

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

VICTOR HUGO DICKOW

**AVALIAÇÃO DE ALGORITMOS DE ROTEAMENTO E
ESCALONAMENTO DE MENSAGENS PARA REDES
WIRELESSHART**

Porto Alegre

2014

VICTOR HUGO DICKOW

**AVALIAÇÃO DE ALGORITMOS DE ROTEAMENTO E
ESCALONAMENTO DE MENSAGENS PARA REDES
WIRELESSHART**

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. Carlos Eduardo Pereira

Porto Alegre

2014

VICTOR HUGO DICKOW

**AVALIAÇÃO DE ALGORITMOS DE ROTEAMENTO E
ESCALONAMENTO DE MENSAGENS PARA REDES
WIRELESSHART**

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. Carlos Eduardo Pereira, UFRGS

Doutor pela Universität Stuttgart – Stuttgart, Alemanha

Banca Examinadora:

Prof. Dr. Walter Fetter Lages, UFRGS

Doutor pelo Instituto Tecnológico de Aeronáutica – São José dos Campos, Brasil

Prof. Dr. João César Netto, UFRGS

Doutor pela Université Catholique de Louvain – Lovaina, Bélgica

Prof. Dr. Ivan Müller, UFRGS

Doutor pela Universidade Federal do Rio Grande do Sul – Porto Alegre, Brasil

Coordenador do PPGEE: _____

Prof. Dr. Arturo Suman Bretas

Porto Alegre, março de 2014.

AGRADECIMENTOS

Aos meus pais, pelo incentivo, apoio e paciência durante todo período do mestrado.

Ao professor Carlos Eduardo Pereira, pela confiança e por prover a oportunidade de pesquisa.

Ao colega Ivan Müller, pela dedicação, auxílio e incentivo nas tarefas e obstáculos durante o desenvolvimento desse trabalho.

A minha namorada Tatiane Martins Machado, pelo apoio incondicional e incentivo na decisão de iniciar o mestrado.

Aos professores Sérgio Vidal Garcia Oliveira e Fábio Rafael Segundo pela indicação ao ingressar no mestrado.

Aos demais amigos e colegas que foram importantes durante este período.

RESUMO

A aplicação de redes sem fio vem crescendo consideravelmente nos últimos anos. Protocolos baseados nesta tecnologia estão sendo desenvolvidos para uma grande variedade de aplicações. A confiabilidade é um dos principais requisitos dos protocolos de comunicação nos ambientes industriais. Interferências, ambiente ruidoso e o grande risco das aplicações industriais que são monitoradas são fatores que elevam os níveis de exigência no que se refere à confiabilidade, redundância e segurança do protocolo. O protocolo *WirelessHART* é um padrão de comunicação sem fio desenvolvido especificamente para monitoramento e controle de processos com os requisitos necessários para ser utilizado em ambientes industriais. A norma do *WirelessHART* define diversos aspectos técnicos a serem utilizados no desenvolvimento de algoritmos. Os algoritmos de roteamento e escalonamento de mensagens são de grande relevância para o cumprimento dos requisitos temporais, de confiabilidade e segurança. Requisitos de roteamento e escalonamento são especificados, porém, os algoritmos a serem utilizados não são definidos. O objetivo nessa dissertação é analisar alguns dos principais algoritmos que tem sido propostos especificamente para o protocolo *WirelessHART* e apresentar um conjunto capaz de ser aplicado nesse protocolo. Análises e comparações entre os algoritmos são realizadas proporcionando um estudo aprofundado sobre seus impactos no desempenho do protocolo.

Palavras-chaves: Protocolo *WirelessHART*, Algoritmos de roteamento, Algoritmos de escalonamento, Redes sem fio industriais.

ABSTRACT

The application of wireless networks has grown considerably in recent years. Protocols based on this technology are being developed for a great variety of applications. Reliability is one of the main requirements for communication protocols in industrial environments. Interferences, noisy environment and high risk processes that are monitored are factors that increase the levels of requirements in terms of reliability, redundancy and security of the protocol. The WirelessHART protocol is a wireless communication standard specifically designed for process monitoring and control applications with the necessary requirements for to be used in industrial environments. The WirelessHART standard defines several technical aspects to be used in the development of the algorithms. The algorithms of routing and scheduling messages are highly relevant to meeting the timing requirements of reliability and safety. Routing and scheduling strategies are specified, however, the routing and scheduling algorithms are not defined for use. The goal of this dissertation is to analyze some of the main algorithms that have been proposed specifically for the WirelessHART protocol and to present a set able to be applied in this protocol. Analyzes and comparisons between algorithms are realized by providing a detailed study of their impacts on the protocol performance.

Keywords: WirelessHART Protocol, Routing algorithms, scheduling algorithms, Industrial wireless networks.

SUMÁRIO

1	INTRODUÇÃO	11
1.1	MOTIVAÇÃO	13
1.2	OBJETIVOS	13
2	FUNDAMENTAÇÃO TEÓRICA	15
2.1	TEORIA DOS GRAFOS	15
2.2	O PROTOCOLO <i>WIRELESSHART</i>	20
2.3	ROTEAMENTO NO PROTOCOLO <i>WIRELESSHART</i>	27
2.4	ESCALONAMENTO NO PROTOCOLO <i>WIRELESSHART</i>	32
3	ANÁLISE DO ESTADO DA ARTE	39
3.1	ALGORITMOS DE ROTEAMENTO	39
3.2	ALGORITMOS DE ESCALONAMENTO	40
4	PROPOSTA DE INTEGRAÇÃO ENTRE ALGORITMOS DE ROTEAMENTO E ESCALONAMENTO NO AMBIENTE DE SIMULAÇÃO	43
4.1	AMBIENTE PARA AVALIAÇÃO DOS ALGORITMOS DE ROTEAMENTO	43
4.2	AMBIENTE PARA AVALIAÇÃO DOS ALGORITMOS DE ESCALONAMENTO	44
4.2.1	Desenvolvimento da estrutura de programação para o escalonamento	45
4.2.2	Novas interfaces gráficas do ambiente	46
4.3	ALGORITMOS DESENVOLVIDOS NO AMBIENTE DE ANÁLISE	47
4.3.1	Algoritmo de roteamento Bellman-Ford de dupla execução	48
4.3.2	Algoritmo de roteamento ELHFR	51
4.3.3	Algoritmo de roteamento Han	53
4.3.4	Algoritmo de escalonamento Han	54
4.3.5	Algoritmo de escalonamento Dang	56
4.3.6	Algoritmo de escalonamento Zhang	59
5	PLANEJAMENTO DAS ANÁLISES E DEFINIÇÃO DAS MÉTRICAS PARA AVALIAÇÃO DOS ALGORITMOS	63
5.1.1	Primeiro experimento	63
5.1.2	Segundo experimento	63
5.1.3	Métricas para avaliação e comparação dos grafos de roteamento e das matrizes de escalonamento	65
6	ANÁLISE E RESULTADOS	67
6.1	PRIMEIRO EXPERIMENTO - ANÁLISE DOS ALGORITMOS DE ROTEAMENTO	67
6.2	SEGUNDO EXPERIMENTO - ANÁLISE DOS ALGORITMOS DE ESCALONAMENTO	76
6.3	AVALIAÇÃO DOS MELHORES CONJUNTOS	79
7	CONCLUSÕES	82

LISTA DE ILUSTRAÇÕES

Figura 1 Topologia genérica de um grafo.	16
Figura 2 Exemplo de um caminho P com tamanho $k = 6$	16
Figura 3 Representação de uma árvore.	17
Figura 4 Grafo percorrido pelo algoritmo DFS.	18
Figura 5 Grafo percorrido pelo algoritmo BFS.	18
Figura 6 Coloração de arestas de um grafo.	20
Figura 7 Evolução do protocolo <i>WirelessHART</i>	21
Figura 8 Dispositivos de uma rede <i>WirelessHART</i>	21
Figura 9 Formato do <i>superframe</i> no protocolo <i>WirelessHART</i>	25
Figura 10 Operacionalidade do <i>channel hopping</i> no protocolo <i>WirelessHART</i>	26
Figura 11 Tipos de grafos de roteamento em redes <i>WirelessHART</i>	29
Figura 12 Topologia formada através do roteamento por origem.	30
Figura 13 Representação gráfica do escalonamento de processos.	32
Figura 14 Diagrama de um <i>superframe</i> genérico.	34
Figura 15 Conflitos de <i>timeslot</i> e de <i>canal</i>	35
Figura 16 Topologia genérica de um salto.	36
Figura 17 Representação genérica do escalonamento para a topologia de único salto.	36
Figura 18 Topologia genérica do tipo saltos múltiplos.	37
Figura 19 Representação genérica do escalonamento para a topologia multisalto.	37
Figura 20 Diagrama de blocos do ambiente de avaliação.	43
Figura 21 Formas de visualização dos resultados da execução de um algoritmo de roteamento.	44
Figura 22 Diagrama de blocos do ambiente de avaliação com a integração dos algoritmos de escalonamento.	45
Figura 23 Sequência de execução para gerar o <i>superframe</i>	47
Figura 24 Sequência de evolução do algoritmo de Bellman-Ford (1956).	49
Figura 25 Grafo da topologia da rede e grafo gerado pelo algoritmo BFS.	52
Figura 26 Subgrafo gerado pelo algoritmo ELHFR para o dispositivo H.	52
Figura 27 Topologia de rede de saltos múltiplos.	55
Figura 28 Matriz de escalonamento do algoritmo Han.	56
Figura 29 Grafo da rede visto a partir do dispositivo H.	59
Figura 30 Matriz de escalonamento do algoritmo Dang.	59
Figura 31 Grafo otimizado $\diamond\diamond'$ para o dispositivo H.
62	
Figura 32 Matriz de escalonamento do algoritmo Zhang.	62
Figura 33 Topologia completa da rede.	68
Figura 34 Grafos de <i>uplink</i> gerados pelos algoritmos de roteamento.	68
Figura 35 Comparação da média de número de saltos.	69
Figura 36 Comparação do máximo número de saltos.	70
Figura 37 Comparação da média de número de vizinhos.	71

Figura 38	Comparação do máximo número de vizinhos.	71
Figura 39	Comparação do gradiente de vizinhos: (a) Algoritmo de roteamento Han, (b) Algoritmo de roteamento Bellman-Ford de dupla execução.	72
Figura 40	Comparação da quantidade de dispositivos roteadores.	73
Figura 41	Comparação da quantidade de dispositivos confiáveis.	74
Figura 42	Comparação da quantidade de dispositivos distantes.	75
Figura 43	Comparação da quantidade de links gerados pelos grafos.	76
Figura 44	Comparação gráfica da taxa de escalonamento entre os algoritmos de escalonamento quando utilizado o algoritmo de roteamento Han.	77
Figura 45	Comparação gráfica da taxa de escalonamento entre os algoritmos de escalonamento quando utilizado o algoritmo de roteamento Bellman-Ford de dupla execução.	78
Figura 46	Comparação gráfica da taxa de escalonamento entre os algoritmos de escalonamento quando utilizado o algoritmo de roteamento ELHFR.	78
Figura 47	Comparação gráfica da taxa de escalonamento entre os algoritmos de roteamento quando utilizado o algoritmo de escalonamento Han.	80

LISTA DE TABELAS

Tabela 1 Faixa de frequências utilizada pelo protocolo WH	24
Tabela 2 Configuração dos parâmetros do primeiro experimento.....	67
Tabela 3 Configuração dos parâmetros do segundo experimento	77
Tabela 4 Comparação dos melhores conjuntos para cada algoritmo de roteamento.	80

LISTA DE ABREVIATURAS

ACK: *Acknowledgment*

ASN: *Absolut Slot Number*

BFS: *Breadth-First Search*

C-LLF: *Conflict-Aware Least Laxity First*

DFS: *Depth-First Search*

ELHFR: *Enhanced Least-Hop First Routing*

GW: *Gateway*

HCF: *HART Communication Foundation*

IEC: *International Electrotechnical Commission*

ISA: *International Society of Automation*

ISM: *Industrial, Scientific and Medical*

LLF: *Least Laxity First*

ORMGR: *Optimized Reliable Multipath Graph Routing*

RF: *Radio Frequency*

RSL: *Received Signal Level*

TDMA: *Time Division Multiple Access*

WH: *WirelessHART*

1 INTRODUÇÃO

Recentes avanços na tecnologia de comunicação sem fio e o baixo custo na fabricação de dispositivos sensores levaram ao desenvolvimento das redes de sensores sem fio. Devido à facilidade de implantação e a multifuncionalidade, as redes de sensores sem fio têm sido utilizadas para uma variedade de aplicações, tais como: monitoramento de ambientes, rastreamento de dispositivos, comunicação de processos entre outros. (YICK; MUKHERJEE; GHOSAL, 2008; KUMAR; CHAUHAN, 2011).

Neste contexto, muitos algoritmos de roteamento estão sendo propostos para melhorar o desempenho nas diversas aplicações das redes de sensores sem fio. Diversos algoritmos existentes são baseados na estratégia de encontrar caminhos únicos entre os dispositivos. Esses algoritmos não levam em consideração os efeitos do ambiente no qual se encontram os dispositivos, o volume do tráfego ou as condições adversas. (AL-KARAKI; KAMAL, 2004; AKKAYA; YOUNIS, 2005).

Algoritmos de roteamento que possuem o objetivo de encontrar múltiplos caminhos também são propostos. Segundo (TARIQUE, 2009) esses algoritmos de roteamento apresentam três principais questões a serem resolvidas: como descobrir múltiplos caminhos, como selecionar um caminho e como distribuir o volume do tráfego. Essas questões devem ser levadas em consideração quando for desenvolvida a estratégia de roteamento de um algoritmo.

Nessa mesma linha, diversos algoritmos de escalonamento também são propostos para viabilizar a estratégia de transmissão e recepção das mensagens. O processo de escalonamento tem por objetivo definir quais dispositivos podem transmitir ou escutar no meio de comunicação. A seleção de pacotes mais prioritários, o monitoramento de interferências e a redução da latência fim-a-fim são também metas dos escalonadores.

A usabilidade das redes de sensores sem fio tem sido reconhecida como uma importante tecnologia no campo industrial para controle de processos (WILLIG, 2008). Muitas organizações como ISA, HART e ZigBee tem desenvolvido protocolos de comunicação sem fio para aplicação em automação industrial. Contudo, o monitoramento e controle de processos através de redes sem fio neste ambiente devem atender requisitos rígidos no que se refere à confiabilidade e redundância dos dados (HAN et al., 2011; MÜLLER; NETTO; PEREIRA, 2011).

Frente a esses requisitos, surgiu em 2007 o protocolo *WirelessHART* (WH). Baseado em redes *mesh*, todos os dispositivos de campo são capazes de rotear mensagens (CHEN; NIXON; MOK, 2010). A definição de normas com especificações rígidas relativas à segurança dos dados e o uso de retransmissões por multicaminhos tornam o protocolo atraente para aplicações em ambientes inóspitos. Segundo (MÜLLER, 2012) a robustez do protocolo é o fator determinante para o aumento gradativo das vendas de produtos compatíveis com o WH.

A norma que define a camada de rede do protocolo WH (HCF, 2009) não especifica o algoritmo a ser utilizado para realizar o roteamento das mensagens, porém, define técnicas a serem consideradas, como por exemplo, roteamento por grafos e por fonte (*source*). Do mesmo modo, a norma que define a camada de enlace do protocolo (HCF, 2008) não especifica o algoritmo de escalonamento a ser utilizado. Todavia, define requisitos a serem considerados no momento de desenvolver ou adaptar algoritmos (DICKOW et al., 2013a).

Em uma rede WH os algoritmos de roteamento e escalonamento devem ser capazes de gerar uma rede segura com caminhos alternativos. A rede é centralizada pelo gerenciador da rede que possui dentre muitas funções, executar os algoritmos de roteamento e escalonamento. Apesar de uma rede centralizada depender de um único dispositivo para gerenciar a rede, a centralização da rede proporciona a vantagem da inteligência e robustez se

concentrarem em apenas um dispositivo. Isso torna os dispositivos de campo mais simples, logo, a configuração de hardware é mínima proporcionando baixo consumo de energia e baixo custo financeiro. (HCF, 2007).

1.1 MOTIVAÇÃO

O protocolo WH surge com uma boa opção para comunicação sem fio em ambientes industriais. A falta de algoritmos definidos em norma resulta em muitas propostas de algoritmos de roteamento e de escalonamento por parte da comunidade acadêmica. O problema é que cada proposta é avaliada em cenários específicos. Isso dificulta uma comparação entre os algoritmos apresentados. A aplicabilidade dos algoritmos no protocolo WH também deve ser considerada, pois os requisitos de confiabilidade exigidos não são suportados por algumas propostas.

Esse trabalho propõe adicionar ao ambiente de análise desenvolvido em (KUNZEL, 2012) a integração entre os algoritmos de roteamento e escalonamento de modo a propiciar a análise completa da estratégia de gerenciamento de redes WH. A verificação do melhor escalonador depende do algoritmo de roteamento utilizado na geração dos grafos, pois o escalonamento faz uso dos grafos gerados para realizar a atribuição dos intervalos de tempo. Uma ferramenta que consiga analisar os algoritmos de escalonamento em conjunto com os grafos criados se faz importante, pois, possibilita validar os algoritmos propostos e verificar o comportamento quando executados em conjunto.

1.2 OBJETIVOS

Os principais objetivos desse trabalho são:

- Desenvolver uma extensão no ambiente proposto em (KUNZEL, 2012) com a finalidade de integrar algoritmos de roteamento e escalonamento, permitindo uma análise completa da estratégia de gerenciamento de uma rede WH.

- Comparar os algoritmos de roteamento em conjunto com os algoritmos de escalonamento presentes na literatura e desenvolvidos especificamente para redes WH.
- Verificar os conjuntos aptos a serem aplicáveis em redes WH.

Esta dissertação é dividida da seguinte forma: no capítulo 2 é apresentada a fundamentação teórica sobre grafos, o protocolo WH e os requisitos de roteamento e escalonamento do protocolo WH. O capítulo 3 apresenta uma análise do estado da arte sobre os algoritmos de roteamento e escalonamento propostos na literatura especificamente para o WH. O capítulo 4 apresenta a proposta de integração entre os algoritmos de roteamento e escalonamento e um estudo detalhado das estratégias de cada um dos algoritmos. O capítulo 5 apresenta o planejamento dos experimentos e as métricas utilizadas para avaliação dos algoritmos. O capítulo 6 apresenta a análise dos resultados obtidos. Por fim, o capítulo 7 apresenta as conclusões e propostas de trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta uma conceituação sobre a teoria de grafos amplamente utilizada no desenvolvimento de algoritmos. Um breve estudo sobre o protocolo WH bem como suas estratégias de escalonamento e roteamento também são apresentadas.

2.1 TEORIA DOS GRAFOS

A teoria dos grafos estuda objetos combinatórios para modelar muitos problemas em vários ramos da matemática, informática, engenharia e indústria (FEOFILOFF; KOHAYAKAWA; WAKABAYASHI, 2011). Existe uma grande diversidade de problemas que podem ser modelados através de grafos, como por exemplo, as conexões entre os componentes em circuitos elétricos, os caminhos que as pessoas precisam escolher para chegar a um determinado destino, as conexões de amigos através das redes sociais, entre outros.

Segundo (SEDGWICK; WAYNE, 2011) um grafo G é um conjunto de vértices e uma coleção de arestas que conectam um par de vértices. A representação do grafo G é dada por (V, E) . Os elementos de V são chamados vértices e os elementos de E são chamados de arestas. Uma aresta pode ser representada por $\{v_1, v_2\}$ ou simplesmente v_1v_2 , ou seja, uma aresta é definida como o segmento que interliga dois vértices. Diz-se que os vértices v_1 e v_2 interligados por uma aresta são vértices vizinhos ou adjacentes. A Figura 1 ilustra uma topologia representada através de grafos. São ilustrados os vértices $v_1, v_2, v_3, v_4, v_5, v_6$ e v_7 , e o grafo possui os seguintes elementos: $E = \{v_1v_2, v_2v_3, v_3v_4, v_4v_5, v_5v_6, v_6v_7, v_1v_3, v_2v_4, v_3v_5, v_4v_6, v_5v_7\}$.

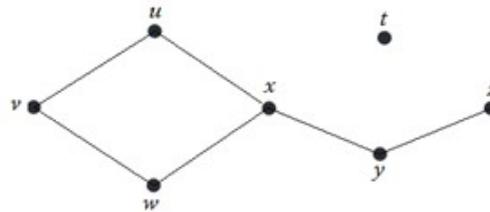
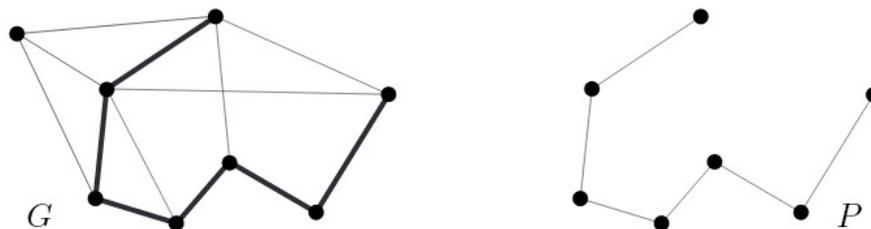


Figura 1 Topologia genérica de um grafo.

O conjunto de vizinhos de um vértice v em G é denotado por $N(v)$ e o grau do vértice é denotado por $d(v)$. O grau é o número de arestas incidentes no vértice e é uma importante métrica utilizada por diversos algoritmos para realizar suas operações. Um vértice de grau igual a zero é dito isolado, ou seja, sem vizinhos. O valor mínimo do grau de um grafo G é denotado por $\delta(G) = \min\{d(v) \mid v \in V\}$ e o valor máximo é denotado por $\Delta(G) = \max\{d(v) \mid v \in V\}$. A Equação 1 apresenta o grau médio de um grafo G , onde $|V|$ é o número de vértices pertencentes ao grafo G (DIESTEL, 2000).

$$\bar{d}(G) = \frac{1}{|V|} \sum_{v \in V} d(v) \tag{1}$$

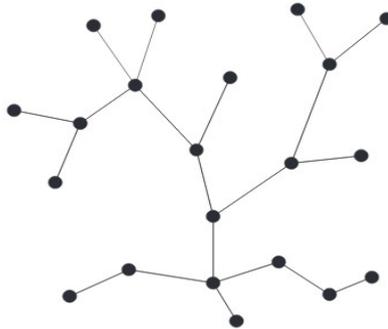
Um caminho P representa o conjunto de arestas e vértices que estão entre os vértices origem e destino de G . O tamanho de um caminho P é a quantidade de arestas pertencentes ao caminho e é denotado por $|P|$. A Figura 2 ilustra um grafo G no qual é destacado um caminho P de tamanho $|P| = 6$.



Fonte: DIESTEL (2000).

Figura 2 Exemplo de um caminho P com tamanho $k = 6$.

Um grafo com caminho único entre cada um dos vértices é chamado de árvore. É conveniente ponderar que um dos vértices pertencentes a uma árvore seja considerado especial. Este vértice em específico é denominado raiz (“*root*”) e é responsável por iniciar a sequência de descoberta de caminhos (também denominado ramos) entre todos os vértices do grafo. A topologia de uma árvore é apresentada na Figura 3.



Fonte: DIESTEL (2002).

