UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

VANESSA ARACY SORIA CORTEZ

ALGORITMO DE ROTEAMENTO DE MENSAGENS CONFIÁVEL COM CONTROLE DE ENERGIA APLICADO ÀS REDES SEM FIO INDUSTRIAIS

Porto Alegre
2021

VANESSA ARACY SORIA CORTEZ

ALGORITMO DE ROTEAMENTO DE MENSAGENS CONFIÁVEL COM CONTROLE DE ENERGIA APLICADO ÀS REDES SEM FIO INDUSTRIAIS

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Rio Grande do Sul como parte dos requisitos para a obtenção do título de Mestre em Engenharia Elétrica.
Área de concentração: Controle e Automação

ORIENTADOR: Prof. Dr. Ivan Müller

CO-ORIENTADOR: Prof. Dr. Marcelo Götz

Porto Alegre
2021

VANESSA ARACY SORIA CORTEZ

ALGORITMO DE ROTEAMENTO DE MENSAGENS CONFIÁVEL COM CONTROLE DE ENERGIA APLICADO ÀS REDES SEM FIO INDUSTRIAIS

Esta dissertação foi julgada adequada para a obtenção do título de Mestre em Engenharia Elétrica e aprovada em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: _____
Prof. Dr. Ivan Müller, UFRGS
Doutor pela Universidade Federal Do Rio Grande Do Sul – Porto Alegre, Brasil

Banca Examinadora:

Prof. Dr. Flávio Morais de Assis Silva, UFBA
Doutor pela Universidade Técnica de Berlin, Alemanha

Prof. Dr. João César Netto, UFRGS
Doutor pela Universite Catholique de Louvain – Louvain-la-Neuve, Bélgica

Prof. Dr. Carlos Eduardo Pereira, UFRGS
Doutor pela Universidade de Stuttgart – Stuttgart, Alemanha

Coordenador do PPGEE: _____
Prof. Dr. Sérgio Luís Haffner

Porto Alegre, dezembro de 2021.

AGRADECIMENTOS

Aos professores Ivan Müller e Marcelo Götz pela confiança e oportunidade de trabalhar nesta área de pesquisa e permitir desenvolver este trabalho.

Aos colegas do LASCAR que foram um suporte fundamental nó só na base do desenvolvimento deste trabalho mas também na partilha diária de conhecimentos e experiências.

A minha família, meus pais e irmã que são sempre o suporte em cada etapa da minha vida e que constantemente me incentivam a continuar crescendo e progredindo pessoal e profissionalmente.

A meu noivo que tem sido sempre um grande apoio de vida, ainda mais nos últimos dois anos onde os dois, fora do nosso país, crescemos e aprendemos mais e mais um com o outro.

À CAPES pela provisão da bolsa de mestrado.

Ao Programa de Pós-Graduação em Engenharia Elétrica da UFRGS por tudo o apoio acadêmico e assistência em todas as etapas do processo de mestrado.

Finalmente a Deus por ser o gestor principal de todo na minha vida.

RESUMO

As redes sem fio industriais (RSFI) têm sido altamente estudadas e pesquisados principalmente por demostrar flexibilidade em sua implementação e baixos custos de manutenção, e isso tem incrementado o interesse pelo uso destas redes em aplicações simples onde o número de pontos (sensores e atuadores) no processo é baixo (entre 10 e 100) até processos em que se precisa controlar e monitorar de cem a milhares de pontos. Mas, com o aumento da complexidade dos processos aumenta também as exigências dentro da rede: segurança, confiabilidade na entrega de pacotes, latência e eficiência no consumo da energia são algumas das características onde as redes sem fio devem apresentar um controle mais robusto para assegurar um ótimo desempenho. Uma das etapas mais complexas dentro da comunicação sem fio é o roteamento de dados, processo de troca de informação entre dois ou mais dispositivos da rede onde devem ser determinados os caminhos para a correta comunicação entre estes dispositivos. Este trabalho apresenta a implementação de diferentes protocolos de roteamento aplicados a RSFI assim como propõe um algoritmo para a criação de rotas confiáveis e com controle no estado de energia dos nós. O algoritmo está dividido em duas etapas, primeiro a formação da árvore de comprimento total mínimo entre todos os nós da rede, onde, nessa formação, um valor de custo é atribuído às arestas da rede; na segunda etapa a formação de rotas confiáveis com redundância para os grafos *uplink* e *downlink* onde a escolha dos nós para a formação dos caminhos tem em consideração um valor de custo associado a características dos dispositivos. O algoritmo proposto leva em consideração características dos dispositivos assim como os parâmetros dos relatórios de estado dos dispositivos que são compartilhados com o gerenciador de rede. Este trabalho foi validado em simulações onde se estabelecem condições diferentes para o estado dos nós e os resultados mostram a adaptabilidade do algoritmo em estabelecer as melhores rotas disponíveis.

Palavras-chave: Redes sem fio industriais, roteamento, confiabilidade, QoS, controle energia, grafos.

ABSTRACT

Industrial wireless networks (IWN) have been highly studied and researched mainly for demonstrating flexibility in their implementation and low maintenance costs, this has increased the interest in the use of these networks in simple applications where the number of devices in the process is low (10 -100) to processes where you need to control and monitor from one hundred to thousands of devices. But with the increase in the complexity of the processes, the demands within the network also increase: security, latency, reliability in packet delivery and energy consumption efficiency are some of the characteristics where wireless networks must have a more robust control to ensure optimum performance. One of the most complex steps in wireless communication is data routing, a process of exchanging information between two or more network devices where the paths between devices that want to establish communication to achieve data transfer must be determined. This work implements different routing protocols applied to IWN as well as proposes an algorithm for the creation of reliable routes with control over the energy state of the nodes. The algorithm is divided into two steps, first the formation of the minimum total length tree among all the nodes of the network, in this formation a cost value is assigned to the edges of the network; in the second stage, the formation of reliable routes with redundancy for the *uplink* and *downlink* graphs where the choice of nodes for the formation of paths takes into account a cost value associated with device characteristics. The proposed algorithm considers device characteristics as well as device status report parameters that are shared with the network manager. This work was validated in simulations where different conditions are established for the state of the nodes and the results show the adaptability of the algorithm in establishing the best available routes.

Keywords: Industrial wireless networks, Routing, reliability, Qos, graphs.

LISTA DE ILUSTRAÇÕES

Figura nº 1 – Exemplo Rede Sensores sem Fio 11
Figura nº 2 – Rede representada por grafo *G = (V_m, E_n)* 16
Figura nº 3 – Exemplo Rede Malha Sensores sem Fio 19
Figura nº 4 – Tipo de grafo Broadcast 24
Figura nº 5 – Tipo de grafo Uplink 25
Figura nº 6 – Tipo de grafo Downlink 26
Figura nº 7 – Processos executáveis 38
Figura nº 8 – Diagrama de classes 39
Figura nº 9 – Diagrama de fluxo do módulo 40
Figura nº 10 – Estrutura de rede 41
Figura nº 11 – Dispositivo de campo 42
Figura nº 12 – Sniffer Wi-Analys 43
Figura nº 13 – Blocos principais modelo proposto 43
Figura nº 14 – Topologia de rede com 50 dispositivos 65
Figura nº 15 – Topologia de rede com 120 dispositivos 65
Figura nº 16 – Topologia de rede com 180 dispositivos 66
Figura nº 17 – Valor de Salto Máximo 67
Figura nº 18 – Valor de Salto Médio 68
Figura nº 19 – Dispositivos Distantes 68
Figura nº 20 – Dispositivos Com Rotas Redundantes 69
Figura nº 21 – Quantidade de enlaces 70
Figura nº 22 – Grafo *uplink* proposta de Han 71
Figura nº 23 – Grafo *downlink* proposta de Han 72
Figura nº 24 – Grafo *uplink* proposta de Kunzel 73
Figura nº 25 – Grafo *downlink* proposta de Kunzel 74
Figura nº 26 – Grafo *uplink* proposta ELHFR 74
Figura nº 27 – Grafo *downlink* proposta ELHFR 75
Figura nº 28 – Grafo *uplink* proposta do trabalho 75
Figura nº 29 – Grafo *downlink* proposta do trabalho 75
Figura nº 30 – Diagrama Fluxo: Teste funcionalidade 76
Figura nº 31 – Execução algoritmo proposto: 1 dispositivo 77
Figura nº 32 – Informação sniffer: um dispositivo, cmd969 78
Figura nº 33 – Execução algoritmo proposto: 2 dispositivos 79
Figura nº 34 – Informação sniffer: dois dispositivos, cmd969 80

LISTA DE TABELAS

Tabela nº 1 – Comparação rápida de requisitos 22
Tabela nº 2 – Trabalhos relacionados 36
Tabela nº 3 – Informações da rede 64
Tabela nº 4 – Resumo resultados obtidos 70

LISTA DE ABREVIATURAS

ELHFR	*Enhanced least-hop first routing*
IIoT	*Industrial Internet of Things*
ISA	*International Society of Automation*
IWN	*Industrial Wireless Networks*
OSI	*Open Systems Interconnection*
QoS	*Quality of Service*
RSFI	Redes Sem Fio Industriais
RSL	*Received Signal Level*
WIA-PA	*Wireless Networks for Industrial Automation Process*
WH	*WirelessHart*

SUMÁRIO

1 INTRODUÇÃO 10
1.1 Motivação 13
1.2 Objetivos 14

2 FUNDAMENTAÇÃO TEÓRICA 15
2.1 Conceitos básicos sobre grafos 15
2.2 Redes industriais sem fio 16
2.2.1 Dispositivos específicos da rede 17
2.2.2 Redes industrias sem fio em malha 18
2.2.3 Gerenciamento centralizado 21
2.3 Roteamento em Redes sem Fio Industriais em malha 22

3 ANÁLISE DO ESTADO DA ARTE 29

4 MATERIAIS E MÉTODOS 37
4.1 Materiais 37
4.1.1 Gerenciador de rede 37
4.1.2 Módulo de simulação 38
4.1.3 Gateway 40
4.1.4 Ponto de acesso 41
4.1.5 Dispositivos de campo 42
4.1.6 *Sniffer* 42
4.2 Métodos 42

5 IMPLEMENTAÇÃO 46
5.1 Algoritmos de roteamento implementados 46
5.1.1 Algoritmo de Han 46
5.1.2 Algoritmo de Künzel 51
5.1.3 Algoritmo ELHFR 53
5.1.4 Algoritmo de roteamento proposto 56

6 ANÁLISE DE RESULTADOS 64
6.1 RSFI simulada 64
6.2 Teste de funcionalidade prático 76

7 CONCLUSÃO 81

REFERÊNCIAS 83

1 INTRODUÇÃO

Com o passar dos anos, as redes sem fio têm sido cada vez mais aplicadas no monitoramento de ambientes e na busca por procedimentos com controle mais dinâmico e preciso, especialmente por apresentarem baixo custo de implementação. Nesse sentido, a tecnologia possibilitou o desenvolvimento e integração de dispositivos de sensoriamento e atuação com características de baixo consumo de potência, tamanho reduzido, autoconfiguração e cada vez mais robustos (RUAY-SHIUNG; CHIA-JOU, 2006). Estes avanços deram lugar à formação de redes de sensores sem fio, um conceito que vem sendo de interesse e que tem como objetivo incrementar eficiência em aplicações diversas (PENG *et al.*, 2006).

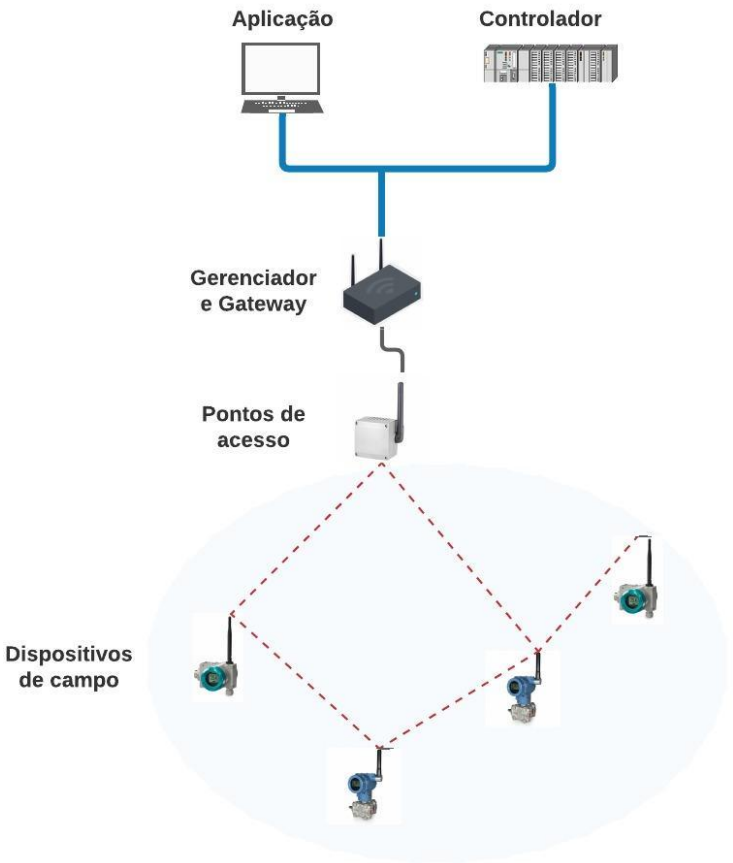
O emprego das redes sem fio vão desde aplicações civis até militares, industriais, residenciais, saúde, vigilância e reconhecimento, segurança, entre outras. Uma das principais vantagens dessas redes é que permitem flexibilidade e baixo custo na instalação e manutenção, devido em grande parte à facilidade na inserção de dispositivos, além da redução considerável dos custos de cabeamento. No entanto, muitas pesquisas estão sendo encaminhadas para prover uma qualidade de serviço (QoS) cada vez maior para este tipo de redes (NOBRE; SILVA; GUEDES, 2015).

As vantagens das redes de sensores sem fio as colocam na mira dos processos industriais, isso as torna uma tendência atual conseguindo aos poucos que as tecnologias cabeadas presentes na indústria estejam sendo mudadas assim como também novos padrões de redes sem fio sejam estabelecidos para atender os requerimentos das plantas (NOBRE; SILVA; GUEDES, 2015). O *WirelessHART* (WH) é um desses padrões, lançado oficialmente em setembro de 2007 e aprovado pela IEC (International Electrotechnical Commission) em 2008, sendo um padrão sem fio aberto e interoperável para automação de processos.

O desempenho dos dispositivos dentro da RSFI irá determinar a segurança e robustez da rede. Em geral, este tipo de rede é composta por quatro elementos principais com tarefas delimitadas: *dispositivos de campo, pontos de acesso, gateway e gerenciador*. Os dispositivos de campo são responsáveis por executar funções de detecção ou atuação, o ponto de acesso é responsável pela conexão dos dispositivos de campo ao gateway, e o *gateway* atua como ponte entre o gerenciador, os pontos de acesso e protocolos externos, e finalmente o gerenciador é responsável pela configuração da rede, agendamento de comunicações, roteamento, escalonamento, entre outras tarefas (ZUO; LING; YUAN, 2013) (CHEN; NIXON; MOK, 2010).

Os dispositivos de campo vêm apresentando grandes evoluções e podem ter um comportamento autônomo com capacidade para formar redes e atuar de acordo com os diferentes modelos e arquiteturas (KÜNZEL, 2012). Na figura 1 é apresentada uma topologia simples com os dispositivos representativos de uma rede sem fio industrial.

Figura 1 – Exemplo Rede Sensores sem Fio



Fonte: autor

Embora o comportamento autônomo dos dispositivos de campo seja importante para a robustez da rede, ter um gerenciamento centralizado otimiza o desempenho da rede em muitas das aplicações. No gerenciamento centralizado se tem uma comunicação e troca constante de dados entre o gerenciador e os dispositivos finais (dispositivos de campo), os nós não são capazes de gerar seus próprios parâmetros de configuração, é o gerenciador o encarregado desta tarefa e outras, como a geração de rotas de dados, controle de adição e saída de dispositivos na rede, controle de prevenção de colisões e distribuição de carga (CHEN; NIXON; MOK, 2010).

As tarefas do gerenciador para um controle centralizado são essenciais ainda mais para uma rede com topologia em malha onde soluções são empregadas para aumentar a confiabilidade da comunicação, com características de auto-organização e autoconfiguração para um maior desempenho, por exemplo. Além das vantagens das RSFI analisadas em muitas pesquisas e contidas na literatura estas redes também apresentam desafios em função da restrição de recursos (NOBRE; SILVA; GUEDES, 2015). Como antes mencionado, uma das tarefas do gerenciador é a programação geral das comunicações da rede (CHEN; NIXON; MOK, 2010).

Os algoritmos de roteamento em princípio podem ser classificados em três categorias: algoritmos proativos, algoritmos reativos e algoritmos híbridos. Nos algoritmos proativos os nós tentam manter atualizada a informação das rotas enquanto os algoritmos reativos só atualizam o gerar a informação em determinados momentos como na falha de uma rota ou quando um novo dispositivo estabelece sessão. Finalmente, os algoritmos híbridos combinam as melhores características dos algoritmos proativo e reativo (ZUO; LING; YUAN, 2013).

Apesar dos esforços em muitos trabalhos de pesquisa por otimizar o processo de transferência de dados em RSFI, os protocolos de roteamento continuam sendo um grande objeto de estudo para colocar as redes de comunicação sem fio ao nível atual das redes cabeadas nas aplicações industriais aumentando sua robustez.

Os recursos envoltos no roteamento tornam este processo um componente crítico da rede que precisa de maior controle sobre confiabilidade, determinismo, latência, consumo e utilização eficiente de recursos, entre outros dentro da transferência de dados. Uma rede em malha sem fio com gerenciamento centralizado apresenta vantagens consideráveis para o roteamento, pelo fato de se poder estabelecer várias rotas redundantes, o que incre-

menta a confiabilidade da rede. Aliado a isto, o roteamento consciente do gasto energético pelos nós, leva a um maior tempo de existência dos dispositivos na rede, até a troca da bateria (JHA; APPASANI; GHAZALI, 2019).

1.1 Motivação

As demandas industriais crescem cada vez mais e necessitam tecnologias que possam atender as exigências de processos cada vez mais complexos e dinâmicos, com maior segurança e melhor confiabilidade. O processo de roteamento em RSFI é crítico e seu correto funcionamento possibilita a melhora do desempenho da rede. Atualmente, este processo ainda apresenta desafios, que demandam estudos com o fim de apresentar alternativas viáveis para aumentar a confiabilidade das redes e melhorar características gerais de funcionamento.

O gerenciador de rede em RSFI tem uma quantidade considerável de tarefas sendo executadas e controladas paralelamente, o que faz com que muitos dos algoritmos de roteamento aplicados nos protocolos procurem ser eficientes e simples sem muitas considerações adicionais, embora essas considerações possam aumentar a robustez com o custo de aumentar a complexidade da implementação e execução. Mas, na atualidade, nem sempre essa visão atende todas as aplicações na indústria, dado que muitas considerações podem ser deixadas de fora, sendo que elas podem ser essenciais para o funcionamento robusto da planta.

Atualmente os processos na indústria estão sendo avaliados por indicadores relacionados ao funcionamento dos equipamentos na planta pelo que cada vez mais se leva em consideração todas as possíveis soluções para ter eficiência e otimização no uso de recursos. Com o objetivo de otimizar o uso de recursos e melhorar o tempo de vida da rede, muitos algoritmos de roteamento surgiram para atender diversas propostas: escolha de caminhos ótimos(HAN *et al.*, 2011), algoritmo equilibrado em carga (CHU *et al.*, 2009) ou equilibrado em energia (ZHANG; YAN; MA, 2013a), assim como muitos trabalhos na literatura apresentam um estudo comparativo dos protocolos de roteamento analisando as métricas que possam determinar um melhor desempenho para rede (URBINA *et al.*, 2015) (PIECHOWIAK *et al.*, 2016). Todos eles levam em consideração um parâmetro específico da rede, o que acarreta em melhorias em certos recursos ao mesmo tempo que outros podem apresentar características não desejáveis.

Energia é um recurso limitado em grande parte dos dispositivos de uma RSFI e ao mesmo tempo um dos recursos mais usados pelo processo de roteamento. A confiabilidade dos dispositivos segundo seu estado e histórico de funcionamento pode ser relevante para a decisão de adição ou não de um dispositivo em uma rota. Mas estas características nem sempre são avaliadas na formação de grafos e definição de caminhos. A proposta deste trabalho é definir um algoritmo de roteamento que leve em consideração parâmetros de estado e algumas características dos dispositivos para a formação de rotas com a maior confiabilidade possível, que possa ser aplicado a RSFI.

1.2 Objetivos

Da análise dos desafios que apresenta o roteamento, surge o interesse por este processo, o que leva a estabelecer os objetivos deste trabalho. O objetivo principal é desenvolver um algoritmo de roteamento para construção de grafos *uplink* e *downlink* com rotas confiáveis visando sua aplicação em RSFI em malha.

Os objetivos específicos são:

- Implementar diferentes algoritmos de roteamento para RSFI;
- Propor uma estratégia de roteamento que analise parâmetros de confiabilidade e características dos nós;
- Avaliar os algoritmos implementados na construção dos grafos da rede;
- Avaliar o algoritmo proposto em diferentes cenários;
- Implementar o algoritmo proposto em uma RSFI prática.

A organização dos capítulos deste trabalho é a seguinte: no capítulo 2 são apresentados fundamentos e conceitos importantes para o processo de roteamento; o capítulo 3 apresenta uma breve análise por casos de estudo relevantes no estudo e desenvolvimento de algoritmos para o roteamento; os materiais usados para o desenvolvimento deste trabalho assim como os métodos propostos são apresentados no capítulo 4; já no capítulo 5 se mostra como foi feita a implementação dos algoritmos junto com a análise dos resultados obtidos; e para finalizar, o capítulo 6 com as conclusões do trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, procura-se estabelecer e discutir conceitos importantes como base para o desenvolvimento desta dissertação. O capítulo apresenta definições básicas sobre a formação de grafos, RSFI em malha, uma breve revisão do conceito de camadas de um protocolo de comunicação e mais especificamente a camada de rede, por ser a encarregada do processo de roteamento, protocolos de roteamento e alguns dos algoritmos mais relevantes desenvolvidos para este processo.

2.1 Conceitos básicos sobre grafos

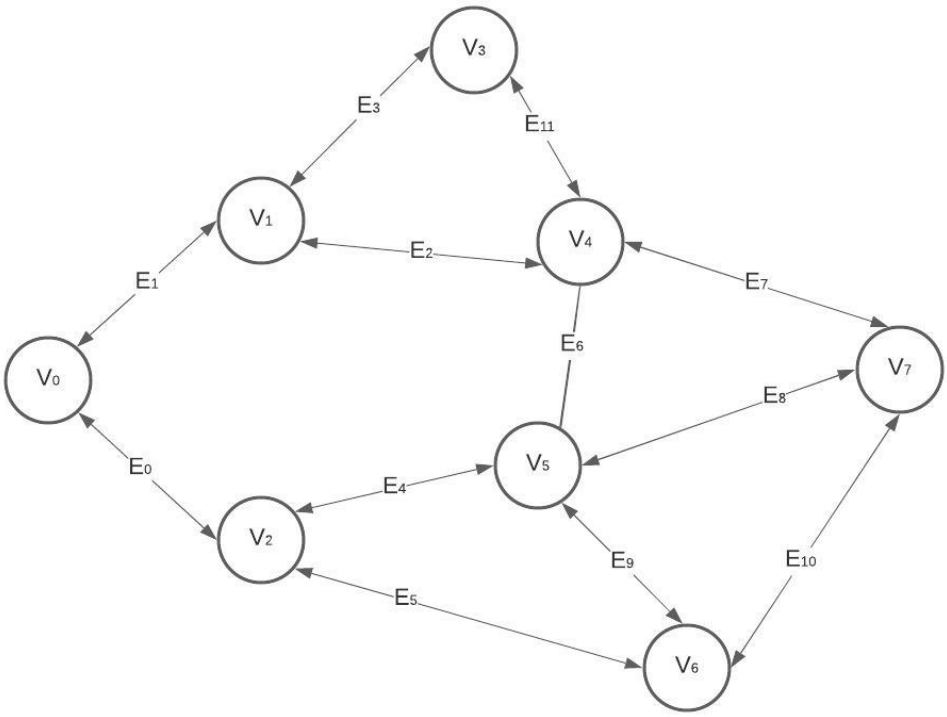
Nas redes de comunicação sempre procura-se ter uma modelagem o mais perto do sistema real que permita avaliar corretamente métricas ou padrões de comportamento de interesse, e nesse sentido, a representação de uma rede por grafos é muito utilizada independentemente de ser esta uma rede cabeada ou sem fio.

Segundo (CHEN; NIXON; MOK, 2010) um grafo é uma coleção de caminhos que conectam os nós da rede e uma rede modelada formalmente por um grafo é representada como *G = (V, E)* onde *V* é o conjunto de vértices da rede que representa aos dispositivos e *E* é o conjunto de arestas da rede que representa as ligações entre os dispositivos, sendo importante levar em consideração que estas só existem se os dispositivos da ligação são capazes de se comunicar (CHRIS; GORDON, 2001).

Na figura 2 tem-se uma topologia de rede representada por grafos com os V_m vértices ou dispositivos conectados pelas arestas E_n. As setas nas arestas representam o sentido em que a comunicação pode acontecer, no caso da figura 2 todas as arestas têm setas nas suas duas extremidades mas também é possível encontrar arestas com um só sentido ou representações de grafos sem setas nas arestas, e neste caso, assume-se que existe

comunicação nos dois sentidos e não existe a comunicação só em sentido único.

