1Viz” e “5Viz” que representam a quantidade de vizinhos do *sink* que cada uma delas possui.



**Figura 12: Número de informações obtidas pelos algoritmos**

**Fonte: Autoria própria**

Com base nos dados da Figura 12, pode-se observar que o *Distance-Based* Alterado foi o que apresentou o melhor resultado em todas as redes. Já se esperava que esta versão tivesse um desempenho melhor que o *Distance-Based* original, pois os custos de enlace calculados por este algoritmo eram menores que a versão original do *Distance-Based*.

O Alg2 - MBFSPA alterna de posição com o *Distance-Based* original dependendo das redes impostas. Pode-se observar que em redes onde há uma quantidade maior de vizinhos, tanto apenas em relação ao *sink* quanto em relação à todos os nós da rede, o Alg2 - MBFSPA se torna uma melhor opção, chegando próximo dos resultados obtidos pelo *Distance-Based* Alterado.

No caso onde a rede possui 25 nós e com o *sink* tendo 5 vizinhos a diferença entre os dois algoritmos que apresentam melhor resultado é baixa: 25167 informações recebidas em média pelo *Distance-Based* alterado contra 22792 do Alg2 - MBFCPA e o mesmo ocorre quando a rede possui 70 nós (38911 do primeiro e 36239 do segundo).

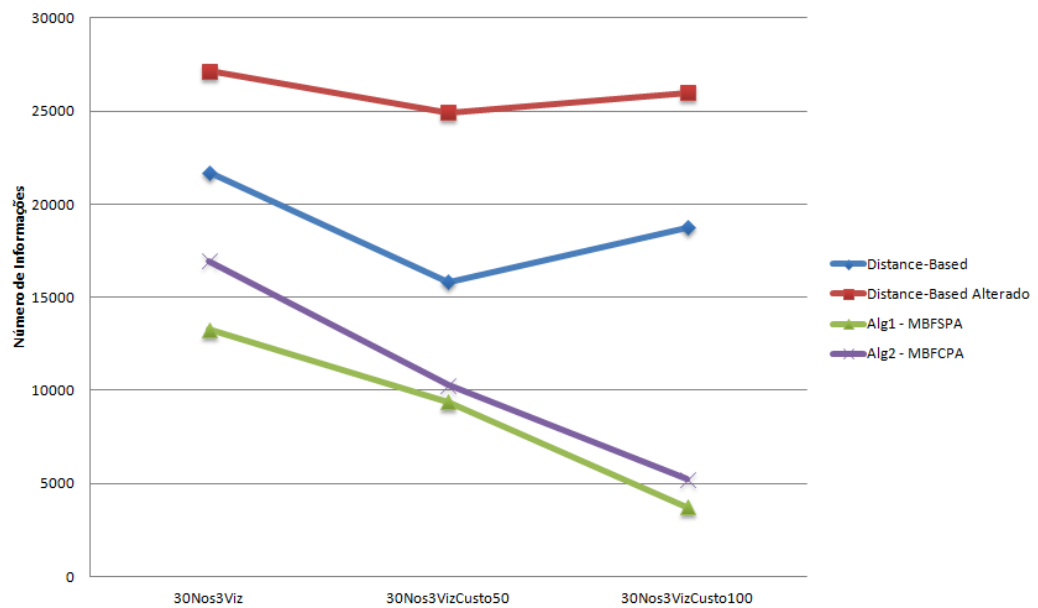
Além disso, o Alg2 - MBFCPA sempre obteve resultado superior ao Alg1 - MBFSPA, atingindo um dos objetivos secundários propostos. A diferença entre os dois algoritmos que utilizam a SPT do MBF cresce quando se aumenta o número de vizinhos da rede, como pode se observar na rede de 50 nós: 27147 mensagens recebidas pelo Alg2 - MBFCPA e 18703 do Alg1 - MBFSPA, um aumento de quase 50%, o que significa que em redes mais densas, a efetividade do método da troca de pais alternativos aumenta.

Como prova disto, quando os nós da rede possuem menos vizinhos, a diferença de desempenho entre eles se torna menor, a ponto de redes menos densas (como a rede de 100 nós com 2 vizinhos em cada nó) o desempenho se tornar semelhante, com 8311 informações recebidas Alg2 - MBFCPA e 7901 do Alg1 - MBFSPA. Nesta topologia, a diferença deles quando comparados ao *Distance-Based* original é grande, já que este algoritmo obteve 19129 informações nos testes realizados, mais que o dobro do obtido pelo Alg2 - MBFCPA.