Figura 3 Representação de uma árvore.

Dentre os algoritmos mais simples que percorrem uma árvore, destacam-se os algoritmos busca em profundidade (ou busca em profundidade-primeiro, também conhecido em inglês por *Depth-First Search* (DFS)) e busca em largura (ou busca em amplitude, também conhecido em inglês por *Breadth-First Search* (BFS)). Ambos executam suas operações considerando um dos vértices o raiz da topologia.

O algoritmo DFS é utilizado para realizar uma travessia em uma árvore. Intuitivamente, o algoritmo começa em um nó considerado raiz que explora tanto quanto possível um dos seus ramos, antes de retroceder (*backtracking*). O algoritmo encerra quando não existem mais ramos a serem explorados a partir do vértice raiz (SEDGWICK; WAYNE, 2011). A Figura 4 apresenta a ordem de visita em um grafo através do algoritmo DFS. Os valores inseridos internamente ao vértice referem-se à sequência de visita do algoritmo e o vértice em destaque representa o vértice raiz do grafo.

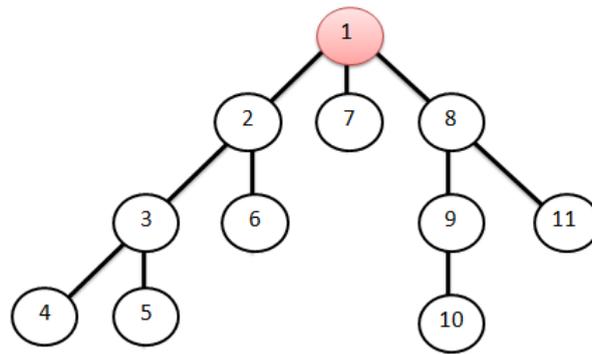


Figura 4 Grafo percorrido pelo algoritmo DFS.

O algoritmo BFS também é utilizado para realizar uma travessia em uma árvore. Entretanto, o vértice raiz explora primeiramente todos os vértices vizinhos. Posteriormente, para cada vértice vizinho, são explorados os vértices vizinhos inexplorados e assim sucessivamente. O algoritmo encerra quando não existem mais vértices vizinhos a serem explorados (SEdgeWICK; WAYNE, 2011). A Figura 5 ilustra a ordem de visita em uma árvore através do algoritmo BFS. Os valores inseridos internamente ao vértice referem-se à sequência de visita do algoritmo e o vértice na cor vermelha representa o vértice raiz da árvore.

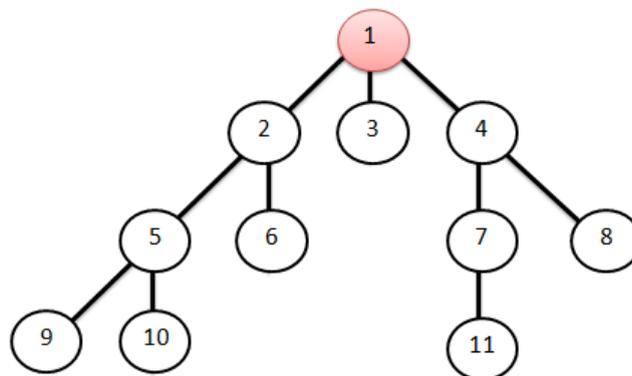


Figura 5 Grafo percorrido pelo algoritmo BFS.

Segundo (GORMEN et al., 2002), o processo de ϵ -criação de um grafo leva em consideração a existência de um grafo orientado ponderado $G = (V, E, w)$. Este tipo de grafo

possui uma direção e um peso w para cada aresta. O peso de um caminho $P = (v_0, v_1, v_2, \dots, v_n)$ é o somatório dos pesos de suas arestas constituintes e é apresentada através da Equação 2.

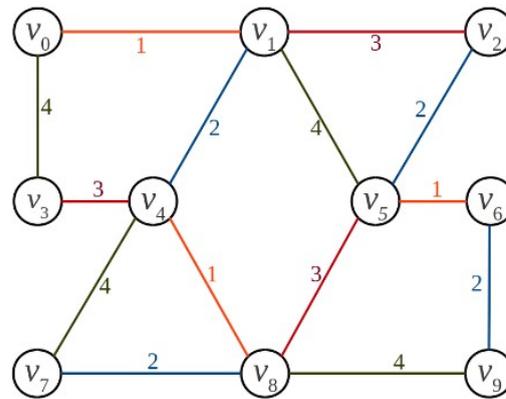
$$w(P) = \sum_{i=1}^n w(v_{i-1}, v_i) \quad (2)$$

O valor do peso do caminho mais curto $\delta_{u,v}$ entre dois vértices u e v é definido como $\delta_{u,v} = \min\{w(P) : P : u \rightarrow v\}$ caso exista tal caminho, ou $\delta_{u,v} = \infty$ caso o caminho não exista. Para cada aresta $(u, v) \in E$ existe a seguinte relação: $\delta_{u,v} \leq \delta_{u,s} + w(s, v)$, onde s representa um ou mais vértices intermediários no grafo.

Baseado nesses conceitos, diversos algoritmos utilizam a técnica chamada relaxamento. Esta técnica consiste em verificar se o caminho $P = (v_0, v_1, v_2, \dots, v_n)$ é um caminho mais curto de v_0 até v_n . Caso sim, as arestas são “relaxadas” na ordem $\mathcal{P} = \{(v_0, v_1), (v_1, v_2), \dots, (v_{n-1}, v_n)\}$. Esta técnica diminui progressivamente a estimativa do peso de um caminho curto de uma origem até cada vértice até encontrar o peso real do caminho mais curto.

Diversos algoritmos também fazem uso da técnica de coloração das arestas. Segundo (FEOFILOFF; KOHAYAKAWA; WAKABAYASHI, 2011) essa técnica consiste em atribuir cores às arestas, de forma que arestas adjacentes recebam cores diferentes. A representação das arestas coloridas de um grafo G é uma partição de E em emparelhamentos. Se $\{E_1, \dots, E_k\}$ é uma partição, é possível afirmar que cada emparelhamento E_i é uma cor e k é o número de cores. O índice cromático de um grafo G é dado por $\chi'(G)$. Tal índice refere-se ao número de cores mínimo necessário para colorir um grafo G . O Teorema de Vizing (VIZING, 1964) definiu que o índice cromático $\chi'(G)$ está limitado entre os valores Δ e $\Delta + 1$, onde Δ é o grau máximo do grafo. A Figura 6 apresenta um exemplo de um problema de organização de

reuniões entre dois em um grupo de dez indivíduos. É utilizada a teoria de grafos em conjunto com os conceitos de coloração de arestas.



Fonte: JANUÁRIO (2011).

Figura 6 Coloração de arestas de um grafo.

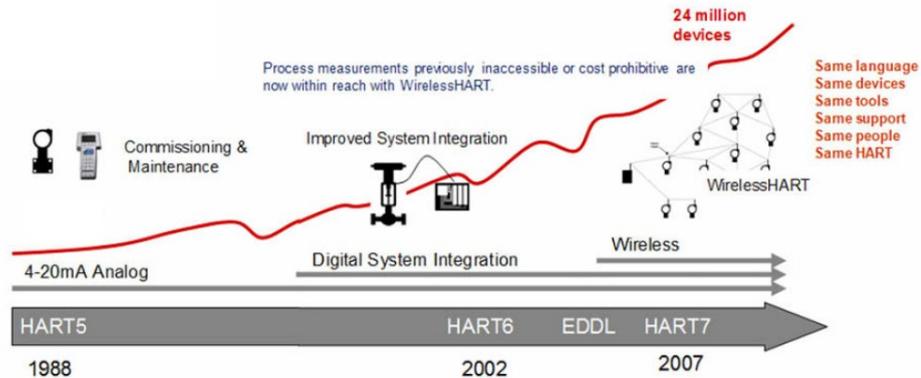
No exemplo da Figura 6, para evitar os conflitos de horários, as reuniões devem ser agendadas em tempos distintos. A coloração de arestas é uma boa estratégia para resolver tal problema. As arestas ou cores representam o horário no qual a reunião ocorrerá e os vértices de cada aresta representam as duas pessoas envolvidas (JANUÁRIO, 2011).

Em uma rede WH, os dispositivos são representados pelos vértices e os *links* pelas arestas. Essa analogia torna os conceitos utilizados na teoria de grafos aplicáveis aos mais diversos algoritmos propostos para o protocolo.

2.2 O PROTOCOLO WIRELESSHART

O monitoramento e controle de processos através de redes sem fio tem se mostrado uma importante tecnologia no ambiente industrial (HAN et al., 2011; MÜLLER et al., 2010). O protocolo WH é o primeiro protocolo de comunicação sem fio aberto aprovado pela *International Electrotechnical Commission* (IEC) para monitoramento e controle de processos (IEC, 2010). O protocolo é uma extensão do protocolo cabeado HART, o que permite a

compatibilidade com sistemas legados. A Figura 7 apresenta a evolução do protocolo desde sua origem.



Fonte: HAN et al., 2011.

Figura 7 Evolução do protocolo *WirelessHART*.

O WH foi desenvolvido com o objetivo de manter um padrão já bem consolidado e disponibilizá-lo na forma sem fio. Com o avanço da tecnologia sem fio, o WH mantém a confiabilidade na transmissão e paralelamente propicia redução nos custos de instalação e protege os investimentos realizados no HART. A Figura 8 ilustra os dispositivos que formam uma rede WH.



Fonte: KUNZEL (2012).

Figura 8 Dispositivos de uma rede *WirelessHART*.

Os dispositivos de campo são os elementos ligados diretamente ao processo que se deseja monitorar ou controlar. São geradores de dados e também capazes de rotear pacotes atuando como simples roteadores. Dispositivos de campo são tanto dispositivos que utilizam a tecnologia sem fio como dispositivos que utilizam a tecnologia cabeada do protocolo HART. Para prover à comunicação entre estas duas tecnologias são utilizados dispositivos adaptadores. O uso de simples roteadores tem por objetivo aumentar o alcance da rede e minimizar a perda de energia de dispositivos à bateria. A energia desperdiçada ocorre devido ao dispositivo ter que rotear muitas mensagens oriundas de dispositivos vizinhos. A colocação de um simples roteador para realizar esta tarefa libera o dispositivo de campo de um alto consumo. O uso de simples roteadores não é estritamente necessário em uma rede WH, uma vez que todos os dispositivos de campo têm capacidade de realizar roteamento (CHEN; NIXON; MOK, 2010).

O dispositivo responsável por prover às aplicações da planta de automação o acesso aos dados dos dispositivos sem fio é o *gateway* (GW). Esse dispositivo é utilizado como a fonte de sincronização de relógio da rede e possui conexão física com o gerenciador da rede. O gerenciador da rede é responsável por criar, controlar, gerenciar e otimizar a rede WH. O gerenciador da rede possui a informação completa da rede, tais como: dispositivos ativos, *links* configurados, *superframes* entre outros parâmetros. Dentre as principais funções do gerenciador estão a execução do escalonamento e roteamento. Os algoritmos destas duas funções são executados no gerenciador e, posteriormente, os dispositivos são programados. (CHEN; NIXON; MOK, 2010).

Para aumentar a confiabilidade da rede ou promover uma maior vazão no fluxo dos dados, múltiplos pontos de acesso podem ser utilizados. Os pontos de acessos devem ser instalados em regiões críticas, como por exemplo, regiões com baixa intensidade do sinal de rádio frequência (RF) e locais onde as variáveis de processo são mais relevantes para o

monitoramento e controle da planta. Os pontos de acessos também propagam o sincronismo da rede. Portanto, à medida que aumenta o número de pontos de acesso, aumenta a responsabilidade do GW em manter estes pontos sincronizados entre si (CHEN; NIXON; MOK, 2010).

Um dispositivo portátil chamado de *Handheld* é necessário para que os operadores da planta de automação possam comissionar os dispositivos de campo. É também utilizado para configurar, controlar, monitorar e realizar a manutenção dos dispositivos de rede. A aplicação deste tipo de dispositivo é um assunto amplo e tem sido alvo de muitas pesquisas que visam monitorar o estado da rede e dos dispositivos.

Em (HCF, 2007) é especificado o padrão a nível elétrico e físico para todos os dispositivos WH. Características tais como o tipo da antena, nível de potência, tensões, taxa de dados, distância máxima de transmissão entre outros parâmetros são definidas. A camada física do protocolo WH é um subconjunto simplificado da norma IEEE 802.15.4 com pequenas modificações e restrições nas definições.

A norma IEEE 802.15.4 define a área da rede operando em um espaço de 10m². O protocolo WH utiliza uma área maior dependendo do ambiente onde se encontram os dispositivos. Em um ambiente interno a norma estima que para uma potência de transmissão igual a 0 dBm, a distância esperada de comunicação é de 35 metros. Já para uma potência de transmissão igual a +10 dBm a distância esperada de comunicação é de 75 metros. Em ambientes externos a distância esperada é mais ampla. Para uma potência de transmissão igual a 0 dBm a distância de comunicação esperada é de 100 metros e para uma potência de transmissão igual a +10 dBm, a distância é de 200 metros.

A norma esclarece que as estimativas citadas acima são baseadas em uma antena omni-direcional, com sensibilidade do receptor igual a -82 dBm e sem interferências. Para ambientes internos é considerado que não há linha de visada direta entre os dispositivos. Já

para ambientes externos é considerada comunicação em campo aberto e com linha de visada direta entre as antenas. A comunicação ocorre na faixa de frequência *Industrial, Scientific and Medical* (ISM) de 2,4 GHz. São utilizados 15 canais nesta banda conforme especificado na Tabela 1.

Tabela 1 Faixa de frequências utilizada pelo protocolo WH

Índice	Canal 802.15.4	Frequência (MHz)
0	11	2405
1	12	2410
2	13	2415
3	14	2420
4	15	2425
5	16	2430
6	17	2435
7	18	2440
8	19	2445
9	20	2450
10	21	2455
11	22	2460
12	23	2465
13	24	2470
14	25	2475
15	26	Não usado

Fonte: Adaptado de HCF (2007).

Ao utilizar 15 canais, o protocolo torna possível a utilização de múltiplos canais no mesmo instante de tempo. Portanto, existe a possibilidade de ocorrer quinze diferentes tipos de comunicação entre diferentes dispositivos. O canal 26 não é utilizado devido à restrição de sua usabilidade em diversos países.

A responsabilidade de detectar e corrigir erros oriundos do nível físico é responsabilidade da cama de enlace. Em (HCF, 2008) são especificadas as características necessárias para manter a confiabilidade na comunicação entre os dispositivos. O protocolo utiliza a técnica *Time Division Multiple Access* (TDMA) para controlar o acesso ao meio. Essa técnica permite que o acesso ao meio ocorra de forma determinística, o que garante uma

transmissão livre de colisões através do uso de *timeslots* (intervalos de tempo pré-definidos onde um dispositivo estará apto a transmitir ou escutar o meio). O conjunto destes *timeslots* forma um *superframe*. O conceito de *superframes* utilizado pelo WH é o agrupamento de vários *timeslots* que se repetem periodicamente ao longo do tempo. O tamanho do *superframe* pode variar e todos os dispositivos da rede devem suportar múltiplos tamanhos, repetidos a um período durante todo o ciclo de vida da rede. A Figura 9 ilustra o conceito de *superframe*.

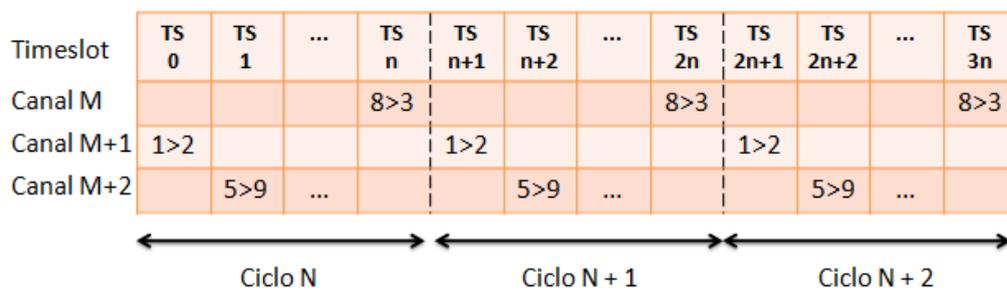


Figura 9 Formato do *superframe* no protocolo *WirelessHART*.

Em um mesmo *timeslot* dois dispositivos são atribuídos a realizar a transmissão e recepção de dados. No primeiro momento, o dispositivo de origem encaminha a mensagem ao dispositivo vizinho e, posteriormente, o dispositivo vizinho responde através de um *acknowledgment* (ACK) o recebimento da mensagem. Essas duas transmissões ocorrem dentro do período de 10ms.

2.2.1.1 Tipos de quadro da camada de enlace

Existem cinco tipos de mensagens no WH que possuem funções específicas. Os quadros do tipo *Data* possuem a informação do processo monitorado a ser transmitida até o dispositivo final. Os quadros do tipo *keep-alive* trocados entre os dispositivos vizinhos possuem informações de manutenção da rede. As mensagens de *keep-alive* têm por objetivos

principais manter a sincronização da rede e confirmar a conectividade entre os dispositivos vizinhos. As mensagens do tipo *advertise* possuem as informações necessárias para prover o ingresso de novos dispositivos na rede. O dispositivo antes de ingressar na rede escuta estas mensagens e faz uso de suas informações para sincronizar com a rede. Os quadros de ACK são transmitidos para confirmar o recebimento correto das mensagens, exceto quando os quadros são de *broadcast* ou do próprio ACK. Os quadros do tipo ACK também carregam informação de ajuste de sincronismo no seu campo de *payload*. Por último, existem os quadros do tipo *Disconnect*, que possuem informações de desconexão e são utilizadas quando um dispositivo deseja sair da rede (HCF, 2008).

O protocolo WH faz uso da técnica *channel hopping* possibilitando ao protocolo a utilização de diversos canais para transmissão em cada *timeslot*. Da mesma forma que o transmissor salta entre canais diferentes para realizar a transmissão, o receptor faz uso do mesmo algoritmo para saltar no mesmo padrão. Esta técnica é executada a cada *timeslot*, ou seja, cada mensagem será propagada por um canal diferente resultando em diversidade em frequência. O salto de canais tem como principal objetivo evitar interferências. A Figura 10 apresenta a operacionalidade da técnica *channel hopping*. A área em destaque refere-se ao canal utilizado para a comunicação.



Figura 10 Operacionalidade do *channel hopping* no protocolo *WirelessHART*.

Os dispositivos utilizam o *channel hopping* de forma sincronizada através da atribuição de diferentes *offsets* de canal para os dispositivos. Os *offsets* são índices utilizados para realizar o cálculo do canal real no qual o dispositivo irá transmitir ou receber seus dados. No cálculo também são considerados o *Absolut Slot Number* (ASN) e o número de canais ativos. O ASN é um valor inteiro inicializado desde que a rede foi originada. O número de canais ativos é a quantidade de canais disponíveis que não constam na *blacklist*. A *blacklist* contém a lista de canais inseridos pelo próprio administrador da rede com o objetivo de evitar que a rede utilize tais frequências devido a interferências detectadas (NOBRE, 2011; KIM, 2008). O salto de canais é baseado nas Equações 3, 4 e 5.

$$\text{Canal_Ativo} = \text{ChannelOffset} + \text{ASN} \% \text{Número_de_canais_ativos} \quad (3)$$

$$\text{Vetor_de_canais_Ativos[]} = \text{conjunto dos canais ativos} \quad (4)$$

$$\text{Canal} = \text{Vetor_de_Canais_Ativos}[\text{Canal_Ativo}] \quad (5)$$

A variável Canal_ativo da Equação 3 retorna um valor utilizado como índice na coleta do canal real no Vetor_de_canais_ativos[]. O Vetor_de_canais_ativos[] da Equação 4 é um vetor que contém todos os canais ativos disponíveis. Posteriormente, a variável Canal da Equação 5 retorna o canal real utilizado para transmitir a mensagem.

2.3 ROTEAMENTO NO PROTOCOLO *WIRELESSHART*

Segundo (TANENBAUM; WETHERALL, 2011) a camada de rede está relacionada a transferências de pacotes da origem para o destino. Chegar ao destino pode exigir vários saltos em roteadores intermediários ao longo do percurso. Portanto, a principal função desta camada é realizar o roteamento de pacotes do dispositivo de origem para o dispositivo de destino. Para isso, algoritmos de roteamento são executados e visam decidir sobre por quais rotas o pacote deve trafegar para atingir seu destino final.

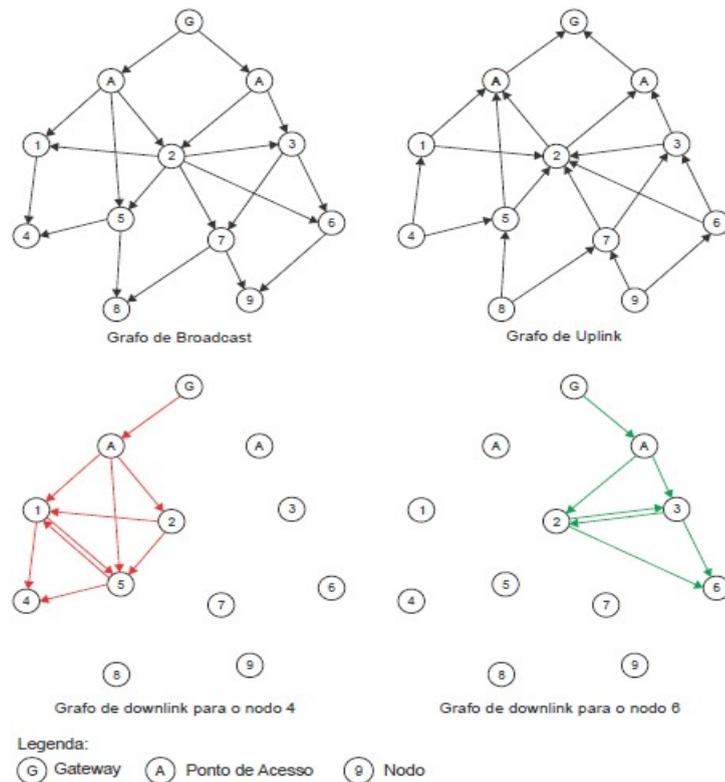
Uma vez a rede formada, espera-se que ela funcione continuamente por um longo período de tempo sem apresentar falhas. Entretanto, durante este período possivelmente ocorrerão problemas, como por exemplo, o desligamento devido à falta de bateria, bloqueios ou interferências devido à coexistência. Frente a estas possíveis falhas, o WH necessita ter robustez na configuração de rotas, para que os dados continuem trafegando por outro caminho pré-determinado. A formação de múltiplos caminhos para comunicação é realizada por algoritmos de roteamento, sendo que, conforme já mencionado anteriormente, a norma do protocolo WH não apresenta nenhum algoritmo a ser utilizado. São especificados apenas os tipos de roteamento suportados pelo protocolo e os requisitos que os algoritmos a serem desenvolvidos necessitam levar em consideração.

Segundo a norma (HCF, 2009) o protocolo WH suporta quatro tipos de roteamento: (i) o roteamento por grafo, (ii) roteamento por origem, (iii) roteamento por *superframes* e (iv) roteamento por *Proxy*. Os quatro tipos de roteamento devem ser suportados por todos os dispositivos da rede WH.