Figura 2 – Rede representada por grafo $G = (V_m, E_n)$



Fonte: autor

Na representação de redes por grafos é comum definir pesos ou custos que caracterizam o estado das ligações entre os dispositivos (arestas) normalmente associados com métricas de confiabilidade do enlace. Esse custo é um valor calculado e na figura do grafo pode ser escrito acima das arestas, o que proporcionaria um grafo valorado com maior informação visível (KÜNZEL, 2012).

2.2 Redes industriais sem fio

A globalização da produção requer cada vez mais indústrias com processos automatizados de alto desempenho mas com custos baixos e controláveis. A utilização de tecnologias de redes sem fio proporciona instalações simples, flexíveis e de baixo custo além da confiabilidade e segurança no tratamento da informação.

Uma rede industrial sem fio é o conjunto de dispositivos e aplicações trabalhando dentro de diversos processos em uma planta e que proporciona estratégias especificas para melhorar o desempenho e eficiência da indústria. Este tipo de redes é amplamente

usado para coletar dados dos sistemas que permitam detectar mudanças nos processos e tomar decisões que levam a ações dentro das aplicações e que em conjunto garantam um controle efetivo (KHAKPOUR; SHENASSA, 2008).

Algumas das características das RSFI são:

Redução de custos: Tem-se uma redução de custo considerável ao eliminar toda engenharia e materiais relacionada à instalação e manutenção no uso de cabos.

Flexibilidade: Uma rede sem fio proporciona maior flexibilidade na adição, exclusão ou mudança de lugar de equipamentos da rede inicial.

Eficiência: Este tipo de rede permite o tratamento de um volume grande de informação assim como o monitoramento remoto em tempo real que permite uma tomada de decisões mais rápida que garante o melhor desempenho da rede (em comparação com outros tipos de redes sem fio).

Segurança: Permite maior segurança na hora de coletar informação em locais de difícil acesso ou de condições não aptas para humanos.

Os constantes estudos e avanços no desenvolvimento das RSFI permitem estar de acordo com conceitos atualmente muito importantes da indústria como, indústria 4.0, fábrica inteligente(em inglês *Smart Factory*) ou Internet das coisas industrial (em inglês *Industrial Internet of Things(IIoT)*)

2.2.1 Dispositivos específicos da rede

Dentro de um sistema de controle de processo tradicional, normalmente se tem um dispositivo com a capacidade de obter leitura da variáveis dos processos, denominado sensor, um controlador, que faz a análise dos dados coletados e determina se um outro dispositivo denominado atuador, deve executar alguma ação para manter ou modificar a variável do processo. Essa é uma descrição rápida e simples do processo de controle, mas a rede toda tem mais alguns dispositivos importantes com funções bem estabelecidas que garantem o funcionamento correto, propiciando comunicação eficaz. Na figura 3 pode-se observar alguns destes, descritas em detalhes de acordo com (CHEN; NIXON; MOK, 2010).

Gerenciador da rede: É o centro da rede e o desempenho da mesma depende em grande parte da operação do gerenciador, responsável pelas configurações e monitoramento da rede, gerenciamento e manutenção das rotas de comunicação entre os dispositi-

vos, gerenciamento dos relatórios da integridade da rede, e permite o acesso às informações da rede por parte das aplicações.

Gateway: Este dispositivo estabelece a comunicação entre o gerenciador e os dispositivos de campo. Além de ser o responsável por coletar as mensagens de resposta de todos os dispositivos da rede, o *gateway* também é ser usado para converter um protocolo de comunicação a outro. As funcionalidades do *gateway* também podem variar. como por exemplo, quando uma rede tem diversos pontos de acesso, eles tem uma comunicação direta com o gateway.

Ponto de acesso: O ponto de acesso fornece comunicação direta com qualquer dispositivo da rede sem fio para o qual o gerenciador de rede estabeleceu uma rota. Na criação da rede é dito também que a rede está ativa quando o primeiro ponto de acesso começa a transmissão das mensagens de anúncio, mensagens notificam a existência da rede para todos os dispositivos dentro do alcance do rádio.

Dispositivos de campo: Em geral os dispositivos de campo são os sensores e atuadores que vão monitorar e atuar de forma a garantir o correto desempenho da aplicação industrial.

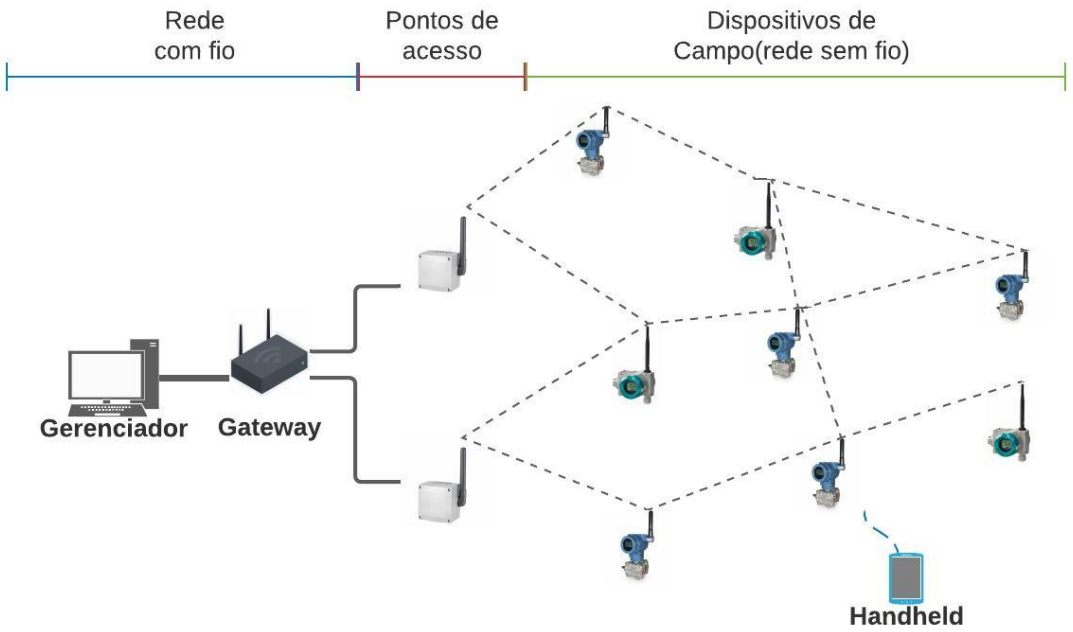
Dispositivo portátil: Também conhecido por seu nome em inglês *handheld*, é um sistema computacional portátil que permite a configuração, calibração dos dispositivos de campo, assim como a execução de diagnósticos. Em teoria, todas essas funções são feitas *in situ* ao dispositivo de interesse.

Todos estes dispositivos são parte integral de uma rede industrial sem fio e o correto funcionamento de cada uma de suas tarefas durante o funcionamento da rede vão determinar o nível de desempenho e eficiência da mesma. Na figura 3 se apresenta a topologia de uma rede em malha, que é uma das topologias mais utilizadas em RSFI, com seus dispositivos básicos.

2.2.2 Redes industrias sem fio em malha

Atualmente cada vez mais plantas adotam o uso de redes sem fio como uma resposta para ter processos mais dinâmicos e eficientes. As vantagens que apresentam as redes indústrias sem fio podem ser diversas, e são potencializadas pela topologia escolhida no processo de engenharia para a criação da rede. A topologia em malha apresenta grande vantagem quando aplicada em redes industriais, que beneficia a robustez necessária pela

Figura 3 – Exemplo Rede Malha Sensores sem Fio



Fonte: autor

severidade e precisão dos processos de plantas industriais, que precisam de requisitos cada vez mais rigorosos para atingir um nível ótimo de desempenho (HAN *et al.*, 2011).

As redes em malha foram pensadas para prover uma infraestrutura de comunicação que adapte-se à dinâmica dos processos e promova o fortalecimento da qualidade do serviço da rede (QoS). Nesta topologia, os dispositivos de campo têm conexões uns com os outros, o que permite que além da sua capacidade de enviar e receber dados, também têm a capacidade de atuar como roteadores, isto é, eles podem retransmitir mensagens para dispositivos com os quais tenham estabelecido um enlace de comunicação. Graças este recurso um pacote de dados vai encontrar uma rota até seu destino passando por nós em um caminho previamente definido pelo gerenciador da rede (LINDHORST; LUKAS; NETT, 2013).

Autoconfiguração: Uma característica importante da rede em malha é sua capacidade de autoconfiguração, ela não precisa de um administrador humano para cada decisão que a rede precise tomar; se um dispositivo se deslocar ou precisa entrar na rede, descobre ele e adiciona automaticamente ao sistema existente, respeitando o cumprimento de certos requisitos importantes, mas sem intervenção externa, o que a converte em uma rede altamente adaptável.

Auto-organização: Sua capacidade de auto-recuperação fornece uma alta eficiência, se algum dos dispositivos da rede ficara inativo a rede conseguirá identificar a falha e usar caminhos alternativos para a transmissão de mensagens. A rede não precisa de intervenção para realizar o processo de recuperação de comunicação e retransmissão de mensagens por falha.

Redundância: A redundância é muito importante na confiabilidade da rede e atualmente é sempre um requisito nos sistemas de monitoramento e controle, dado que nas redes industriais, manter um processo com funcionamento contínuo é uma necessidade não só pelo nível de desempenho mas também para manter a segurança dos processos. O desenho da topologia em malha permite que cada nó se conecte com mais de um dispositivo, o que permite à rede seguir operando ainda quando um dispositivo esteja inativo ou com falhas. O nível de redundância está diretamente associado à quantidade de dispositivos dentro da rede. Se a rede é muito densa, isso ajudará a ter maior redundância, mas , por outro lado, pode afetar outras características importantes se isso não for implementado com o devido cuidado. É importante ter em consideração que os dispositivos de campo não são os únicos capazes de proporcionar redundância na rede, com engenharia adequada, os recursos suficientes, ter mais de um gateway, vários pontos de acesso ou um gerenciador de rede em reserva a mais, vai permitir a rede atuar em caso de falhas nos dispositivos principais sem deixar a rede cair (CHEN; NIXON; MOK, 2010).

Escalabilidade: Dado que os nós em uma rede em malha não só são capazes de receber e enviar dados mas também de retransmitir mensagens para seus vizinhos, a rede pode ser escalável. Uma rede em malha tem a capacidade de aumentar consideravelmente o número de nós ativos ou diminuir e se adaptar a estas mudanças de escalabilidade sem precisar que grandes mudanças na configuração da rede aconteçam. Em resumo, ao ser uma rede escalável, consegue dar suporte adequado no crescimento da rede sem ter grandes prejuízos no desempenho dos dispositivos da rede original. Além disso, esta característica permite o maior uso de redes sem fio para aplicações que requerem um controle mais dinâmico. A escalabilidade da rede em malha também tem muita relação com o fato dela permitir a integração de novas características e melhoras que possam regular o comportamento da rede. Isso é importante dado que muitas pesquisas ainda estão sendo encaminhadas para melhorar o desempenho da rede.

Robustez: É uma consequência proporcionada por todas as outras características da

rede industrial em malha, sua capacidade de autoconfiguração, auto-recuperação, redundância e escalabilidade, fazem que esta topologia possa estabelecer comunicações robustas, de extrema importância na área industrial, dado que se tem processos de controle rigorosos.

Com o tempo, o uso das RSFI em malha tem se convertido também em um grande foco de pesquisa. O crescimento do uso deste tipo de redes e os requisitos cada vez mais rigorosos das indústrias tem levado aos pesquisadores a estudar e estabelecer melhoras para ter maior controle da latência na transmissão de dados, o consumo de energia dos dispositivos e outras características para melhorar cada vez mais a eficiência da rede. Isso só afirma o nível de importância deste tipo de rede na atualidade.

2.2.3 Gerenciamento centralizado

O tipo de controle usado para o gerenciamento de uma rede pode apresentar características, que podem ser muito favoráveis para o sistema melhorando a qualidade de serviço da rede. Nesta seção é apresentada uma análise dos pontos mais favoráveis do controle centralizado, e assim como alguns detalhes estudados sobre a aplicabilidade do controle distribuído.

No gerenciamento centralizado, as tarefas e serviços da rede toda são administrados por um só dispositivo na rede. Os dispositivos da rede não tem a capacidade de lidar com questões como agendamento de mensagens, roteamento de seus próprios pacotes de dados, ocupação de banda e requisitos de tempo real, essas demandas são administradas pelo gerenciador central e permitem a diminuição de conflitos na rede. Este tipo de controle de rede favorece também os requisitos de comunicações em tempo real, fundamentais em aplicações industriais de processo (CHEN; NIXON; MOK, 2010).

O fato de que os dispositivos não executem tarefas complexas de gerenciamento é uma das razões pelas quais os dispositivos para este tipo de rede têm um baixo custo e simplicidade. Neste tipo de controle o gerenciador tem disponibilidade de todas as informações da rede, pelo que cada decisão tomada é baseada neste conjunto de dados. As redes com gerenciamento central permitem engenheira com soluções de ótimo desempenho e com maior segurança das comunicações.

Já no uso de controle de rede descentralizado, supõe-se a necessidade de uma independência das funções e serviços. As decisões neste tipo de controle podem ser as mais

variadas, e podem ser tomadas e gerenciadas localmente, o que pode resultar em perda de pacotes durante a comunicação, problemas de determinismo ou até mesmo segurança. Embora seja uma proposta que visa melhorar a latência e dinamicidade das comunicações, ainda é estudada para prover seu uso com maior desempenho (LAW *et al.*, 2010). Na tabela 1 é apresentada uma comparação sucinta de alguns dos requisitos para o gerenciador e o nível de satisfação dos mesmos segundo o tipo de gerenciamento, seja centralizado ou distribuído.

Tabela 1 – Comparação rápida de requisitos

Requisitos	**Gerenciamento Centralizado**	**Gerenciamento Distribuído**
Segurança	√√√	√
Determinismo	√√√	√
Latência	√	√√√
Dinamicidade	√	√√√
Simplicidade	√√√	√
Custo	√√√	√

2.3 Roteamento em Redes sem Fio Industriais em malha

Um processo fundamental em uma rede de comunicação é o roteamento, um conceito simples define este processo como o encarregado de enviar dados ou pacotes de dados de um dispositivo para outro dentro da rede mas, embora exista uma forma simples de definilo, há uma série de etapas e requerimentos que têm que ser cumpridos para que esta tarefa seja executada corretamente. Pelo seu objetivo no funcionamento da rede, o processo de roteamento também define muitas das métricas de desempenho da rede, e com isso sua qualidade de serviço. Muitas pesquisas e técnicas são desenvolvidas nesta área específica para obter novos algoritmos com propostas para garantir níveis de desempenho maior.

O roteamento em RSFI é levado com cuidado, e muitos protocolos de roteamento foram criados para este tipo de redes, dado que os processos indústrias geralmente são muito rigorosos e o tratamento de informação é uma tarefa crítica dentro dos sistemas das plantas.

Existe uma classificação base para os algoritmos de roteamento. Em geral eles estão

divididos em três grupos importantes: proativos, reativos e híbridos.

a) *Proativos:* Os algoritmos nesta divisão mantém informação atualizada dos caminhos usados para o processo de roteamento fazendo atualizações periodicamente. Isso facilita sua adaptação a mudanças de topologia ou carga na rede. Este tipo de algoritmo pode muitas vezes ajudar a que o tempo de latência na comunicação seja menor, dado que a informação de roteamento está disponível e atualizada em todo momento (URBINA *et al.*, 2015).

b) *Reativos:* Estos algoritmos são também conhecidos como "sob demanda" dado que eles calculam suas rotas só quando for necessário (ZAKRZEWSKA *et al.*, 2010) e isso pode aumentar a latência na comunicação mas também pode significar uma economia de energia significativa.

c) *Híbridos:* Os protocolos de roteamento híbrido usam algoritmos com características proativas de forma local e reativas em um nível geral da rede. Existem premissas dentro do protocolo para a escolha do algoritmo (proativo ou reativo) a usar em cada momento, mas isso vai depender do protocolo implementado.

Muitos algoritmos de roteamento tem servido como base para pesquisas e desenvolvimento de protocolos, e eles têm que cumprir com algumas propriedades que possam garantir sua eficiência e confiabilidade.

Os algoritmos têm que ser corretos na sua funcionalidade, indiferente de suas especificações, o algoritmo tem sempre que cumprir sua função; têm que ter estabilidade em toda a operação da rede, onde os erros não podem afetar de forma crítica o processo de envio e recepção de dados; conforme a tecnologia avança e os processos aumentam seus níveis de exigência, os algoritmos de roteamento tem que estar cada vez mais otimizados nas suas funcionalidades e métricas de avaliação. Finalmente tudo isso tem que levar a obter um algoritmo robusto, sendo essa uma das propriedades mais difíceis de se obter e mais desejadas nos protocolos de roteamento.

O tipo de roteamento usado vai depender da complexidade das aplicações e da dinâmica e topologia da rede, e na escolha do algoritmo, se considera também o tipo de comunicação que é preciso estabelecer entre os dispositivos. Em RSFI, a comunicação entre os dispositivos geralmente é feita das seguintes formas:

- *Unicast:* Neste tipo de comunicação um dispositivo envia um dado ou pacote de dados a outro dispositivo na rede.

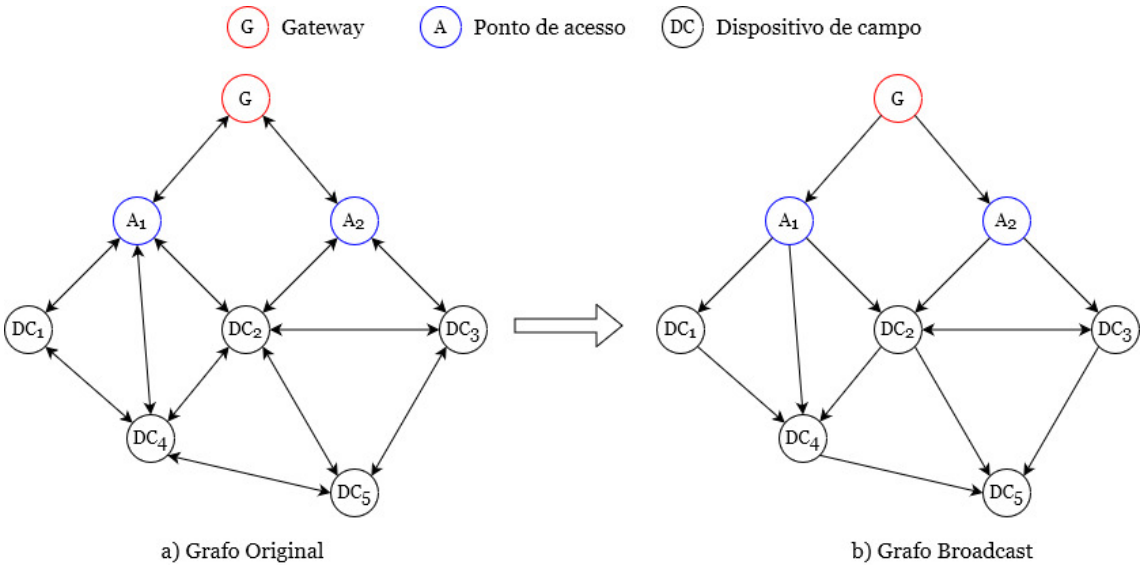
- *Broadcast:* Na comunicação *broadcast* um dispositivo envia um dado ou pacote de dados para todos os outros dispositivos que formam parte da rede.

- *Multicast:* Quando um dispositivo deseja estabelecer comunicação e enviar dados para só um conjunto de dispositivos da rede.

Os grafos são de grande utilidade no entendimento do processo de roteamento e sua simplicidade leva-os a serem muito usados nas RSFI. No desenvolvimento deste trabalho, o uso de grafos é fundamental dado que os algoritmos implementados constroem rotas a partir da formação de um grafo inicial e a informação obtido da execução do algoritmo é também representada por grafos, por isso é importante definir os tipos de grafos mais comuns no roteamento e algumas da suas utilidades.

- *Grafo broadcast:* O grafo *broadcast* estabelece caminhos a partir do grafo original para possibilitar o envio de pacotes desde o *gateway* até todos os demais dispositivos da rede. Este tipo de grafo costuma ser usado para enviar comandos de configuração até os dispositivos de campo.

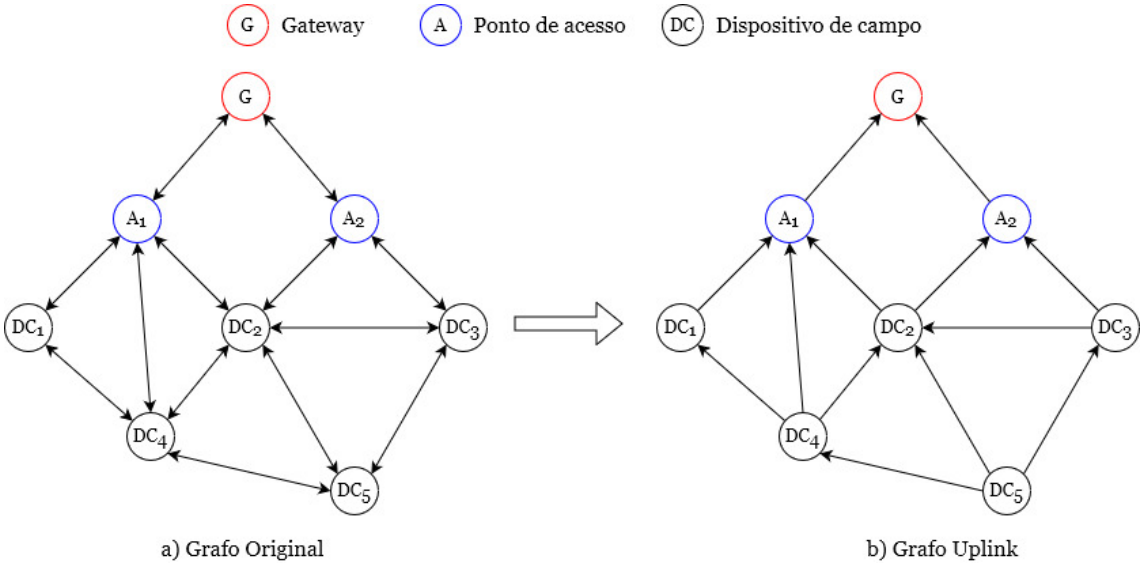
Figura 4 – Tipo de grafo Broadcast



Fonte: Adaptado de (HAN *et al.*, 2011)

- *Grafo uplink:* No grafo *uplink* são apresentados os caminhos que possibilitam aos dispositivos de campo o envio de dados até o gateway, geralmente é usado para o envio de respostas por parte dos dispositivos de campo o envio de leituras de parâmetros ou informações do estado da rede.

Figura 5 – Tipo de grafo Uplink



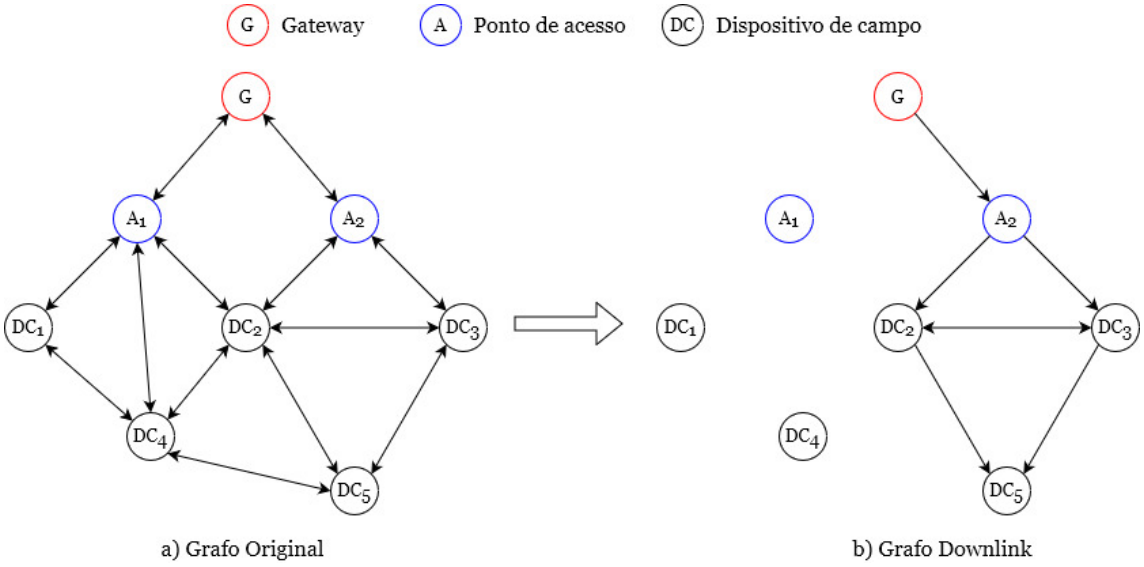
Fonte: Adaptado de (HAN *et al.*, 2011)

- *Grafo downlink:* As rotas estabelecidas no grafo *downlink* permitem que o *gateway* se comunique com um único dispositivo da rede, isso pode ser usado também para o envio de comandos de configuração ou comandos para dispositivos atuadores. A figura 6 apresenta um exemplo com dois grafos, o primeiro representa o grafo original e o segundo a rota desde o *gateway* até o dispositivo DC_5.