O Alg1 - MBFSPA conseguiu resultados ligeiramente superiores ao *Distance-Based* original, como na topologia de 50 nós em que 18703 informações foram enviadas pelo *sink* pelo primeiro algoritmo contra 17867 do segundo. Porém em redes de pouca vizinhança, o *Distance-Based* apresenta resultados bem melhores.

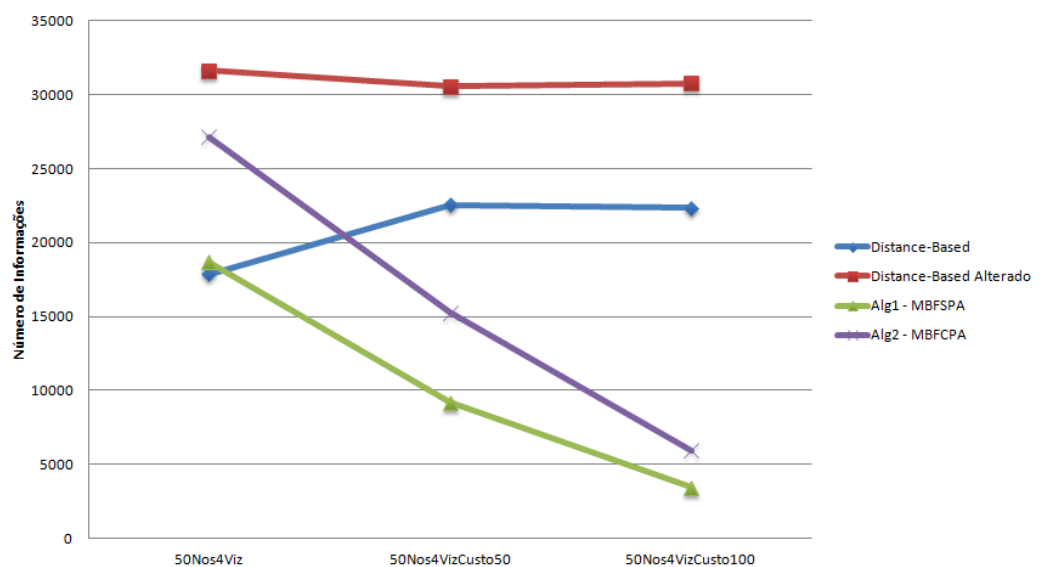
As Figuras 13 e 14 exibem testes realizados com as redes de 30 e 50 nós, onde se variou o custo de enlace dos nós. Os nós têm alcance de 20 a 30 metros em locais fechados, podendo chegar de 75 a 100 metros em locais abertos (MEMSIC, 2015). Os testes mostrados na Figura 12 exibem redes onde os custos não passam de 25 metros. Nos testes seguintes, escolheu-se alterar o custo para no máximo 50 metros e para no máximo 100 metros nas duas topologias citadas anteriormente, tendo desta forma três redes iguais de custos máximos nos enlaces variando entre elas.

Assim, a topologia o sufixo “Custo50” e “Custo100” representam as topologias onde os enlaces possuíam custo de no máximo 50 e 100, respectivamente. A topologia que não apresenta nenhum destes sufixos é a mesma topologia considerada na Figura 12, onde o custo dos enlaces não ultrapassaram o valor de 25.



**Figura 13: Alteração de custos dos enlaces na rede de 30 nós**

Fonte: Autoria própria



**Figura 14: Alteração de custos dos enlaces na rede de 50 nós**

Fonte: Autoria própria

É notável a diminuição do número de informações dos algoritmos de Alg1 – MBFSPA e Alg2 - MBFCPA, mostrando que as estratégias apresentadas em ambos não são eficazes para redes com sensores colocados em uma distância muito alta uns dos outros.

Conforme se aumenta o custo dos nós, o Alg2 - MBFCPA vai apresentando um desempenho cada vez mais similar ao Alg1 - MBFSPA: na rede com 50 nós criada inicialmente, a diferença entre eles era de  $27147 - 18703 = 8444$  nós recebidos a mais pelo primeiro. Quando a rede de 50 nós possui custo de no máximo 100, a diferença entre eles passa a ser  $5959 - 3441 = 2518$ . Uma perda de mais de 70% do total efetivo que se era ganho pela utilização de pais alternativos no Alg2 - MBFCPA.