O roteamento por grafos consiste na utilização dos conceitos da teoria dos grafos para representar a topologia de rede. Os vértices representam os dispositivos e as arestas representam os *links* da rede. Os dispositivos intermediários mantêm uma lista de todos os *links* que podem ser utilizados para encaminhar os pacotes ao longo do grafo. Esta lista é criada e propagada pelo gerenciador da rede a todos os dispositivos. A fim de prover a confiabilidade necessária para protocolos de redes industriais, os dispositivos necessitam possuir no mínimo dois vizinhos (CHEN; NIXON; MOK, 2010), de forma a garantir a existência de caminhos redundantes.

Segundo (HAN et al., 2011) existem três tipos de grafos de roteamento definidos no protocolo WH, o grafo de *downlink*, *broadcast* e *uplink*. Grafos de *downlink* representam as rotas para mensagens de dados configuradas na direção do GW aos dispositivos. Grafos de

broadcast representam as rotas para mensagens de configuração comum ou de controle configuradas na direção do GW aos dispositivos. Os grafos de *uplink* são as rotas configuradas na direção dos dispositivos ao GW. As rotas formadas devem ser redundantes e unidirecionais. Exemplo dos tipos de grafos é apresentado na Figura 11.



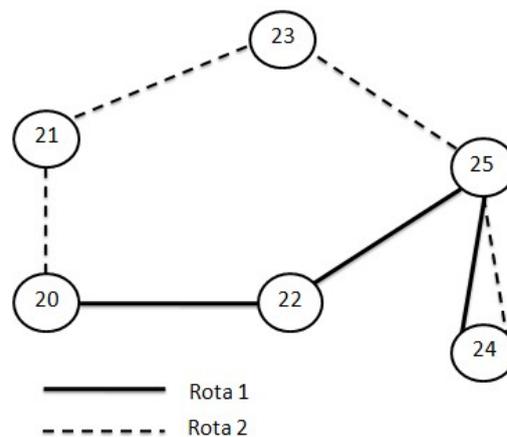
Fonte: KUNZEL (2012).

Figura 11 Tipos de grafos de roteamento em redes *WirelessHART*

Na Figura 11 é apresentado grafo de *broadcast* indicando por onde passará a mensagem do tipo *broadcast* oriundas do GW. O grafo de *uplink* propaga as mensagens de processos e dados com destino o GW. São apresentados dois grafos de *downlink*, um referente ao dispositivo 4 e outro referente ao dispositivo 6.

No roteamento por origem, uma rota é especificada no próprio pacote desde sua origem através de uma lista. Esta lista é inserida em um campo opcional da mensagem e contém a rota salto a salto no qual o pacote precisa trafegar até alcançar o destino final. Ao receber um pacote, cada dispositivo intermediário verifica a lista do endereço da rota. Caso o

endereço de destino não seja o do próprio dispositivo, o pacote será encaminhado para o próximo dispositivo contido na lista. Conseqüentemente, dispositivos intermediários não necessitam conhecer as rotas previamente, necessitando apenas analisar a lista e encaminhar o pacote para o próximo dispositivo especificado. Este tipo de roteamento cria uma rota direta entre um dispositivo de origem e um dispositivo de destino. Portanto, caso ocorra alguma falha em um *link* intermediário o pacote será perdido. A Figura 12 ilustra duas rotas criadas quando o roteamento por origem é utilizado.



Fonte: Adaptado de CHEN; NIXON; MOK (2010).

Figura 12 Topologia formada através do roteamento por origem

A Figura 12 exemplifica duas transmissões que possuem o mesmo dispositivo origem 20 e o mesmo dispositivo de destino 24. Baseado no roteamento por origem, apenas uma rota é criada entre os dispositivos. Para a rota 1, os dados são transmitidos através dos seguintes saltos: 20 > 22 > 25 > 24. Já para a rota 2, a sequência de saltos é 20 > 21 > 23 > 25 > 24. Portanto, este tipo de roteamento não oferece redundância na comunicação o que reduz a confiabilidade em relação ao roteamento por grafos. Como em redes industriais é requisito ter alta confiabilidade na transmissão dos dados, é definido que o roteamento por grafos é mais

adequado em redes WH. O roteamento por origem deve ser utilizado somente em testes e *troubleshooting* na rede (CHEN; NIXON; MOK, 2010).

Uma forma especial de rotear os pacotes através da rede é chamada de roteamento por *superframes*. Nesse tipo de roteamento, um dispositivo propaga uma mensagem de acordo com o identificador do *superframe* construído pelo gerenciador da rede. Todo dispositivo que é associado a qualquer *link* pertencente ao *superframe* precisa receber a informação sobre o *superframe* e o *link*. Portanto, os pacotes são enviados pelo dispositivo através de qualquer *link* disponível pertencente ao *superframe*. Para este tipo de roteamento, os dispositivos intermediários não precisam obter informações sobre as rotas, apenas os *links* com os seus vizinhos do *superframe*.

Existe também o roteamento por *Proxy*, somente utilizado quando um dispositivo deseja ingressar na rede. Através das mensagens do tipo *advertise*, o dispositivo ingressante consegue conectividade com outro dispositivo já pertencente à rede. Este último é denominado *Proxy* e tem a responsabilidade de fazer a comunicação entre o gerenciador de rede e o dispositivo que deseja ingressar na rede.

Apesar de não determinar um algoritmo de roteamento a norma especifica uma série de requisitos que devem ser considerados, tais como:

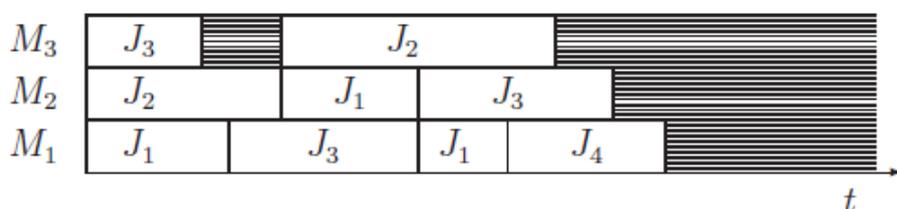
- o gerenciador da rede deve manter uma representação interna da rede contendo todos os dispositivos e *links* presentes;
- no roteamento por grafos devem existir os grafos de *uplink*, *downlink* e *broadcast*;
- se existir um caminho com apenas um salto até o GW, este caminho deverá ser utilizado;
- o número máximo de saltos que um dispositivo pode estar a partir do GW é igual a 4;

- os dispositivos são confiáveis se e somente se possuírem no mínimo dois dispositivos vizinhos dos quais possa receber os dados de forma redundante.

O gerenciador da rede é responsável por manter uma representação lógica de toda a rede. Esta representação é utilizada para executar o roteamento tanto por origem como por grafos. Informações como o nível de intensidade do sinal são utilizadas para ajustar as rotas entre os dispositivos. Não podem ser criadas rotas circulares, ou seja, rotas que saem de um dispositivo de origem não podem retornar novamente ao mesmo dispositivo. Algoritmos de roteamento são executados pelo gerenciador da rede e tem por objetivo construir rotas eficientes de forma a otimizar o tráfego na rede (WINTER, 2013).

2.4 ESCALONAMENTO NO PROTOCOLO *WIRELESSHART*

O escalonamento é uma sequência de execução de processos ou de transmissão de mensagens. Para cada processo ou transmissão, existe uma alocação de um ou mais intervalos de tempo para uma ou mais máquinas (BRUCKER, 2007). Para representar um escalonamento é possível utilizar gráficos do tipo *Gantt* conforme ilustrado na Figura 13, onde \mathcal{M} máquinas \mathcal{M}_i ($i = 1, \dots, \mathcal{M}$) precisam executar \mathcal{J} processos \mathcal{J}_i ($i = 1, \dots, \mathcal{J}$).



Fonte: BRUCKER (2007).

Figura 13 Representação gráfica do escalonamento de processos.

A Figura 13 demonstra que nenhum processo \mathcal{J}_i é executado no mesmo intervalo de tempo por diferentes máquinas. As máquinas \mathcal{M}_i concorrem entre si e executam cada processo

em intervalos de tempo distintos selecionados pelo escalonador. Cada máquina \diamond possui requisitos de tempo para executar determinado processo $\diamond\diamond$. Portanto, existem parâmetros de tempo importantes que compreendem um processo de escalonamento.

Uma importante métrica de avaliação de um escalonador é através do teste de escalonabilidade. Esse teste verifica a taxa que o escalonador consegue executar os processos sem perder os respectivos *deadlines*. O *deadline* se refere ao instante de tempo limite que o processo pode ser finalizado. Este é um dos parâmetros mais importantes e significa que se o processo não for finalizado antes do *deadline*, o processo não terá sido executado como o esperado (HERRTWICH, 1990), (ZAND et al., 2012). A Equação 6 apresenta o cálculo de escalonabilidade de um escalonador. Quanto mais próximo ao valor 1 for a taxa, melhor é o escalonador.

$$\text{Taxa de Escalonabilidade} = \frac{\text{Quantidade de processos sem perda de } \diamond\diamond\diamond\diamond\diamond\diamond\diamond\diamond}{\text{Quantidade de todos os processos}} \quad (6)$$

No protocolo WH o escalonador de *links* tem por objetivo atribuir *timeslots* aos dispositivos para que as comunicações possam ser realizadas de forma correta sem afetar o *deadline* das mensagens. A operação do escalonador é complexa devido a eventos como mudanças de prioridades nos pacotes, modificações de *links* e ativação e desativação de *superframes* (HCF, 2008). O escalonador não depende do tipo de roteamento utilizado pelo protocolo WH, e sim apenas nos links disponibilizados pelo algoritmo de roteamento aptos a serem escalonados.

O escalonamento no protocolo WH é centralizado pelo gerenciador da rede e se ajusta ao longo do tempo baseado nas informações recebidas dos dispositivos (DICKOW et al., 2013b). O algoritmo deve ser executado no gerenciador da rede devido a este possuir o conhecimento total da rede. O escalonador cria os *superframes* baseando-se nos períodos de

publicação de cada dispositivo ativo na rede. A Figura 14 ilustra um exemplo genérico de um *superframe* com período de 250 ms. Nesta ilustração os dispositivos são representados através das letras.



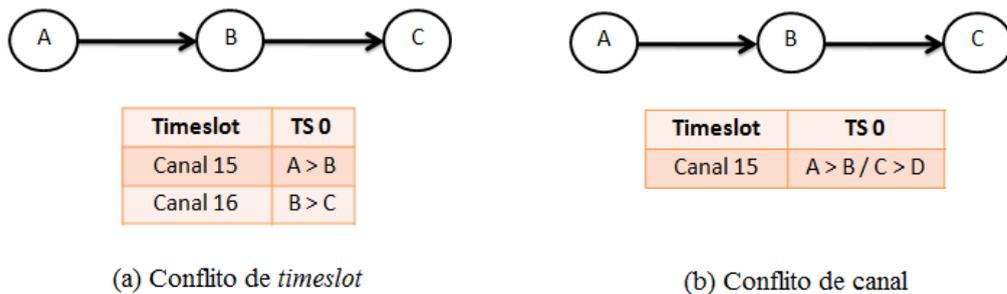
Figura 14 Diagrama de um *superframe* genérico.

O número de saltos até o GW, a quantidade de caminhos redundantes, a latência e o consumo de energia são parâmetros importantes para otimização do escalonamento. Apesar de não determinar um algoritmo de escalonamento, a norma (HCF, 2008) do protocolo especifica uma série de requisitos que devem ser considerados, tais como:

- o gerenciador da rede deve conhecer a período de atualização de cada dispositivo da rede informada pelo próprio dispositivo durante o pedido de serviço;
- para cada transmissão escalonada deverá ser escalonada uma retransmissão por outro caminho, caso este outro caminho exista;
- os períodos de publicação de dados são definidas por $2^{\diamond} - 2 \leq \diamond \leq 9$ segundos;
- o número máximo de canais ativos é 15 e diminui com o ingresso de canais na *blacklist*;

- nenhum dispositivo pode ser escalonado para transmitir ou receber duas vezes no mesmo *timeslot*;
- os dispositivos com o menor número de saltos deverão ser escalonados primeiramente;
- o tamanho do *superframe* é baseado o período de publicação dos dados;
- o destino dos dados dos dispositivos é sempre o GW ou o gerenciador;
- os timeslots são alocados sempre do período de publicação mais rápida para a mais lenta.

Existem dois conflitos que necessitam ser evitados durante a realização do escalonamento em redes WH. O primeiro é o conflito de *timeslots*. Um dispositivo não pode ser escalonado para transmitir ou receber duas vezes no mesmo *timeslot*. Outro conflito a ser evitado deve ser o de canal. Uma comunicação não pode ser escalonada no mesmo canal junto com outra comunicação. A Figura 15 apresenta estes dois conflitos.

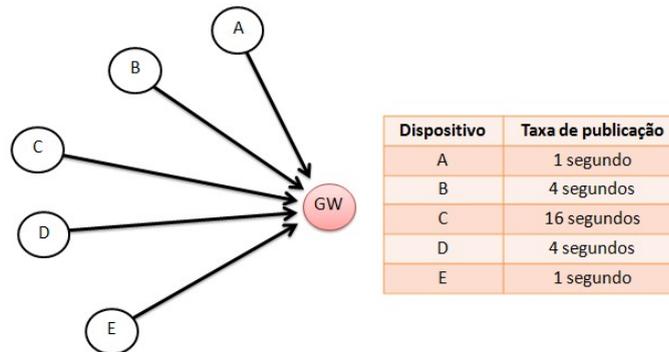


Fonte: Adaptado de DANG et al. (2013).

Figura 15 Conflitos de *timeslot* e de canal.

Baseado no período de publicação dos dados de cada dispositivo, a alocação de *timeslots* realizada pelo escalonamento pode ser executada. A Figura 16 exemplifica uma

topologia de rede formada por dispositivos distante apenas um salto do GW. Posteriormente, a Figura 17 apresenta uma representação genérica de como seria o escalonamento para esta rede.



Fonte: Adaptado de CHEN; NIXON; MOK (2010).

Figura 16 Topologia genérica de um salto.

Superframe 1 (Tamanho = 100 TS)

Timeslot	TS 0	TS 1	TS 2	...	TS 99
Canal 1	A > GW	E > GW			

Superframe 2 (Tamanho = 400 TS)

Timeslot	TS 0	TS 1	TS 2	TS 3	...	TS 399
Canal 1			B > GW	D > GW		

Superframe 3 (Tamanho = 1600 TS)

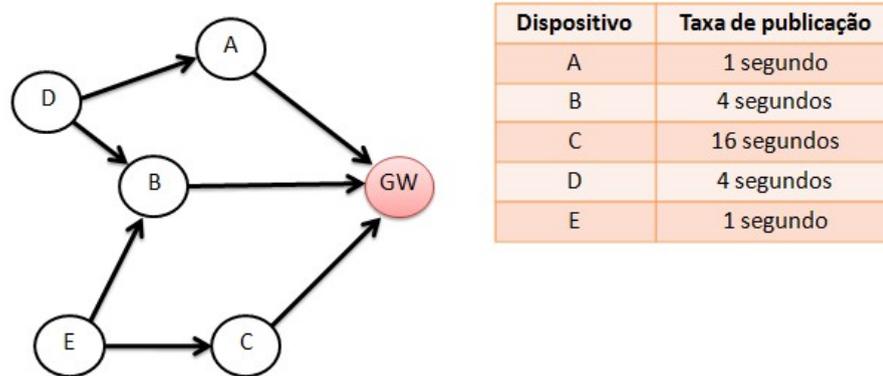
Timeslot	TS 0	TS 1	TS 2	TS 3	TS 4	...	TS 1599
Canal 1					C > GW		

Fonte: Adaptado de CHEN; NIXON; MOK (2010).

Figura 17 Representação genérica do escalonamento para a topologia de único salto.

O escalonamento apresentado na Figura 17 leva em consideração o período de publicação de dados de cada dispositivo. Esse período é importante para que o escalonador possa calcular os timeslots para cada dispositivo sem ocasionar a perda de deadline das mensagens. Os dispositivos A e E possuem período de atualização de 1 segundo e os dispositivos B e D publicam dados a cada 4 segundos. O dispositivo C possui período de atualização de dados de 16 segundos. Portanto, são criados três superframes. É importante analisar que não há sobreposição de timeslot. Cada timeslot é reservado especificamente para um dispositivo, evitando colisões nos próximos ciclos independentemente do superframe.

Todavia, em uma aplicação real haverá vários dispositivos que não possuem conectividade direta com o GW, logo a rede apresentará comunicações em saltos múltiplos. Este tipo de topologia é apresentado na Figura 18 e posteriormente, a Figura 19 apresenta uma representação genérica do escalonamento para esta rede.



Fonte: Adaptado de CHEN; NIXON; MOK (2010).

Figura 18 Topologia genérica do tipo saltos múltiplos.

Superframe 1 (Tamanho = 100 TS)

Timeslot	TS 0	TS 1	TS 2	TS 3	TS 4	TS 5	...	TS 99
Canal 1	A > GW	E > C			A > GW			
Canal 2	E > B							

Superframe 2 (Tamanho = 400 TS)

Timeslot	TS 0	TS 1	TS 2	TS 3	TS 4	TS 5	TS 6	...	TS 399
Canal 1		B > GW	D > B			B > GW	B > GW		
Canal 2		D > A							

Superframe 3 (Tamanho = 1600 TS)

Timeslot	TS 0	TS 1	TS 2	TS 3	TS 4	TS 5	TS 6	TS 7	...	TS 1499
Canal 1				C > GW				C > GW		

Fonte: Adaptado de CHEN; NIXON; MOK (2010).

Figura 19 Representação genérica do escalonamento para a topologia multisalto.

Em uma topologia de saltos múltiplos o escalonamento torna-se mais complexo devido à existência de dispositivos intermediários. Os dispositivos possuem o mesmo período de publicação da topologia apresentada na rede com um único salto ilustrada na Figura 16.

Entretanto, o *superframe* para uma topologia de saltos múltiplos possui mais *timeslots* atribuídos, isto porque os dispositivos intermediários precisam transmitir seus próprios dados e também através de outros *timeslots* realizarem a transmissão dos dados oriundos de dispositivos vizinhos que não possuem conectividade direta com o GW. À medida que a rede cresce na quantidade de dispositivos distantes do GW, a complexidade da construção dos *superframes* aumenta e, portanto, a taxa de escalonabilidade tende a diminuir. Na Figura 19 os *timeslots* TS 4, TS 5, TS 6 e TS 7 são utilizados para o roteamento.

3 ANÁLISE DO ESTADO DA ARTE

A definição do estado da arte relacionada com esta dissertação é realizada através de uma análise nos trabalhos desenvolvidos especificamente para o protocolo WH no campo de roteamento e escalonamento de mensagens.

3.1 ALGORITMOS DE ROTEAMENTO

Comparadas com as redes sem fio tradicionais, as redes sem fio para fins industriais necessitam de alta confiabilidade e robustez para que não ocorram falhas. A norma que especifica o protocolo WH não determina qual algoritmo de roteamento a ser utilizado, porém são apresentados os requisitos necessários para que o protocolo possua confiabilidade e robustez. Frente a isto, trabalhos sobre o tema roteamento têm sido propostos na comunidade científica, cada qual com suas particularidades.

O trabalho proposto em (HAN et al., 2011), tese de doutorado na Universidade do Texas em Austin, aborda o desenvolvimento de algoritmos de roteamento através da construção de grafos de *broadcast*, *downlink* e *uplink*. Os três tipos de grafos são especificados de tal forma que cada dispositivo deva possuir no mínimo dois vizinhos para que seja possível receber dados de forma segura. Os algoritmos possuem como métrica o número de saltos na construção das rotas com o objetivo de reduzir o número de saltos dos dispositivos até o GW da rede. Um escalonador de mensagens também é proposto no trabalho.

Uma proposta para adicionar rotas de forma redundante é apresentada em (MA et al., 2012). É proposto um modelo de roteamento no qual um dispositivo vizinho é atribuído como sendo o roteador principal e outro dispositivo como um roteador secundário. A escolha de qual dispositivo vizinho é o roteador principal baseia-se no valor do nível de sinal recebido dos dispositivos, também conhecido em inglês por *Received Signal Level* (RSL).

O trabalho proposto em (JINDONG; ZHENJUN; YAOPEI, 2009) apresenta um algoritmo de roteamento denominado *Enhanced Least-Hop First Routing* (ELHFR). Tem por objetivo criar uma árvore para cada dispositivo através do algoritmo BFS e, posteriormente, criar rotas redundantes. Ao final, cada dispositivo terá seu próprio subgrafo com diferentes rotas até o GW.

Em (GAO; ZHANG; LI, 2013) é apresentado o algoritmo de roteamento denominado *Optimized Reliable Multipath Graph Routing* (ORMGR). Este algoritmo faz uso do algoritmo ELHFR para gerar os subgrafos e, ao criar as rotas redundantes são levadas em considerações estimativas de qualidade de *link*. Tais estimativas são estatísticas que os dispositivos recebem temporariamente de seus vizinhos e são utilizadas para aceitar ou não que determinado *link* seja utilizado como rota redundante.

Trabalhos no campo do roteamento considerados na literatura como algoritmos clássicos também foram analisados. Os trabalhos publicados em (BELLMAN, 1956) e (FORD, 1956) formam um dos algoritmos mais conhecidos na área de redes, o Bellman-Ford. Este algoritmo visa encontrar um caminho único mínimo em uma árvore com atribuição de pesos nas arestas. A técnica de relaxamento é utilizada para buscar um caminho sempre mais curto baseado na métrica escolhida.

3.2 ALGORITMOS DE ESCALONAMENTO

Um algoritmo de escalonamento não é especificado na norma, mas, assim como no roteamento, os algoritmos de escalonamento propostos devem seguir uma série de requisitos. O assunto tem sido alvo de pesquisas na comunidade científica e cada trabalho possui suas respectivas estratégias.

Os trabalhos apresentados em (ZHANG; SOLDATI; JOHANSSON, 2009a; ZHANG; SOLDATI; JOHANSSON, 2009b; ZHANG; SOLDATI; JOHANSSON, 2011) abordam dois

algoritmos de escalonamento. O primeiro algoritmo é chamado de *ConvergeCast Line One Buffer* e pressupõe que cada dispositivo possui um *buffer* limitado a apenas um pacote, seja ele de transmissão ou recepção. O segundo algoritmo é chamado de *ConvergeCast Multiple Buffer* e considera que os dispositivos podem possuir vários pacotes no *buffer*. Ambos os algoritmos possuem a restrição de serem utilizados apenas em topologias lineares.

O trabalho publicado em (SAIFULLAH et al., 2010) apresenta uma modificação do tradicional algoritmo *Least Laxity First* (LLF). A proposta adiciona uma habilidade para evitar conflitos de transmissões frequentes na tecnologia sem fio. O algoritmo é denominado de *Conflict-Aware Least Laxity First* (C-LLF) e tem o *deadline* da mensagem como parâmetro a ser utilizado como uma variável de decisão quando pacotes disputam o mesmo *timeslot*.