Roteamento de caminhos mínimos, distância vectorial, roteamento por inundação ou *link state* são só alguns dos algoritmos que apresentam características de interesse no desempenho das comunicações da rede e têm sido importantes no desenvolvimento de protocolos e de pesquisas inovadoras. Eles também apresentam uma diversidade de conceitos para o entendimento de processos de roteamento desde diferentes modelos de avaliação.

Atualmente protocolos de comunicação para redes sem fio têm sido especificamente desenvolvidos para atender a exigências das aplicações industriais. Esses protocolos apresentam uma série de características de funcionamento que visam melhorar a dinamicidade das plantas com equipamentos para a automação de processos.

Figura 6 – Tipo de grafo Downlink



Fonte: Adaptado de (HAN *et al.*, 2011)

A seguir é apresentada uma breve descrição de alguns protocolos para RSFI, fazendo ênfase na camada de rede e seu processo de roteamento.

2.3.0.1 WirelessHart

O WirelessHart (WH) foi lançado em setembro do 2007 como uma evolução do padrão Hart para comunicações sem fio com tecnologia robusta, mas de implementação simples. Uma rede em malha WH apresenta características de auto-organização e auto-reparação que proporcionam grande flexibilidade para se ajustar a mudanças na infraestrutura da planta. WH é baseado no padrão IEEE 802.15.4, e trabalha apenas na faixa de frequência de 2,4GHz com 15 canais disponíveis. A potência de transmissão neste padrão é programável de -10 dBm até 10 dBm atingindo distâncias máximas de transmissão de até 300m.

Na camada de rede do padrão, é executado o processo de roteamento e seu objetivo é a entrega de dados confiável, que suporta principalmente roteamento por grafo, roteamento de origem e roteamento *proxy*. O roteamento *proxy* é usado durante o processo de adição de um dispositivo na rede, já o roteamento de origem, que estabelece só uma rota para a comunicação entre dois dispositivos, é especialmente usado pelo gerenciador e o gateway, que têm total conhecimento da topologia, para testar a rede validando o funcionamento correto dos dispositivos. O roteamento por grafo é o mais usado durante o funcionamento

normal da rede proporcionando redundância de rotas, maior confiabilidade e segurança na troca de dados. Neste último, o encaminhamento das mensagens é feita segundo um *graph ID* que vai representar o grafo direcional segundo o qual os dispositivos sabem o seguinte dispositivo ao qual encaminhar a mensagem até chegar ao destino. Esse grafo direcional é construído pelo gerenciador a partir das informações compartilhadas dos dispositivos sobre seus vizinhos (CHEN; NIXON; MOK, 2010).

2.3.0.2 WIA-PA

WIA-PA (Wireless Networks for Industrial Automation-Process Automation) foi desenvolvido pelo Consorcio Wireless Industrial Chinês(CIWA) no ano 2007. De acordo com o modelo *Open System Interconnection* (OSI) define 4 camadas: física, enlace, rede e aplicação. A camada física é totalmente baseada no padrão IEEE 802.15.4 com duas bandas de rádio possíveis: 868/915 MHz e 2,4 GHz. A organização da comunicação na rede é centralizada, os roteadores são conectados em uma topologia em malha enquanto a comunicação entre os dispositivos de campo e roteadores segue uma topologia em estrela, ou seja, o protocolo tem uma topologia de rede em dois níveis.

WIA-PA tem um roteamento redundante estático, e esse processo é desenvolvido na camada de rede onde todos os dispositivos têm um endereço de 16 bits que é usado para todas as comunicações na rede. O gerenciador de rede é o encarregado da formação das tabelas de roteamento que são construídas após a obtenção das informações de todos os vizinhos dos roteadores. As tabelas são construídas para obter uma topologia em malha e garantindo que exista redundância de rotas entre os roteadores e o *gateway*. Neste protocolo, os dispositivos são programados para o envio de informação ao *gateway* sobre falhas nos enlaces, estado dos vizinhos e nível de energia. O gerenciador de rede atualiza as informações das tabelas de roteamento baseado nas informações recebidas.

2.3.0.3 ISA 100.11a

Em 2009, este padrão foi aprovado pela Sociedade Internacional de Automação (ISA) como um novo padrão de comunicação sem fio para aplicações industriais com controle de comunicação centralizado. Foi criado baseado nas exigências dos usuários para aplicações com latência de até 100 ms, apresentando 5 das 7 camadas do modelo OSI: física, enlace, rede, transporte e aplicação. ISA 100.11a trabalha na faixa de frequência de 2,4GHz e permite a utilização de 16 canais. A camada de rede neste padrão especifica a

transmissão de pacotes IPv6 em uma rede IEEE 802.15.4 permitindo conexão IP entre os dispositivos e a construção de redes com características de alta escalabilidade.

O protocolo de comunicação industrial utilizado neste trabalho é o WH, pelo fato de ser o mais amplamente empregado, e também por ser objeto de estudos do grupo de pesquisas pertencente a este programa de pós graduação.

3 ANÁLISE DO ESTADO DA ARTE

Neste capítulo são analisados trabalhos que foram desenvolvidos na área de roteamento para redes indústrias sem fio. Foram avaliados uma série de algoritmos propostos para melhorar o processo de encaminhamento de pacotes dentro da rede, e estudos sobre problemas mais comuns nas RSFI e possíveis soluções para melhorar condições de confiabilidade, desempenho e segurança.

Um dos primeiros trabalhos analisados foi (WANG; QUAN, 2010) que propõe um modelo de cadeias de Markov finitas para a predição da confiabilidade na comunicação, sendo que o modelo proposto também pode ser usado como um algoritmo de programação de roteamento em redes sem fio industriais com gerenciamento centralizado e topologia em malha. Os autores consideram que problemas na comunicação em redes sem fio podem ser mitigados pelo uso de avaliações de confiabilidade e tecnologias de predição, pelo que eles desenvolveram um modelo de estados finitos o qual permite avaliar a confiabilidade das rotas possíveis entre dois dispositivos para a entrega de pacotes, dado que o modelo foi configurado com base nas taxas de sucesso do pacote de cada link e na prioridade do grafo. Dessa forma, os autores criaram uma equação que permite calcular a probabilidade de sucesso na comunicação entre dois dispositivos da rede.

Para validar o trabalho, os autores definiram uma rede industrial sem fio com topologia em malha e simularam em *Matlab*. Eles se asseguraram que o grafo proposto tivesse mais de uma rota possível na comunicação entre um dispositivo e o *gateway* e aplicaram o modelo para calcular o caminho com maior probabilidade de sucesso na comunicação, sendo que o caminho com o maior valor foi certamente a rota com as melhores condições para o encaminhamento de pacotes. O calculo da probabilidade é feita em base de um só parâmetro dos dispositivo o que poderia em redes práticas não dar uma boa resposta sendo que na construção da rede esse parâmetro é igual em todos os dispositivos o que

poderia levar ao algoritmo a precisar de mais tempo de funcionamento antes de ter uma resposta esperada.

O trabalho de (TAN; PRASETIADI; KIM, 2012) procura alcançar o balanço do tráfego de dados, pela detecção de áreas congestionadas e distribuição de pacotes em caminhos diferentes. Os autores neste trabalho desenvolveram um modelo para estabelecer um algoritmo de roteamento baseado em gradiente, onde os dispositivos da rede tem um índice de gradiente que contêm duas informações importantes: o custo da distância entre a fonte e o destino e a informação de tráfego dos nós vizinhos. O algoritmo implementado no trabalho está dividido em três etapas chave: na primeira etapa, as informações sobre o tráfego de informação é distribuída para todos os nós, e se definem os vizinhos dos nós e os custos dos enlaces; na segunda etapa, os caminhos para roteamento são calculados e os pacotes de dados podem ser enviados; e uma última etapa, onde as informações de roteamento são atualizadas e o rendimento do modelo depende em grande parte da escolhia acertada de dois fatores numéricos definidos no modelo (α e β).

Para a obtenção de resultados foram realizadas simulações e feitas comparações com o protocolo padrão de roteamento pelo caminho mais curto(SPF pelas suas siglas em inglês), uma topologia aleatória foi gerada e foram usadas a camada física e camada de enlace do padrão IEEE 802.15.4. Os resultados numéricos mostram uma melhora no atraso geral do pacote com taxas de tráfego variadas e o rendimento da rede baseada na taxa de transferência é 18% maior que o protocolo padrão. Mas o trabalho não apresenta uma análise para a escolha de valores ótimos dos fatores do modelo pelo que o desempenho todo da rede pode não ser o mais eficiente.

O trabalho apresentado em (ZHAO *et al.*, 2012) apresenta um estudo aprofundado dos problemas apresentados pelos protocolos de roteamento em RSFI, onde são discutidos seus desempenhos e algumas das mais importantes características e desafios que apresentam. Neste trabalho, os autores fazem uma análise de questões chave, tais como: confiabilidade, comunicação em tempo real, custo de energia e segurança, propondo para cada um destes tópicos, um modelo de descrição. Na data da publicação do artigo, os autores já estabelecem possíveis direções de melhora para trabalhos futuros, tais como, gerenciamento centralizado, mecanismos de auto adaptação e roteamento por grafo. Já em (WINTER *et al.*, 2013), é apresentado um estudo mais específico do protocolo WH, com suas principais características, e foca-se principalmente nos mecanismos de roteamento.

O trabalho pretende evidenciar os problemas e soluções dos mecanismos de roteamento no protocolo WH. Logo após de ter analisado os detalhes, os autores estabelecem um caso de estudo que inclui um experimento com uma rede WH e dispositivos que atendem os requerimentos do protocolo. O uso de comunicação com múltiplos saltos é aplicada, e se faz uma análise de roteamento apresentando a relação entre os dispositivos e seus vizinhos, seguida de uma análise do tempo de roteamento de mensagens entre os dispositivos, além de outras considerações no processo de envio de pacotes, como o mecanismo de roteamento baseado em *superframe*. Os resultados da experiencia permitiram a análise de fatores quanto aos mecanismos de roteamento em WH, e se determinou que o tempo de roteamento de pacotes é desde 110 ms até 2,8 s em uma comunicação de dois saltos, e ainda se concluiu que as estratégias usadas pelo gerenciador de rede comercial para roteamento de mensagens no garante a melhor programação para a transferência de dados. O desenvolvimento apresenta um estudo abrangente e de utilidade mas não chega a definir uma estratégia de roteamento.

Trabalhos onde os valores de peso nos nós são utilizadas para fornecer informações de direcionamento também tem sido foco de pesquisas, como em (YU *et al.*, 2013), em que o objetivo é melhorar significativamente a confiabilidade e desempenho da rede. O algoritmo de roteamento proposto tem duas partes: a manutenção das rotas e o encaminhamento de pacotes. A cada nó é atribuído um valor de peso e uma lista de dispositivos relacionados. A importância de proposta está em que os nós não precisam entrar em contato periodicamente com seus vizinhos para atualizar suas informações de roteamento, dado que é proposto um mecanismo de roteamento por inundação controlado com critérios de encaminhamento específicos. O valor de peso atribuído para os nós é baseado no indicador de força de sinal de RF recebido, e se estabelecem seis critérios para um encaminhamento eficiente de informação. Os resultados atingidos da aplicação do modelo proposto foram obtidos a partir da simulação de uma rede com vinte dispositivos em três cenários diferentes e em comparação com três protocolos de roteamento tradicionais. Os autores conseguiram demostrar que as taxas de entrega de pacotes para o modelo proposto atingem até 60% e os protocolos tradicionais são ainda menores do que 10%, assim como também que os atrasos médios em *uplink* e *downlink* foram menores em comparação com os outros,resultados com respostas maiores que os protocolos comparados mas segundo o estabelecido no trabalho o número de dispositivos da rede é baixo (20) o que nem sempre

é verdade para uma rede industrial.

O controle de consumo de energia em RSFI é muito importante para melhorar a eficiência da rede. Em (ZHANG; YAN; MA, 2013b) se propõe um algoritmo de roteamento por grafo com equilíbrio de energia (EBGRA, pelas suas siglas em inglês) com o fim de aumentar os níveis de confiabilidade do protocolo WH. A proposta dos autores se baseia no cálculo de um coeficiente para cada nó da rede como sendo uma métrica essencial para a escolha do dispositivo do seguinte salto na geração de caminhos. Isso determinará, baseado nas métricas, o caminho mais curto e confiável possível. A implementação do algoritmo está dividido em duas partes: na primeira, definem-se três princípios para estabelecer a topologia inicial da rede; o segundo passo é o cálculo da matriz de coeficientes robustos e atualizações de tabelas roteamento. O coeficiente tem em consideração: energia residual, ciclos de comunicação, frequência de comunicação, frequência dos enlaces de comunicação e dissipação de energia. Os resultados obtidos pelos autores a partir de simulações feitas em *Matlab* validam dois fatores importantes para um protocolo de roteamento, que são, o tempo de vida da rede e a robustez do algoritmo. As simulações comprovam que o tempo de vida da rede é consideravelmente maior com o protocolo proposto em comparação com outro protocolo muito usado no WH, além disto, este tempo aumenta conforme aumenta o número de nós na rede. Na análise de robustez, foi demostrado que o algoritmo proposto é mais confiável por definir rotas com nós previamente analisados, que melhora a garantia de sucesso dos caminhos no roteamento. As duas avaliações propostas neste trabalho foram comparadas cada uma com um algoritmo diferente o que não deixa claro o desempenho geral do algoritmo ao ser escolhido um algoritmo diferente para ser comparado com uma e outra métrica.

O trabalho de (WU *et al.*, 2016) propõe um novo algoritmo de roteamento por grafo que tem como objetivo prolongar o tempo de vida da rede mas sem afetar os benefícios de confiabilidade estabelecidos já pelo roteamento por grafo. Antes de apresentar uma solução para o problema de maximização do tempo de rede, os autores estabelecem o modelo da rede e o mesmo é dividido em três modelos: o modelo de roteamento, onde é estabelecido o uso de roteamento por grafo pela sua maior confiabilidade, o modelo de programação de transmissão, e o modelo de consumo de energia, onde se definem equações para o cálculo de consumo de energia dos dispositivos para envio e recepção de pacotes. O problema de maximização do tempo de vida da rota do gráfico, tendo em consideração que

esse tempo corresponde ao intervalo antes do primeiro dispositivo de campo esgotar sua bateria, baseia-se em encontrar rotas que maximizem o tempo de vida da rede dentro do conjunto de caminhos possíveis entre dois dispositivos. Os autores definiram o problema como sendo um problema NP-difícil, onde a versão de decisão foi: *"dado um tempo de vida T para uma rede, esse tempo de vida pode ser satisfeito pela rede?"* e partiram disso para estabelecer um processo de otimização para a escolha de rotas de carga mínima onde a ideia básica é permitir que os dispositivos com maior capacidade de bateria transportem mais tráfego de rede na configuração de roteamento de gráfico. No final, algoritmos com seus pseudocódigos são apresentados para serem avaliados em experimentos de campo e simulações. A implementação do modelo foi obtida de experimentos de campo e simulações, e comparados com o algoritmo de roteamento proposto em (HAN *et al.*, 2011) e o algoritmo de escolha do caminho mais curto de Dijkstra. O resultado obtido mostra que o modelo consegue um tempo de vida da rede 37% maior em comparação com o modelo de Dijkstra e um 85% do tempo de vida útil ideal. Embora o tempo de vida da rede é uma métrica de interesse o algoritmo não está avaliando a confiabilidade das rotas ao focar seu análise só em energia.

Outra proposta baseada em nós com atribuição de pesos é apresentada no algoritmo de (KÜNZEL; CAINELLI; PEREIRA, 2017), que gera os caminhos para um grafo *broadcast* com uma métrica de escolha de nós para a formação das rotas, que é determinada em base de várias métricas da rede. Os autores têm uma proposta para a formação de rotas *broadcast* em redes WH. Na execução do algoritmo, o programa escolhe um dispositivo a cada vez para a construção das rotas de cada dispositivo do grafo *broadcast* a partir de uma função de custos onde são considerados fatores tais como número de saltos ao ponto de acesso, tipo de fonte de energia, nível de sinal recebido de vizinhos e número de vizinhos. A equação proposta para a atribuição de custos nos dispositivos relaciona sete parâmetros, entre eles constantes de pesos os quais foram modificados em vários cenários para avaliar o comportamento do algoritmo frente a essas variações. Testes foram realizados dentro de uma rede simulada e com ajuda de uma plataforma de avaliação de algoritmos de roteamento foram comparados com o algoritmo proposto em (HAN *et al.*, 2011), obtendo como conclusão que o algoritmo proposto tem um melhor ajuste às características de roteamento *broadcast*, sendo também que ao considerar alguns fatores para a formação das rotas ele permite alongar o tempo de vida da rede. As respostas obtida da análise do

algoritmo foram positivas, mas ele só explora a formação do grafo *broadcast*, sendo que a formação dos grafos *uplink* e *downlink* são também muito úteis nas RSFI.

Uma característica importante na decisão de escolhe de uma RSFI é a qualidade de serviço (QoS), e essa característica é considerada no desenvolvimento de (ZHANG *et al.*, 2018). A proposta deste trabalho é estabelecer parâmetros de confiabilidade para a escolha de nós candidatos que serão parte das rotas de encaminhamento de informação. Essa formação de caminhos tem como objetivo equilibrar o consumo de energia. Os autores têm em consideração que as aplicações industriais não são apenas tolerantes ao atraso, mas também críticos em relação ao tempo pelo que no desenvolvimento do algoritmo foi importante na análise de QoS. O algoritmo proposto leva em consideração três aspetos: confiabilidade, oportunismo e consumo de energia. Os autores descrevem como é feito o cálculo dos dois primeiros aspetos e o parâmetro de energia é a primeira consideração em ser feita e leva em consideração uma relação com a distância entre os dispositivos. Os resultados das simulações feitas com o algoritmo proposto mostram que a proposta tem uma melhor taxa de entrega de pacotes em comparação com outros algoritmos e que os dados de eventos de alta pontualidade podem ser transmitidos de forma mais confiável.

O trabalho proposto por (HAN; MA; CHEN, 2019) apresenta um algoritmo que gera dois caminhos de roteamento, um principal e um caminho redundante. A construção da topologia da rede e coleta do consumo de energia de cada nó é baseado no algoritmo hierárquico de busca em largura (BSF pela suas siglas em inglês). O algoritmo BSF vai construindo a rede ao estabelecer um valor a cada novo nó que vai sendo descoberto. Partindo de um nó, ele vai visitando a os vizinhos do nó e os vizinhos dos vizinhos até obter a topologia da rede completa. Depois do processo de construção da rede é analisado o consumo de energia dos nós da rede para garantir um balanço no consumo e considerando que o algoritmo BSF é baseado na técnica de busca do caminho mais curto sua aplicação garante que cada nó tem um caminho ao *gateway* com o mínimo número de salto além da análise de energia. No desenvolvimento do trabalho se descreve o pseudocódigo da geração das duas rotas para o encaminhamento de pacotes entre dois dispositivos, e os resultados obtidos por simulação na plataforma de Matlab em comparação com um algoritmo de escolha do caminho mais curto mostraram mediante gráficos que o tempo de vida da rede foi melhorado.

O planejamento de uma rede é abordado em (CHEN *et al.*, 2019), onde os autores

estudam o custo, confiabilidade e baixa latência para um RSFI em malha baseados em alguns aspectos, entre eles a confiabilidade do processo de roteamento. No desenvolvimento do trabalho, foi definida a importância de estabelecer uma rede em malha confiável, pelo que foi adotada uma rede com uma estrutura de células triangulares, o que garante três rotas diferentes entre cada dispositivo da rede e o gateway. Estabelecer uma rede confiável começou com estabelecer a localização ótima de cada dispositivo da rede, assim, o algoritmo de Dijkstra foi utilizado para definir inicialmente qual seria a localização dos pontos de acesso com o fim de obter caminhos com o menor número de saltos possíveis para cada dispositivo. A partir disso, a escolha das rotas foi analisada em três propostas diferentes, das quais foram estabelecidos os três algoritmos desenvolvidos no trabalho: algoritmo de menor número de saltos, algoritmo de menor número de roteadores e um algoritmo que pretende equilibrar duas propostas anteriores. Os autores fizeram simulações para comparar os três algoritmos propostos baseado em três parâmetros de interesse e os resultados demostraram que para um número pequeno de dispositivos o desempenho dos algoritmos foi muito similar, mas conforme o número de dispositivos na rede aumenta, há parâmetros em que o algoritmo de saltos mínimos supera o algoritmo de menos roteadores e vice-versa, tanto que o desempenho do terceiro algoritmo sempre se manteve entre os outros dois.

Em (EL-FOULY; RAMADAN, 2020) é abordada a importância de comunicações em tempo real dentro de aplicações em RSFI, onde o algoritmo de roteamento proposto busca apresentar um balanço entre eficiência energética e confiabilidade que consiga um bom desempenho dentro de aplicações em tempo real, além de ter maior controle na utilização de nós com carga insuficiente, para melhorar assim o desempenho da rede e reduz os efeitos e congestionamento.

Embora os trabalhos apresentados nesta seção tem aportações importantes para o processo de roteamento em diferentes tipos de redes e para a formação dos diferentes grafos da rede, na sua maioria eles centralizam a análise em uma métrica de interesse dos dispositivos o que poderia enfraquecer com maior facilidade o desempenho dos algoritmos ao ser testados em condições um pouco diferentes às propostas. Na tabela 2 são apresentados 5 trabalhos desenvolvidos nos na área nos últimos anos, esses trabalhos foram escolhidos pela sua proposta de definir valores de custo associados aos dispositivos para a formação dos caminhos de roteamento, eles consideram mais de uma métrica de análise e tiveram

um desempenho superior quando comparados com outros protocolos de roteamento, eles também foram avaliados em RSFI pelo que foram escolhidos para ter uma comparação mais próxima do algoritmo proposto.

Tabela 2 – Trabalhos relacionados

Estado de arte			
Autores	**Ano**	**Tipo**	**Proposta**
(KÜNZEL; CAINELLI; PEREIRA, 2017)	2017	*Broadcast*	- Uso de pesos predefinidos para escolha de rotas - Relaciona 4 parâmetros dos dispositivos
(ZHANG *et al.*, 2018)	2018	*Uplink*	- Classificação do tipo de dados - Definição parâmetro de confiabilidade - Definição taxa de transmissão - Análise de energia.
(HAN; MA; CHEN, 2019)	2019	*Uplink*	- Construção de topologia de rede hierárquica - Definição rota principal e rota redundante - Análise de energia
(CHEN *et al.*, 2019)	2019	*Broadcast, Downlink, Uplink*	- Planejamento da rede - Estruturas para transmissão confiável - Destaque na engenharia prévia
(EL-FOULY; RAMADAN, 2020)	2020	*Downlink, Uplink*	- Comunicação em tempo real - Maximização confiabilidade das rotas - Análise de energia
PROPOSTA	2021	*Downlink, Uplink*	- Algoritmo: rotas mínimas e análise dados de confiabilidade - Rotas redundantes - Ciente do estado energia dos nós

4 MATERIAIS E MÉTODOS

Neste capítulo são apresentados os materiais usados neste trabalho e os métodos para o desenvolvimento do algoritmo de roteamento proposto.

4.1 Materiais

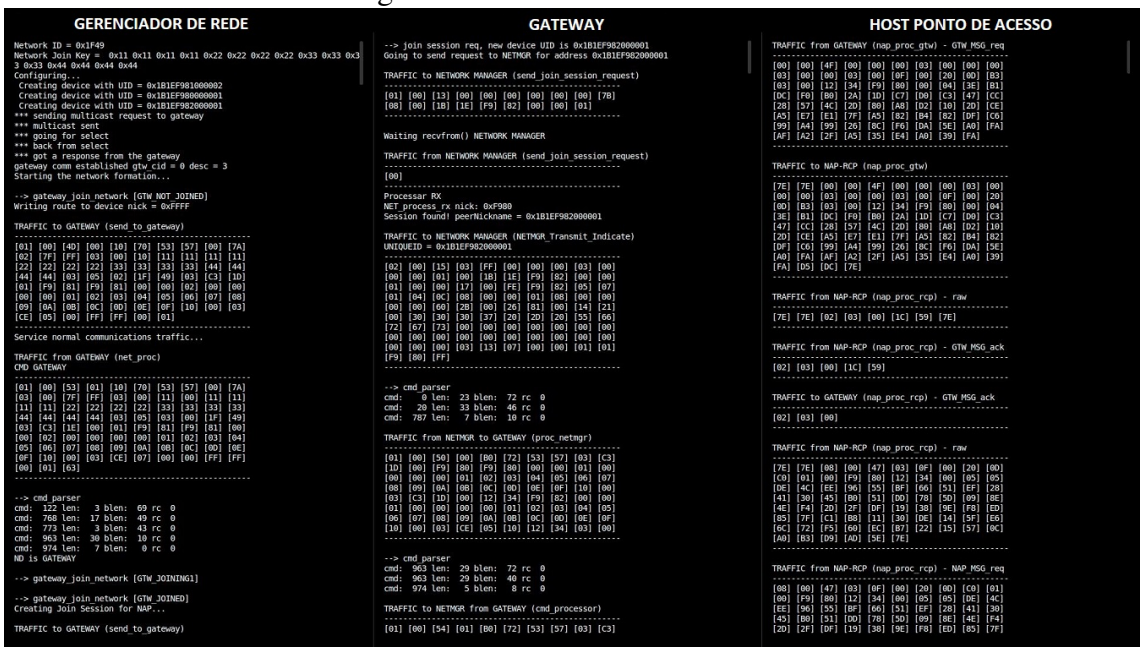
No desenvolvimento deste trabalho foram utilizados alguns materiais fundamentais para o cumprimento dos objetivos. Os algoritmos de roteamento foram implementados sobre o software de um gerenciador de rede que segue o protocolo WH e é apresentado em (CAINELLI *et al.*, 2020). Este gerenciador de rede WH foi desenvolvido com programação C e C++ e é executado no sistema operacional Linux. Todos os códigos correspondem à programação orientada a objetos e foram estruturados de forma modular o que facilita testes de tarefas ou funções individuais.