Pode-se observar também uma alteração nos desempenhos do *Distance-Based*. Porém, as informações obtidas não foram suficientes para se obter uma conclusão sobre o comportamento deste algoritmo.

## 7. CONCLUSÃO

Os algoritmos criados para manutenção de topologia através das estratégias pensadas para o MBF podem ser utilizados para aumentar o tempo de vida de uma aplicação em RSSF. Os testes realizados na seção 6 comprovam que em determinadas situações eles se tornam superiores em desempenho em relação ao *Distance-Based* original. Principalmente se comparado com o Alg2 – MBFCPA, pois este algoritmo chega a transmitir quase 50% mais informação que o *Distance-Based* original em algumas topologias.

Ao analisar as diferentes redes, percebe-se que redes mais densas, com pouca distância entre os nós, são mais atrativas tanto para Alg1 – MBFSPA, quanto para Alg2 – MBFCPA, o que significa que suas utilizações em aplicações de redes que se encontram em locais fechados são as melhores alternativas a serem exploradas.

A variação da quantidade de nós vizinhos foi determinante para atingir o outro objetivo secundário: em redes com baixa vizinhança entre os nós, a diferença de desempenho dos algoritmos Alg2 – MBFCPA e Alg1 – MBFSPA é pequena, podendo chegar a menos de 10%, enquanto que em redes mais densas, o primeiro chega a transmitir cerca de 50% a mais em informações que o segundo.

A maior complexidade do Alg2 – MBFCPA, evidenciada durante sua descrição na seção 5.2, representa um grande obstáculo para sua implementação em larga escala. Assim, abrem-se oportunidades para que medidas preventivas sejam tomadas a fim de torná-lo mais escalável.

Desta forma, conclui-se que o Alg2 – MBFCPA é uma boa alternativa durante aplicações que demandam redes de sensores para aplicações em locais pequenos e redes densas. O Alg1 – MBFSPA apesar de ter resultados bons, é inferior ao seu concorrente e aos modelos de *Distance-Based* apresentados durante este trabalho, o tornando dispensável.



## 7.1 TRABALHOS FUTUROS

Com a realização deste trabalho, alguns tópicos interessantes se tornam relevantes a serem estudados a fim de expandir o que foi apresentado. Pode-se notar principalmente os seguintes itens:

- Inserção de um algoritmo de reorganização de topologias de redes durante uma aplicação para aproveitamento de nós com baterias restantes que não possuem pais alternativos como mencionado na seção 5.2;
- Expansão de testes ao aplicar outro algoritmo de manutenção de topologia de rede e de outras topologias como, por exemplo, com a utilização de múltiplos *sink*;
- Proposição de novas métricas para calcular o desempenho dos algoritmos Alg2 – MBFCPA e Alg1 – MBFSPA.
- Estudo de técnicas para melhorar a escalabilidade do Alg2 – MBFCPA.

## REFERÊNCIAS

AKYILDIZ, Ian F.; SU, Weilian; SANKARASUBRAMANIAM, Yogi; CAYRCI, Erdal. Wireless Sensor Networks: A Survey. **Computer Networks**. Atlanta, v.38, n. 4, p. 393-422, dez. 2001

AKYILDIZ, Ian; VURAN, Mehmet C. **Wireless Sensor Networks**. Atlanta: John Wiley & Sons Ltd. 2010.

ANASTASI, Giuseppe; CONTI, Marco; FRANCESCO, Mario Di; PASSARELLA, Andrea. Energy Conservation in Wireless Sensor Networks: A Survey. **Ad Hoc Networks**, Pisa, v.07, n. 03, p. 537 – 568, jul, 2008.

AWERBUCH, Baruch; AMOTZ, Bar-Noy; MADAN, Gopal. Approximate distributed Bellman-Ford algorithms. **IEEE Transactions on Communications**, Cambridge , v. 42, n. 08, p. 2515 – 2517, ago. 1994

AZIZ, Azrina A.; SEKERCIOGLU, Y. Ahmet; FITZPATRICK, Paul; IVANOVICH, Milosh. A Survey on Distributed Topology Control Techniques for Extending the Lifetime of Battery Powered Wireless Sensor Networks. **IEEE Communications Surveys & Tutorials**, Melbourne, v. 15, n. 01, p. 121 – 144, abr. 2012.

BERTSEKAS, D.; GALLAGER, R. **Data Networks**. 2 ed. Upper Saddle River: Prentice-Hall. 1992.