O trabalho de (HAN et al., 2011) aborda um algoritmo de escalonamento onde o *superframe* é dividido igualmente em quatro períodos. O primeiro quarto do *superframe* é destinado ao escalonamento dos grafos de *uplink* para a transmissão dos dados e o segundo quarto é destinado à retransmissão dos dados transmitidos no primeiro quarto do período. O terceiro período é destinado ao escalonamento dos grafos de *downlink* para a transmissão dos dados e o último quarto é destinado à retransmissão dos dados transmitidos no terceiro quarto do período. O algoritmo implementa uma técnica particular para solucionar o escalonamento quando detecta a existência de dois caminhos para se transmitir os dados. Na presença destes dois caminhos, o período de publicação dos dados é reduzido para metade do período original.

Em (DANG et al., 2013) é proposto um algoritmo de escalonamento que utiliza canais simultâneos e múltiplas retransmissões das mensagens. O objetivo é realizar tanto a transmissão como a retransmissão da mensagem através de todos os *links* disponíveis. O algoritmo utiliza uma variável booleana para executar a transmissão duas vezes na mesma rota.

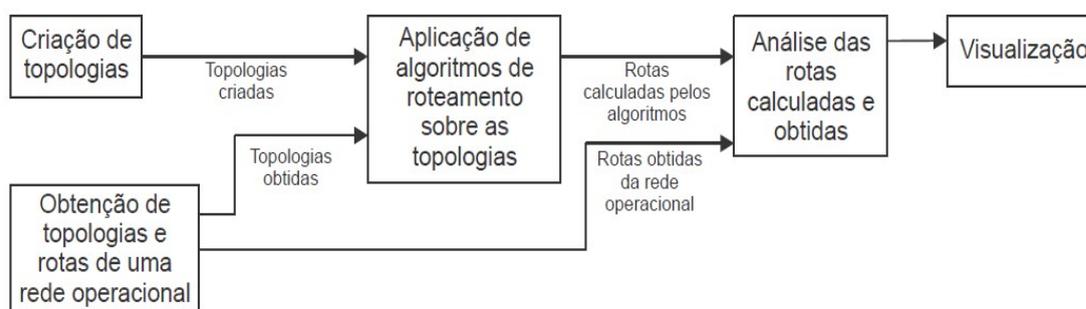
O trabalho apresentado em (ZHANG et al., 2013) faz uso da técnica de coloração de arestas para identificar os *timeslots* disponíveis no escalonamento. A estratégia do algoritmo tem por objetivo verificar a profundidade e a rota ótima de cada dispositivo até o GW. O trabalho apresenta duas transmissões da mesma mensagem através da rota ótima e uma retransmissão por um caminho alternativo.

4 PROPOSTA DE INTEGRAÇÃO ENTRE ALGORITMOS DE ROTEAMENTO E ESCALONAMENTO NO AMBIENTE DE SIMULAÇÃO

Este capítulo apresenta a descrição do ambiente de simulação usado como base para este trabalho o que permite o estudo e a análise dos algoritmos de roteamento. A integração de uma extensão nesse ambiente é proposta para que seja possível analisar os algoritmos de roteamento e escalonamento em conjunto tornando o ambiente mais completo. Um estudo sobre os algoritmos desenvolvidos no ambiente também é apresentado.

4.1 AMBIENTE PARA AVALIAÇÃO DOS ALGORITMOS DE ROTEAMENTO

O ambiente para avaliação dos algoritmos de roteamento foi desenvolvido no âmbito de uma dissertação de mestrado do PPGEE e é apresentado em (KUNZEL, 2012). Este ambiente possibilita o desenvolvimento de diferentes algoritmos de roteamento para redes WH e permite a análise das rotas construídas. Segundo (KUNZEL, 2012) o ambiente é dividido em cinco blocos principais: (i) criação de topologias para aplicação em algoritmos de roteamento; (ii) obtenção de topologias e rotas através de captura de pacotes de uma rede operacional; (iii) aplicação de algoritmos de roteamento sobre as topologias construídas e obtidas; (iv) análise das rotas construídas e obtidas e (v) visualização das análises. A representação dos blocos principais do ambiente é ilustrada na Figura 20.



Fonte: KUNZEL (2012).

Figura 20 Diagrama de blocos do ambiente de avaliação.

O ambiente de análise permite que o usuário crie uma topologia, baseada no número de dispositivos, ou também que as topologias sejam criadas a partir da monitoração de uma rede WH em uso. Para o segundo caso é necessário um equipamento de captura de tráfego de mensagens. Esse equipamento realiza a gravação dos dados em um arquivo e posteriormente o ambiente lê o arquivo e gera a topologia e grafos da rede (HAN et al., 2009).

O resultado da execução do algoritmo de roteamento em uma rede é apresentado em forma de estatísticas, histogramas e visualização de grafos de *uplink* e *downlink*. Na Figura 21 são apresentadas as três formas citadas anteriormente:

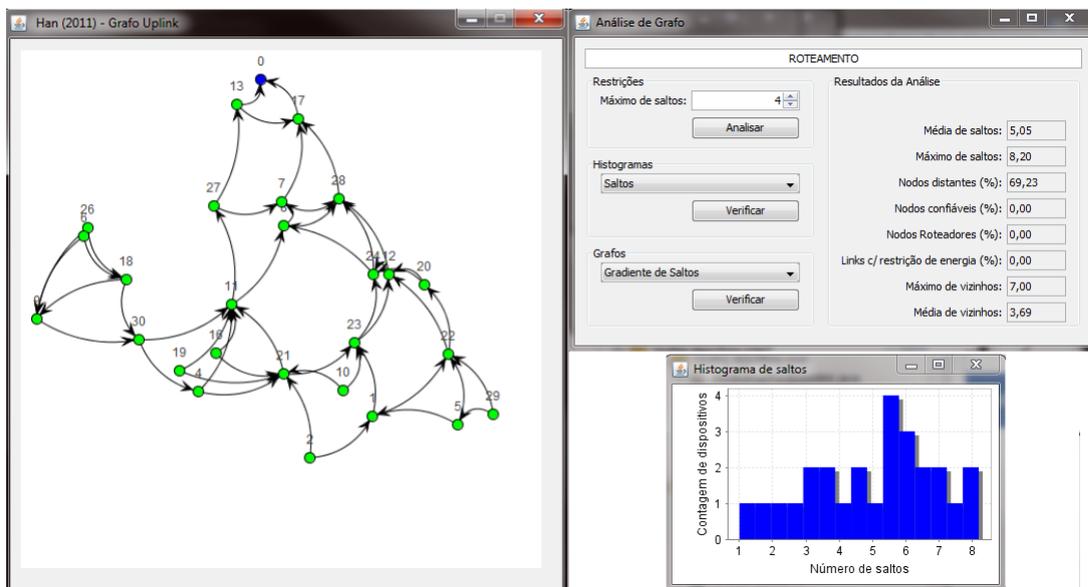


Figura 21 Formas de visualização dos resultados da execução de um algoritmo de roteamento.

4.2 AMBIENTE PARA AVALIAÇÃO DOS ALGORITMOS DE ESCALONAMENTO

Ao ambiente proposto em (KUNZEL, 2012) foi integrada uma extensão para que algoritmos de escalonamento também pudessem ser analisados. A partir dos grafos de roteamento já gerados pelo ambiente, algoritmos de escalonamento podem ser executados possibilitando a integração entre os algoritmos. A integração permite uma análise mais

profunda na operabilidade de uma rede WH frente aos algoritmos utilizados. A Figura 22 apresenta a nova sequência de operação do ambiente.

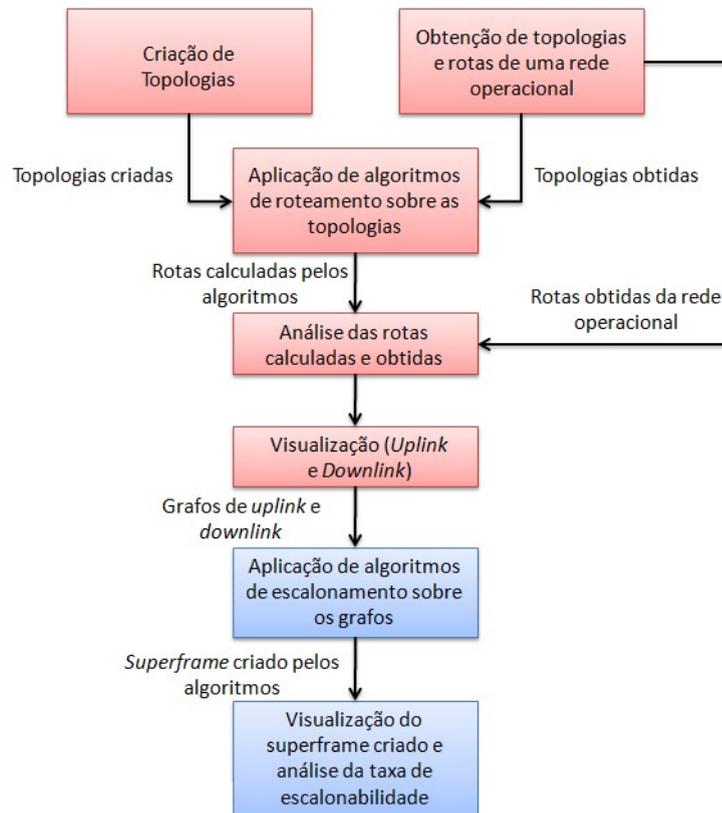


Figura 22 Diagrama de blocos do ambiente de avaliação com a integração dos algoritmos de escalonamento.

A sequência básica de execução do ambiente de análise possui em cor vermelha as funções originais do ambiente e em cor azul as novas funções desenvolvidas. Para que um algoritmo de escalonamento seja analisado, obrigatoriamente um algoritmo de roteamento deve ser previamente executado.

4.2.1 Desenvolvimento da estrutura de programação para o escalonamento

Novas classes e funções foram desenvolvidas na linguagem Java. O código do ambiente possui muitos pacotes de forma organizada, facilitando o entendimento e a

programação para futuras extensões. Mantendo a organização foram criados dois novos pacotes: *scheduling.algorithms* e *scheduling.tasks*. O pacote *scheduling.algorithms* contém os algoritmos de escalonamento desenvolvidos, já o pacote *scheduling.tasks* possui as tarefas que os algoritmos de escalonamento compartilham.

Todos os algoritmos de escalonamento desenvolvidos no pacote *scheduling.algorithms* recebem como parâmetros os grafos de *uplink* e *downlink* de um dos algoritmos de roteamento executado previamente. A partir das informações sobre os dispositivos e *links* contidos nos grafos é que o algoritmo de escalonamento pode ser executado possibilitando a criação do respectivo *superframe*. Tarefas compartilhadas como o processo de salto de canais, criação do *superframe*, período de publicação de dados e canais presentes na *blacklist* estão presentes no pacote *scheduling.tasks*.

4.2.2 Novas interfaces gráficas do ambiente

Novas interfaces foram criadas para realizar a configuração do escalonamento. É possível no ambiente escolher o período de publicação dos dispositivos. Porém, o ambiente é limitado à escolha de apenas um único período de publicação que será aplicada em todos os dispositivos do grafo. Dentre os períodos de publicação disponíveis, estão: 250ms, 500ms, 1s, 2s, 4s, 8s, 16s, 32s, 64s, 128s, 256s e 512s.

Além disso, também é possível adicionar canais à *blacklist*. Tal opção restringe o escalonador de utilizar os canais marcados. Por fim é gerado um *superframe* com os dispositivos escalonados nos seus respectivos canais e *timeslots*. A taxa de escalonamento também é apresentada informando a eficiência do algoritmo. A Figura 23 ilustra a sequência de janelas para geração do *superframe* dos algoritmos de escalonamento. Verifica-se na Figura 23 que os canais 12, 16 e 24 foram adicionados a *blacklist*, devido a esta razão, estes canais não são apresentados no *superframe* gerado.

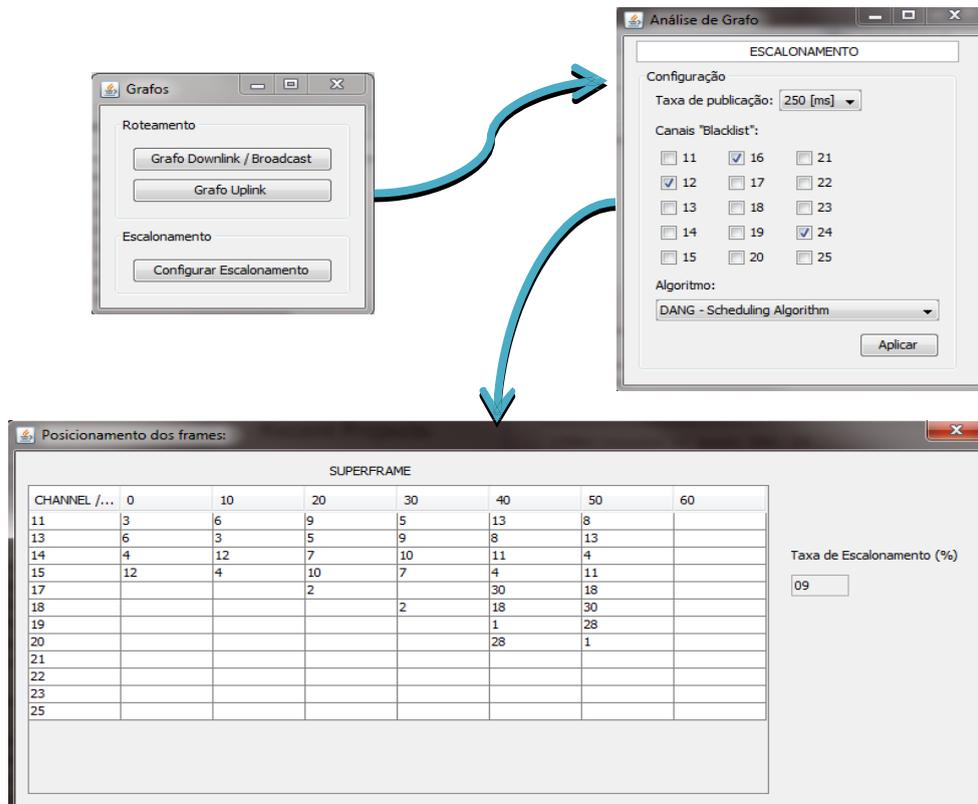


Figura 23 Sequência de execução para gerar o *superframe*.

4.3 ALGORITMOS DESENVOLVIDOS NO AMBIENTE DE ANÁLISE

Esta seção apresenta a descrição e o estudo dos algoritmos desenvolvidos no ambiente de simulação. Foram desenvolvidos três algoritmos de roteamento: (i) roteamento Bellman-Ford de dupla execução; (ii) roteamento Han proposto em (HAN et al., 2011); e (iii) ELHFR proposto em (JINDONG; ZHENJUN; YAOPEI., 2009). Outros três algoritmos de escalonamento foram desenvolvidos: (i) escalonamento Han proposto em (HAN et al., 2011); (ii) escalonamento Dang proposto em (DANG et al., 2013); e (iii) escalonamento Zhang proposto em (ZHANG et al., 2013).

4.3.1 Algoritmo de roteamento Bellman-Ford de dupla execução

O algoritmo de Bellman-Ford é a integração dos trabalhos descritos em (BELLMAN, 1956) e em (FORD, 1956). O objetivo é encontrar um caminho mínimo em uma árvore com atribuição de pesos nas arestas, inclusive pesos com valores negativos. O algoritmo soluciona o problema de caminhos mais curtos em um grafo orientado ponderado $G = (V, E, w)$. Durante a execução do algoritmo, um valor booleano é retornado apontando a existência ou não de um ciclo com pesos negativos. O pseudocódigo é apresentado no Algoritmo 1. A variável v é considerada o vértice raiz, ou seja, o vértice no qual inicia a execução do algoritmo.

Algoritmo 1 Roteamento Bellman-Ford (1956)

Entrada: Grafo da topologia completa $G = (V, E, w)$
Saída: Grafo de Downlink $G' = (V, E', w')$

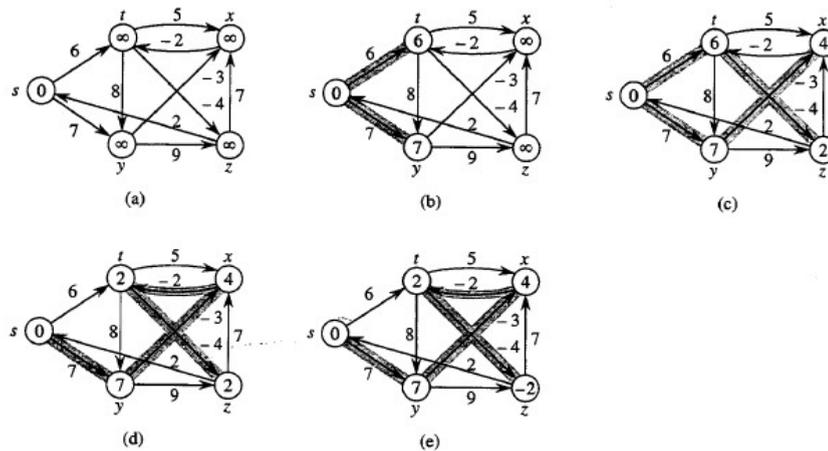
```

1   $\delta_{v,v} = 0$   $\delta_{v,v} = \infty$ 
2   $\delta_{v,v} = \infty$   $\delta_{v,v} \neq \infty$ 
3   $\forall u \in V$   $\delta_{v,u} = 1$  até  $\delta_{v,u} = V$ 
4  |  $\forall$  cada aresta  $(u, v) \in E$ 
5  |   relaxamento  $(u, v, w)$  em  $\delta_{v,u}$ 
6  |    $\delta'_{v,u} = \text{adiciona}(\delta_{v,u}, w(u, v))$ 
7  |  $\delta_{v,u} = \delta'_{v,u}$ 
8  |  $\delta_{v,u} = \delta_{v,u}$ 
9  |  $\forall$  cada aresta  $(u, v) \in E$ 
10 |  $\delta_{v,u} > \delta_{v,u} + w(u, v)$ 
11 |   retorna FALSO
12 |  $\delta_{v,u} = \delta_{v,u}$ 
13 |   retorna VERDADEIRO
14 |  $\delta_{v,u} = \delta_{v,u}$ 
15  $\delta_{v,u} = \delta_{v,u}$ 

```

No início do algoritmo é atribuído o peso $\delta_{v,v} = 0$ caso o vértice origem seja o próprio destino, ou seja, $v = v$. Se o caminho real de v até v ainda não é conhecido, então o peso é considerado infinito, ou seja, $\delta_{v,v} = \infty$. Um loop é criado com o objetivo de visitar todos os vértices do grafo na ordem de $|V| - 1$. Para cada visita, a técnica de relaxamento é executada em todas as arestas do vértice. A técnica de relaxamento é utilizada para diminuir progressivamente a estimativa do peso de um caminho mais curto da origem até cada

vértice $v \in V$, visando encontrar o peso real $\delta_{s,v}$ do caminho mais curto. Após visitar todos os vértices do grafo G , a existência de um ciclo de peso negativo é verificada. O algoritmo que seja VERDADEIRO se e somente se G não possui um ciclo de peso negativo que seja acessível a partir da origem s . A Figura 24 ilustra a sequência de operações do algoritmo de Bellman-Ford.



Fonte: CORMEN, (2002).

Figura 24 Sequência de evolução do algoritmo de Bellman-Ford (1956).

Para o vértice v da Figura 24, inicialmente, são executadas as linhas 1 e 2 do Algoritmo 1, ou seja, é atribuído peso igual a zero para o vértice v e peso igual a infinito aos vértices não conhecidos. Essa representação é ilustrada na Figura 24 (a). Na sequência de ilustrações (b), (c), (d) e (e), as linhas 3 a 8 do Algoritmo 1 são executadas. Cada vértice v é visitado uma vez e suas arestas são relaxadas a fim de encontrar o caminho de peso mínimo. Por fim, da linha 9 a 14 do Algoritmo 1 é verificada a existência de um ciclo de peso negativo.

Como a norma do protocolo WH exige que os dispositivos tenham no mínimo dois vizinhos a fim de prover redundância, o algoritmo de Bellman-Ford não pode ser utilizado em redes WH práticas. Portanto, o uso deste algoritmo bem como outros algoritmos clássicos como (DIJKSTRA, 1959; JOHNSON, 1977) e Floyd-Warshall originado das propostas (FLOYD, 1962) e (WARSHALL, 1962) que seguem a mesma estratégia de encontrar

caminhos únicos para cada dispositivo, também não são utilizáveis. Em contrapartida é possível utilizar esses algoritmos clássicos através de execuções sucessivas, evitando a reutilização das rotas já encontradas nas execuções anteriores (DICKOW et. al., 2013a).

Quando as execuções sucessivas ocorrem, os algoritmos geram rotas redundantes tornando-se bons algoritmos a serem estudados. Frente a esta consideração, é proposto o algoritmo de roteamento Bellman-Ford de dupla execução com modificação simples em relação à estratégia original. O algoritmo 2 apresentado a seguir, é uma implementação do algoritmo de Bellman-Ford sendo executado na sua essência por duas vezes consecutivas.

Algoritmo 2 Roteamento BellmanFord de dupla execução

Entrada: Grafo da topologia completa $\mathcal{G}(\mathcal{V}, \mathcal{E})$

Saída: Grafo de Downlink $\mathcal{G}'(\mathcal{V}', \mathcal{E}')$

```

1   $\delta_{\mathcal{S}, \mathcal{S}} = 0$   $\delta_{\mathcal{S}, \mathcal{D}} = \infty$ 
2   $\delta_{\mathcal{S}, \mathcal{D}} = \infty$   $\delta_{\mathcal{D}, \mathcal{D}} = 0$ 
3   $\mathcal{Q} = \mathcal{S}$  até  $\mathcal{Q} = \mathcal{D}$ 
4   $\mathcal{Q} = \mathcal{S}$  até  $\mathcal{Q} = \mathcal{V}$ 
5  para cada aresta  $(\mathcal{U}, \mathcal{V}) \in \mathcal{E}[\mathcal{Q}]$ 
6  para cada  $\mathcal{W} \in \mathcal{V} \setminus \mathcal{Q}$ 
7  relaxamento  $(\mathcal{U}, \mathcal{V}, \mathcal{W})$  em  $\mathcal{Q}$ 
8   $\mathcal{Q} = \text{adiciona}(\mathcal{Q}, \mathcal{W})$ 
9
10  $\mathcal{Q} = \mathcal{D}$  até  $\mathcal{Q} = \mathcal{V}$ 
11 para cada aresta  $(\mathcal{U}, \mathcal{V}) \in \mathcal{E}[\mathcal{Q}]$ 
12 para cada  $\mathcal{W} \in \mathcal{V} \setminus \mathcal{Q}$ 
13 se  $\delta_{\mathcal{U}, \mathcal{W}} > \delta_{\mathcal{U}, \mathcal{V}} + c_{\mathcal{U}, \mathcal{V}, \mathcal{W}}$ 
14 então FALSO
15  $\delta_{\mathcal{U}, \mathcal{W}} = \delta_{\mathcal{U}, \mathcal{V}} + c_{\mathcal{U}, \mathcal{V}, \mathcal{W}}$ 
16 então VERDADEIRO
17  $\mathcal{Q} = \text{adiciona}(\mathcal{Q}, \mathcal{W})$ 
18  $\mathcal{Q} = \mathcal{D}$  até  $\mathcal{Q} = \mathcal{V}$ 
19  $\mathcal{G}' = \text{constrói}(\mathcal{G}, \mathcal{Q})$ 

```

A primeira execução do algoritmo encontra o caminho de custo mínimo entre cada dispositivo, ou seja, é a execução na forma original. Os caminhos encontrados são salvos em uma fila \mathcal{Q} . O algoritmo é executado novamente evitando o uso dos caminhos já encontrados durante a primeira execução. Portanto, para cada dispositivo são encontrados os dois caminhos de menor custo.