São três os processos principais que compõem o gerenciador: o gerenciador de rede em si que proporciona todas as funções que permitem a criação e controle da rede, o gateway, que encaminha os pacotes de dados e permite a comunicação dos dispositivos de campo com o gerenciador e aplicações, e finalmente o ponto de acesso. O ponto de acesso é formado pelo dispositivo físico e pelo *host* e sua comunicação é feita por meio de uma porta USB. Na figura 7 são apresentados os processos principais durante a execução no PC.

4.1.1 Gerenciador de rede

O gerenciador foi desenvolvido para a formação e manutenção das rede e seus dispositivos. Ele é baseado no padrão WH e na arquitetura de rede deste trabalho ele se conecta diretamente ao *gateway* que possibilita a conexão com as aplicações e o *host* do ponto de acesso (CAINELLI *et al.*, 2020).

Figura 7 – Processos executáveis



Fonte: Autor

As diferentes classes que compõem o código do gerenciador de rede possibilitam a realização de tarefas como: adição de novos dispositivos, escalonamento, roteamento e processos para diagnóstico da rede. Essas classes foram fundamentais para o desenvolvimento e implementação dos algoritmos de roteamento deste trabalho que passaram a formar parte dos códigos de processo do gerenciamento. Uma breve descrição das classes mais fundamentais é apresentada abaixo:

netdevice.cc: Atributos e métodos que permitem a definição de objetos para cada dispositivos da rede.

session.cc: Atributos e métodos que permitem a criação e armazenamento das sessões criadas para a comunicação entre os dispositivos.

link.cc: Características dos enlaces criados entre os dispositivos.

Graph.cc: Atribuições e métodos que permitem a criação e armazenamento do grafo inicial da rede.

4.1.2 Módulo de simulação

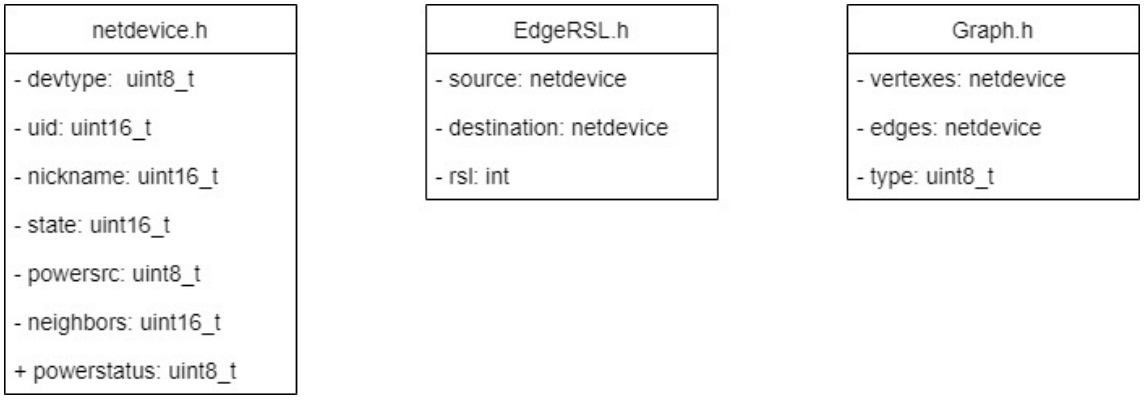
Este modulo foi criado nessa dissertação para obter-se uma rede de dispositivos simulados. Os dispositivos criados neste módulo são objetos da classe *netdevice.cc* que tem atributos como tipo de dispositivo, *unique ID*, *nickname*, localização, vizinhos, sessões, tipo de fonte de energia, vida da bateria entre outros. Todos os dispositivos criados têm

características atribuídas aleatoriamente ou que foram dados seguindo alguma premissa que tinha como objetivo avaliar um estudo de caso. Os dispositivos criados são aleatoriamente dispostos em uma área de tamanho designado como entrada do módulo, e a quantidade de dispositivos criados é também um parâmetro de entrada no módulo e vai depender da análise que se deseja fazer.

Depois da criação dos dispositivos é preciso definir os enlaces entre eles e para isso se faz uso da classe *EdgeRSL.cc* que tem como atributos: *source* e *destination* que correspondem ao dispositivo fonte e destino do enlace respetivamente. Outro atributo importante é o valor de RSL do enlace.

Neste ponto os dispositivos já foram criados e os enlaces foram atribuídos, agora essa informação dos enlaces deve ser levada aos atributos dos dispositivos e isso é feito com um dos métodos da classe *netdevice.cc* que vai permitir a designação dos dispositivos vizinhos de cada nó. Uma vez que as informações dos dispositivos estão completas, elas são usadas como ponto de partida para a execução de um método da classe *Graph.cc*. Nesta classe as informações dos dispositivos são revistas uma por uma para definir os atributos de um novo objeto que corresponde ao grafo inicial da rede. Na figura 8 se apresenta um diagrama das principais classes usadas pelo módulo de simulação.

Figura 8 – Diagrama de classes



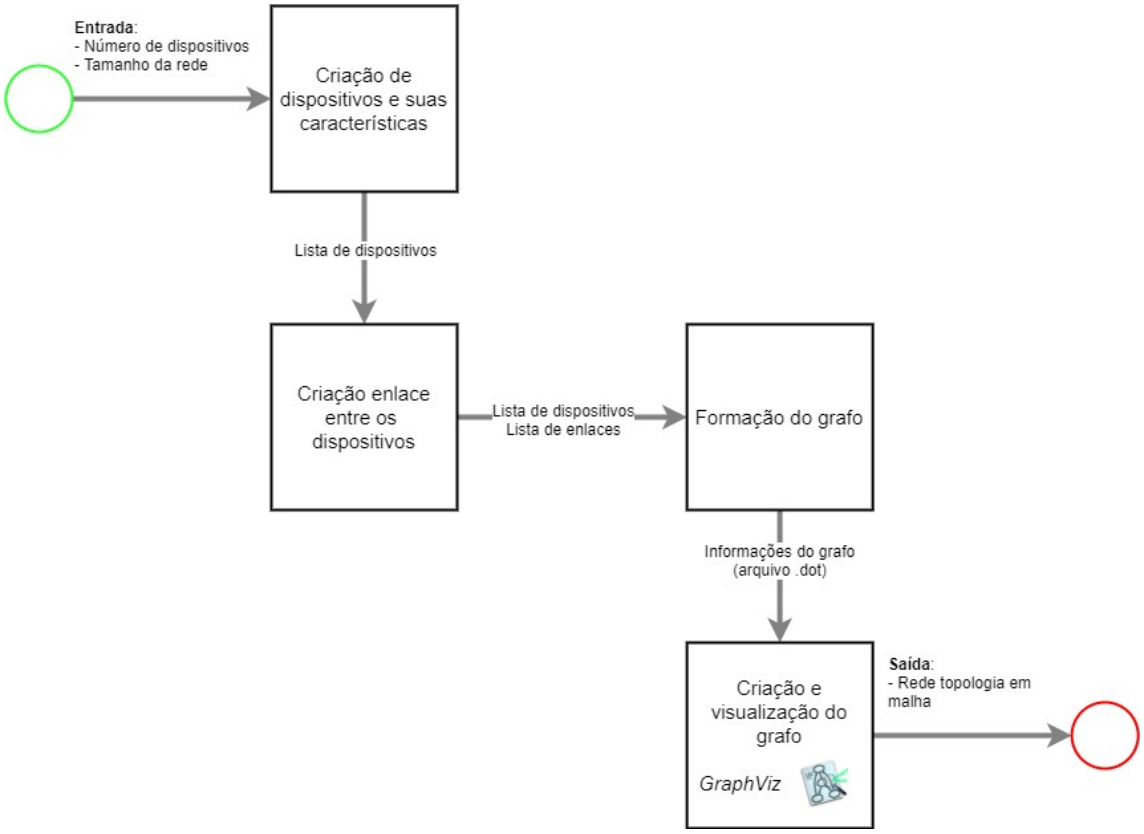
Fonte: Autor

Durante todo o processo do módulo de simulação a informação gerada está sendo armazenada, e a partir de essa informação, um código com extensão *.dot* é escrito e é executado no final do processo, o que permite visualizar por meio de grafos o estado da rede construída.

GraphViz é a extensão de código aberto usada para a visualização de grafos e que tem

importantes aplicações em redes, engenharia de software, banco de dados, aprendizado de máquina e em interfaces visuais para outros domínios técnicos. Todo o processo do módulo é representado na figura 9 como um fluxo. A última etapa deste processo que

Figura 9 – Diagrama de fluxo do módulo



Fonte: Autor

corresponde ao armazenamento das informações em um arquivo e posterior execução do mesmo na extensão *GraphViz* nó só é usado neste módulo, esse processo também foi implementado nos algoritmos apresentados neste trabalho para ajudar visualizar os grafos obtidos de cada um deles.

4.1.3 Gateway

O *gateway* tem como função principal possibilitar a conexão entre o ponto de acesso e o gerenciador assim como a comunicação com aplicações. Neste trabalho ele é um processo executável no computador. Na sua função de direcionamento do tráfego, as mensagens enviadas para os dispositivos de campo são encaminhadas pelo *gateway* ao ponto de acesso físico através do *host* do ponto de acesso, enquanto que as mensagens enviadas para o gerenciador seguem o caminho inverso: primeiro são enviadas ao ponto

de acesso físico, posteriormente ao *host* do ponto de acesso e finalmente ao *gateway*, sendo ao final encaminhadas ao gerenciador.

4.1.4 Ponto de acesso

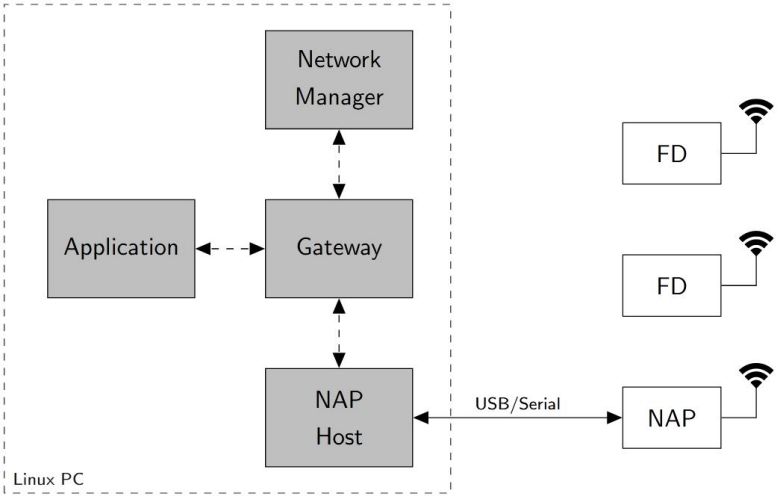
Mencionado anteriormente neste trabalho, o *host* do ponto de acesso desenvolvido é composto por dois elementos, o ponto de acesso físico e o *host* do ponto de acesso, que permite a conexão entre o ponto de acesso físico e o gateway.

Host ponto de acesso: O software do ponto de acesso é uma entidade desenvolvida para rodar em Linux que forma parte dos processos principais do gerenciador. Ao iniciar seu funcionamento, ele procura pelo *gateway* para estabelecer uma conexão TCP. Depois de encontrá-lo, o *host* se conecta ao ponto de acesso físico por uma conexão serial. O *host* ficará a partir daí monitorando continuamente cada canal de conexão.

Ponto acesso físico: Um hardware baseado na plataforma *freescale* MC1322x foi usado para o ponto de acesso físico. Dispositivos de campo já tinham sido desenvolvidos anteriormente em (MÜLLER *et al.*, 2010), implementando as pilhas do protocolo WH. Para a obtenção do ponto de acesso foram feitas algumas mudanças nestes dispositivos, principalmente a criação e modificação da inicialização de tarefas (HAHN; NETTO, 2011).

Na figura 10 é apresentada a estrutura construída com todos os dispositivos disponíveis no laboratório para o correto funcionamento da rede durante testes práticos.

Figura 10 – Estrutura de rede



Fonte: (CAINELLI *et al.*, 2020)

4.1.5 Dispositivos de campo

Os dispositivos de campo considerados são rádios que para a formação da rede WH são conectados aos sensores e atuadores da planta. Eles são compatíveis com WH, já que possuem a pilha do protocolo e estão compostos pelo *freescale* MC13224 que foi apropriado para o desenvolvimento dos dispositivos de campo por ter um microcontrolador ARM 7 de 32bits, um transceptor de rádio IEEE 802.15.4 integrado e vários periféricos e módulos de interesse (MÜLLER *et al.*, 2010).

Figura 11 – Dispositivo de campo

Fonte: (MÜLLER *et al.*, 2010)

4.1.6 *Sniffer*

Wi-Analys é o *sniffer* compatível com o protocolo WH usado como suporte no desenvolvimento do trabalho. Sua função é a captura de pacotes de dados em uma rede dentro de seu rádio de alcance, sendo capaz de decriptar e armazenar as mensagens em seus registros. No software do equipamento, as informações dos pacotes podem ser apresentadas em tabelas com dados tais como: origem e destinatário do pacote, tipo de pacote, data de transmissão, nível de sinal recebido entre outras informações.

4.2 Métodos

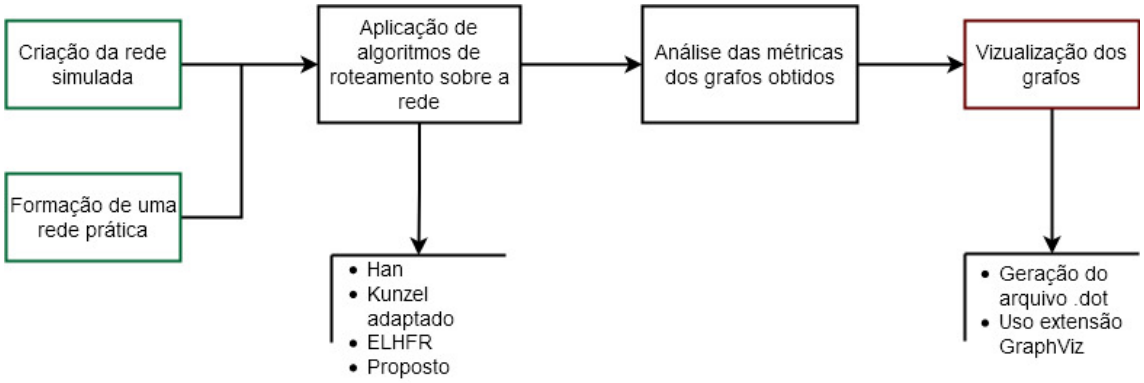
Neste capítulo vai ser apresentado o modelo usado no desenvolvimento deste trabalho: fluxo de execução, algoritmos implementados, métricas de avaliação e como será feita a apresentação dos resultados. A figura 13 é a representação dos blocos principais do

Figura 12 – Sniffer Wi-Analys

Fonte: (CAINELLI *et al.*, 2020)

modelo de implementação proposto.

Figura 13 – Blocos principais modelo proposto



Fonte: Autor

Como descrito na figura, primeiro é feita a criação da rede: no modelo simulado, a rede é formada com a disposição aleatória de nós em uma área estabelecida. O algoritmo implementado possui uma função denominada *generationGraph* encarregada da criação de um número estabelecido de objetos da classe *netdevice.cc*. As características dos objetos são também escolhas aleatórias, tendo só controle de que o número de *gateway* criados sejam um e o número de pontos de acesso igual a um ou dois, dependendo da topologia de rede que se deseja simular. Essa mesma função é a encarregada pela geração de enlaces entre os dispositivos formando objetos da classe *EdgeRSL.cc*, dessa forma tem-se as informações necessárias para a construção do grafo da topologia da rede por parte do gerenciador. Também é feita uma implementação prática onde a formação da rede é feita

com ajuda dos dispositivos de campo descritos na secção de materiais.

Após a formação da topologia da rede são executados sobre ela, três algoritmos além do proposto para a formação dos grafos *uplink* e *downlink*. Neste trabalho foram implementados algoritmos de roteamento que em princípio foram desenvolvidos para RSFI em malha. Esses algoritmos foram escolhidos já que usam uma atribuição de custo nos dispositivos como critério para a escolha, dentre um conjunto de nós candidatos, o nó com melhores características para formar parte das rotas que vão sendo construídas. Os algoritmos implementados são:

- Algoritmo de roteamento de *Han*: Estabelece uma função para calcular o valor de salto de cada dispositivos ao ponto de acesso, as rotas vão sendo construídas pela escolha dos dispositivos segundo seu valor de salto, além na formação do grafo *downlink* três critérios devem se cumprir para garantir o objetivo do algoritmo (HAN *et al.*, 2011).

- Algoritmo de roteamento de *Künzel*: Baseado no algoritmo de roteamento de *Han* sua função de custo estabelece uma relação entre quatro parâmetros dos dispositivos: número de saltos ao ponto de acesso, tipo de fonte de energia, RSL e número de vizinhos (KÜNZEL; CAINELLI; PEREIRA, 2017).

- Algorismo de roteamento *Exhenced least-hop first routing* (ELHFR): Usa o conceito de distância mínima sendo o valor dos saltos uma métrica para a escolha dos dispositivos na formação das rotas (JINDONG; ZHENJUN; YAOPEI, 2009).

Os três algoritmos implementados são avaliados junto com o algoritmo proposto em diferentes cenários de rede, esses algoritmos foram escolhidos por apresentar como base na construção de seus grafos funções de custo que consideram métricas de avaliação do estado dos dispositivos ao igual que a proposta deste trabalho. Sobre os grafos obtidos da execução de cada um dos algoritmos são calculadas métricas que servem para fazer uma comparação entre esses algoritmos e fazer uma avaliação dos resultados obtidos. A avaliação dos algoritmos implementados é feita a partir das seguintes métricas:

- Valor de salto máximo h_{max}: Maior número de saltos entre todos os dispositivos da rede;

- Valor médio de salto h_{med}: Corresponde à média da quantidade de saltos entre todos os dispositivos da rede, desde o nó até o ponto de acesso;

– Dispositivos distantes $h_{d>4}$: Corresponde à porcentagem de dispositivos na rede que estejam a mais de quatro saltos do ponto de acesso;

– Quantidade rotas redundantes n_R: Corresponde ao porcentagem de dispositivos na rede que tem duas arestas de saída pelas quais encaminhar pacotes de dados;

– Quantidade de enlaces Q_{links}: Corresponde à quantidade de enlaces criados pelo algoritmo de formação do grafo.

Também, é possível visualizar a formação dos grafos da rede: paralela à execução dos algoritmos de roteamento e construção das rotas, é formado um arquivo *.dot* com códigos que são interpretados pela extensão *GraphViz* que é uma ferramenta de visualização que toma o arquivo *.dot* e converte-o em uma ilustração gráfica da rede com seus nós representados como círculos e os enlaces por arestas que conectas os nós.

5 IMPLEMENTAÇÃO

Neste capítulo são apresentados os algoritmos implementados neste trabalho detalhadamente, assim como o desenvolvimento do algoritmo proposto destacando suas características mais relevantes. A implementação desses algoritmos foi feita no gerenciador de rede descrito anteriormente (CAINELLI *et al.*, 2020), onde cada algoritmo foi implementado individualmente fazendo uso de algumas das classes já disponíveis, visando sua aplicação posterior numa rede real. Desta forma, o gerenciador de rede é adaptado para funcionar como um simulador, e posteriormente, tem sua funcionalidade original reestabelecida.

5.1 Algoritmos de roteamento implementados

Os algoritmos implementados têm como objetivo fazer uma avaliação do processo de roteamento em uma RSFI em malha em diferentes cenários. Nesta secção são apresentados alguns algoritmos de roteamento com propostas relevantes na construção dos grafos com rotas redundantes para tráfego de dados.

5.1.1 Algoritmo de Han

A proposta do algoritmo de roteamento de Han pretende garantir confiabilidade e ótimo desempenho para aplicações que precisam de comunicação em tempo real. O algoritmo foi desenvolvido para RSFI em malha e estabelece como devem ser construídos os grafos que definem as rotas de comunicação dos dispositivos. O algoritmo cria os grafos *broadcast*, *uplink* e *downlink* garantindo redundância de rotas e alta confiabilidade. A implementação dos algoritmos não começou de zero, havia uma base implementada sobre os algoritmos de *Han* no gerenciador apresentado por (CAINELLI *et al.*, 2020), e, foram feitas mudanças e revisões no código para conseguir executá-lo com sucesso.

Grafo Uplink: Sua construção pretende minimizar o número de saltos de cada nó até o *gateway* e garantir que todos os dispositivos tenham dois caminhos pelos quais enviar pacote de dados. O algoritmo de Han define o número de saltos de um dispositivo como uma média de saltos dada pela equação (1) e a construção do grafo é feita adicionando um por um os dispositivos escolhendo o dispositivo com menor número de saltos do conjunto em análise.

$$h_i = \frac{\sum_{k \epsilon P_i} h_k}{P_i} + 1 \tag{1}$$

Onde:

h_i : Valor do salto do dispositivo *i*;

P_i: Vizinhos do dispositivo *i* no grafo *uplink*.

O Algoritmo 1 apresenta a estrutura da proposta de Han para a construção do grafo *uplink*, A partir do grafo inicial da rede $G(V, E)$ é construído o grafo invertido $G^R(V, E^R)$ correspondente ao grafo inicial com arestas no sentido oposto, nesta primeira etapa é também definido as condições iniciais do vetor de vértices do grafo *uplink*que está sendo formado (V_U) contendo o *gateway* e pontos de acesso da rede e o vetor de arestas do grafo *uplink* (E_U) contendo todos os enlaces disponíveis entre esses dispositivos. O algoritmo vai criando as rotas até que os dispositivos em V_U sejam todos os dispositivos da rede (V) para isso todos os dispositivos que ainda não pertencem ao grafo *uplink* ($V - V_U$) vão ir sendo divididos a cada novo análise em dois subgrupos: S' que corresponde aos dispositivos que têm ao menos duas arestas vindo de dispositivos em V_U e o segundo S" aqueles que têm uma aresta vindo de V_U.

Para os todos os dispositivos v em S', se armazena em ordem crescente em $e_{u,v}$ as arestas vindas de V_U segundo seu valor de salto h_u calculado da equação **??** após são escolhidas as duas arestas com o menor valor de salto e calculado o novo valor de salto do dispositivo v em análise. No final será adicionado ao vetor V_U o dispositivo v com o menor valor de salto h_v e suas arestas $e_{u_1,v}$, $e_{u_2,v}$ adicionadas ao vetor de arestas E_u (Linhas 2-11). Para os dispositivos v em S" é calculado seu valor de salto h_v e procurado o número n_v de arestas desde v até os dispositivos que ainda não pertence ao grafo *uplink* ($V - V_U$), deste subgrupo o dispositivo que vai formar parte de V_U é o dispositivo com o valor máximo de n_v (Linhas 12-23). Todo este processo vai ser repetido até V_U conter todos os dispositivos da rede para finalmente obter o grafo *uplink* $G_U(V_U, E_U)$ completo.

Algorithm 1 Construção Grafo *uplink* Han $G_U(V_U, E_U)$

In: $G(V, E)$ **Out:** $G_U(V_U, E_U)$

Construir $G^R(V, E^R)$ o grafo invertido de $G(V, E)$

Inicialmente $V_U = g \cup V_{AP}$ e E_U contem todos os enlaces de g a V_{AP}

while $V_U \neq V$ **do**
  Encontrar $S' \subseteq V - V_U : \forall v \epsilon S'$, v tem ao menos duas arestas que vêm V_U
  if $S' \neq \emptyset$ **then**
    for $\forall v \, \epsilon \, S'$ **do**
      Ordenar as arestas $e_{u,v}$ segundo h_u
      Escolha as primeiras duas arestas $e_{u_1,v}$ e $e_{u_2,v}$
      $h_v = \frac{h_{u1}+h_u 2}{2} + 1$
    end for
    Escolha o nó v com menor h_v
    Adicione v a V_U e $e_{u_1,v}$ e $e_{u_2,v}$ a E_U
  else
    Encontrar S" $\subseteq V - V_U : \forall v \epsilon S$", v tem uma aresta de V_U
    if S" $\neq \emptyset$ **then**
      for $\forall v \, \epsilon \, S$" **do**
        $h_v = h_u + 1$
        Calcular n_v como número de arestas de saída para $V - V_U$
      end for
      Escolha o nó v com máximo n_v, quebre o empate usando h_v
      Adicione v a V_U e $e_{u,v}$ a E_U
    else
      Construção do grafo falhou
    end if
  end if
end while
A construção do grafo $G_U(V_U, E_U)$ está completo
Construa o grafo final $G_U(V_U, E_U)$ revertendo as arestas E_U

Grafo Downlink: Neste algoritmo, a escolha dos dispositivos que vão sendo adicionados ao conjunto vértices não só vai depender do número e saltos mas também do cumprimento de três restrições definidas como *C1*, *C2* e *C3*. *C1* verifica se o nó em análise tem ao menos dois nós pais (u_1, u_2) do conjunto de nós que já formam parte do grafo *downlink*; *C2* valida que entre os nós pais u_1 e u_2 encontrados em *C1* existe uma conexão; finalmente *C3* verifica se $u_1(u_2)$ tem ao menos um pai no grafo *downlink* de $u_2(u_1)$.