CHANG, Jae-Hwan; TASSIULAS, Leandros. Energy Conserving Routing in Wireless Ad-hoc Networks. **Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies**. Tel Aviv, v. 01, p. 22-31, mar. 2000.

CHANG, Chih-Yung; CHANG, Hsu-Ruey. Energy-aware Node Placement, Topology

Control and MAC Scheduling for Wireless Sensor Networks. **Computer Networks**, Taiwan, v. 52, n. 11, p. 2189-2204, ago. 2008

DIEDRICH, Isabel, DRESSLER, Falko. On the Lifetime of Wireless Sensor Networks. **ACM Transactions on Sensor Networks**, New York, v. 05, n. 01, p. 39-78, fev. 2009.

HOWITT, Ivan.; WANG, Jing. Energy Efficient Power Control Policies for the Low Rate WPAN. **Sensor and Ad Hoc Communications and Networks**. Charlotte, v. 01, p. 527-536, out. 2004.

HUSSAIN, Sajid; ISLAM, Obidul. An Energy Efficient Spanning Tree Based Multi-Hop Routing in Wireless Sensor Networks. **Wireless Communications and Networking Conference**. Kowloon, v. 01, p. 4383 – 4388, mar. 2007.

INTANAGONWIWAT, Chalermek; GOVINDAN, Ramesh; ESTRIN, Deborah. Directed diffusion: a scalable and robust communication paradigm for sensor networks. **Sixth International conference on Mobile Computing and Networking**. Boston, p. 56–67, ago. 2000.

ISHMANOV, Farruh; MALIK, Aamir S.; KIM, Sung W. Energy Consumption Balancing Issues and Mechanics in Wireless Sensor Networks. **European Transactions on Telecommunications**, v. 22, n. 04, p. 151-167. Jun, 2011.

KAHN, Joseph M.; KATZ, Randy. H.; PISTER, Kristofer S. J. Next century challenges: mobile networking for “smart dust.” **Fifth International Conference on Mobile Computing and Networking**. Seattle, p. 271–278, ago. 1999.

MEMSIC. MICAz Sensor Datasheet. Disponível em:  
<[http://www.memsic.com/userfiles/files/Datasheets/WSN/micaz\\_datasheet-t.pdf](http://www.memsic.com/userfiles/files/Datasheets/WSN/micaz_datasheet-t.pdf)>

LACHOWSKI, Rosana. **Algoritmos Distribuídos para Construção de Spanning Trees em Redes de Sensores Sem Fio**. 2014. 32f. Dissertação (Mestrado em Informática) – Pontífica Universidade Católica do Paraná, Curitiba, 2014.

PANTAZIS, Nikolaos; NIKOLIDAKIS, Stefanos; VERGADOS, Dimitrios. Energy-Efficient Routing Protocols in Wireless Sensor Networks. **IEEE Communications Surveys & Tutorials**, Athens, v. 15, n. 02, p. 551-591. Jun, 2012.

POLASTRE, Joseph; HILL, Jason; CULLER, David. Versatile Low Power Media Access for Wireless Sensor Networks. **2<sup>nd</sup> International Conference on Embedded Networked Sensor Systems**, Baltimore, p. 95-107. Mar, 2004.

POTTIE, Gregory J.; KAISER, William J. Wireless integrated network sensors. **Communications of the ACM**. New York, v. 43, n. 05, p. 51–58, mai. 2000.

RAPPAPORT, Theodore. **Wireless Communications: Principles and Practices**. New Jersey: Prince Hall, 1996.

ROUNDY, S.; WRIGHT, P. K.; RABAEY, J. M. **Energy Scavenging for Wireless Sensor Networks: with Special Focus on Vibrations**. Berlin: Springer, 2003.

SALEH, Ahmed S.; ALI, Borhanuddin M.; RASID, Mohd F.; ISMAIL, Alyani. A survey on Energy Awareness mechanisms in Routing Protocols for Wireless Sensor Networks using optimization methods. **Transactions on Emerging Telecommunications Technologies**, v. 25, n. 12, p. 1184-1207. Dez, 2014.

ZHOU, Hai-Ying; LUO, Dan-Yan; GAO, Yan; ZUO, De-Cheng. Modeling of Node Energy Consumption for Wireless Sensor Networks. **Wireless Sensor Network**. Harbin, v. 03, n. 01, p. 18 – 23, jan. 2001.