4.3.2 Algoritmo de roteamento ELHFR

O algoritmo proposto em (JINDONG; ZHENJUN; YAOPEI, 2009) faz uso do algoritmo BFS para gerar um grafo de caminho único entre todos os dispositivos da rede. O GW é considerado o dispositivo raiz e gera a árvore $V_{\diamond\diamond\diamond\diamond}$ com os demais dispositivos. Nessa árvore $V_{\diamond\diamond\diamond\diamond}$, cada dispositivo da rede terá um único caminho P até o GW. O caminho P é baseado no menor número de saltos $\diamond(\diamond)$ do dispositivo \diamond até o GW gerado pelo BFS.

A confiabilidade é um requisito do protocolo WAVE para um caminho. Para resolver este problema, o algoritmo ELHFR tem como ideia principal a construção de subgrafos $\diamond\diamond, \diamond$ para cada dispositivo da árvore $V_{\diamond\diamond\diamond\diamond}$. A construção dos subgrafos para cada dispositivo é baseada na premissa de:

- se existir um *link* possível de se criar entre o dispositivo origem \diamond e o dispositivo destino \diamond , este somente será criado se, e somente, se o número de saltos $\diamond(\diamond)$ do dispositivo destino for menor em um salto do que o número de saltos $\diamond(\diamond)$ do dispositivo origem.

O pseudocódigo é apresentado no algoritmo 3, sendo possível verificar a condição para criação de *links* na linha 4.

Algoritmo 3 Roteamento ELHFR

Entrada: Grafo da topologia completa $\diamond\diamond, \diamond$

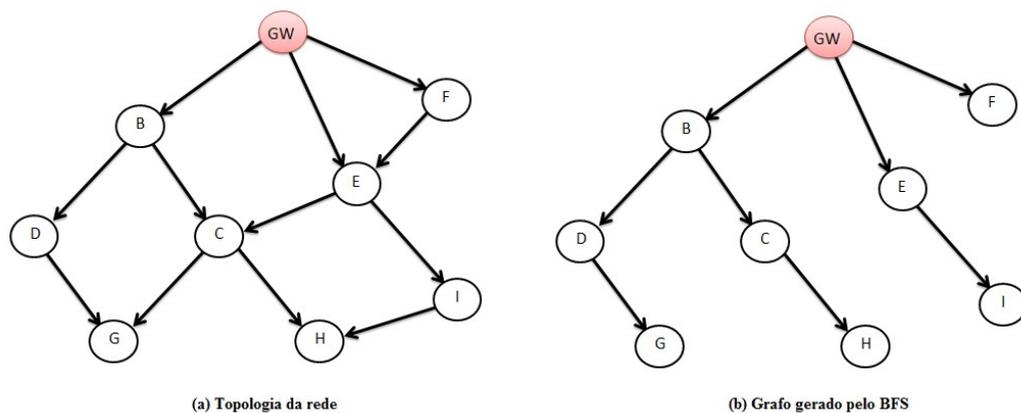
Saída: Grafo de Downlink $\diamond'(\diamond', \diamond')$

```

1   $V_{\diamond\diamond\diamond\diamond}$  = Grafo de caminho único entre todos os dispositivos através do BFS
2   $\diamond\diamond\diamond\diamond$ a  $\diamond\diamond\diamond$  dispositivo  $\diamond \in V_{\diamond\diamond\diamond\diamond}$ 
3  |  $\diamond\diamond\diamond\diamond$ a  $\diamond\diamond\diamond$ a aresta  $\diamond, \diamond$ ,  $\diamond\diamond\diamond\diamond \diamond$  = vizinhos de  $\diamond$ 
4  |  $\diamond\diamond\diamond \diamond = \diamond \diamond - 1$ 
5  | | cria a conexão  $\diamond\diamond\diamond(\diamond, \diamond)$ 
6  |  $\diamond\diamond\diamond \diamond$ 
7  |  $\diamond\diamond\diamond \diamond\diamond\diamond$ a
8  |  $\diamond\diamond\diamond \diamond\diamond\diamond$ a

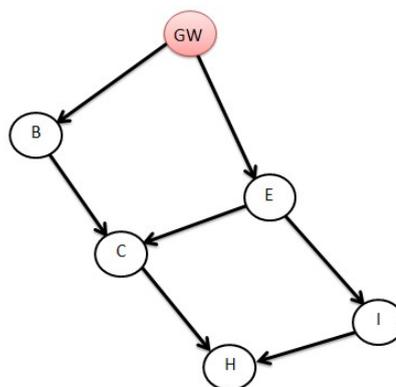
```

Nos subgrafos gerados, todos os caminhos oriundos do dispositivo \diamond até o GW possuem a mesma quantidade de saltos. Isto porque, o próximo salto de cada dispositivo ou é o dispositivo pai (dispositivos conectados através do BFS) ou é o dispositivo tio, formado pelas redundâncias de caminhos. Tanto o dispositivo pai como o dispositivo tio possuem o mesmo número de saltos até o GW. Logo, qualquer caminho no subgrafo a partir do dispositivo \diamond até o GW é o caminho com o menor número de saltos. Com redundância de caminhos através da criação de subgrafos, o ELHFR torna-se um algoritmo de roteamento aplicável ao protocolo WH. A Figura 25 demonstra um exemplo da operação do algoritmo ELHFR. A topologia completa da rede e a topologia do grafo gerado pelo BFS são apresentadas. Já na Figura 26 é ilustrado o subgrafo gerado para dispositivo H.



Fonte: Adaptado de JINDONG; ZHENJUN; YAOPEI (2009).

Figura 25 Grafo da topologia da rede e grafo gerado pelo algoritmo BFS.



Fonte: Adaptado de JINDONG; ZHENJUN; YAOPEI (2009).

Figura 26 Subgrafo gerado pelo algoritmo ELHFR para o dispositivo H.

4.3.3 Algoritmo de roteamento Han

O algoritmo proposto em (HAN et al., 2011) considera que um dispositivo \diamond somente é confiável se, e somente se, possuir ao menos dois vizinhos dos quais possa receber dados. A estratégia tem por objetivo maximizar o número de dispositivos confiáveis e minimizar o número médio de saltos \diamond de cada dispositivo em relação ao GW. O algoritmo faz uso do conjunto V_\diamond para armazenar os dispositivos já analisados pelo algoritmo. Um conjunto \diamond também é criado para armazenar os links já analisados. A cada iteração, o dispositivo analisado é adicionado ao V_\diamond . A execução do algoritmo finaliza quando o conjunto V_\diamond for igual ao conjunto V que corresponde a todos os dispositivos da rede. O algoritmo 4 apresenta o pseudocódigo.

Algoritmo 4 Roteamento Han

Entrada: Grafo da topologia completa $\diamond, \diamond, \diamond$

Saída: Grafo de broadcast $\diamond(V_\diamond, \diamond)$

```

1  Inicie  $V_\diamond = \diamond \cup V_{\diamond_P}$   $\diamond$  com as arestas de  $\diamond$  em direção a  $V_{\diamond_P}$ 
2   $\diamond \neq V$ 
3  Encontre  $\diamond' \subseteq V - V_\diamond: \forall \diamond \in \diamond', \diamond$  tenha duas ou mais arestas provindas de  $V_\diamond$  em  $\diamond$ 
4   $\diamond' \neq \emptyset$ 
5   $a \in \diamond'$ 
6  Classifique as arestas de  $\diamond$  de acordo com o menor número de saltos  $\diamond$  até o GW
7  Escolha as arestas  $\diamond_{1,\diamond}$  e  $\diamond_{2,\diamond}$  com os menores números de saltos  $\diamond$ 
8   $\diamond = \frac{\diamond_{1,\diamond} + \diamond_{2,\diamond}}{2} + 1$ 
9   $a$ 
10 Identifique o nodo  $\diamond$  com o menor custo  $\diamond$  em  $\diamond'$ 
11  $V_\diamond = V_\diamond \cup \diamond$  e  $\diamond = \diamond \cup \diamond_{1,\diamond} \cup \diamond_{2,\diamond}$ 
12  $\diamond$ 
13 Encontre  $\diamond'' \subseteq V - V_\diamond: \forall \diamond \in \diamond'', \diamond$  tenha uma aresta provinda de  $V_\diamond$  em  $\diamond$ 
14  $\diamond'' \neq \emptyset$ 
15  $a \in \diamond''$ 
16  $\diamond = \diamond_{1,\diamond} + 1$ 
17 Calcule  $\diamond$ , o # de arestas que vão em direção a  $V - V_\diamond$ 
18  $a$ 
19 Identifique o dispositivo  $\diamond$  com maior # de arestas em  $\diamond''$ 
20 senão
21 Retorne  $\diamond$  que possui o grafo parcial, sem os dispositivos desconectados
22  $\diamond$ 
23  $\diamond$ 
24  $\diamond$ 

```

algoritmo DFS. Logo, apenas dois caminhos são criados e não todos os caminhos possíveis de um dispositivo até o GW. O pseudocódigo do algoritmo é apresentando a seguir:

Algoritmo 5 Escalonamento Han

Entrada: Grafo gerado pelo roteamento: $\diamond \leftarrow \diamond \diamond, \diamond$
Saída: Matriz de escalonamento: Canais Slots

```

1   $\diamond \diamond \diamond \diamond$  a todo dispositivo  $\diamond \in \diamond \diamond \diamond, \diamond$ 
2   $\diamond \diamond \diamond \diamond$  a todo dispositivo  $\diamond \in \diamond \diamond \diamond \diamond \diamond \diamond \diamond \diamond (\diamond) \diamond \diamond \diamond \diamond \diamond$ 
3   $\diamond \diamond \diamond$  é o único sucessor de  $\diamond \diamond \diamond \diamond \tilde{a} \diamond$ 
4  Alocar o  $\diamond \diamond \diamond \diamond \diamond, \diamond \diamond \diamond$ 
5   $\diamond \diamond \diamond$ 
6   $\diamond \diamond \diamond \tilde{a} \diamond$ 
7   $\diamond \diamond \diamond$  é o primeiro sucessor de  $\diamond \diamond \diamond \diamond \tilde{a} \diamond$ 
8  Alocar o  $\diamond \diamond \diamond \diamond \diamond \diamond \diamond \diamond \diamond'$ 
9   $\diamond \diamond \diamond'$ 
10  $\diamond \diamond \diamond \tilde{a} \diamond$ 
11 Alocar o  $\diamond \diamond \diamond \diamond \diamond \diamond + \diamond, \diamond \diamond \diamond'$ 
12  $\diamond \diamond (\diamond \diamond + \diamond \diamond) \diamond \diamond \diamond'$ 
13  $\diamond \diamond \diamond \diamond \diamond \diamond$ 
14  $\diamond \diamond \diamond \diamond \diamond \diamond$ 
15  $\diamond \diamond \diamond \diamond \diamond \diamond \diamond \diamond$  a
16  $\diamond \diamond \diamond \diamond \diamond \diamond \diamond \diamond$  a
    
```

A operação do algoritmo de escalonamento é exemplificada a seguir. A Figura 27 apresenta uma topologia de rede a ser escalonada. A matriz de escalonamento da topologia é ilustrada na Figura 28. O *superframe* \diamond gerado é um *superframe* de tamanho genérico igual a 10 *timeslots* para exemplificar o funcionamento do algoritmo. Portanto, o *superframe* \diamond' terá um tamanho igual a 20 *timeslots*.

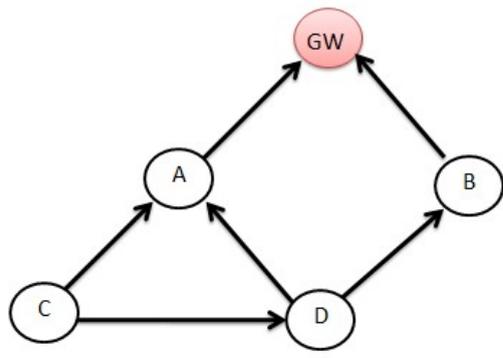
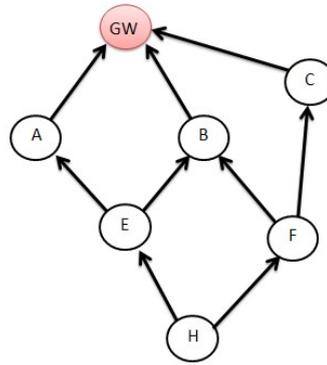


Figura 27 Topologia de rede de saltos múltiplos.

temporária denominada *LastSlot* para cada dispositivo da rede. Esta variável tem seu valor atualizado cada vez que um dispositivo é escalonado. Portanto, ela armazena sempre o último valor do *timeslot* para o qual o dispositivo foi atribuído. Durante o processo de escalonamento, o primeiro *timeslot* disponível após o *LastSlot* será sempre selecionado.

Uma matriz de escalonamento [*Canais*][*Slots*] é utilizada para armazenar os dispositivos escalonados nos seus respectivos canais e *timeslots*. Durante o processo de escalonamento esta matriz é percorrida a fim de localizar o primeiro *timeslot* disponível, tal que, tanto o dispositivo origem, como o dispositivo destino, não estejam atribuídos a este. Da mesma forma, a matriz de escalonamento é percorrida para localizar um canal livre em um dado *timeslot*. Se todos os canais estiverem ocupados no *timeslot* \diamond deverá ser realizada uma nova verificação de canais no *timeslot* $\diamond + 1$ e assim sucessivamente.

Para um dispositivo origem, uma transmissão redundante se faz necessária para prover a confiabilidade na rede. O algoritmo utiliza a variável booleana denominada *First* para tal estratégia. A variável é VERDADEIRA para um *timeslot* inicial e FALSA para *timeslots* de retransmissão. Outra variável denominada *FirstChannel* é utilizada para armazenar o canal que foi atribuído o *timeslot* inicial. Posteriormente, o algoritmo utiliza este valor como uma variável de decisão para que o uso do mesmo canal seja evitado na retransmissão pelo próprio dispositivo. O pseudocódigo da estratégia de escalonamento proposto em (DANG et al., 2013) é apresentado no algoritmo 6.



Fonte: Adaptado de DANG et al. (2013).

Figura 29 Grafo da rede visto a partir do dispositivo H.

Timeslot	TS 0	TS 1	TS 2	TS 3	TS 4	TS 5	TS 6	TS 7	TS 8	TS 9
Canal 1	H > E		E > A		E > B	F > C		C > GW		B > GW
Canal 2		H > E	H > F	E > A	F > C	E > B	F > B	F > B		
Canal 3				H > F	A > GW	A > GW	C > GW		B > GW	

Fonte: Adaptado de DANG et al. (2013).

Figura 30 Matriz de escalonamento do algoritmo Dang.

Através do *superframe* gerado é possível verificar que não há conflitos de *timeslots* e de canais. No TS4 são escalonadas três comunicações simultaneamente, porém, não há interferência entre as mesmas. *Timeslots* redundantes são atribuídos agregando retransmissão para todas as comunicações existentes.

4.3.6 Algoritmo de escalonamento Zhang

O algoritmo proposto em (ZHANG et al., 2013) faz uso da teoria de coloração das arestas para executar o escalonamento. As arestas representam os *links* e as cores correspondem aos *timeslot*. A sequência e o número da cor são importantes para incrementar o desempenho na rede. A estratégia de coloração das arestas leva em consideração os seguintes princípios:

- para evitar conflitos, *links* que se conectam com o GW precisam ser coloridos com diferentes cores. Isto evita que dispositivos tentem transmitir no mesmo instante de tempo para o GW. Já *links* que não possuem como destino o GW, podem ser coloridos com a mesma cor, entretanto, com atribuição de canais diferentes;
- todos os *links* do grafo da rede precisam ser coloridos com pelo menos uma cor para proporcionar a redundância nos dados;
- *links* pertencentes à rota ótima têm prioridades na coloração e possuem a distinção de serem coloridos por duas cores diferentes e em canais diferentes.

A estratégia de escalonamento consiste em gerar um subgrafo G' para o dispositivo sob análise. A profundidade d do dispositivo em relação ao GW é verificada. Esta profundidade tem por objetivo ser uma variável a ser decrementada a cada iteração do algoritmo. Ao chegar ao valor igual à zero, outro dispositivo será analisado. Para cada dispositivo é verificado uma rota ótima única até o GW. A rota ótima é baseada através da verificação dos dispositivos que possuem o menor número de saltos até o GW. Uma variável de decisão para a seleção de dois dispositivos com o mesmo número de saltos é o RSL.

A coloração dos *links* é realizada na direção *uplink*, ou seja, dos dispositivos para o GW. A cor utilizada é representada por um número inteiro e os *links* pertencentes à rota ótima são coloridos duas vezes devido à existência da retransmissão através da rota principal. O algoritmo atribui ao conjunto L todos os *links* l do grafo otimizado G' . Os *links* l de um dispositivo d são representados pelo conjunto L_d . Uma vez o algoritmo executado, o número que representa a cor será acrescido em um e os *links* coloridos serão removidos do conjunto L , exceto o *link* da ótima rota caso este não tenha ainda sido colorido. A atribuição dos canais

ocorre de forma aleatória. O pseudocódigo da estratégia de escalonamento é apresentado pelo algoritmo 7:

Algoritmo 7 Escalonamento Zhang

Entrada: Grafo gerado pelo roteamento: $\mathcal{G} \leftarrow \mathcal{G}, \mathcal{G}$

Saída: Matriz de escalonamento: Canais Slots

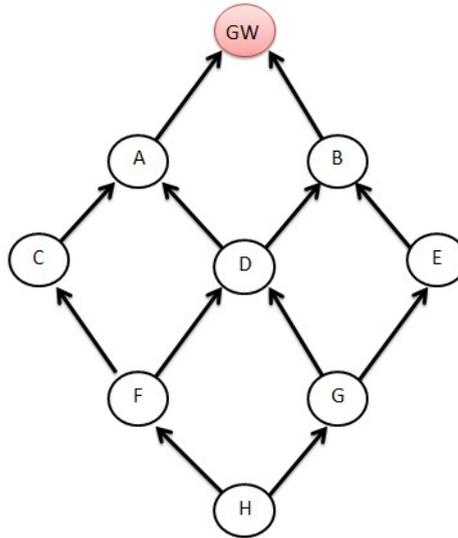
```

1   $\forall a \in \mathcal{D}$  dispositivo  $a \in \mathcal{G}$ 
2  Gere um grafo otimizado  $\mathcal{G}'$  para  $a$ 
3  Calcule a profundidade  $d_a$  de cada dispositivo do grafo  $\mathcal{G}'$ 
4  Inicializa o número serial da cor  $c_a = 0$ 
5  Selecione a melhor rota  $\mathcal{R}_a$  entre o dispositivo  $a$  e o GW
6   $d_a(\mathcal{R}_a) = 0$ 
7  Sair do algoritmo referente ao grafo otimizado  $\mathcal{G}'$ .
8   $\forall a \in \mathcal{D}$ 
9  Adicionar os links  $\mathcal{L}_a$  do dispositivo  $a$  em  $\mathcal{G}'$ 
10  $\forall \mathcal{L}_a \in \mathcal{G}' \neq 0$ 
11 se  $\mathcal{L}_a \in \mathcal{R}_a$  &&  $d_a = 0$ 
12    $c_a = c_a + 1$ 
13    $d_a = d_a - c_a$ 
14    $\forall \mathcal{L}_a \in \mathcal{G}'$ 
15    $c_a = c_a - 1$ 
16    $d_a = d_a - c_a + 1$ 
17    $c_a = c_a - 1$ 
18    $\forall \mathcal{L}_a \in \mathcal{G}'$ 
19   Colorir cada link  $\mathcal{L}_a$  em  $\mathcal{G}'$  utilizando o número serial da cor  $c_a$ 
20   Atribuir diferentes canais para os link  $\mathcal{L}_a$  em  $\mathcal{G}'$  randomicamente
21   Atribuir ao link  $\mathcal{L}_a \in \mathcal{R}_a$  um canal diferente do último caso  $d_a \neq 0$ 
22    $c_a + 1$ 
23    $\forall \mathcal{L}_a \in \mathcal{G}'$ 
24    $c_a = c_a + 1$ 
25  $\forall a \in \mathcal{D}$ 

```

A atribuição de mesmo *timeslot* para diferentes *links* não ocasiona conflitos. Portanto, o algoritmo faz uso de mais de um canal por *timeslot*, recurso escasso no *superframe*. Os *links* pertencentes à rota principal possuem prioridade no escalonamento frente aos demais *links*. Para evitar que muitas retransmissões consumam grande parte da banda disponível do *superframe*, as retransmissões são somente alocadas para os *links* pertencentes à rota principal. Os demais *links* possuem apenas uma comunicação com seu vizinho.

Para exemplificar a operação do algoritmo, a Figura 31 apresenta o grafo otimizado \mathcal{G}' para o dispositivo H possuindo apenas um ponto de acesso. Todos os *links* do grafo \mathcal{G}' são adicionados ao conjunto \mathcal{L} . Já a Figura 32 apresenta a matriz de escalonamento gerado para o grafo otimizado \mathcal{G}' .



Fonte: Adaptado de ZHANG et al. (2013).

Figura 31 Grafo otimizado \mathcal{G}' para o dispositivo H.

Timeslot	TS 0	TS 1	TS 2	TS 3	TS 4	TS 5	TS 6	TS 7	TS 8	TS 9
Canal 1	H > G		G > E		G > D	D > A	C > A		B > GW	
Canal 2		H > G	H > F	G > E	F > C	E > B	D > B			A > GW
Canal 3				F > D	E > B			B > GW		

Fonte: Adaptado de ZHANG et al. (2013).

Figura 32 Matriz de escalonamento do algoritmo Zhang.

As comunicações escalonadas na Figura 32 representam apenas as mensagens que o dispositivo H encaminha até o GW. Os escalonamentos intermediários referem-se ao roteamento das mensagens provindas do dispositivo H.

5 PLANEJAMENTO DAS ANÁLISES E DEFINIÇÃO DAS MÉTRICAS PARA AVALIAÇÃO DOS ALGORITMOS

Esta sessão descreve os procedimentos realizados para a execução dos experimentos utilizando o ambiente de análise. Métricas também são apresentadas para avaliação dos algoritmos de roteamento e escalonamento.

5.1.1 Primeiro experimento

O primeiro experimento envolveu a análise dos algoritmos de roteamento desenvolvidos no ambiente de análise. Os algoritmos de roteamento foram submetidos a uma variação na quantidade de dispositivos na rede entre 50, 100 e 150 dispositivos. Parâmetros como a média de saltos, dispositivos roteadores, dispositivos confiáveis, entre outros, foram analisados para cada algoritmo. Devido ao ambiente de análise gerar aleatoriamente a topologia da rede através de uma distribuição uniforme, os algoritmos foram submetidos a dez topologias diferentes. Os resultados apresentados posteriormente são a média das dez execuções.