Os Algoritmos 2 e 3 apresentam a estrutura da proposta de *Han* para a construção do grafo *downlink*. No algoritmo 2 se apresenta a construção do grafo *downlink* para cada nó da rede $G_D^i(V_D^i, E_D^i)$, inicialmente é definido o conjunto de nós S que vai conter os dispositivos com grafo *downlink* já construído. O *gateway* e os pontos de acesso são os primeiros dispositivos em ter seus grafos definidos pelo que o conjunto $S = g \cup V_{AP}$. O algoritmo será controlado por um laço e será executado até S conter todos os dispositivos da rede. Semelhante à construção do grafo *uplink* nesta construção do grafo *downlink* os dispositivos que ainda não pertencem ao conjunto S vão ir sendo divididos a cada novo análise em dois subgrupos: S' que corresponde aos dispositivos que têm ao menos duas arestas vindo de dispositivos em S e o segundo S" aqueles que têm uma aresta vindo de S. Também é construído um conjunto de S_r que armazena os nós da rede definidos como confiáveis por cumprir as restrições estabelecidas como *C1*, *C2* e *C3*.

Para todos os dispositivos em S' se procura por os pares de arestas $(e_{u1,v}, e_{u2,v})$ que vêm de S, para todos os pares de arestas encontrados o nó v em análise é adicionado a S_r se o par de arestas cumprem com *C1*, *C2* e *C3* e o valor de salto para v é calculado da escolha do par de arestas com menor valor de salto $h_{u1,u2}$(Linhas 2-14). No final da análise dos dispositivos em S', se S_r não está vazio é adicionado ao conjunto S o dispositivo com menor valor de salto h_v do conjunto S_r, caso contrário o novo dispositivo do conjunto S sera escolhido do conjunto S'. A formação do grafo *downlink* G_D^v é feita a partir de ConstructDG(G, G_{u1}, G_{u2}, v) com algoritmo 3 (Linhas 2-20).

Do conjunto S" se calcula para todos os dispositivos o valor do salto h_v e o número de arestas de saída n_v que cada dispositivo tem até os nós que não tem subgrafo *downlink* formado $(V - S)$, o novo dispositivo adicionado ao conjunto S será o que tem maior valor de n_v. Para os dispositivos neste conjunto (S") a chamada da função para a construção do grafo *downlink* do respetivo dispositivo será ConstructDG($G, G_{u1}, NULL, v$) (Linha 22-31).

Algorithm 2 Construção Grafo *Downlink* Han $G_D(V_D, E_D)$

In: $G(V, E)$ **Out:** $G_D(V_D, E_D)$

Definir S como conjunto de nós com grafo *downlink* construído.
Inicialmente $S = g \cup V_{AP}$ e $G_D^g = \{\{g\}, \emptyset\}$
Grafo *downlink* para cada AP i como $G_D^i = \{\{g \cup i\}, \{e_{g,i}\}\}$

1: **while** $S \neq V$ **do**
2: Encontre $S' \subseteq V - S : \forall v \epsilon S'$, v tem ao menos duas arestas que vêm de S
3: Definir S_r como o conjunto de nós confiáveis em S, inicialmente $S_r = \emptyset$
4: **if** $S' \neq \emptyset$ **then**
5: **for** $\forall v \in S'$ **do**
6: **for** $\forall$ par de arestas $(e_{u1,v}, e_{u2,v})$ de S **do**
7: **if** $C1 \wedge C2 \wedge C3$ **then**
8: $S_r = S_r \cup \{v\}$
9: **end if**
10: $h_{u1,u2} = \frac{h_{u1}+h_{u2}}{2}$
11: **end for**
12: Escolha o par de arestas $(e_{u1,v}, e_{u2,v})$ com menor $h_{u1,u2}$
13: $h_v = h_{u1,u2} + 1$
14: **end for**
15: **if** $S_r \neq \emptyset$ **then**
16: Adicionar o nó v em S_r com menor h_v ao conjunto S
17: **else**
18: Adicionar o nó v em S' com menor h_v ao conjunto S
19: **end if**
20: Construção do grafo G_D^v com ConstructDG(G, G_{u1}, G_{u2}, v)
21: **else**
22: Encontre $S" \subseteq V - S : \forall v \epsilon S"$, v tem uma aresta $e_{u,v}$ que vêm de S
23: **if** $S" \neq \emptyset$ **then**
24: **for** $\forall v \in S"$ **do**
25: $h_v = h_u + 1$
26: Calcule n_v como o número de arestas de saída para $V - S$
27: **end for**
28: Adicione v com o máximo valor de n_v a S, quebre o empate usando h_v
29: Construção do grafo G_D^v com ConstructDG($G, G_{u1}, NULL, v$)
30: **else**
31: Construção do grafo falhou
32: **end if**
33: **end if**
34: **end while**
35: Retorna o conjunto de grafos *downlink* $G_D(V_D, E_D)$

No Algoritmo 3 se dá a formação completa de cada subgrafo a partir das considerações já feitas sobre os melhores nós candidatos a pai para a formação do subgrafo específico de um dispositivo. Descreve como construir o grafo *downlink* de um dispositivo $G_D^i(V_D, E_D)$ com base nos grafos dos nós pais $G_{u1}(V_{u1}, E_{u1}), G_{u2}(V_{u2}, E_{u2})$. Primeiro é feita a mescla em G_D^i dos vértices e arestas de G_{u1} e G_{u2} e definido S como o conjunto de nós já explorados. Um laço controla a execução do algoritmo até o conjunto de nós em S ser igual ao conjunto de nós em V_v (Linha 1-9). Do conjunto S' e S" que é o conjunto de nós ainda não explorados ($V_v - S$) que tem duas ou mais arestas ou uma aresta a S respetivamente é adicionado em S o nó com menor valor de salto (Linha 10-27).

5.1.2 Algoritmo de Künzel

O algoritmo em (KÜNZEL; CAINELLI; PEREIRA, 2017) é uma proposta para a formação do grafo broadcast, onde se estabelece um valor de custo para os dispositivos. Inicialmente se tem o grafo da topologia da rede completa e os nós que formarão parte do grafo *broadcast* vão sendo escolhidos e adicionados segundo o valor de custo atribuído, ou seja, de um conjunto de nós candidatos é escolhido o nó com menor valor de custo. A construção deste algoritmo é baseado no algoritmo para formação do grafo *broadcast* de *Han* (HAN *et al.*, 2011).

Para a construção do grafo *uplink* e *downlink* se fez uso da função de custo apresentada em *Künzel* para a construção dos grafos *uplink* e *downlink* adaptando o algoritmo de *Han*. As funções custo do algoritmo de *Künzel* são apresentadas a seguir (2-3) e o uso de cada uma vai depender das características do conjunto de dispositivos candidatos:

$$c_i = \frac{w_h \frac{h_i}{h_{max}} + w_p p_i + w_s \frac{s_i}{-85}}{w_h + w_p + w_s} \tag{2}$$

Onde:

c_i: custo do dispositivo i

h_i: número de saltos de i

h_{max}: Máximo número de saltos dos candidatos

p_i: restrição de energia para i

s_i: média do RSL de dois dispositivos de i em V

w_h: peso do número de saltos

w_p: peso da fonte de energia

w_s: peso do RSL médio

Algorithm 3 Construção Grafo *Downlink* Han $G_D^i(V_D, E_D)$

In: $(G(V,E), G_{u1}(V_{u1}, E_{u1}), G_{u2}(V_{u2}, E_{u2}), v)$ **Out:** $G_D^i(V_D, E_D)$

Inicialmente S contem os nós explorados em $G_v(V_v, E_v) : S = \{g, v, u_1, u_2\}$

G_v é construído pela integração de G_{u1}, G_{u2}, v e as arestas entre v, u_1, u_2

$G_v(V_v, E_v) = G_v(V_{u1} \cup V_{u2} \cup \{v\}, E_{u1} \cup E_{u2} \cup \{e_{u1,y}, e_{u2,y}, e_{u1,u2}, e_{u2,u1}\})$

while $S \neq V_v$ **do**
  if V_{AP} tem duas arestas saindo de S em G **then**
    $S = S \cup V_{AP}$
    break
  end if
  for $\forall$ nós $i \epsilon V_v - S$ **do**
    Ordene as arestas de saída de i para S em ordem descendente segundo h_i
  end for
  Encontre $S' \subseteq V_v - S : \forall v \epsilon S'$, v tem ao menos duas arestas a S
  if $S' \neq \emptyset$ **then**
    Adicione nó i com o menor h_i a S
    Adiciones as primeiras duas arestas de i para S a G_v se ainda não existem
    Remova todas as outras arestas de E_v
  else
    Encontre S" $\subseteq V_v - S : \forall v \epsilon S$", v tem uma aresta a S
    if S" $\neq \emptyset$ **then**
      Adiciones nó i com o menor h_i a S
      Adicione a aresta de i a S em G_v, se ainda não existe
    else
      Falha na construção do grafo
    end if
  end if
end while
for $\forall$ nós $i \epsilon V_v - S$ **do**
  Remova i do conjunto V_v e as correspondentes arestas de E_v
end for
Retorna o grafo *downlink* $G_D^i(V_D, E_D)$

A equação (3) leva em consideração o caso em que não existam dois dispositivos de i para o conjunto de nós que já formam parte do grafo *uplink* ou *downlink*.

$$c_i = \frac{wn(1 + \frac{n_i}{n_{ma}}) + w_p p_i}{w_n + w_p} \tag{3}$$

Onde:

n_i: número de arestas de saída de i para $V - V_B$

n_{max}: máximo número de n_i dos candidatos

w_n: peso das arestas de saída.

Os outros termos correspondem à mesma descrição dada na equação (2). No Algoritmo 4 é apresentado o pseudocódigo para a formação do grafo *uplink* modificado, o código neste algoritmo é semelhante ao código de construção do grafo *uplink* de Han a diferença é que para a escolha do nó do conjunto S' que vai formar parte do conjunto V_U é usada a equação (2) de *Künzel* e para a escolha do nó no conjunto S" é usada a equação (2).

No Algoritmo 5 é apresentado o código para a formação do grafo *downlink* considerando os valores de custo atribuídos pela função de Künzel. As mudanças sobre o código de construção de cada subgrafo *downlink* não são apresentadas, visto que são similares ao Algoritmo 3 com a diferença de que nas linhas 9 e 15 a adição do nó i é feita em base ao valor de custo do dispositivo e não ao número de saltos.

5.1.3 Algoritmo ELHFR

ELHFR (*Enhanced least-hop first routing*) é um algoritmo de roteamento por grafo proposto em 2009 para RSFI em malha. O algoritmo trabalha desde a descoberta dos dispositivos na rede usando o conceito de busca em largura tomando ao *gateway* como o dispositivo inicial e gerando a árvore de abrangência do gráfico de topologia da rede. A formação dos grafos de interesse é feita por dispositivo, cada dispositivo da rede procura por dispositivos vizinhos com os quais consegue se comunicar e que tenham um valor de salto menor ao valor do salto ao dispositivo em análise. Essa busca continua até conseguir estabelecer um caminho até o *gateway*. Todas as rotas construídas para um dispositivo desde o até o *gateway* se caracterizam por ter a mesma quantidade de saltos, e isso é uma vantagem, dado que o uso de rotas redundantes pode estar garantindo o mesmo tempo para envio e recepção de pacotes.

O algoritmo ELHFR estabelece três propriedades para os subgrafos construídos:

Algorithm 4 Construção Grafo *uplink* Künzel $G_U(V_U, E_U)$

In: $G(V, E)$ **Out:** $G_U(V_U, E_U)$

Construir $G^R(V, E^R)$ o grafo com arestas invertidas de $G(V, E)$
Inicialmente $V_U = g \cup V_{AP}$ e E_U contem todos os enlaces de g a V_{AP}
while $V_U \neq V$ **do**
  Encontrar $S' \subseteq V - V_U : \forall v \epsilon S'$, v tem ao menos duas arestas que vêm de V_U
  if $S' \neq \emptyset$ **then**
    for $\forall v \, \epsilon \, S'$ **do**
      Armazena todas as arestas de V_U a v como $e_{u_i,v}$
      Armazenar todos os nós fonte das u_i arestas $e_{u_i,v}$
      Ordene todos os nós u_i segundo a equação (2)
      Escolha as arestas $e_{u_1,v}$ e $e_{u_2,v}$ com menor custo segundo (2)
      $h_v = \frac{h_{u1}+h_u 2}{2} + 1$
    end for
    Escolha o nó v com menor custo de acordo com a Eq. 2
    Adicione v a V_U e $e_{u_1,v}$ e $e_{u_2,v}$ a E_U
  else
    Encontrar S" $\subseteq V - V_U : \forall v \epsilon S$", v tem uma aresta de V_U
    if S" $\neq \emptyset$ **then**
      for $\forall v \, \epsilon \, S$" **do**
        $h_v = h_u + 1$
        Calcular n_v como número de arestas de saída para $V - V_U$
      end for
      Escolha o nó v com menor custo, segundo (3)
      Adicione v a V_U e $e_{u,v}$ a E_U
    else
      Construção do grafo falhou
    end if
  end if
end while
A construção do grafo $G_U(V_U, E_U)$ está completo
Construa o grafo final $G_U(V_U, E_U)$ revertendo as arestas E_U

Algorithm 5 Construção Grafo *Downlink* Künzel $G_D(V_D, E_D)$

In: $G(V, E)$ **Out:** $G_D(V_D, E_D)$

Definir S como conjunto de nós com grafo *downlink* construído.

Inicialmente $S = g \cup V_{AP}$ e $G_D^g = \{\{g\}, \emptyset\}$

Inicialmente é construído o grafo *downlink* para cada AP i como $G_D^i = \{\{g \cup i\}, \{e_{g,i}\}\}$

while $S \neq V$ **do**
  Encontre $S' \subseteq V - S : \forall v \epsilon S'$, v tem ao menos duas arestas que vêm de S
  Definir S_r como o conjunto de nós confiáveis em S, inicialmente $S_r = \emptyset$
  if $S' \neq \emptyset$ **then**
    for $\forall v \epsilon S'$ **do**
      for $\forall$ par de arestas $(e_{u1,y}, e_{u2,y})$ de S **do**
        if $C1 \wedge C2 \wedge C3$ **then**
          $S_r = S_r \cup \{v\}$
        end if
        $h_{u1,u2} = \frac{h_{u1}+h_{u2}}{2}$
      end for

      Escolha o par de arestas $(e_{u1,y}, e_{u2,y})$ com menor custo segundo (2)
      $h_v = h_{u1,u2} + 1$
    end for
    if $S_r \neq \emptyset$ **then**
      Adicionar o nó v com menor custo (Eq. 2) ao conjunto S
    else
      Adicionar o nó v em S' com menor custo (Eq. 2) ao conjunto S
    end if
    Construção do grafo G_D^v com ConstructDG(G, G_{u1}, G_{u2}, v)
  else
    Encontre $S" \subseteq V - S : \forall v \epsilon S"$, v tem uma aresta $e_{u,y}$ que vêm de S
    if $S" \neq \emptyset$ **then**
      for $\forall v \epsilon S"$ **do**
        $h_v = h_u + 1$
        Calcule n_v como o número de arestas de saída para $V - S$
      end for
      Adicione v com o menor valor de custo segundo (3)
      Construção do grafo G_D^v com ConstructDG($G, G_{u1}, NULL, v$)
    else
      Construção do grafo falhou
    end if
  end if
end while
Retorna o conjunto de grafos *downlink* $G_D(V_D, E_D)$

1. Em um subgrafo gerado por ELHFR todos os caminhos para um dispositivo até o *gateway* tem igual número de saltos.

2. Qualquer caminho em um subgrafo do nó até o *gateway* tem o mínimo número de saltos.

3. Qualquer nó pode encontrar seu caminho mais curto para o *gateway* a partir dos subgrafos.

Nos Algoritmos 6 e 7 é apresentada a construção do grafo *uplink* baseada na construção de rotas com mínimo de número de saltos possíveis.22

Algorithm 6 Construção Grafo *uplink* ELHFR: Parte I
In: $G(V, E)$ **Out:** $G_H(V_H, E_H)$

Definir S como vetor de dispositivos já adicionados em G_H
$S = g$
while $V - S \neq \emptyset$ **do**
    for $\forall v \, \epsilon \, V - S$ **do**
        Encontre $S' \subseteq V - S : \forall v \epsilon S'$, v tem arestas até $V - S$
        for $\forall u \, \epsilon \, S'$ **do**
            if u não pertence a S **then**
                Adicione u ao conjunto S
                Adicione a aresta $e_{u,v}$ no conjunto E_H
                $h_u = h_v + 1$
            end if
        end for
    end for
end while
$V_H = S$
Retorno do grafo $G_H(V_H, E_H)$

5.1.4 Algoritmo de roteamento proposto

O entendimento do processo de roteamento em RSFI tem sido o ponto de partida para conseguir alcançar o objetivo de se estabelecer um algoritmo de roteamento que proporcione a formação de grafos com rotas confiáveis, tendo em consideração parâmetros e características dos dispositivos na rede. A proposta deste trabalho estabelece a construção dos grafos *downlink* e *uplink* com redundância de rotas, rotas onde os dispositivos

Algorithm 7 Construção Grafo *uplink* ELHFR: Parte II

In: $G_H(V_H, E_H)$; **Out:** $G_U^i(V_U^i, E_U^i)$

Definir $V = V_H$
1: Definir S como conjunto de dispositivos com subgrafo construído
2: **while** $V - S \neq \emptyset$ **do**
3: **for** $\forall v \, \epsilon \, V - S$ **do**
4: Constrói o subgrafo $G_U^i(V_U^i, E_U^i)$
5: Encontre $S' \subseteq V_H : \forall u \epsilon S'$, $h_u = h_v - 1$
6: **for** $\forall u \, \epsilon \, S'$ **do**
7: **if** existe $e_{v,u}$ **then**
8: Adicione $e_{v,u}$ ao conjunto E_U^i
9: Adicione u ao conjunto V_U^i
10: **end if**
11: **end for**
12: Adicione v ao conjunto V_U^i
13: **end for**
14: **end while**
15: Retorno dos subgrafos $G_U^i(V_U^i, E_U^i)$

de um conjunto de candidatos vão sendo escolhidos com as melhores características de confiabilidade, melhor condição de comunicação e ciência do seu estado de energia.

O algoritmo proposto é baseado no algoritmo de ELHFR e dividido em duas etapas bem definidas com objetivos chave para obtenção dos resultados esperados. A primeira etapa do algoritmo é baseada na descrição do problema 1 estabelecido por *Edsger Dijkstra* em (DIJKSTRA, 1959) que estabelece a construção da árvore de comprimento mínimo entre todos os nós da rede. A segunda etapa define a formação das rotas para os grafos *downlink* e *uplink* tendo em consideração a importância na formação de caminhos redundante. Em cada etapa do algoritmo, a escolha dos dispositivos para formação das rotas é baseada em uma função de custo que considera alguns critérios previamente analisados com o objetivo de obter rotas com maior confiabilidade.

Construção da árvore de comprimento mínimo:

A construção da árvore de comprimento mínimo no algoritmo de *Dijkstra* é feita pela atribuição de custos nas arestas de um grafo onde geralmente é considerado o número de saltos entre um dispositivo origem e os demais dispositivos da rede, garantido que a distância entre o *gateway* e qualquer dispositivo da rede seja mínima segundo o valor

de custo usado. Na implementação usada neste trabalho, se estabeleceu uma função de custo para as arestas onde são considerados critérios para garantir o melhor estado entre: distância entre dispositivos, nível de sinal recebido entre dispositivos e confiabilidade do caminho. Os três critérios são relacionados por uma equação para obter a melhor relação entre eles e definir as melhores rotas na construção da árvore de comprimento mínimo.

A equação (4) nesta etapa tem em consideração: o RSL dos vizinhos reportados ao gerenciador por cada um dos dispositivos, a distância em metros entre os dispositivos dispostos na rede e um valor de estatística de confiabilidade. Para a definição da equação foram tomadas em consideração os valores do padrão WH que define um valor mínimo de -85dBm para o valor de RSL e uma faixa de comunicação com alcance de até 100m.

$$d_{i\to j} = \frac{\frac{D_{i\to j}}{100}}{PR - \frac{RSL_{i\to j}-60}{85}} \tag{4}$$

Onde:

$d_{i\to j}$: costo da aresta do dispositivo i para j

$D_{i\to j}$: distância entre o dispositivo i e j

PR : confiabilidade do caminho obtido a partir de métricas do relatório de estado do dispositivo

$RSL_{i\to j:}$ RSL de j em i

O pseudocódigo da implementação desta etapa é apresentado no Algoritmo 8 onde $G(V, E)$ corresponde ao grafo original formado a partir das informações que os dispositivos compartilham com o gerenciador, $G_D(V_D, E_D)$ corresponde à árvore de comprimento total mínimo resultante, o conjunto $SetE_{II}$ são todos as arestas que estão sendo consideradas e que têm probabilidade de formar parte do conjunto E_D, o conjunto $SetV_{II}$ os dispositivos que ainda não foram analisados e não formam parte do conjunto V_D. No início do programa, o *gateway* representado como g é adicionado à lista de vértices da árvore V_D e as arestas que vão até o *gateway* desde os pontos de acesso são adicionadas no conjunto $SetE_{II}$. Os dispositivos de campo são representados como V_{fd} e os pontos de acesso como V_{ap}. Os dispositivos têm um parâmetro que armazena o custo da sua rota até o *gateway*, e, para o início do programa, este valor é infinito e é atualizado conforme os dispositivos vão sendo adicionados ao grafo final. Os passos que serão descritos na continuação são seguidos até que os conjuntos $SetE_{II}$ e $SetV_{II}$ estejam vazios.

1. O conjunto de arestas $SetE_{II}$ é ordenado de forma crescente segundo equação (4)

e a aresta com o menor valor é removida do conjunto e adicionada ao conjunto E_D. Do resultado desta operação o dispositivo destino da aresta é também adicionado ao conjunto V_D.

2. Novamente são considerados todas as arestas que tem como nó fonte o dispositivo recentemente agregado ao conjunto V_D, e como dispositivo destino aos nós que ainda não foram considerados o que significa que formam parte do conjunto $SetV_{II}$, esse conjunto de arestas formará parte de $SetE_{III}$. É calculado valor do custo segundo equação (4) e adicionado ao valor do dispositivo fonte: se a aresta entre o dispositivo fonte e destino considerados no momento existe no conjunto $SetE_{II}$ e se o valor é maior que o correspondente no conjunto $SetE_{II}$, a aresta não é considerada, mas se é menor, a aresta substitui a correspondente no conjunto $SetE_{II}$. Se a aresta não existe ela é adicionada e se volta à execução do passo 1.

Após a formação completa da árvore é feita a atribuição do valor de salto para cada um dos dispositivos. O *gateway* e os pontos de acesso são considerados com um valor se salto igual a zero, os dispositivos que segundo a árvore construída estão conectados ao ponto de acesso tem um valor de salto igual a 1 e continuando esse esquema vão sendo definidos os valores de salto de todos os dispositivos da rede.

Construção de subgrafos com redundância de rotas confiáveis:

Na primeira etapa foi garantida a construção da árvore com rotas únicas até cada dispositivo, onde os diapositivos que conformam os caminhos tem as melhores condições dos critérios considerados. Mas, numa RSFI, a redundância de rotas é de grande importância e pode chegar a ser um requisito de escolha para o algoritmo de roteamento a ser usado. Nesta segunda etapa será feita a construção dos subgrafos para cada nó com a maior quantidade possível de dispositivos com rotas redundantes. A construção feita na etapa um vai garantir que todas as rotas redundantes definidas nesta etapa tenham a mesma quantidade de saltos, o que pode ajudar a ter mais controle do tempo no envio de pacotes de dados, indiferentemente da rota escolhida.

Esta etapa é baseada no algoritmo de criação de rotas ELHFR apresentado em (ZHANG; YAN; MA, 2013a). Ele parte da construção do grafo obtida na primeira etapa junto com a informação dos vizinhos dos dispositivos da rede. Nesta etapa é usada uma nova função de custo atribuída a cada dispositivo e tem em consideração parâmetros de energia e estatísticas de confiabilidade dos dispositivos obtidas pelo gerenciador de rede. O objetivo

Algorithm 8 Construção árvore de comprimento total mínimo

In: $G(V, E)$

Out: $G_D(V_D, E_D)$

Inicialmente $V_D = g$ e $SetE_{II} =$ aretas que vão de g a V_{ap}

while (!$SetV_{II}.empty()$) **do**
    Ordenar o vetor $SetE_{II}$ segundo equação (4)
    Adiciona e_{min} o primeiro elemento de $SetE_{II}$ em E_D
    Adiciona v o novo nó em V_D a partir de e_{min}
    Armazena em $SetE_{III}$ arestas que vão de v a $SetV_{II}$.
    for (todos e_i em $SetE_{III}$) **do**
        for (todos e_j em $SetE_{II}$) **do**
            if (e_i != todos em $SetE_{II}$) **then**
                Adiciona e_i em $SetE_{II}$
            else if (e_i==e_j) **then**
                Compara valores da métrica segundo equação (4)
                Se e_i<e_j então e_j = e_i
            end if
        end for
    end for
end while
$g.setnivel(0)$
$netdev = g$
while (!$netdev.empty()$) **do**
    for (todos v em $netdev$) **do**
        Armazena em e_{out} aretas com fonte v
        for todas e em e_{out} **do**
            $e.setnivel() = 1 + v.getnivel()$
            Armazena em $netdev_{new}$ o destino de e
        end for
    end for
    $netdev = netdev_{new}$
end while
Retorna o G_D completo

é conseguir estabelecer rotas com dispositivos com as melhores condições de energia e ótimos históricos de confiabilidade. A equação de custo usada nesta etapa define uma relação entre tipo de fonte de energia do dispositivo, estado da fonte de energia, dados de confiabilidade do caminho, e dados de confiabilidade. Para os parâmetros de energia, tem-se em consideração a descrição feita pelo protocolo WH em (FOUNDATION, 2011), onde é atribuído um código para cada estado dos parâmetros usados na função de custo. Neste trabalho usa-se esse código como uma designação numérica para possibilitar o cálculo. As estatísticas de confiabilidade são valores dados em porcentagens e calculados a partir dos relatórios de estado enviados ao gerenciador desde cada dispositivo. Adicionalmente, nesta equação foram usados dos valores constantes que ajudam em testes onde se deseja ter maior consideração de um dos termos da equação, sejam estes os parâmetros de energia ou os parâmetros de confiabilidade. A função de custos é definida pela equação (5).

$$e_i = x_e * \frac{Tsrc_i}{Pst_i + 1} + x_c * \left(\frac{1}{2} - \frac{DR * PR}{DR + PR}\right) \tag{5}$$

Onde:

$Tsrc_i$: tipo de fonte de energia usado pelo dispositivo i considerando: 0. Fonte de alimentação, 1. Bateria.

Pst_i : estado da fonte de energia do dispositivo i considerando: 1. Crítico-Baixo, 2. Recarga baixa, 3. Baixo, 4. Recarga alta, 5. Nominal.

DR : confiabilidade dos dados obtido a partir de métricas do relatório de estado do dispositivo

PR : confiabilidade do caminho obtido a partir de métricas do relatório de estado do dispositivo

x_e : peso estado de energia

x_c : peso estatísticas de confiabilidade

O pseudocódigo do algoritmo proposto é apresentado no Algoritmo 9 e tem como entrada o grafo original $G(V, E)$ e o grafo criado na primeira etapa $G(V_D, E_D)$ e gera como resultado os subgrafos *uplink* de cada dispositivo $G_U^i(V_U^i, E_U^i)$. Para começar o algoritmo, é escolhido um nó v do conjunto V_D para o qual o grafo $G_U^i(V_U^i, E_U^i)$ será construído, o nó v com valor de salto n é adicionado ao conjunto V_U e os seguintes passos são seguidos até obter o grafo *uplink* de todos os dispositivos.

1. Obter os nós de V_D com valor de salto $n - 1$ que tem arestas desde v e ordenar eles

acordo com a métrica dada por 5 com o fim de obter os três melhores candidatos e agregar essas arestas ao conjunto E_U e os vértices correspondentes em V_U.

2. Criar um vetor V_{temp} a partir dos dispositivos destino das arestas encontradas em 1 e fazer com cada um deles o processo executado em 1.

3. Se o vetor de destinos encontrado no passo 1 tem um só elemento e esse elemento é o ponto de acesso é procurado em V_D, um dispositivo com o qual v tenha enlace e tenha o menor valor de custo é escolhido e agregado em V_U, sendo que este será do mesmo nível que v, para garantir sempre a existência de quanto menos duas rotas.

4. Enquanto V_{temp} não esteja vazio, o processo é repetido desde o passo 1, caso contrário um novo dispositivo do conjunto V_D é escolhido e o processo de criação do seu grafo *uplink* é iniciado.

Após a execução desses algoritmos, tem-se como resultado o grafo *uplink* da rede Para a obtenção do grafo *downlink* é preciso utilizar do algoritmo 9 considerando que: o grafo de entrada $G(V, E)$ ao início do programa deverá ter as arestas invertidas $G^R(V, E^R)$ e o programa será executado com essa nova versão do grafo como grafo de entrada. No final, cada subgrafo *uplink* $G_U^i(V_U, E_U)$ deverá inverter o sentido das arestas $G_U^{R_i}(V_U, E_U^{R_i})$, e este último corresponderá ao grafo *downlink*.

Algorithm 9 Construção rotas confiáveis

In: $G(V, E)$, $G(V_D, E_D)$

Out: $G_U^i(V_U^i, E_U^i)$

Criação de vetor de dispositivos já explorados da rede S

Criação do grafo G_i para os pontos de acesso onde $V_{ui} = g \cup ap_i$

1: $notUsedV = V_D - S$
2: **while** $!notUsedV.empty()$ **do**
3: Escolher um $v \subset notUsedV$
4: Adiciona v ao conjunto S
5: Adiciona v ao conjunto V_U
6: Obter V_{temp} de dispositivos com arestas que vem de v
7: Ordenar V_{temp} segundo equação (5)
8: **if** $V_{temp}.size() == 1 \;\&\&\; == V_{AP}$ **then**
9: Obter lista de dispositivos com nível n
10: Ordenar lista de dispositivos obtidos segundo equação (5)
11: Adicionar a V_{temp} melhor opção
12: **else if** $V_{temp}.size() > 2$ **then**
13: Ordenar V_{temp} segundo equação (5)
14: Deixar em V_{temp} as duas melhores opções
15: **end if**
16: **for** todos v_v em V_{temp} **do**
17: Se existe aresta ente v e v_v
18: Adiciona a aresta ao conjunto E_U
19: Adiciona o vetor v_v ao conjunto V_U
20: Repete o processo todo para o dispositivo v_v
21: **end for**
22: Atualiza $notUsedV = V_D - S$
23: **end while**
24: Retorna a lista de grafos $G_U^i(V_U, E_U)$

6 ANÁLISE DE RESULTADOS

Os algoritmos implementados são executados sobre topologias de rede em malha com o objetivo de analisar e comparar seus comportamentos no momento da construção dos grafos. Os resultados são apresentados na forma de gráficos em barra para a avaliação das métricas.

6.1 RSFI simulada

Dentro do ambiente do gerenciador de rede foram construídas três topologias de rede em malha com diferentes quantidades de dispositivos: 50, 120 e 180. Os dispositivos foram criados e localizados aleatoriamente na área disposta para a rede. Toda a informação de cada uma das redes foi armazenada em arquivos de texto para conseguir avaliar cada algoritmo sobre as mesmas redes. As informações sobre as redes simuladas são apresentadas na tabela 3.

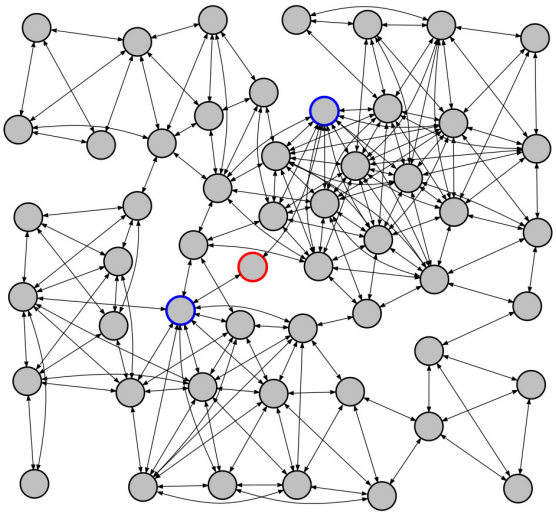
Tabela 3 – Informações da rede

Descrição	**Valor**
Dimensões da área	450x450
Número de dispositivos	50, 120, 180
Número de pontos de acesso	2
Máximo alcance de comunicação	100m
Mínimo RSL para conexão	-85dBm

Na figuras 14 a 16 são apresentadas respectivamente a topologia inicial de cada uma das redes. Em cada grafo o nó com borda vermelha representa o *gateway*, os nós com

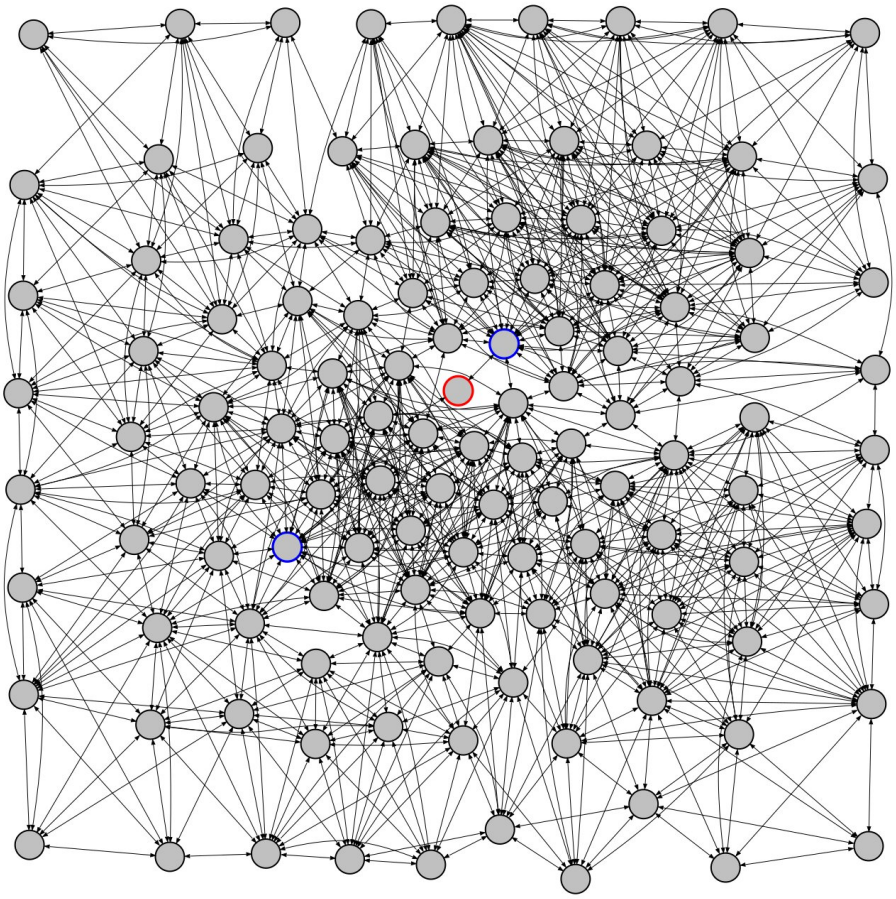
borda azul estão representando os pontos de acesso e os demais nós em preto, os dispositivos de campo (sensores ou atuadores distribuídos na planta).

Figura 14 – Topologia de rede com 50 dispositivos



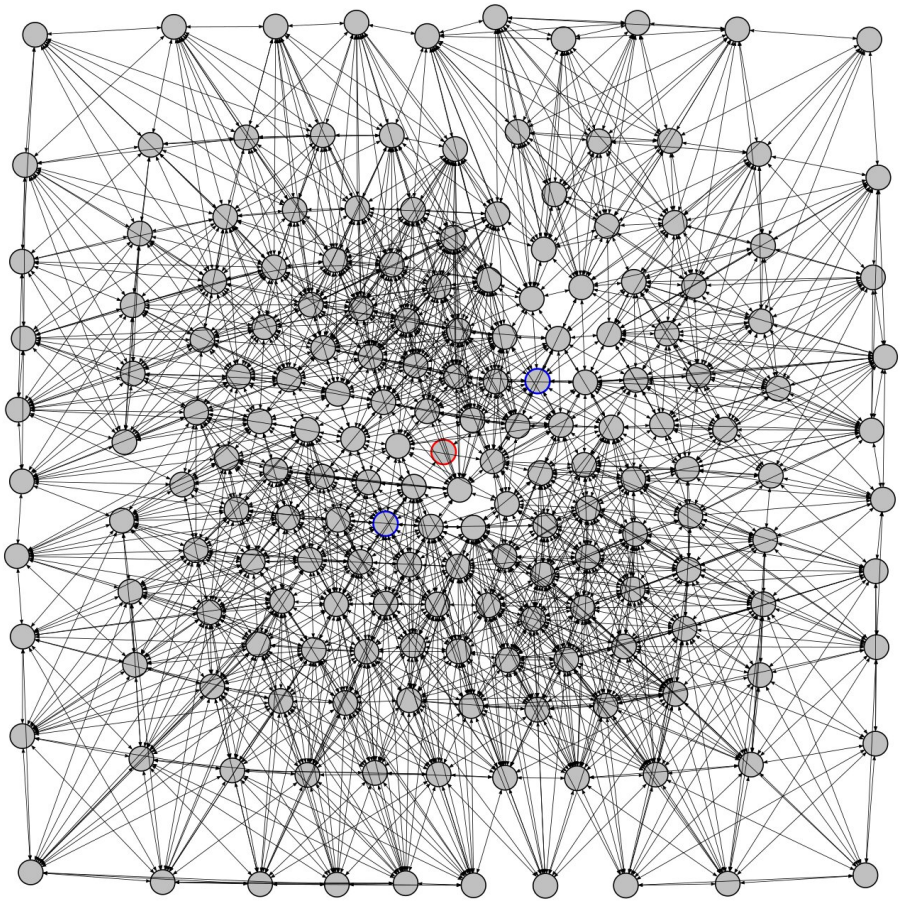
Fonte: Autor

Figura 15 – Topologia de rede com 120 dispositivos



Fonte: Autor

Figura 16 – Topologia de rede com 180 dispositivos



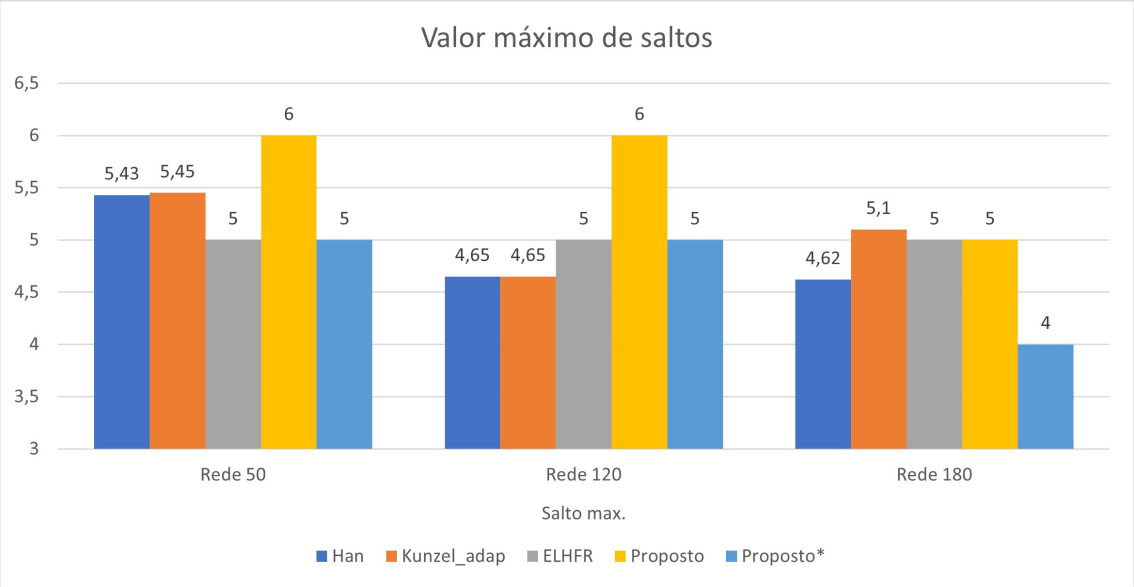
Fonte: Autor

Uma avaliação comparativa entre os algoritmos implementados é realizada e apresentada como gráficos de barra que permitem visualizar de forma direta os resultados obtidos. O algoritmo proposto foi implementado considerando que os nós são dispositivos com geolocalização, seja porque eles têm essa capacidade ou porque durante sua instalação se teve um controle sobre sua localização e é uma informação disponível. Nos gráficos seguintes será apresentado como *Proposto* o algoritmo que leva em consideração a localização real dos dispositivos para o cálculo da distância que é usada na equação (4) e como *Proposto** o mesmo algoritmo, mas neste caso, o valor de distância é considerado como uma constante.

A primeira métrica avaliada e apresentada na Figura 17 foi o valor de salto máximo. Neste gráfico, é possível observar como esse valor diminui em quase todas as implementações com o aumento da densidade da rede, e isso pode ser porque aumento de dispositivos na rede pode estar criando rotas mais diretas até os dispositivos mais distantes, diminuindo a quantidade de saltos por dispositivo. Quando a rede é pequena, não tem um

grande número de dispositivos candidatos para a formação das rotas, o que pode provocar que alcançar um dispositivo implique uma maior quantidade de saltos até ele.

Figura 17 – Valor de Salto Máximo



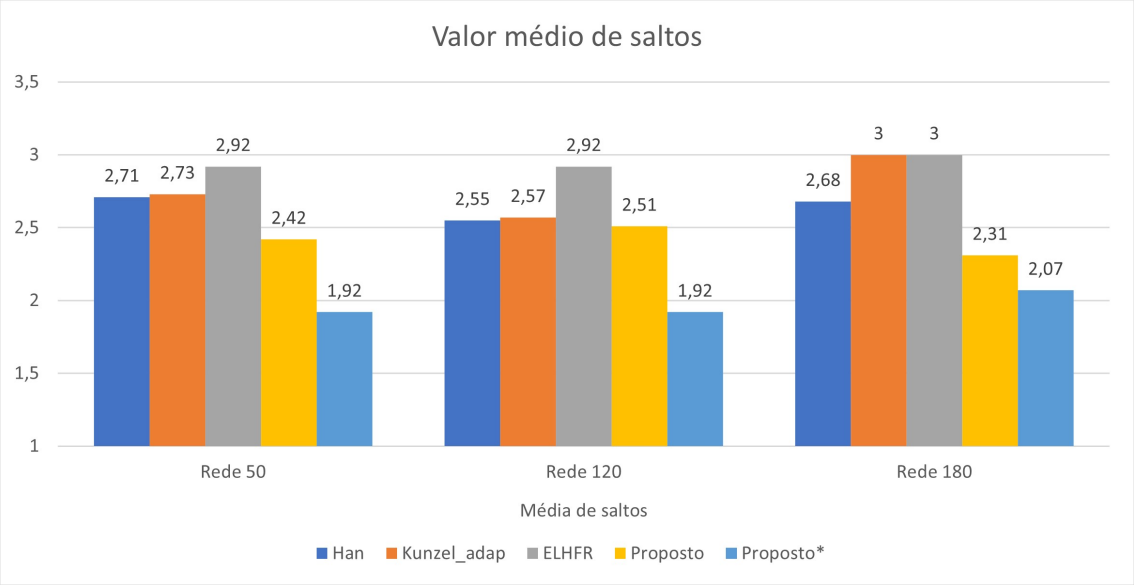
Fonte: Autor

Na Figura 18 o valor médio de saltos no algoritmo proposto é menor em todos os cenários de rede em comparação com as outras implementações. Isso pode ser devido a proposta do trabalho levar em consideração a distância entre os dispositivos dentro da equação de escolha de rotas mínimas, o que pode ser uma vantagem, dado que o tempo de envio e recepção de dados por parte dos dispositivos nessa rede pode ser menor em comparação com os outros algoritmos. Esta constatação implica em uma melhora no processo de roteamento da rede.

A disposição da rede foi sobre uma área de 450m x 450m sendo que o máximo alcance de comunicação é de até 100m. Analisando essas duas características neste trabalho, considera-se como dispositivos distantes aqueles nós que levam mais de quatro saltos para alcançar um ponto de acesso. Na Figura 19 se observa que o número de dispositivos distantes diminui quando o número de dispositivos na rede aumenta e isso acontece porque com maior número de nós na rede, tem-se maior probabilidade de encontrar rotas com menor número de saltos até os dispositivos mais distantes.

O algoritmo proposto obteve os menores valores de dispositivos distantes. Ele consegue estabelecer uma distribuição do grafo com as características dos nós da rede, o que possibilita uma menor quantidade de saltos entre os dispositivos de campo e os pontos de

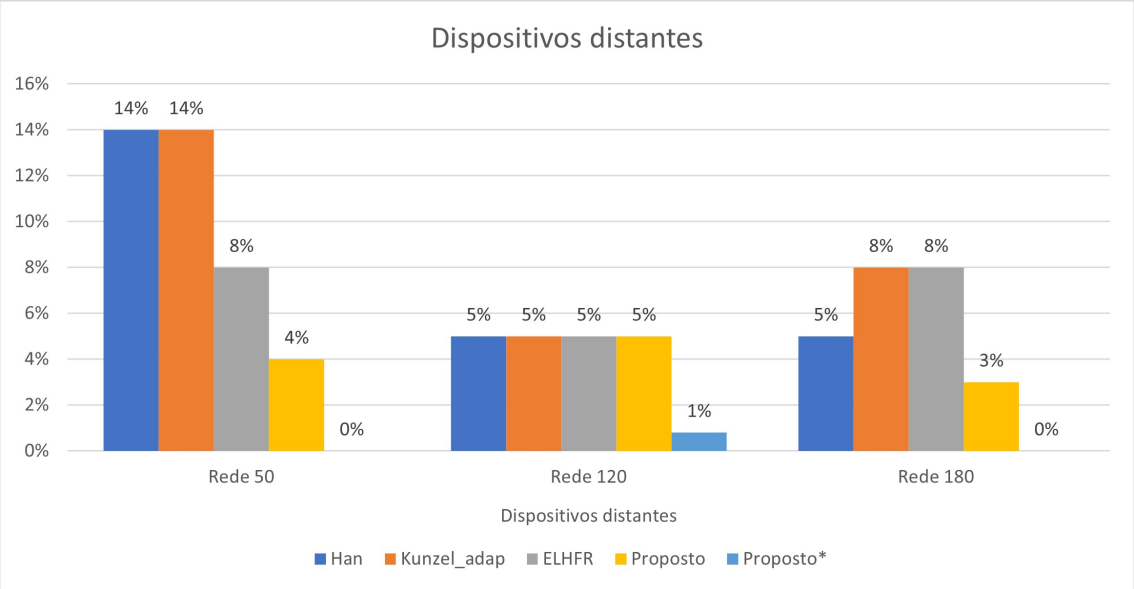
Figura 18 – Valor de Salto Médio



Fonte: Autor

acesso.

Figura 19 – Dispositivos Distantes



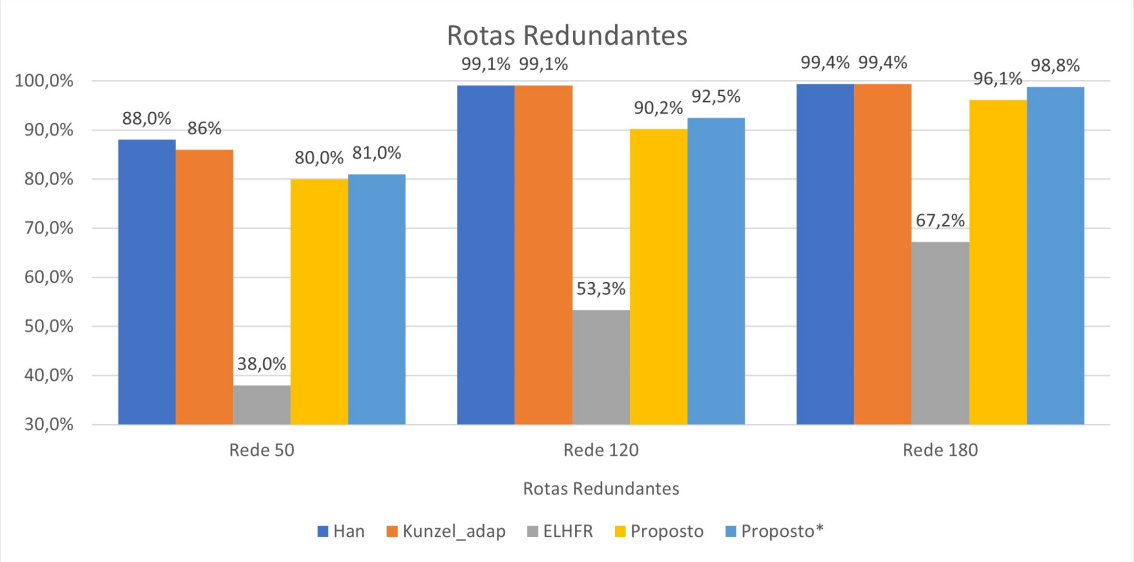
Fonte: Autor

Rotas redundantes em RSFI aumentam consideravelmente a confiabilidade da rede e são vitais em muitos processos e aplicações nas plantas. Essa redundância está sendo medida como a porcentagem de dispositivos confiáveis que são os dispositivos que dispõem de mais de uma rota para o envio ou recepção de pacote de dados. Na figura 20 os resultados mostram como o número de dispositivos confiáveis aumenta conforme o a den-

sidade da rede aumenta: com uma maior quantidade de dispositivos, a possibilidade de formação de rotas redundantes é maior, dado que cada dispositivo tem um maior número de vizinhos pelos quais encaminhar ou receber dados. Neste gráfico o algoritmo *ELHFR* apresenta um baixo número de dispositivos confiáveis, isso porque neste algoritmo todos os dispositivos diretamente conectados ao ponto de acesso estabelecem essa única rota para transferência de dados.