5.1.2 Segundo experimento

O segundo experimento envolveu a análise dos algoritmos de escalonamento desenvolvidos no ambiente de análise. A análise dos algoritmos de escalonamento é executada somente após a obtenção dos grafos de *downlink* e *uplink*, ou seja, um algoritmo de roteamento é necessário para que seja possível escalonar determinado grafo. Portanto, este experimento consiste em verificar a taxa de escalonamento de cada algoritmo de escalonamento utilizando cada algoritmo de roteamento. O período de publicação dos dados também é variada, influenciando no tamanho do *superframe*. Um *superframe* de tamanho maior possibilita o escalonamento de uma maior quantidade de dispositivos.

Os algoritmos de escalonamento foram todos desenvolvidos levando em consideração o processo de escalonamento descrito em (HAN et al., 2011). O processo define dois tipos de *superframes*: *superframe* de dados e *superframe* de gerenciamento. O *superframe* de dados é utilizado para suportar transmissões de dados dos processos monitorados entre os dispositivos e o GW, enquanto o *superframe* de gerenciamento é utilizado para suportar a troca de mensagens de gerenciamento do protocolo. O número de *superframes* ativos na rede é determinado pelo número de diferentes períodos de publicação utilizado pelos dispositivos. No ambiente de análise, todos os dispositivos publicam com o mesmo período, portanto, sempre apenas um *superframe* é gerado. O algoritmo do processo utilizado por todos os escalonadores desenvolvidos é apresentado no pseudocódigo a seguir.

Algoritmo 8 Processo de escalonamento básico

```

1  ♦♦♦♦a ♦♦♦♦ ♦♦♦♦♦♦
2  | Gere um superframe de dados ♦♦
3  | ♦♦♦♦a ♦♦♦♦ dispositivo ∈ ♦♦, ♦♦♦♦♦♦
4  | // Fase de detecção
5  | Escalonar os links de uplink de 0 até 0,25♦♦
6  | Escalonar os links de uplink de 0,25♦♦, até 0,5♦♦
7  | // Fase de controle
8  | Escalonar os links de downlink de 0,5♦♦, até 0,75♦♦
9  | Escalonar os links de downlink de 0,75♦♦, até ♦♦
10 | ♦♦♦♦ ♦♦♦♦a
11 | ♦♦♦♦ ♦♦♦♦a

```

O algoritmo possui duas fases, a fase de detecção e a de controle. A fase de detecção aloca os *links* principais e de retransmissão ao longo do grafo de *uplink*. No primeiro quarto do *superframe* são alocados todos os *links* possíveis e a comunicação é do tipo exclusiva, ou seja, não haverá competição pelo *timeslot* ♦ com demais transmissões. No segundo quarto do *superframe* são alocados todos os *links* possíveis e a comunicação é do tipo compartilhada, ou seja, é possível que várias transmissões compitam entre si para serem transmitidas no *timeslot* ♦. Segundo (HAN et al., 2011), até cinco transmissões podem competir por um *timeslot* do tipo

compartilhado. A fase de controle realiza a mesma estratégia, entretanto, o escalonamento é realizado na direção do GW aos dispositivos. Nessa fase, dados de gerenciamento são transmitidos do GW para os demais dispositivos.

Para este experimento, apenas o primeiro quarto do *superframe* disponível é utilizado, ou seja, apenas a linha 5 do algoritmo 8 é executada, as linhas 6, 8 e 9 não foram programadas. Portanto, os *links* são escalonados de forma exclusiva e o tamanho do *superframe* fica limitado a um quarto do tamanho original do *superframe* na rede.

5.1.3 Métricas para avaliação e comparação dos grafos de roteamento e das matrizes de escalonamento

As métricas propostas para avaliação dos grafos de roteamento são as mesmas definidas em (KUNZEL, 2012) adicionada da quantidade de links de um grafo. Tal parâmetro é importante para realizar posteriormente uma análise com as matrizes de escalonamento.

- Média de saltos: corresponde a média de saltos de cada dispositivo no grafo em relação ao GW. Este parâmetro indica a distância média da rede e o consumo de energia por mensagem.
- Máximo de saltos: corresponde ao maior número de saltos entre todos os dispositivos do grafo em relação ao GW.
- Porcentagem de dispositivos distantes: corresponde à porcentagem dos dispositivos que estão a uma distância maior em número de saltos do que o definido pelo protocolo. O protocolo WH define como sendo quatro o valor máximo de saltos.
- Porcentagem de dispositivos confiáveis: corresponde à porcentagem dos dispositivos que possuem ao menos dois vizinhos pelos quais seja possível transmitir suas mensagens.

- Porcentagem de dispositivos roteadores: corresponde à porcentagem dos dispositivos que recebem mensagens oriundas de dispositivos vizinhos e precisa encaminhá-las em direção ao dispositivo destino.
- Número máximo de vizinhos: corresponde ao valor máximo de links de um dos dispositivos do grafo. O dispositivo com a maior quantidade de links terá que processar muitos pacotes, necessitando ser um dispositivo mais confiável e robusto.
- Média de vizinhos: corresponde ao valor médio de vizinhos de uma rede.
- Quantidade de *links*: corresponde à quantidade de *links* correspondente ao grafo gerado.

As métricas propostas para avaliação das matrizes de escalonamento são baseadas na possibilidade de realizar o escalonamento para determinado dispositivo ou não. Caso um dispositivo tenha um período de publicação de 1 segundo, e esta mensagem não consiga chegar até o GW dentro do *superframe* de 1 segundo, esta mensagem será considerada não escalonável. Os tempos de processamento dos algoritmos não foram computados, porém, caso os algoritmos sejam implementados em um gerenciador real, este será um teste válido e importante. A taxa de escalonamento para os algoritmos implementados utiliza exatamente a Equação 7 apresentada a seguir e baseia-se na Equação 6 já apresentada.

$$\text{Taxa de Escalonabilidade (\%)} = \frac{\text{SchedPositive} * 100}{(\text{SchedPositive} + \text{SchedNegative})} \quad (7)$$

Os parâmetros *SchedPositive* e *SchedNegative* são variáveis do tipo inteiro. A cada dispositivo escalonado a variável *SchedPositive* é incrementada em um e a cada dispositivo não escalonado a variável *SchedNegative* é incrementada em um.

6 ANÁLISE E RESULTADOS

Este capítulo descreve os resultados obtidos nos experimentos propostos. São apresentadas análises referentes a cada comparação realizada entre os algoritmos de roteamento e escalonamento. Por fim, uma análise nos melhores conjuntos aptos a serem utilizados pelo protocolo WH é discutida baseando-se nos resultados encontrados dos experimentos propostos.

6.1 PRIMEIRO EXPERIMENTO - ANÁLISE DOS ALGORITMOS DE ROTEAMENTO

Neste experimento, os grafos gerados pelos algoritmos de roteamento foram analisados no ambiente. As dimensões da área e o modelo de conectividade são os mesmos utilizados nos trabalhos de (HAN et al., 2011) e (KUNZEL, 2012). A Tabela 2 apresenta as configurações dos parâmetros da rede para o primeiro experimento.

Tabela 2 Configuração dos parâmetros do primeiro experimento

Dimensões da área	450 x 450 m
Alcance máximo de comunicação	100 m
Número de dispositivos	50, 100, 150
Pontos de Acesso	1

A Figura 33 ilustra o exemplo de uma topologia completa gerada para a análise dos algoritmos. Posteriormente, a Figura 34 apresenta os grafos de *uplink* gerados pelos algoritmos de roteamento Han, Bellman-Ford de dupla execução e ELHFR. Nos grafos apresentados, o ponto de acesso é representado na cor azul e os dispositivos de campo na cor verde.

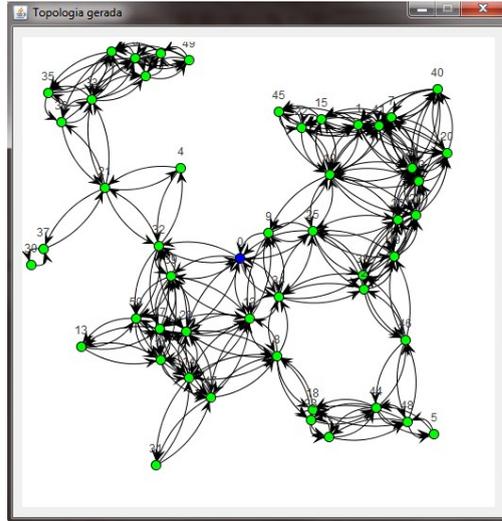


Figura 33 Topologia completa da rede.

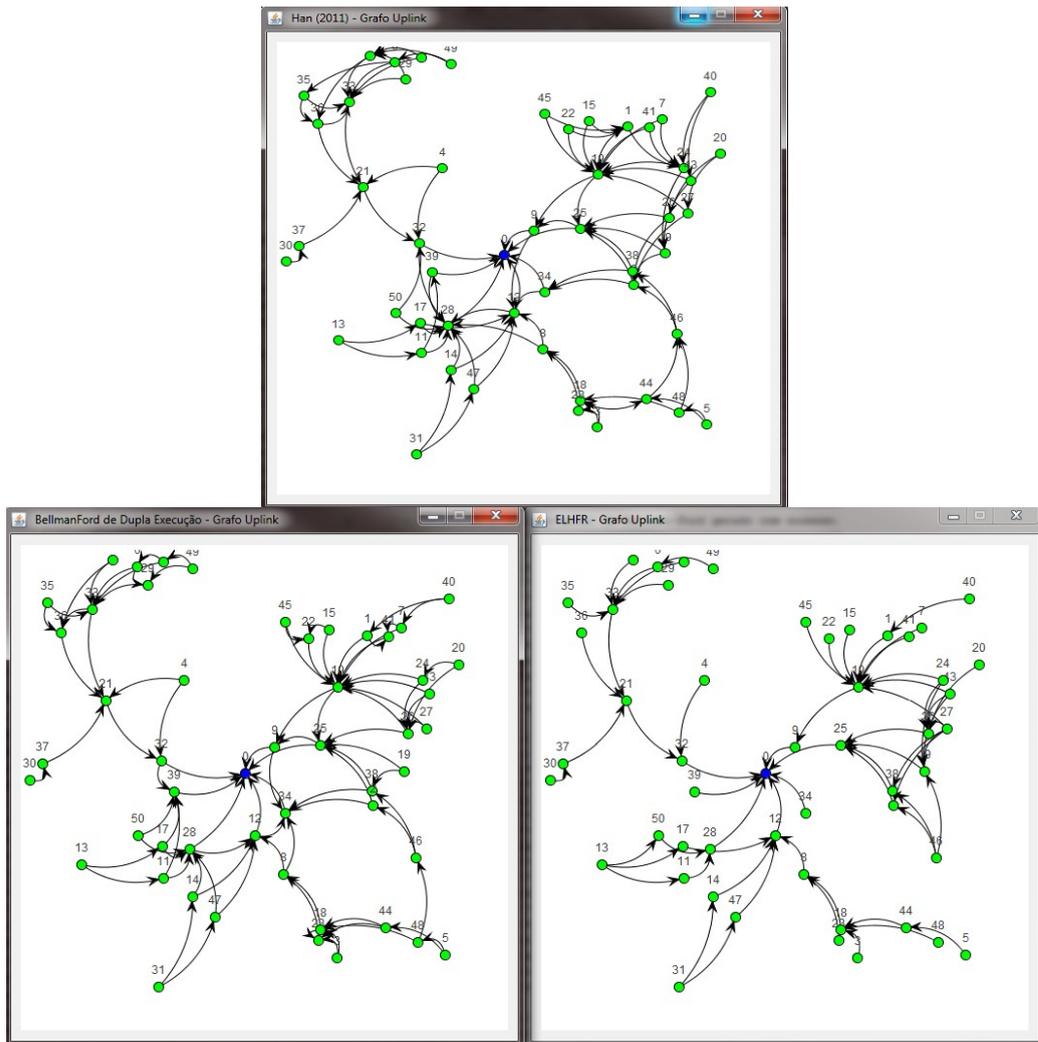


Figura 34 Grafos de *uplink* gerados pelos algoritmos de roteamento.

O número de saltos representa uma importante métrica de análise de algoritmos de roteamento. Um menor número de saltos normalmente leva a uma diminuição na latência da entrega final da mensagem e a um menor consumo de energia. Todavia, aumenta a distância entre os dispositivos ocasionando uma redução no RSL. O resultado é um aumento da probabilidade de ocorrer erros nos dados. (MISHRA; MISHRA, 2011). Frente a essas considerações, afirma-se que redes com grande número de saltos podem ser prejudicadas no desempenho devido ao aumento da probabilidade do pacote ser perdido em algum dos enlaces. O protocolo WH define o valor igual a quatro como o máximo número de saltos possíveis para um dispositivo ser considerado participante da rede. Este valor evita que dispositivos fiquem distantes do GW.

A Figura 35 apresenta uma comparação na quantidade média de saltos obtidos a partir dos algoritmos de roteamento sobre as topologias geradas. Já a Figura 36 apresenta uma comparação no máximo número de saltos que um dos dispositivos presentes no grafo gerado obteve.

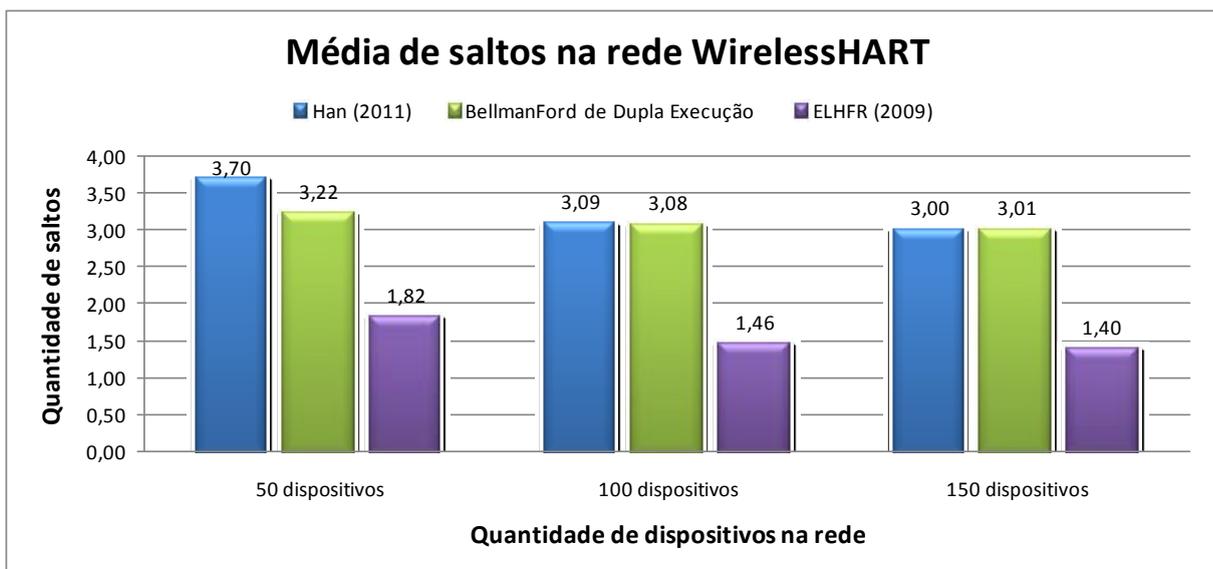


Figura 35 Comparação da média de número de saltos.

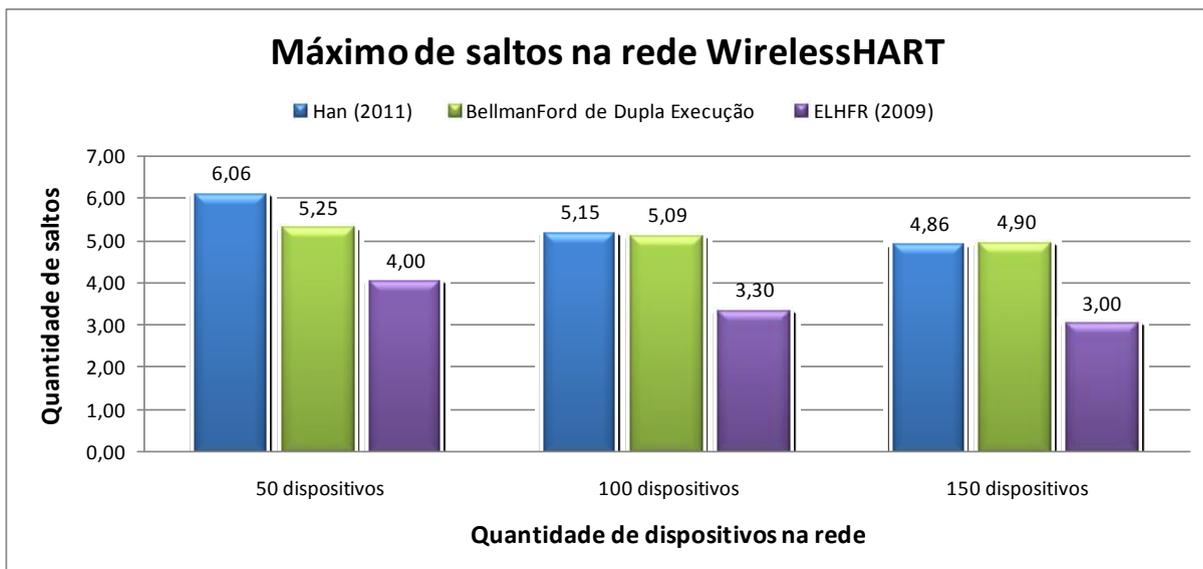


Figura 36 Comparação do máximo número de saltos.

Através da Figura 35 e da Figura 36 verifica-se que à medida que a quantidade de dispositivos aumenta, a média de saltos tende a diminuir em todos os algoritmos de roteamento. Esse comportamento ocorre devido ao tamanho da área utilizada nas simulações. Em uma área de 450m² com alcance de comunicação em 100m para cada dispositivo, a probabilidade de existir dispositivos mais próximos do GW é maior do que dispositivos distantes em quatro saltos.

O algoritmo ELHFR obteve a melhor média comparada com os demais analisados. Isso se deve ao fato do algoritmo criar *links* apenas quando um dispositivo está a um salto de diferença do dispositivo vizinho. A impossibilidade de criar *links* entre dispositivos a mesmo nível de salto ajuda este algoritmo a obter tal média.

Outro importante parâmetro de avaliação dos algoritmos de roteamento é o número de vizinhos. À medida que um dispositivo se torna concentrador de muitos *links*, a rede tende a ficar menos confiável. Isso porque na ocorrência de falha no dispositivo, muitos pacotes serão perdidos relativos aos dispositivos vizinhos. Quando a quantidade de vizinhos é alta, o processamento do dispositivo roteador é superior aos demais. O alto consumo de energia

também é um fator a ser considerado e indica que o dispositivo não deva possuir alimentação a bateria. Conseqüentemente, uma rede com uma quantidade de vizinhos distribuídos de forma homogênea entre os dispositivos é mais adequada para prover a confiabilidade necessária que o protocolo WH exige. A Figura 37 e a Figura 38 apresentam uma comparação na quantidade de vizinhos dos dispositivos obtidos a partir dos algoritmos de roteamento sobre as topologias geradas.

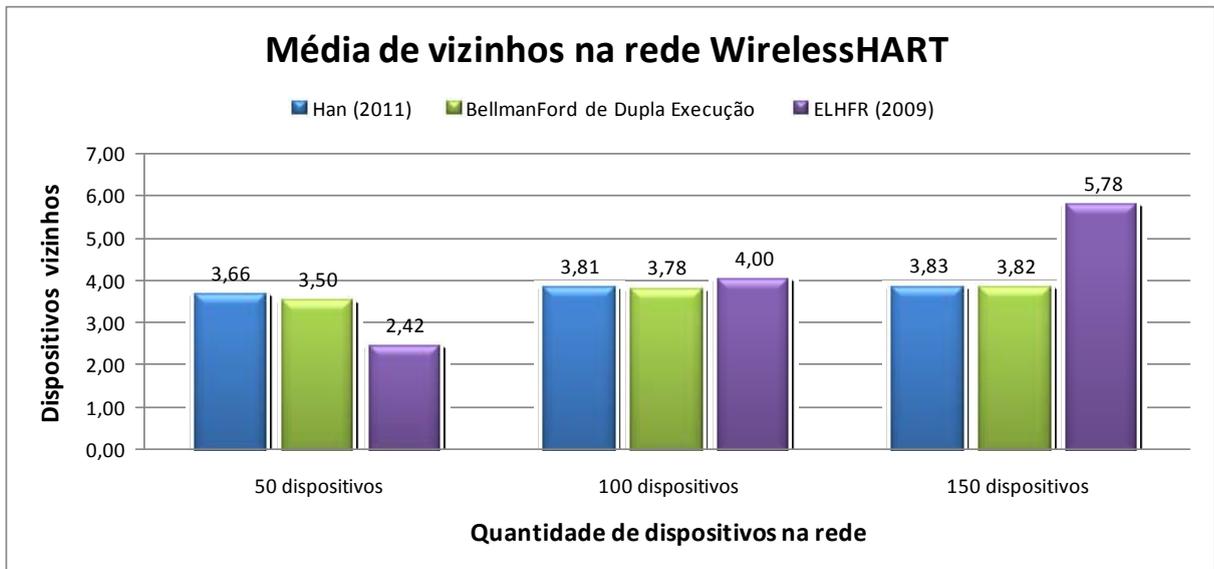


Figura 37 Comparação da média de número de vizinhos.

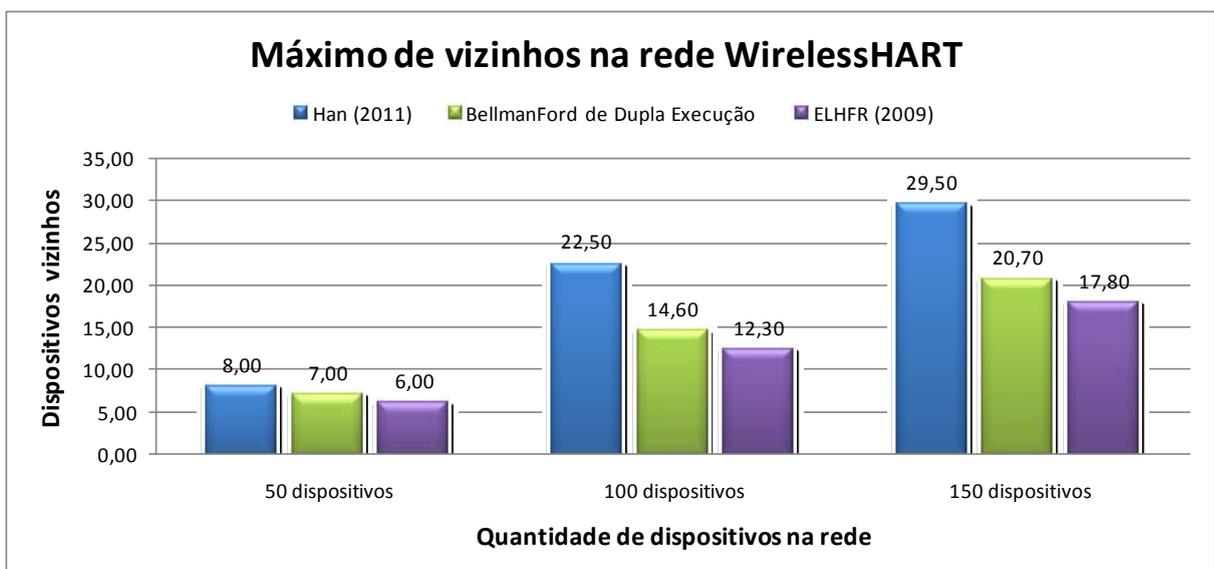


Figura 38 Comparação do máximo número de vizinhos.

O algoritmo de roteamento Han obteve o maior valor referente ao máximo número de vizinhos. Este valor transmite a ideia de que o algoritmo concentra muitos *links* em poucos dispositivos. Esta consideração é confirmada através da comparação do gradiente de vizinhos apresentada na Figura 39. A comparação é realizada entre os algoritmos de roteamento Han e Bellman-Ford de dupla execução proposto. Este último foi o que obteve o segundo maior máximo de vizinhos.

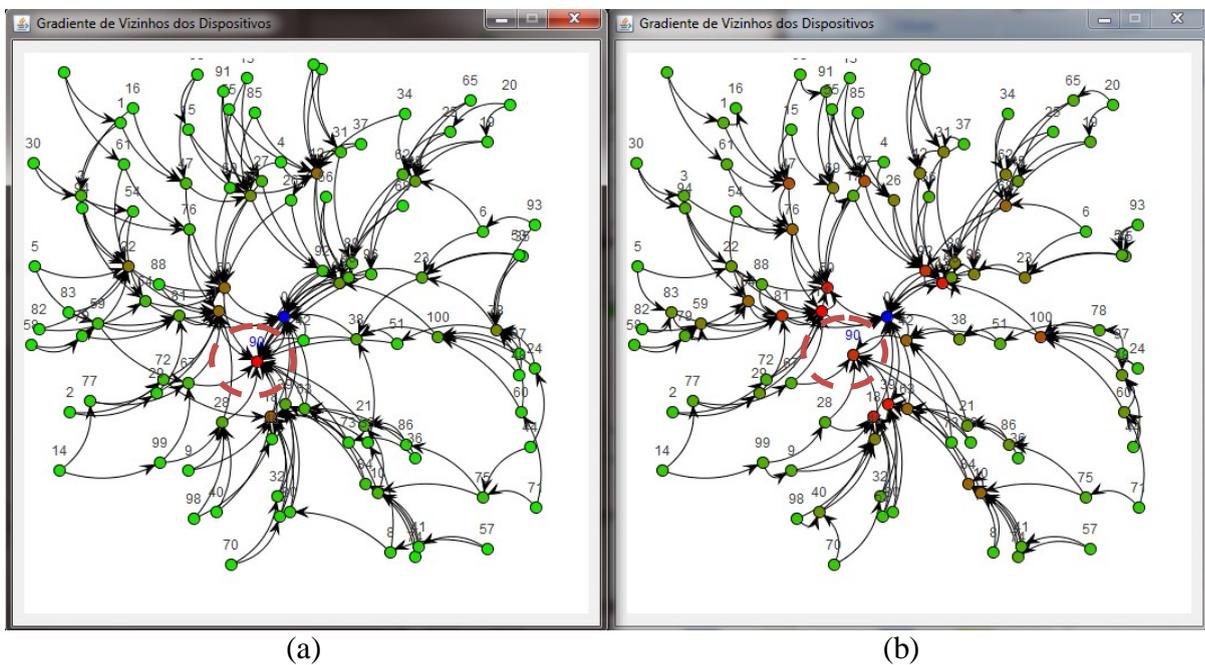


Figura 39 Comparação do gradiente de vizinhos: (a) Algoritmo de roteamento Han, (b) Algoritmo de roteamento Bellman-Ford de dupla execução.

Através do grafo (a) apresentado na Figura 39, verifica-se que o dispositivo destacado identificado pelo número 90 possui o maior número de vizinhos, concentrando alto tráfego na rede. Isto não ocorre para o grafo (b), visto que o mesmo dispositivo 90 não apresenta tamanha quantidade de vizinhos (a cor que tende para o vermelho denota alto número de vizinhos). Nota-se que o grafo (b) possui um espalhamento dos dispositivos com muitos vizinhos em torno do GW. À medida que os dispositivos se afastam do GW, a tendência é uma diminuição na quantidade de vizinhos.

Dispositivos roteadores possuem a função de encaminhar os pacotes oriundos de dispositivos mais distantes do GW. A Figura 40 apresenta a comparação entre os algoritmos de roteamento analisados frente à quantidade de dispositivos considerados roteadores nos grafos.

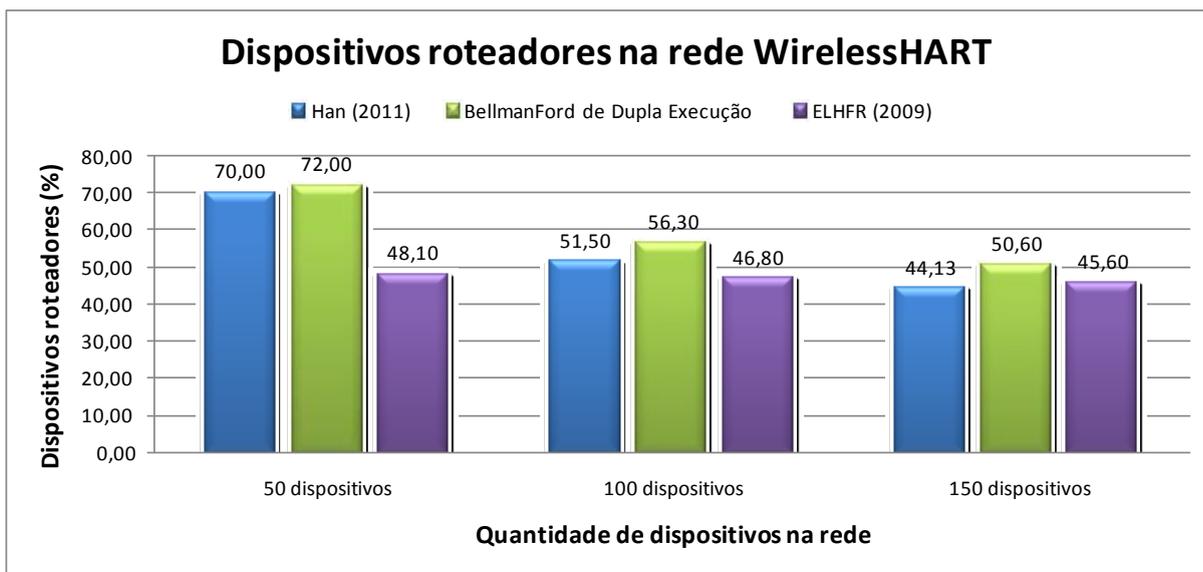


Figura 40 Comparação da quantidade de dispositivos roteadores.

A partir dos resultados obtidos na Figura 40 é possível afirmar que à medida que aumenta o número de dispositivos na rede, a quantidade de dispositivos roteadores diminui. Esta afirmação vai de encontro aos resultados obtidos nas comparações de vizinhos ilustradas anteriormente pela Figura 37 e Figura 38. A comparação de vizinhos indica que à medida que a rede cresce, persistem os mesmo roteadores na rede ocorrendo uma sobrecarga em dispositivos específicos. Portanto, um aumento na quantidade de dispositivos, a tendência é a redução no número de dispositivos roteadores.

A norma do protocolo WH exige que dispositivos de rede tenham pelo menos dois vizinhos para que se possa gerar redundância na comunicação. Os dispositivos que cumprem à essa exigência são considerados dispositivos confiáveis pelo ambiente de análise.

Conseqüentemente, uma comparação na quantidade destes dispositivos na rede é de suma importância. A quantidade de dispositivos confiáveis na rede demonstra o quanto um algoritmo é confiável, robusto e apto a ser aplicável no protocolo WH. A Figura 41 apresenta essa comparação entre os algoritmos analisados.

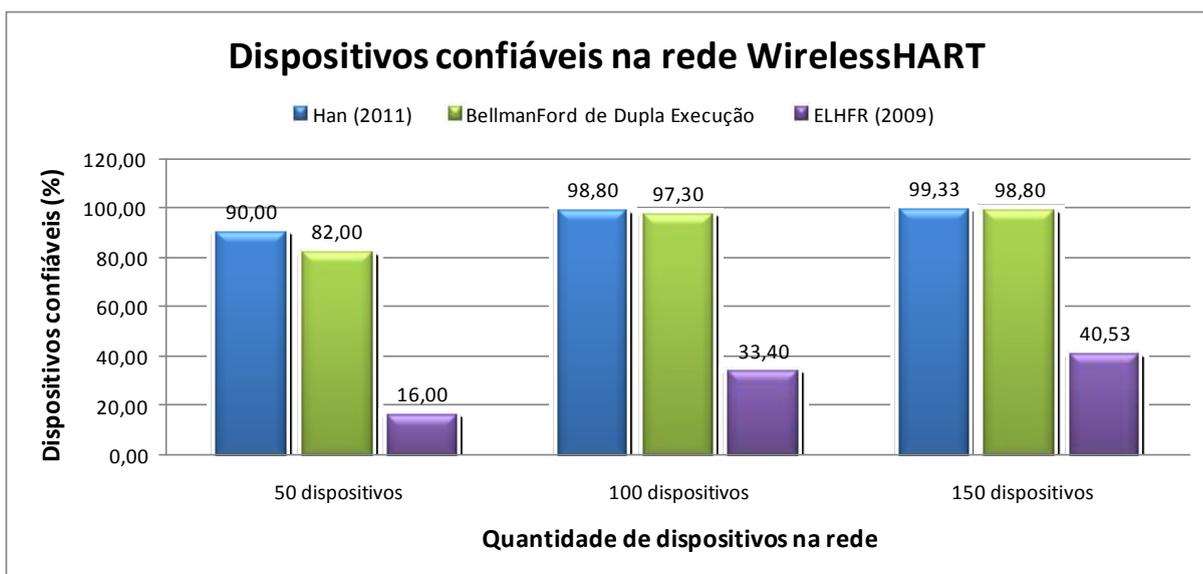


Figura 41 Comparação da quantidade de dispositivos confiáveis.

À medida que cresce a rede em quantidade de dispositivos, o número da porcentagem de dispositivos confiáveis aumenta. O algoritmo de roteamento Han e Bellman-Ford de dupla execução apresentaram ótimo desempenho nesta comparação. Portanto, frente à exigência de existir dispositivos confiáveis, os dois algoritmos são fortemente indicados a serem aplicáveis em redes WH. O algoritmo de roteamento ELHFR obteve um resultado baixo quando comparado com os demais. Essa baixa quantidade de dispositivos confiáveis ocorre devido a sua estratégia de encontrar rotas. Caso os dispositivos estejam apenas a um salto do GW, será disponibilizado apenas o *link* direto com o GW, sem agregar redundância.

A Figura 42 apresenta a comparação referente à porcentagem de dispositivos considerados distantes. São dispositivos que possuem o número de saltos em relação ao GW superior a quatro.

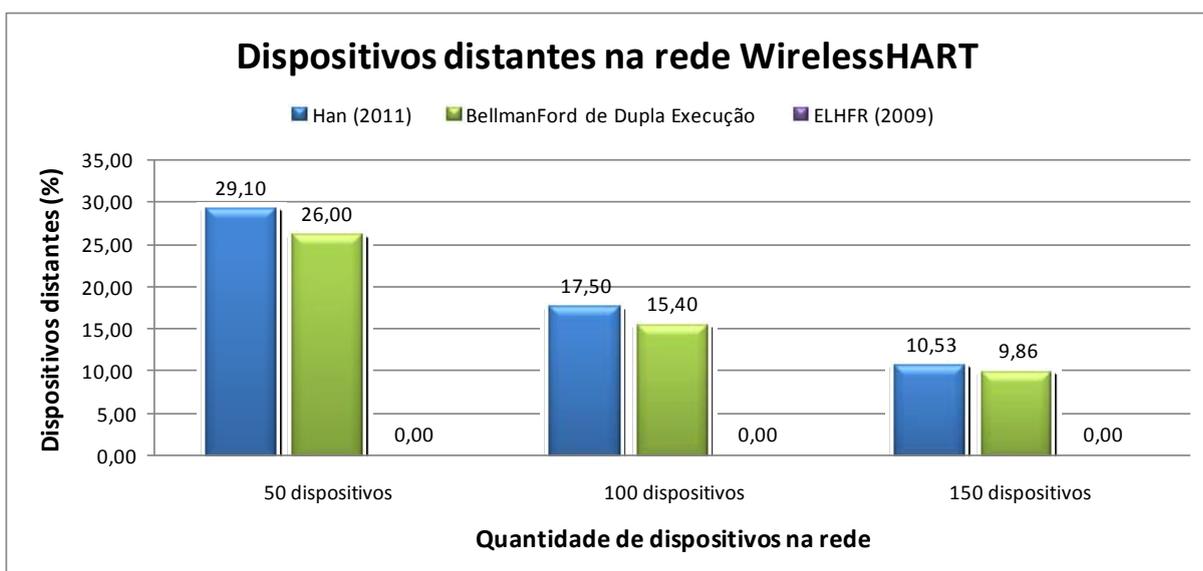


Figura 42 Comparação da quantidade de dispositivos distantes.

O resultado apresentado pela Figura 42 vai ao encontro dos resultados encontrados pelas comparações ilustradas na Figura 35 e na Figura 36. A comparação do número de saltos indica que à medida que a rede cresce, o número médio de saltos diminui. Conseqüentemente, a quantidade de dispositivos distantes também deve diminuir.

A Figura 43 apresenta a quantidade de *links* gerados pelos grafos. Esta comparação tem por objetivo verificar qual dos algoritmos gera mais *links* em seus respectivos grafos. Uma maior quantidade de *links* indica que o algoritmo é mais robusto devido a diversos caminhos redundantes possíveis.

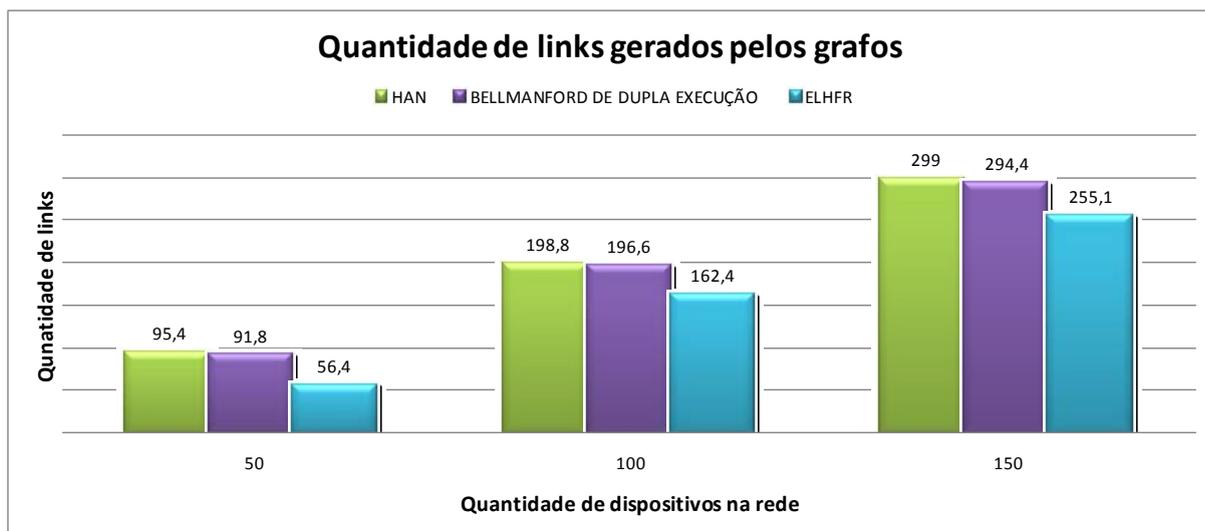


Figura 43 Comparação da quantidade de links gerados pelos grafos.

O resultado obtido indica que os algoritmos de roteamento Han e o Bellman-Ford de dupla execução possuem quantidade de *links* semelhantes. O algoritmo ELHFR demonstra criar poucos *links* para redes com baixo número de dispositivos. À medida que a quantidade de dispositivos aumenta, o algoritmo ELHFR tende a criar um maior número de *links*.

6.2 SEGUNDO EXPERIMENTO - ANÁLISE DOS ALGORITMOS DE ESCALONAMENTO

Neste experimento, os grafos gerados pelos algoritmos de roteamento foram submetidos aos algoritmos de escalonamento. O objetivo foi verificar a taxa de escalonabilidade que determinado conjunto apresenta. As dimensões da área e o modelo de conectividade são os mesmos utilizados no primeiro experimento. Foi fixada a quantidade de 50 dispositivos para as simulações. O período de publicação foi variada a fim de verificar a progressão do conjunto em relação à variação do período. A Tabela 3 apresenta as configurações dos parâmetros da rede para o segundo experimento.

Tabela 3 Configuração dos parâmetros do segundo experimento

Dimensões da área	450 x 450
Alcance máximo de comunicação	100 m
Número de dispositivos	50
Período de publicação	Variável
Pontos de Acesso	1

A taxa de escalonabilidade é o fator de comparação para esse experimento. Quanto maior essa taxa, mais dispositivos estão aptos a transmitir sem ocorrer conflitos na rede. A Figura 44, Figura 45 e Figura 46 apresentam a comparação entre os algoritmos de escalonamento. A Figura 44 ilustra os resultados obtidos utilizando o algoritmo de roteamento Han. A Figura 45 apresenta os mesmos resultados, porém, quando os grafos foram submetidos ao algoritmo Bellman-Ford de dupla execução. Da mesma forma, a Figura 46 apresenta quando utilizado o algoritmo ELHFR.

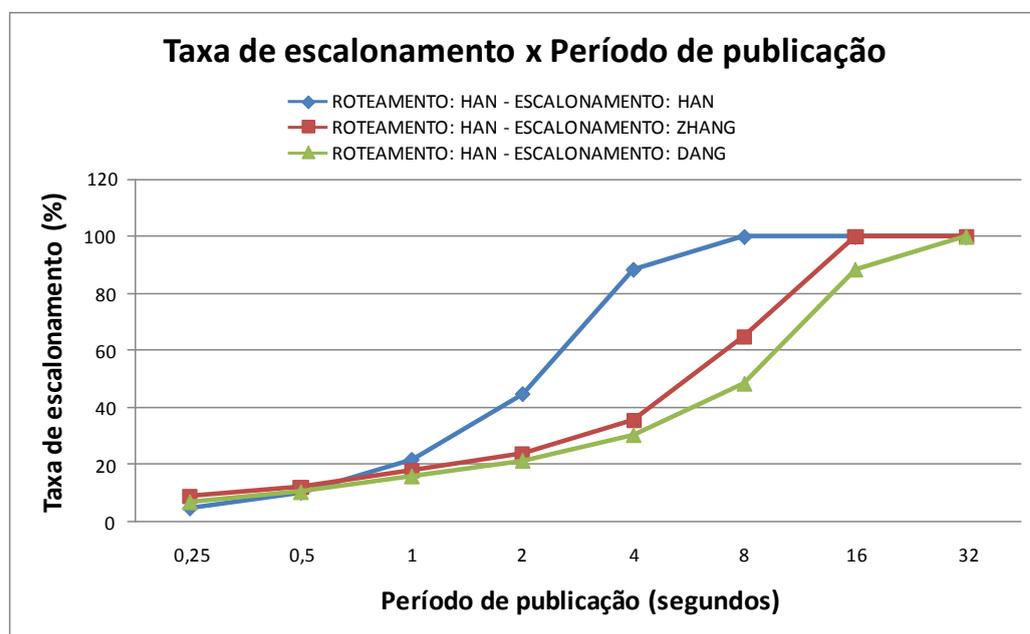


Figura 44 Comparação gráfica da taxa de escalonamento entre os algoritmos de escalonamento quando utilizado o algoritmo de roteamento Han.

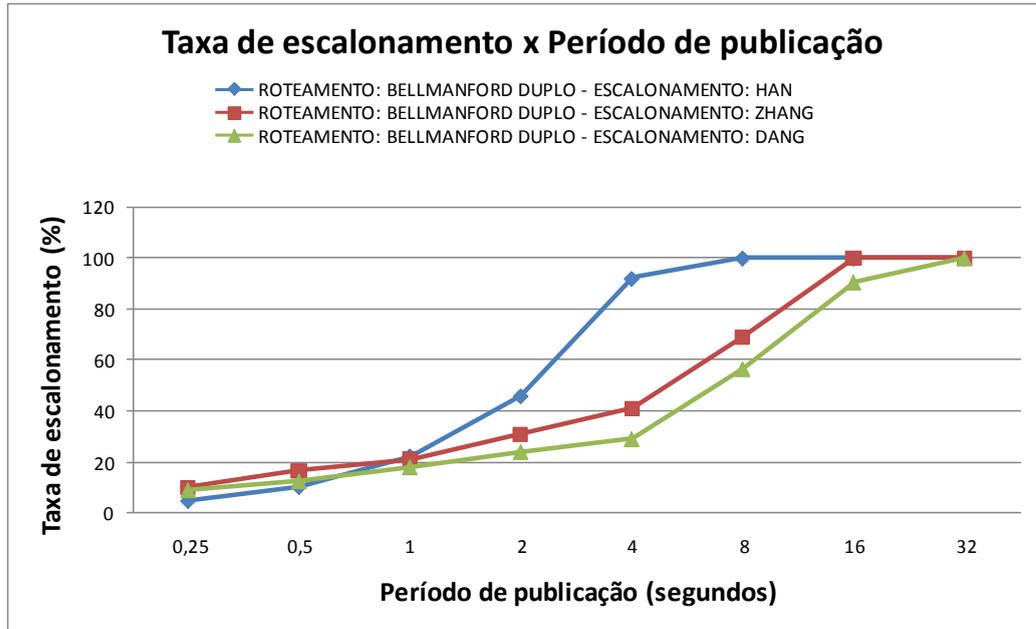


Figura 45 Comparação gráfica da taxa de escalonamento entre os algoritmos de escalonamento quando utilizado o algoritmo de roteamento Bellman-Ford de dupla execução.

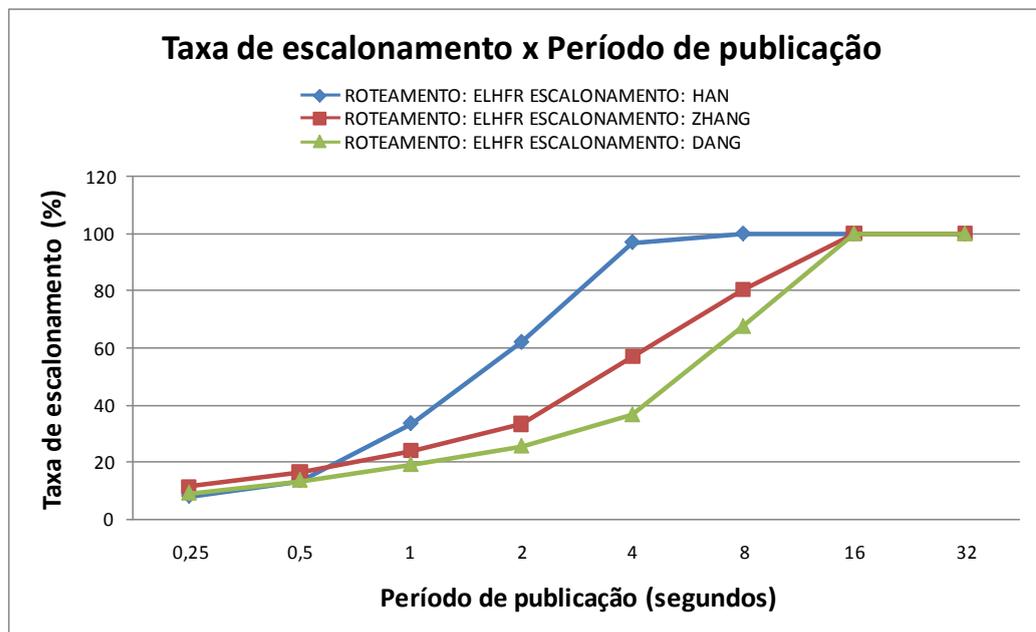


Figura 46 Comparação gráfica da taxa de escalonamento entre os algoritmos de escalonamento quando utilizado o algoritmo de roteamento ELHFR.