O modelo proposto, embora tenha uma base no algoritmo *ELHFR* procura sempre estabelecer rotas redundantes em que todos os dispositivos perto dos pontos de acesso procuram dispositivos vizinhos com sua mesma quantidade de saltos por onde construir outra rota, sem perder totalmente o objetivo alcançado com a etapa um do algoritmo.

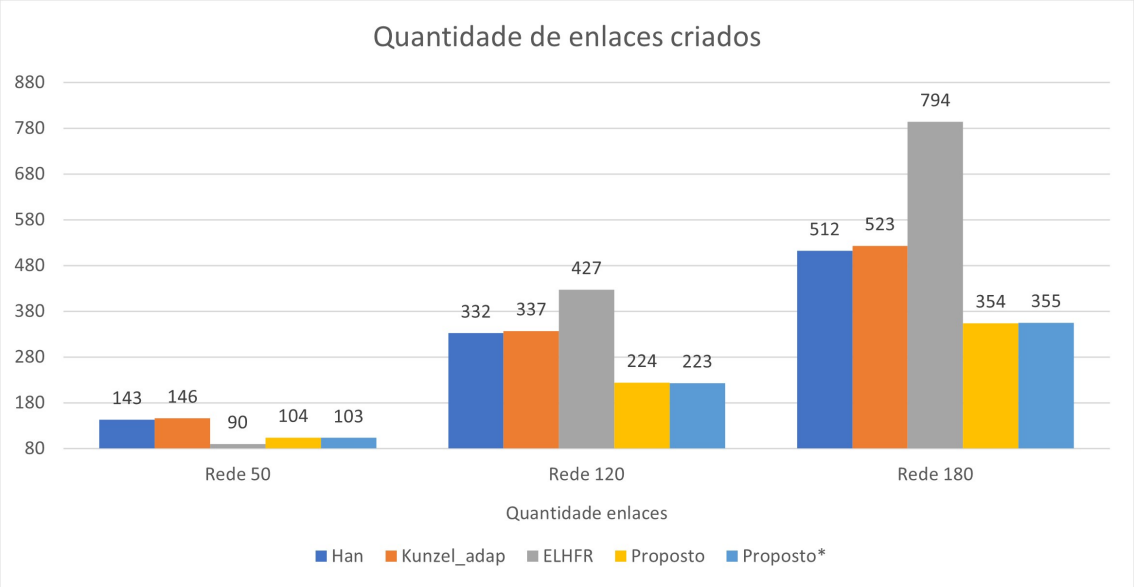
Figura 20 – Dispositivos Com Rotas Redundantes



Fonte: Autor

O algoritmo proposto em comparação com os demais algoritmos implementados apresenta uma menor quantidade de enlaces criados e isso pode ser muitas vezes relacionado com a robustez das rotas, mas é importante levar em consideração que a proposta já garante rotas redundantes. Desta forma, considera-se que algoritmo é robusto, além de apresentar tempo de resposta na construção das rotas menor que os demais algoritmos ao ter uma construção de caminhos com um número de ramos adequado, melhorando provavelmente o desempenho do gerenciador.

Figura 21 – Quantidade de enlaces



Fonte: Autor

Na tabela 4 é apresentado um resumo dos resultados obtidos e discutidos nesta seção com a finalidade de ter uma comparativa mais rápida.

Tabela 4 – Resumo resultados obtidos

Algoritmo	h_{max}			h_{med}			$h_{d>4}$			n_R			Q_{links}		
	50	**120**	**180**	**50**	**120**	**180**	**50**	**120**	**180**	**50**	**120**	**180**	**50**	**120**	**180**
Han	5,43	4,65	4,62	2,71	2,55	2,68	14%	5%	5%	88,0%	99,1%	99,4%	143	332	512
Kunzel	5,45	4,65	5,1	2,73	2,57	3	14%	5%	8%	86%	99,1%	99,4%	146	337	523
ELHFR	5	5	5	2,92	2,92	3	8%	5%	8%	38,0%	53,3%	67,2%	90	427	794
Proposto	**6**	**6**	**5**	**2,42**	**2,51**	**2,31**	**4%**	**5%**	**3%**	**80,0%**	**90,2%**	**96,1%**	**104**	**224**	**354**
Proposto*	**5**	**5**	**4**	**1,92**	**1,92**	**2,07**	**0%**	**1%**	**0%**	**81,0%**	**92,5%**	**98,8%**	**103**	**223**	**355**

Lembrando que:

– h_{max}: Valor de salto máximo da rede.

– h_{med}: Média da quantidade de saltos entre todos os dispositivos da rede;

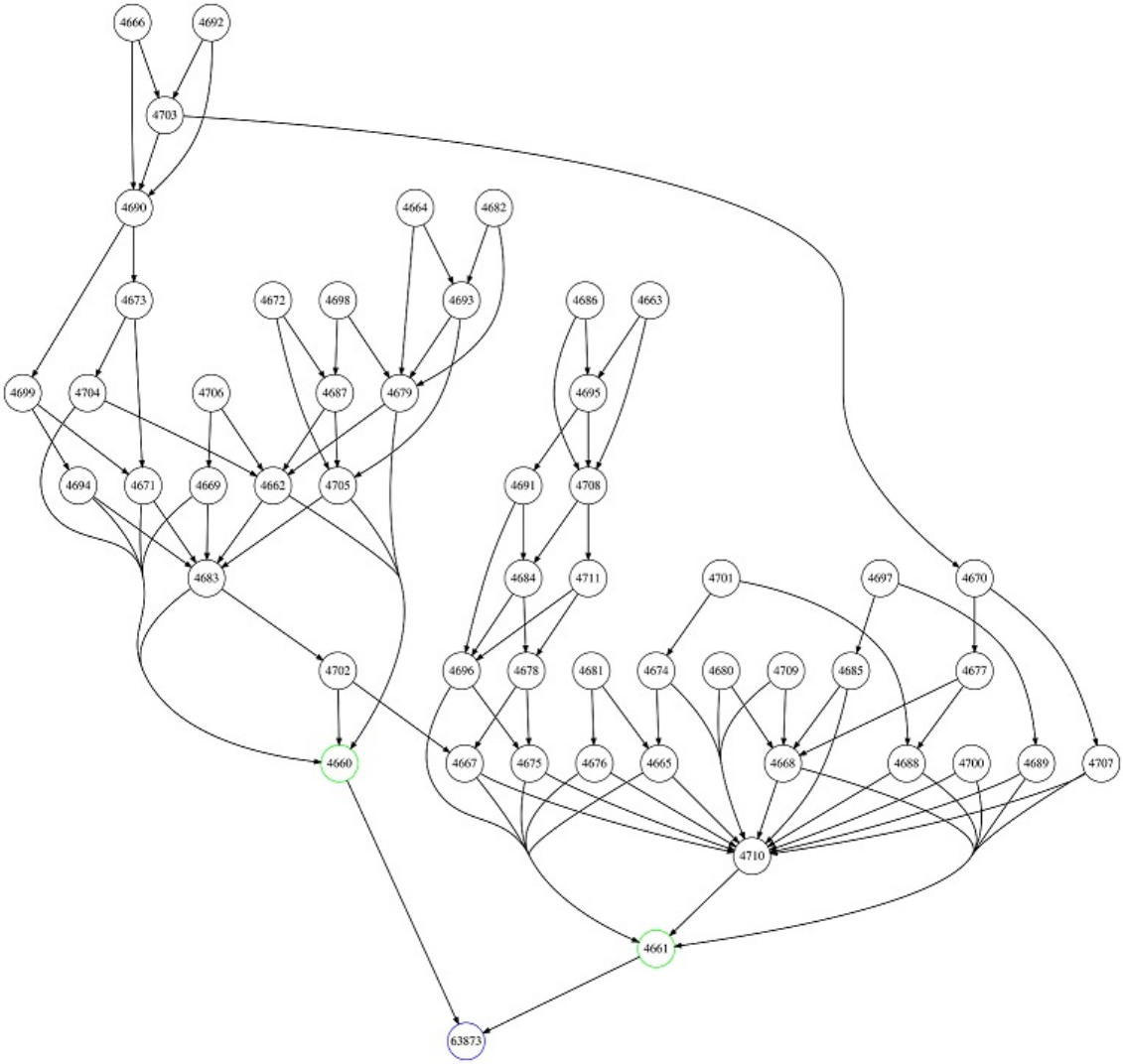
– $h_{d>4}$: Porcentagem de dispositivos na rede que estejam a mais de quatro saltos do ponto de acesso;

– n_R: Porcentagem de dispositivos na rede que tem duas arestas de saída pelas quais encaminhar ou/e receber pacotes de dados;

– Q_{links}: Quantidade de enlaces criados pelo algoritmo de formação do grafo.

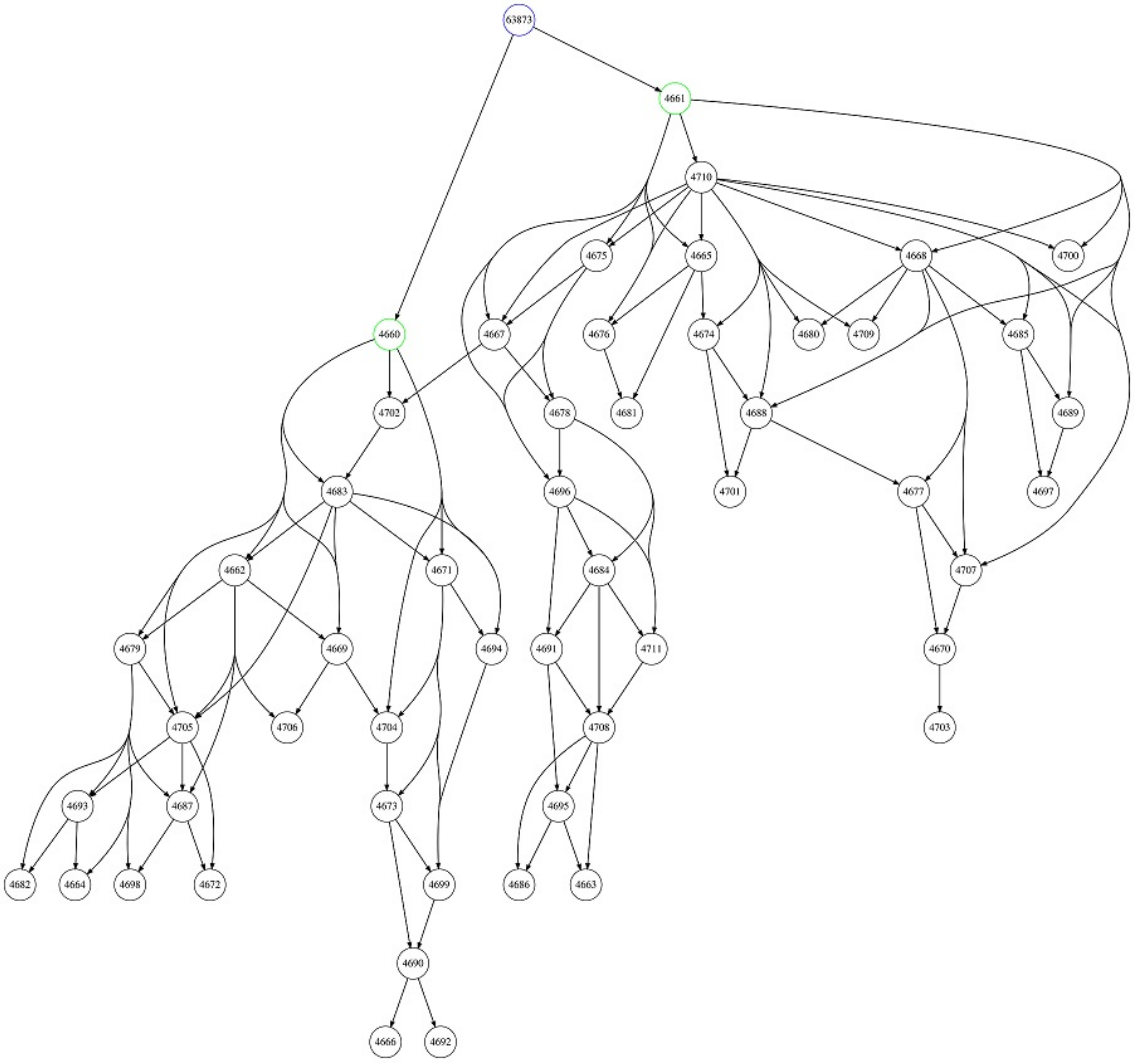
A continuação são apresentados os grafos *downlink* e *uplink* obtidos pela execução dos algoritmos implementados para uma rede de 50 dispositivos.