Os resultados obtidos demonstram que o algoritmo de escalonamento Han é superior na taxa de sucesso de escalonamento frente aos outros dois algoritmos. Esta superioridade é mais acentuada quando o período de publicação dos dados é superior a 1s. Isto ocorre devido ao método de escalonamento que o algoritmo utiliza. A estratégia de criar um *superframe* com o dobro do tamanho do original na existência de dois vizinhos sucessores aumenta consideravelmente o sucesso de escalonamento dos dispositivos.

O algoritmo de escalonamento Zhang foi superior ao escalonamento Dang. Isto se deve ao fato da estratégia utilizada pelo escalonamento Zhang em realizar as retransmissões apenas no caminho considerado principal. Em contrapartida, o escalonamento Dang realiza a retransmissão por todos os *links* encontrados do subgrafo.

6.3 AVALIAÇÃO DOS MELHORES CONJUNTOS

Os algoritmos de escalonamento têm por objetivo maximizar a taxa de escalonabilidade sem afetar a confiabilidade da rede. Logo, os resultados obtidos da taxa de escalonabilidade devem ser confrontados com os resultados obtidos dos algoritmos de roteamento a fim de avaliar o melhor conjunto.

Uma comparação entre os melhores conjuntos é apresentada nesta subseção. Para todos os algoritmos de roteamento, o escalonamento Han foi o que apresentou melhores resultados na taxa de sucesso de escalonamento. A Tabela 4 apresenta os resultados dos algoritmos de roteamento utilizando o escalonamento Han e a Figura 47 apresenta uma ilustração gráfica dos mesmos resultados.

Tabela 4 Comparação dos melhores conjuntos para cada algoritmo de roteamento.

Período de Publicação (segundos)	Escalonamento: Algoritmo HAN.		
	Roteamento:		
	Algoritmo HAN (%)	Algoritmo BellmanFord de dupla execução (%)	Algoritmo ELHFR (%)
0,25	5	5	8
0,5	10,5	10,5	13,5
1	22	22,5	33,5
2	45	46	62
4	88,5	92	97
8	100	100	100
16	100	100	100
32	100	100	100

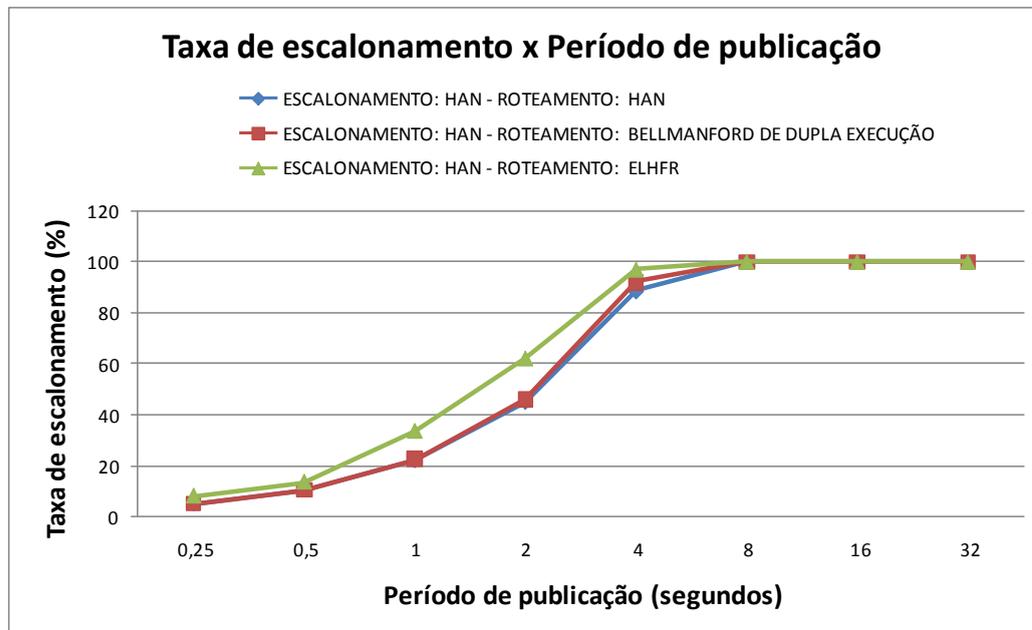


Figura 47 Comparação gráfica da taxa de escalonamento entre os algoritmos de roteamento quando utilizado o algoritmo de escalonamento Han.

Os resultados indicam que o melhor conjunto em termos de taxa de escalonamento é o composto pelo escalonamento Han proposto em (HAN et al., 2011) utilizando o algoritmo de escalonamento ELHFR proposto em (JINDONG; ZHENJUN; YAOPEI, 2009). As superioridade frente os demais conjuntos fica mais evidente nos períodos de publicação de 1 e 2 segundos. Nesta faixa, o conjunto chegou a apresentar uma taxa de escalonamento 16%

superior aos outros dois conjuntos. Contudo, a análise somente da taxa de escalonamento para classificar um conjunto como o melhor aplicável a redes WH não é suficiente. Os resultados obtidos no primeiro experimento para cada algoritmo de roteamento devem ser considerados nesta análise.

O protocolo WH exige que os dispositivos sejam confiáveis para que a rede sem fio contenha os requisitos necessários para ambientes industriais. Verifica-se que o algoritmo de roteamento ELHFR obteve fraco desempenho neste parâmetro analisado através da Figura 41 em relação aos demais algoritmos. A quantidade de *links* gerados pelo roteamento ELHFR também foi inferior (a Figura 43 apresentou essa análise). Isto demonstra que o algoritmo de escalonamento teve que escalonar uma menor quantidade de *links* resultando em uma melhor taxa de escalonamento.

A partir dos resultados obtidos nas Figura 41 e Figura 43, fica claro que o algoritmo de roteamento ELHFR somente torna-se atraente quando a rede possui uma grande quantidade de dispositivos. Pelas análises realizadas em redes de até 150 dispositivos e o baixo desempenho em confiabilidade e criação de *links*, as expectativas e indicativos de utilizá-lo em um ambiente real ficam muito reduzidas.

Confrontando os resultados obtidos no primeiro experimento com os do segundo experimento, o conjunto mais apropriado a ser aplicado em redes WH é composto pelo algoritmo de roteamento Bellman-Ford de dupla execução e o algoritmo de escalonamento Han proposto em (HAN et al., 2011). Outro conjunto que apresentou ser aplicável a redes WH é formado por ambos os algoritmos propostos em (HAN et al., 2011). O algoritmo de roteamento Han obteve resultados similares ao algoritmo de roteamento Bellman-Ford de dupla execução, tornando-o um algoritmo confiável ao protocolo WH.

7 CONCLUSÕES

A presente dissertação abordou a integração dos algoritmos de escalonamento no ambiente proposto em (KUNZEL, 2012). Através desse desenvolvimento foi possível analisar mais profundamente o protocolo WH. A partir de um estudo no estado da arte foi possível verificar os algoritmos de roteamento e escalonamento propostos especificamente para o WH. Foram implementados três algoritmos de roteamento e três algoritmos de escalonamento. O objetivo foi analisar o conjunto apto a ser utilizado na rede WH.

O ambiente permite analisar grafos gerados pelos algoritmos de roteamento. Métricas ilustradas em forma de histogramas e tabela são ilustradas. A integração dos algoritmos de escalonamento permite verificar a taxa de escalonabilidade de cada escalonador proposto. Através dessa integração é possível analisar mais profundamente os algoritmos em conjunto propostos para o protocolo WH.

O primeiro experimento abordou uma comparação nos algoritmos de roteamento desenvolvidos no ambiente. Os algoritmos foram submetidos à variação no número de dispositivos e uma análise foi realizada. Observou-se que aumentando a quantidade de vizinhos o número de dispositivos roteadores diminui, ou seja, os algoritmos analisados concentram o tráfego sempre nos mesmos roteadores. Isso sobrecarrega os dispositivos, tornando-os essenciais para a confiabilidade da rede.

A quantidade de dispositivos confiáveis é a métrica mais importante para analisar os algoritmos de roteamento. Isto porque, o protocolo WH exige que os dispositivos tenham pelo menos dois vizinhos a quem possam encaminhar os dados. Observou-se que o algoritmo de roteamento proposto em (HAN et al., 2011) e o algoritmo Bellman-Ford de dupla execução obtiveram ótimos resultados nessa métrica. Consequentemente são os algoritmos mais indicados para serem aplicados em redes WH.

O segundo experimento realizado abordou uma análise na taxa de escalonabilidade dos algoritmos de escalonamento desenvolvidos no ambiente. Para realizar as simulações de escalonamento foi necessário gerar grafos através dos algoritmos de roteamento desenvolvidos. Portanto, todos os algoritmos de escalonamento foram analisados para cada um dos algoritmos de roteamento. Os resultados obtidos demonstraram que o algoritmo de escalonamento Han obteve o melhor desempenho em todos os algoritmos de roteamento utilizado.

Baseado no segundo experimento, uma análise nos resultados apenas dos conjuntos que utilizam o escalonamento Han foi realizado. Observou-se que mesmo obtendo a melhor taxa de escalonabilidade, o conjunto formado pelo algoritmo de roteamento ELHFR e pelo escalonamento Han não foi suficiente para ser considerado apto a ser utilizado em redes WH. O algoritmo ELHFR foi considerado ineficaz para utilização devido ao baixo desempenho na quantidade de dispositivos confiáveis. Logo, a taxa de escalonamento por si só não foi suficiente para determinar um conjunto aplicável ao protocolo WH. Foi necessário que os resultados tanto do primeiro experimento como do segundo fossem confrontados e analisados em conjunto.

Os algoritmos desenvolvidos neste trabalho podem ser programados diretamente em um gerenciador real, permitindo analisar uma rede em funcionamento no qual o gerenciador fará uso dos algoritmos para rotear e escalonar as mensagens.

Dentre as possíveis atividades que buscam dar continuidade ao trabalho citam-se dois caminhos a se seguir. O primeiro caminho é utilizar o ambiente como base para o desenvolvimento de novos algoritmos tanto de roteamento como de escalonamento. Atualmente, não há algoritmos ótimos para o protocolo WH. Conseqüentemente, propostas de algoritmos têm sido realizadas constantemente pela comunidade acadêmica.

Outra linha de pesquisa visa aumentar a complexidade de análise do ambiente a fim de buscar resultados mais precisos baseados em aplicações reais. Em ambientes práticos, dispositivos possuem períodos de publicação diferentes. O ambiente de análise, atualmente, possui a restrição de configuração do mesmo período para todos os dispositivos de campo. Períodos de publicação aleatórios geram diferentes *superframes* na rede. Um estudo no comportamento dos conjuntos de algoritmos frente a múltiplos *superframes* é interessante. Tornar o ambiente mais dinâmico é outra atividade sugerida. Após a criação da topologia da rede não há possibilidade de alterar a localização dos dispositivos, dos pontos de acessos. A adição de novos dispositivos e pontos de acesso também são interessantes para um estudo mais aprofundado dos conjuntos frente às modificações na rede.

REFERÊNCIAS

- AKKAYA, K.; YOUNIS, M. A survey on routing protocols for wireless sensor networks. **Ad Hoc Networks**, Cancun, v. 3, n. 3, p. 325-349, 2005.
- AL-KARAKI, J. N.; KAMAL, A. E. Routing techniques in wireless sensor networks: a survey. **Wireless Communications**, New York, v. 11, n. 6, p. 6-28, 2004.
- BELLMAN, R. **On a routing problem**. RAND Corporation, Santa Monica, 1956. 8 p. Disponível em: <<http://www.dtic.mil/dtic/tr/fulltext/u2/606258.pdf>> Acesso em: 8 jul. 2013.
- BRUCKER, P. **Scheduling algorithms**. 5th. ed. New York: Springer, 2007.
- CHEN, D.; NIXON, M.; MOK, A.K. **WirelessHART: Real-Time Mesh Network for Industrial Automation**. New York: Springer, 2010.
- CORMEN, T. H. et al. **Introduction to Algorithms**. 2nd. ed. Massachusetts: Elsevier, 2002.
- DANG, K. et al. A Graph Route-Based Superframe Scheduling Scheme in *WirelessHART* Mesh Networks for High Robustness. In: WIRELESS PERSONAL COMMUNICATIONS (WPC), 2012, New York. **Proceedings...** New York: IEEE, 2012. v.71, p. 2431-2444.
- DICKOW, V. H. et al. Avaliação da aplicabilidade de algoritmos clássicos de roteamento em redes *WirelessHART*. In: SIMPÓSIO BRASILEIRO DE AUTOMAÇÃO INTELIGENTE (SBAI), 11., 2013, Recife. **Anais...** [S.l.]: IEEE, 2013a. p. 1-6.
- DICKOW, V. H. et al. Análise de algoritmos de escalonamento para redes *WirelessHART*. In: SIMPÓSIO BRASILEIRO DE ENGENHARIA DE SISTEMAS COMPUTACIONAIS (SBESC), 3., 2013, Niterói. **Anais...** [S.l.]: IEEE, 2013b. p. 1-6.
- DIESTEL, R. **Graph theory**. 2nd. ed. New York: Springer-Verlag, 2000.
- DIJKSTRA, E. W. A note on two problems in connexion with graphs. **Numerische mathematik**, Amsterdam, v. 1, n. 1, p. 269-271, 1959.
- FEOFILOFF, P.; KOHAYAKAWA, Y.; WAKABAYASHI, Y. **Uma introdução sucinta à teoria dos grafos**. IME-USP, 2011. 61 p. Disponível em: <<http://www.ime.usp.br/~pf/teoriadosgrafos/>> Acesso em: 2 out. 2013.
- FLOYD, R. W. Algorithm 97: shortest path. **Communications of the ACM**, New York. v. 5, n. 6, p. 345, 1962.
- FORD, L.R. **Network Flow Theory**. RAND Corporation, Santa Monica, 1956. 8 p. Disponível em: <<http://www.rand.org/pubs/papers/P923.html>> Acesso em: 8 jul. 2013.

GAO, G.; ZHANG, H.; LI, L. A Reliable Multipath Routing Strategy for *WirelessHART* Mesh Networks Using Subgraph Routing. **Journal of Computational Information Systems**, New York, v. 9, n. 5, p. 2001-2008, 2013.

HAN, S. et al. Reliable and Real-Time Communication in Industrial Wireless Mesh Networks. In: REAL-TIME AND EMBEDDED TECHNOLOGY AND APPLICATIONS SYMPOSIUM (RTAS), 17., 2011, Chicago. **Proceedings...** [S.l.]: IEEE, 2011. p. 3-12.

HAN, S. et al. Wi-HTest: Compliance Test Suite for Diagnosing Devices in Real-Time *WirelessHART* Network. In: REAL TIME AND EMBEDDED TECHNOLOGY AND APPLICATIONS SYMPOSIUM (RTAS), 15., 2009, São Francisco. **Proceedings...** [S.l.]: IEEE, 2009. p 327-336

HART COMMUNICATION FOUNDATION. **HCF_SPEC-065**: 2.4GHz DSSS O-QPSK Physical Layer Specification. Austin, 2007. 20 p.

HART COMMUNICATION FOUNDATION. **HCF_SPEC-075**: TDMA Data Link Layer Specification. Austin, 2008. 76 p.

HART COMMUNICATION FOUNDATION. **HCF_SPEC-085**: Network Management Specification. Austin, 2009. 98 p.

HERRTWICH, R. G. **An Introduction to Real-Time Scheduling**. International Computer Science Institute, Berkeley, 1990. 29 p. Disponível em: <<http://www.icsi.berkeley.edu/pubs/techreports/tr-90-035.pdf>> Acesso em: 29 jul. 2013.

INTERNATIONAL ELECTROTECHNICAL COMMISSION. **IEC-62951**: Industrial communication networks - Wireless communication network and communication profile – *WirelessHART*. Ed 1.0. Geneva, 2010. 446 p.

JANUÁRIO, T. O. **Implementação e análise de algoritmos para coloração de arestas**. 2011. 47 p. Dissertação (Mestrado em ciências) – Programa de Pós-Graduação em Ciências da Computação, Universidade Federal de Minas Gerais, Belo Horizonte, 2011.

JINDONG, Z.; ZHENJUN, L.; YAOPEI, Z. ELHFR: A graph routing in industrial wireless mesh network. In: INFORMATION AND AUTOMATION (ICIA), 2009, Macau. **Proceedings...** [S.l.]: IEEE, 2009. p. 106-110.

JOHNSON, D. B. Efficient algorithms for shortest paths in sparse networks. **Journal of the ACM**, New York. v. 24, n. 1, p. 1-13, 1977.

KIM, A. N. et al. When HART goes wireless: Understanding and implementing the *WirelessHART* standard. In: EMERGING TECHNOLOGIES AND FACTORY AUTOMATION (ETFA), 2008. Hamburg. **Proceedings...** New York: IEEE, 2008. p. 899-907.

KUMAR, S.; CHAUHAN, S. A survey on scheduling algorithms for wireless sensor networks. **International Journal of Computer Applications**, New York, v. 20, n. 5, p. 7-13, 2011.

KUNZEL, G. **Ambiente para avaliação de estratégias de roteamento para redes *WirelessHART***. 2012. 95 p. Dissertação (Mestrado em engenharia) – Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2012.

MA, X. et al. Design and implementation of one redundancy function in wireless tree networks. In: MEASUREMENT, INFORMATION AND CONTROL INTERNATIONAL CONFERENCE (MIC), 2012, Harbin. **Proceedings...** [S.l.]: IEEE, May 2012. p. 1-4.

MISHRA, M. K., MISHRA, S. Performance Based Optimization Routing Technique for Wireless Mesh Network. **Journal of Current Computer Science and Technology**, Nova Deli, v. 1, n. 8, p. 442-451, 2011.

MÜLLER, I. et al. Development of *WirelessHART* Compatible Field Devices. In: INSTRUMENTATION AND MEASUREMENT TECHNOLOGY CONFERENCE (I2MCT), 2010, Austin. **Proceedings...** [S.l.]: IEEE, 2010. p. 1430-1434.

MÜLLER, I., NETTO, J.C., PEREIRA, C.E. *WirelessHART* Field Devices. **Instrumentation & Measurement Magazine**, New York, v. 14, p. 20-25, 2011.

MÜLLER, I. et al. Towards *WirelessHART* Protocol Decentralization: A Proposal Overview. In: INTERNET OF THINGS, SMART SPACES, AND NEXT GENERATION NETWORKING (NEW2AN/ruSMART), 2012, St. Petersburg. **Proceedings...** [S.l.]: IEEE, 2012. p. 248-259.

MÜLLER, I. **Gerenciamento Descentralizado de redes sem fio industriais segundo padrão *WirelessHART***. 2012. 104 p. Tese (Doutorado em engenharia) – Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2012.

NOBRE, M. H. **Análise de Desempenho Baseada em Simulação de Redes *WirelessHART***. 2011. 73 p. Dissertação (Mestrado em ciências) – Programa de Pós-Graduação em Engenharia Elétrica e Computação, Universidade Federal do Rio Grande do Norte, Natal, 2011.

SAIFULLAH, A. et al. Real-time scheduling for *WirelessHART* networks. In: REAL-TIME SYSTEMS SYMPOSIUM (RTSS), 31., 2010. San Diego. **Proceedings...** New York: IEEE, 2010. p. 150-159.

SEDGEWICK, R., WAYNE, K. **Algorithms**. 4th. ed. Boston: Addison-Wesley Professional, 2011.

TANENBAUM, A. S., WETHERALL, D. J. **Redes de computadores**. 5. ed. São Paulo: Pearson, 2011.

TARIQUE, M. et al. Survey of multipath routing protocols for mobile ad hoc networks. **Journal of Network and Computer Applications**, New York, v. 32, n. 6, p. 1125-1143, 2009.

VIZING, V. G. **On an estimate of the chromatic class of a p-graph.** Discrete Analysis, Novosibirsk, 1964. 6 p. disponível em: <http://www.researchgate.net/publication/242530345_On_an_estimate_of_the_chromatic_class_of_a_p-graph> Acesso em: 10 dez. 2013.

WARSHALL, S. A theorem on boolean matrices. **Journal of the ACM**, New York. v. 9, n. 1, p. 11-12, 1962.

WILLIG, A. Recent and emerging topics in wireless industrial communications: A selection. **Industrial Informatics, IEEE Transactions on**, New York, v. 4, n. 2, p. 102-124, 2008.

WINTER, J. M. et al. Study of routing mechanisms in a *WirelessHART* network. In: INDUSTRIAL TECHNOLOGY (ICIT), 2013. Cape Town. **Proceedings...** [S.l.]: IEEE, 2013. p. 1540-1545.

YICK, J.; MUKHERJEE, B.; GHOSAL, D. Wireless sensor network survey. **Computer networks**, New York, v. 52, n. 12, p. 2292-2330, 2008.

ZAND, P. et al. A distributed scheduling algorithm for real-time (D-SAR) industrial wireless sensor and actuator networks. In: EMERGING TECHNOLOGIES & FACTORY AUTOMATION (ETFA), 17., 2012. Krakow. **Proceedings...** [S.l.]: IEEE, 2012. p. 1-4.

ZHANG, H.; SOLDATI, P.; JOHANSSON, M. **Efficient link scheduling and channel hopping for convergecast in *WirelessHART* networks.** Technical Report: School of Electrical Engineering, Royal Institute of Technology, Stocolmo, 2009a. 39 p. Disponível em: <<http://www.cs.otago.ac.nz/staffpriv/haibo/papers/TR1.pdf>> Acesso em: 29 jul. 2013.

ZHANG, H.; SOLDATI, P.; JOHANSSON, M. Optimal link scheduling and channel assignment for convergecast in linear *WirelessHART* networks. In: MODELING AND OPTIMIZATION IN MOBILE, AD HOC, AND WIRELESS NETWORKS. 7., 2009. Seoul. **Proceedings...** [S.l.]: IEEE, 2009b. p. 1-8.

ZHANG, H.; SOLDATI, P.; JOHANSSON, M. Time-and channel-efficient link scheduling for convergecast in *WirelessHART* networks. In: COMMUNICATION TECHNOLOGY (ICCT), 13., 2011. Jinan. **Proceedings...** [S.l.]: IEEE, 2011. p. 99-103.

ZHANG, S. et al. A highly reliable link scheduling strategy for *WirelessHART* networks. In: ADVANCED TECHNOLOGIES FOR COMMUNICATIONS (ATC), 2013. Ho Chi Minh City. **Proceedings...** [S.l.]: IEEE, 2013. p. 39-43.