Figura 22 – Grafo *uplink* proposta de Han



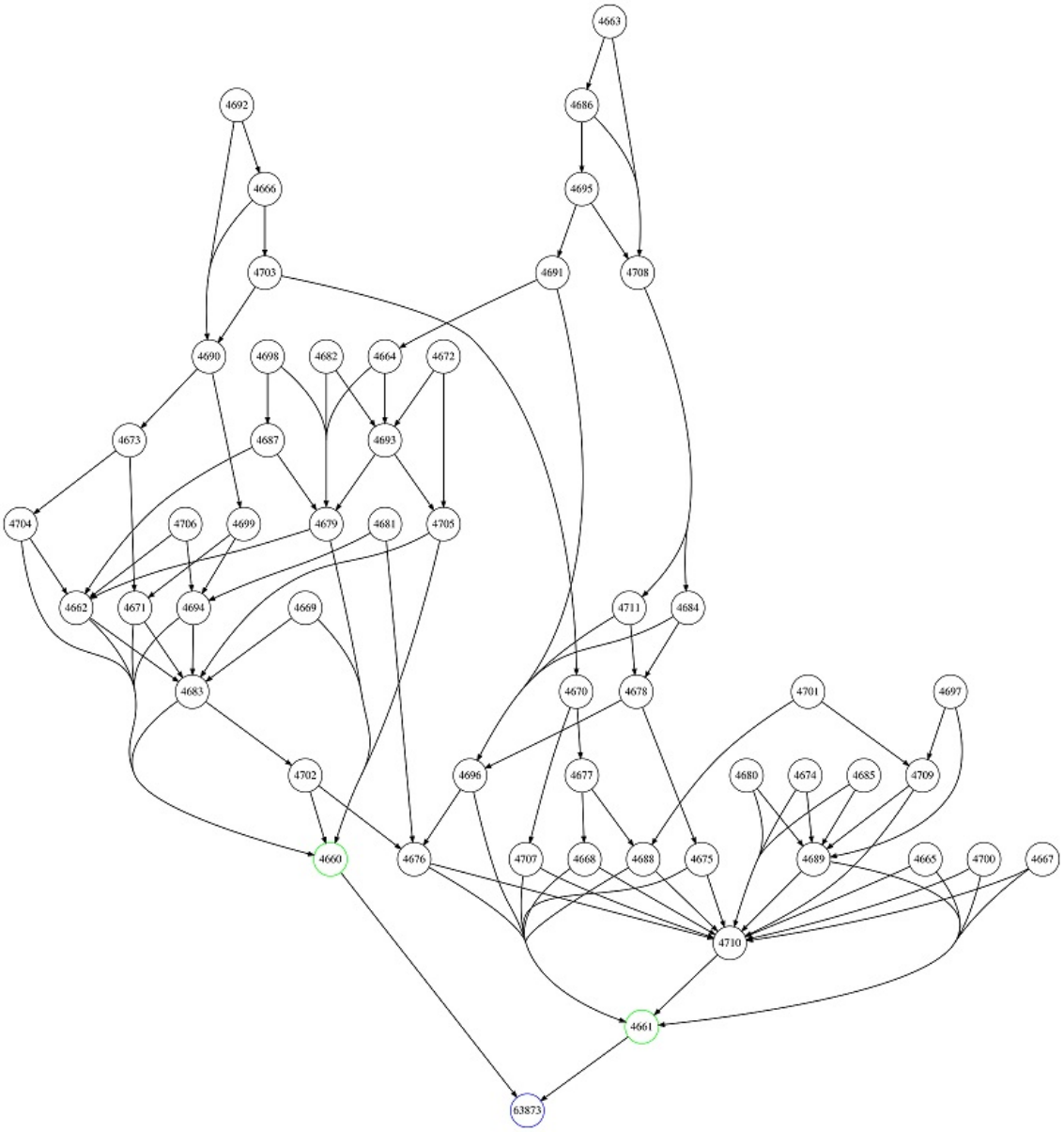
Fonte: Autor

Figura 23 – Grafo *downlink* proposta de Han



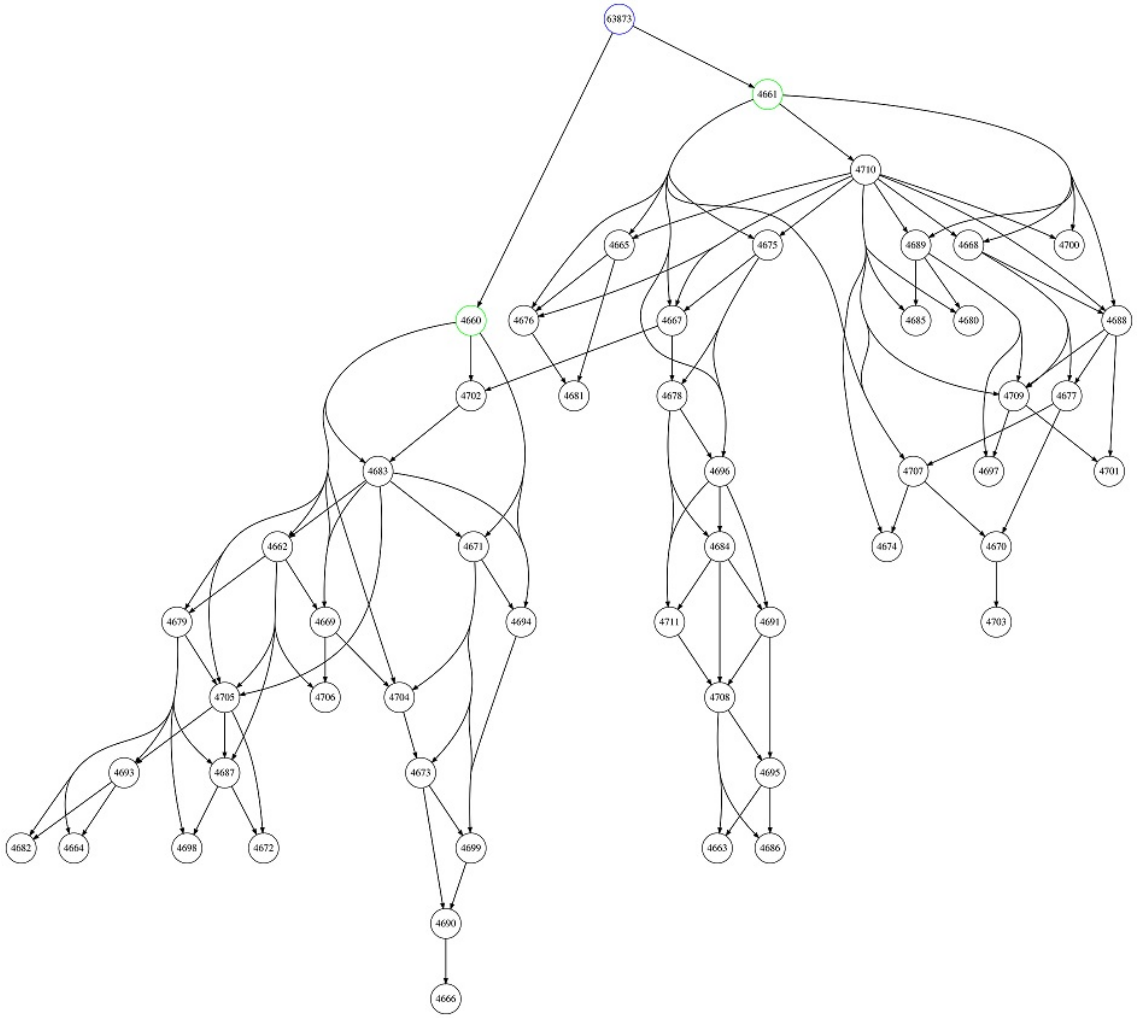
Fonte: Autor

Figura 24 – Grafo *uplink* proposta de Kunzel



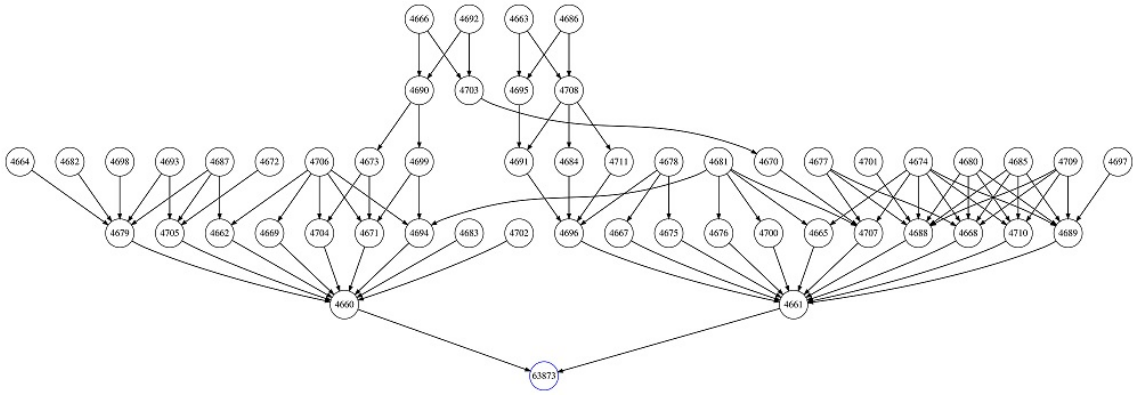
Fonte: Autor

Figura 25 – Grafo *downlink* proposta de Kunzel



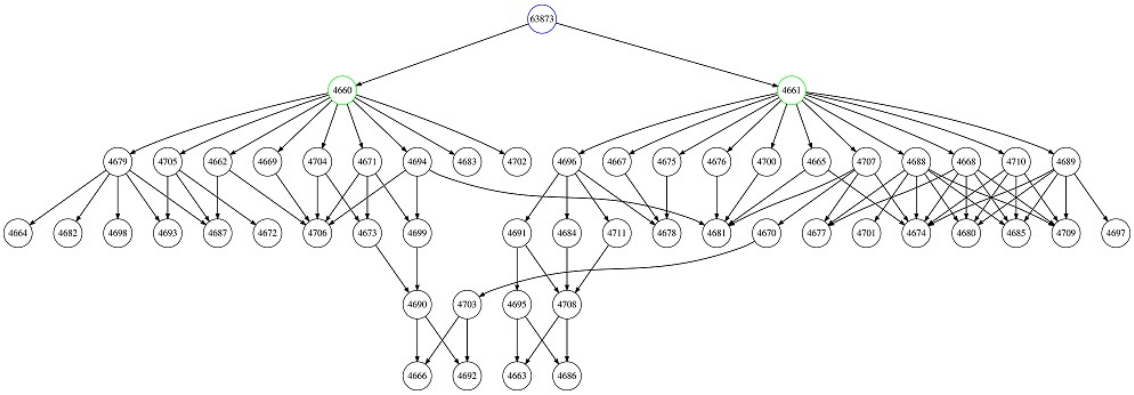
Fonte: Autor

Figura 26 – Grafo *uplink* proposta ELHFR



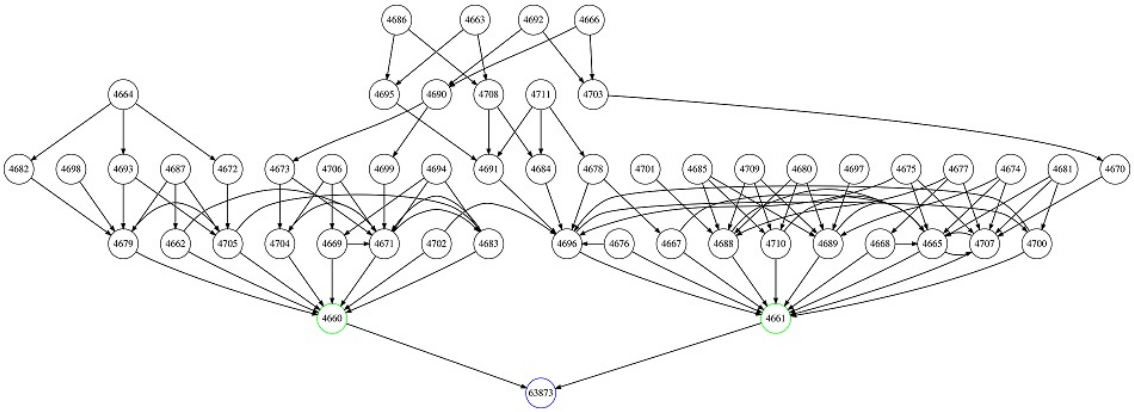
Fonte: Autor

Figura 27 – Grafo *downlink* proposta ELHFR



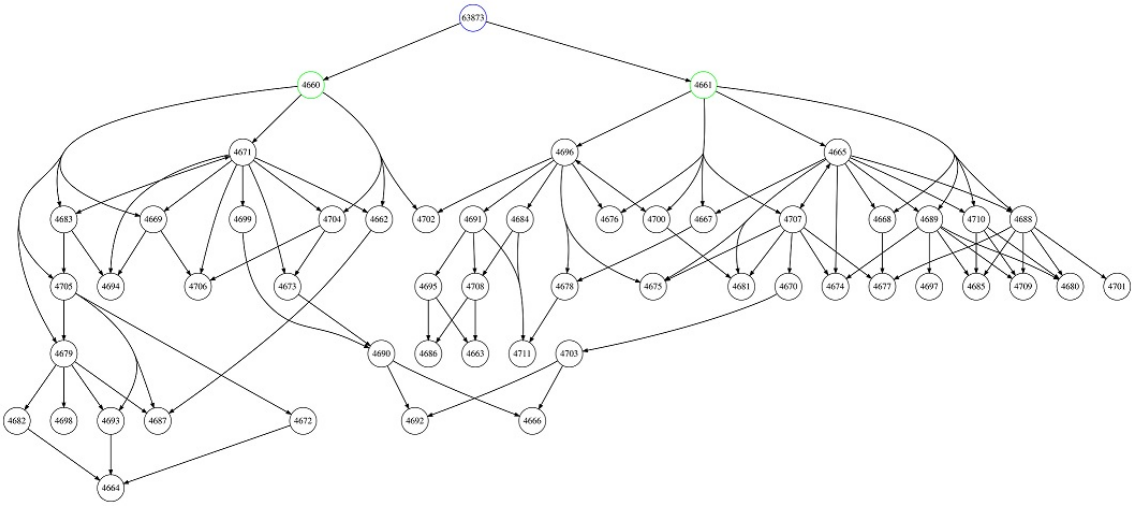
Fonte: Autor

Figura 28 – Grafo *uplink* proposta do trabalho



Fonte: Autor

Figura 29 – Grafo *downlink* proposta do trabalho

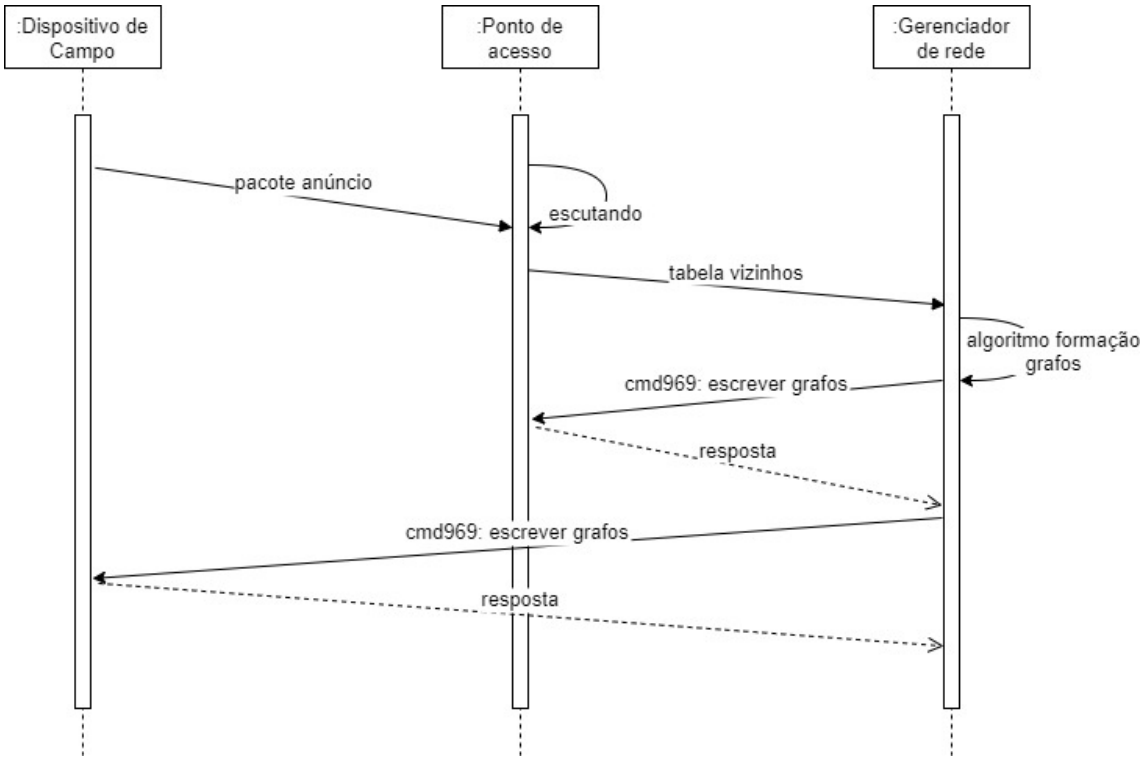


Fonte: Autor

6.2 Teste de funcionalidade prático

No desenvolvimento deste trabalho foi efetuado um teste de funcionalidade básico. O algoritmo proposto foi embarcado em um dispositivos de campo WH. A execução do algoritmo de formação de grafos foi verificada pelo envio do *graphID* para um dispositivo e registro da resposta obtida no processo. Algumas funcionalidades nos dispositivos e o gerenciador ainda estão sendo implementadas e melhoradas com a finalidade de permitir um teste com maior número de dispositivos. É importante ressaltar que os testes foram feitos com os mesmos códigos dos dispositivos usados durante uma execução *online*, sendo que a diferença é a obtenção de características e informações nos dispositivos mais perto de uma rede industrial real (sinais reais). Na figura 30 é apresentado o digrama de sequência geral de um dos processos implementados durante os testes.

Figura 30 – Diagrama Fluxo: Teste funcionalidade



Fonte: Autor

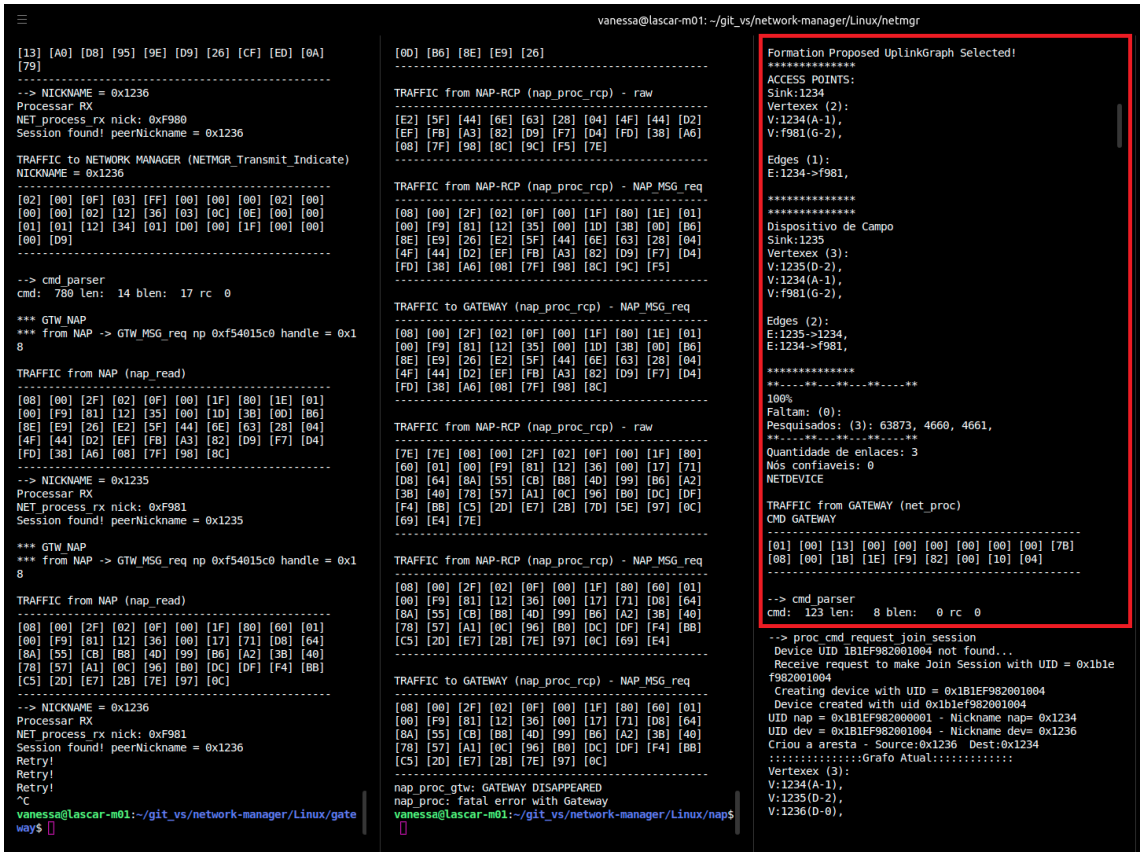
Dois testes foram feitos, com um e dois dispositivos de campo além do ponto de acesso, para avaliar a execução correta do algoritmo de roteamento proposto a partir das informações compartilhadas pelos dispositivos. Após os dispositivos entrarem na rede, eles compartilham com o gerenciador de rede a lista de dispositivos vizinhos de cada um, o gerenciador analisa a informação e cria listas de vértices e arestas com as quais será

criado o grafo de condições iniciais. Esse grafo é o ponto de partida para a execução dos algoritmos de roteamento que construíram os grafos segundo cada implementação, *graphID* são atribuídos para os grafos formados e compartilhados com os dispositivos correspondentes. Neste teste foi utilizado o comando WH 969 (*write graph*) para que o gerenciador de rede adicione uma atribuição de conexão para um dispositivo de rede específico.

Um dispositivo de campo

Na figura 31 é apresentada a execução do algoritmo proposto no gerenciador. Pode-se observar os vértices e arestas que fazem parte de cada subgrafo correspondente a cada dispositivo da rede. Já na figura 32 são as informações coletadas pelo *sniffer* durante a troca de mensagens na rede, especificamente está apresentando a requisição e resposta obtida pelo uso do comando 969.

Figura 31 – Execução algoritmo proposto: 1 dispositivo



Fonte: Autor

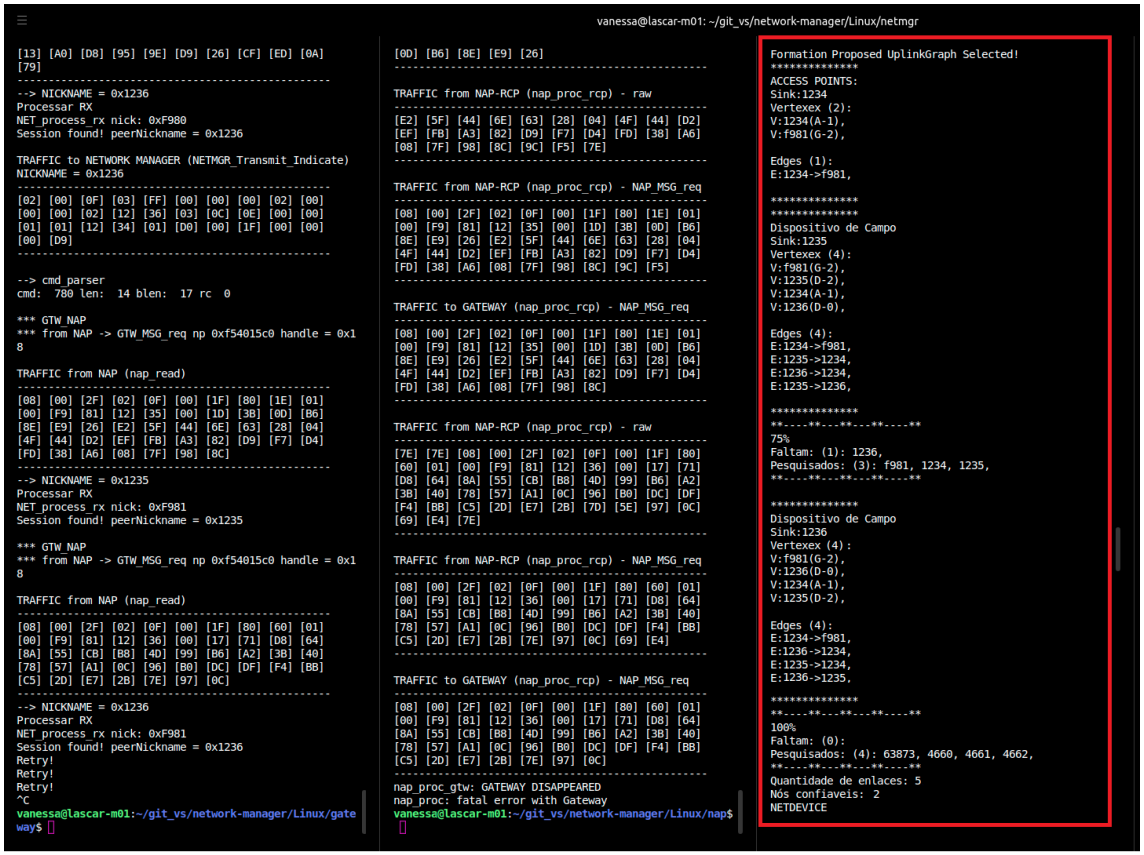
Figura 32 – Informação sniffer: um dispositivo, cmd969

Fonte: Autor

Dois dispositivos de campo

Um novo dispositivo é adicionado e depois de ter entrado na rede, novas informações devem ser compartilhadas por parte dos dispositivos da rede, atualizando as listas de vizinhos e/ou reportando dispositivos próximos com os quais não se tenha criado enlace. Uma vez mais o gerenciador é o encarregado de analisar as novas informações, estabelecer um novo grafo inicial e executar os algoritmos de roteamento. Na figura 33 é apresentada a execução do algoritmo proposto para as novas condições da rede.

Figura 33 – Execução algoritmo proposto: 2 dispositivos



Fonte: Autor

Figura 34 – Informação sniffer: dois dispositivos, cmd969

Fonte: Autor

7 CONCLUSÃO

Este trabalho apresentou a implementação de algoritmos de roteamento para RSFI objetivando aumento de confiabilidade nas comunicações industriais. O estudo desses algoritmos e a análise de alguns dos problemas mais comuns nas RSFI levou-nos a estabelecer uma proposta de algoritmo para a criação de grafos no processo de roteamento que pudesse garantir confiabilidade, e que levasse em consideração parâmetros dos dispositivos que pudessem ajudar na melhora da confiabilidade das rotas de comunicação, tais como o estado de energia dos nós, os níveis de sinal recebido pelos vizinhos do dispositivo, a confiabilidade da rota baseada no número de pacotes recebidos e transmitidos com o sem sucesso, e a confiabilidade de dados baseada no número de pacotes gerados terminados e com falha. Todos esses parâmetros obtidos a partir dos relatórios de estados dos nós.

O consumo de energia nos dispositivos sem fio pode se converter em um problema crítico se não for avaliado com atenção, isso porque para dispositivos que tem funcionamento com bateria, a energia é um parâmetro de grande restrição. No algoritmo proposto, a consideração do estado de energia dos nós permite fazer uso de rotas com dispositivos que têm as melhores condições de energia entre os candidatos, o que proporciona maior segurança no encaminhamento de dados.

Os parâmetros apresentados nos relatórios de estado dos dispositivos sempre têm sido guia para analisar a saúde dos dispositivos e seu comportamento na rede. Foram consideradas algumas dessas informações que ajudaram a escolher dispositivos e rotas com maior confiabilidade ao analisar o número de pacotes gerados, perdidos e com falhas de um dispositivo: a escolha de dispositivos com uma maior porcentagem de confiabilidade melhora a confiabilidade da rota estabelecida.

Dos resultados obtidos na análise comparativa, pode se observar que o algoritmo pro-

posto apresenta uma boa resposta para redes de pequeno e médio porte com um grande número de dispositivos que conseguiram estabelecer rotas redundantes de comunicação, um baixo valor de salto médio e salto máximo, o que indica que se conseguiu estabelecer condições com mínimas rotas possíveis.

O módulo de simulação desenvolvido neste trabalho possibilitou o estudo e a implementação dos algoritmos com uma ótima abordagem para as condições reais.

Trabalhos futuros

Outros parâmetros dos dispositivos poderão ser analisados e considerados para adaptar o modelo do algoritmo proposto, e estudar possíveis melhoras para o uso em diferentes aplicações. Ainda se sugere o estudo e aplicação das propostas em redes práticas onde se têm mais de um ponto de acesso.

REFERÊNCIAS

CAINELLI, G. *et al.* Development of a Network Manager Compatible with WirelessHART Standard. **XXIII Congresso Brasileiro de Automática (CBA 2020)**, [S.l.], v. 2, p. 1–7, 2020.

CHEN, D.; NIXON, M.; MOK, A. **WirelessHart, Real-Time mesh network for industrial automation**. Austin, TX-USA: Springer, 2010. 276 p.

CHEN, Q. *et al.* High Reliability, Low Latency and Cost Effective Network Planning for Industrial Wireless Mesh Networks. **IEEE/ACM Transactions on Networking**, [S.l.], v. 27, n. 6, p. 2354–2362, 2019.

CHRIS, G.; GORDON, F. R. **Algebraic Graph Theory**. New York, NY: Springer, 2001. XIX, 443 p.

CHU, Y.-J. *et al.* Application of load-balanced tree routing algorithm with dynamic modification to centralized wireless sensor networks. *In*: SENSORS, 2009 IEEE, 2009. **Proceedings [...]** [S.l.: s.n.], 2009. p. 1392–1395.

DIJKSTRA, E. W. A note on two problems in connexion with graphs. **Numerische Mathematik**, [S.l.], v. 1, p. 269–271, 1959.

EL-FOULY, F. H.; RAMADAN, R. A. Real-Time Energy-Efficient Reliable Traffic Aware Routing for Industrial Wireless Sensor Networks. **IEEE Access**, [S.l.], v. 8, p. 58130–58145, 2020.

FOUNDATION, H. C. HCF_SPEC-183, Revision 21.0 Common Tables Specification. **WirelessHart**, [S.l.], 2011.

HAHN, D.; NETTO, J. Desenvolvimento de um ponto de acesso para redes wirelessHART. **Projeto de diplomação(Graduação em Engenharia Eléctrica)**, [S.l.], v. Universidade Federal do Rio Grande do Sul, Porto Alegre, p. 1430–1434, 2011.

HAN, S. *et al.* Reliable and Real-Time Communication in Industrial Wireless Mesh Networks. *In*: IEEE REAL-TIME AND EMBEDDED TECHNOLOGY AND APPLICATIONS SYMPOSIUM, 2011., 2011. **Proceedings [...]** [S.l.: s.n.], 2011. p. 3–12.

HAN, X.; MA, X.; CHEN, D. Energy-balancing routing algorithm for WirelessHART. *In*: IEEE INTERNATIONAL WORKSHOP ON FACTORY COMMUNICATION SYSTEMS (WFCS), 2019., 2019. **Proceedings [...]** [S.l.: s.n.], 2019. p. 1–7.

JHA, A. V.; APPASANI, B.; GHAZALI, A. N. Performance Evaluation of Network Layer Routing Protocols on Wireless Sensor Networks. *In*: INTERNATIONAL CONFERENCE ON COMMUNICATION AND ELECTRONICS SYSTEMS (ICCES), 2019., 2019. **Proceedings [...]** [S.l.: s.n.], 2019. p. 1862–1865.

JINDONG, Z.; ZHENJUN, L.; YAOPEI, Z. ELHFR: a graph routing in industrial wireless mesh network. **International Conference on Information and Automation**, [S.l.], p. 22–25, 2009.

KHAKPOUR, K.; SHENASSA, M. Industrial Control using Wireless Sensor Networks. *In*: INTERNATIONAL CONFERENCE ON INFORMATION AND COMMUNICATION TECHNOLOGIES: FROM THEORY TO APPLICATIONS, 2008., 2008. **Proceedings [...]** [S.l.: s.n.], 2008. p. 1–5.

KÜNZEL, G. Ambiente para avaliação de estratégias de roteamento para redes WirelessHART. *In*: 2012. **Proceedings [...]** [S.l.: s.n.], 2012. v. 1, p. 95.

KÜNZEL, G.; CAINELLI, G.; PEREIRA, C. A Weighted Broadcast Routing Algorithm for WirelessHART Networks. *In*: 2017. **Proceedings [...]** [S.l.: s.n.], 2017. p. 187–192.

LAW, K. *et al.* Decentralized Control Strategies with Wireless Sensing and Actuation. , [S.l.], 12 2010.

LINDHORST, T.; LUKAS, G.; NETT, E. Wireless Mesh Network infrastructure for industrial applications — A case study of tele-operated mobile robots. *In*: IEEE 18TH

CONFERENCE ON EMERGING TECHNOLOGIES FACTORY AUTOMATION (ETFA), 2013., 2013. **Proceedings [...]** [S.l.: s.n.], 2013. p. 1–8.

MÜLLER, I. *et al.* Development of a WirelessHART compatible field device. **IEEE INTERNATIONAL INSTRUMENTATION AND MEASUREMENT TECHNOLOGY CONFERENCE**, [S.l.], p. 1430–1434, 2010.

NOBRE, M.; SILVA, I.; GUEDES, L. A. Routing and Scheduling Algorithms for WirelessHARTNetworks: a survey. **Sensors**, [S.l.], v. 15, p. 9703–9740, 05 2015.

PENG, J. *et al.* A Study of Routing Protocols in Wireless Sensor Networks. *In*: WORLD CONGRESS ON INTELLIGENT CONTROL AND AUTOMATION, 2006., 2006. **Proceedings [...]** [S.l.: s.n.], 2006. v. 1, p. 266–270.

PIECHOWIAK, M. *et al.* Comparative analysis of routing protocols for wireless mesh networks. *In*: INTERNATIONAL SYMPOSIUM ON COMMUNICATION SYSTEMS, NETWORKS AND DIGITAL SIGNAL PROCESSING (CSNDSP), 2016., 2016. **Proceedings [...]** [S.l.: s.n.], 2016. p. 1–5.

RUAY-SHIUNG, C.; CHIA-JOU, K. An energy efficient routing mechanism for wireless sensor networks. *In*: INTERNATIONAL CONFERENCE ON ADVANCED INFORMATION NETWORKING AND APPLICATIONS - VOLUME 1 (AINA'06), 20., 2006. **Proceedings [...]** [S.l.: s.n.], 2006. v. 2, p. 5 pp.–.

TAN, D. D.; PRASETIADI, A.; KIM, D.-S. Traffic-balancing routing scheme for industrial wireless sensor networks. *In*: INTERNATIONAL CONFERENCE ON ICT CONVERGENCE (ICTC), 2012., 2012. **Proceedings [...]** [S.l.: s.n.], 2012. p. 595–600.

URBINA, M. *et al.* Analysis and comparison of routing protocols in Wireless Sensor Networks under 802.15.4. *In*: IEEE THIRTY FIFTH CENTRAL AMERICAN AND PANAMA CONVENTION (CONCAPAN XXXV), 2015., 2015. **Proceedings [...]** [S.l.: s.n.], 2015. p. 1–6.

WANG; QUAN, P. A Finite-State Markov Model for Reliability Evaluation of Industrial Wireless Network. *In*: INTERNATIONAL CONFERENCE ON WIRELESS COMMUNICATIONS NETWORKING AND MOBILE COMPUTING (WICOM), 2010., 2010. **Proceedings [...]** [S.l.: s.n.], 2010. p. 1–4.

WINTER, J. M. *et al.* Study of routing mechanisms in a WirelessHART network. *In*: IEEE INTERNATIONAL CONFERENCE ON INDUSTRIAL TECHNOLOGY (ICIT), 2013., 2013. **Proceedings [...]** [S.l.: s.n.], 2013. p. 1540–1545.

WU, C. *et al.* Maximizing Network Lifetime of WirelessHART Networks under Graph Routing. *In*: 2013 , 2016. **Proceedings [...]** [S.l.: s.n.], 2016.

YU, K. *et al.* Reliable real-time routing protocol for industrial wireless sensor and actuator networks. *In*: IEEE 8TH CONFERENCE ON INDUSTRIAL ELECTRONICS AND APPLICATIONS (ICIEA), 2013., 2013. **Proceedings [...]** [S.l.: s.n.], 2013. p. 1895–1901.

ZAKRZEWSKA, A. *et al.* Analysis of Routing Protocol Performance in Wireless Mesh Networks. *In*: INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE AND ITS APPLICATIONS, 2010., 2010. **Proceedings [...]** [S.l.: s.n.], 2010. p. 307–310.

ZHANG, S.; YAN, A.; MA, T. An Energy-Balanced Graph Routing Algorithm for WirelessHART Networks. *In*: INTERNATIONAL CONFERENCE ON INTELLIGENT HUMAN-MACHINE SYSTEMS AND CYBERNETICS, 2013., 2013. **Proceedings [...]** [S.l.: s.n.], 2013. v. 2, p. 557–560.

ZHANG, S.; YAN, A.; MA, T. An Energy-Balanced Graph Routing Algorithm for WirelessHART Networks. *In*: INTERNATIONAL CONFERENCE ON INTELLIGENT HUMAN-MACHINE SYSTEMS AND CYBERNETICS, 2013., 2013. **Proceedings [...]** [S.l.: s.n.], 2013. v. 2, p. 557–560.

ZHANG, W. *et al.* An Energy Efficient and QoS Aware Routing Algorithm Based on Data Classification for Industrial Wireless Sensor Networks. **IEEE Access**, [S.l.], v. 6, p. 46495–46504, 2018.

ZHAO, J. *et al.* Issues of routing protocol for Wireless Industrial Sensor Networks. *In*: IECON 2012 - 38TH ANNUAL CONFERENCE ON IEEE INDUSTRIAL ELECTRONICS SOCIETY, 2012. **Proceedings [...]** [S.l.: s.n.], 2012. p. 3225–3230.

ZUO, Y.; LING, Z.; YUAN, Y. Hybrid multi-path routing algorithm for industrial wireless mesh networks. **Sensors**, [S.l.], 03 2013.