UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE - UERN

UNIVERSIDADE FEDERAL RURAL DO SEMIÁRIDO - UFERSA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO UERN / UFERSA

CREDSON ISAAC LOPES DOS SANTOS

DEPENDABILIDADE APLICADA A PROTOCOLO HIERÁRQUICO BASEADO NO LEACH

NATAL

2017

Credson Isaac Lopes dos Santos

Dependabilidade Aplicada a Protocolo Hierárquico Baseado no LEACH

Dissertação apresentada como requisito para obtenção do título de Mestre em Ciência da Computação, pelo Programa de Pós-graduação em Ciência da Computação (PPgCC), da Universidade do Estado do Rio Grande do Norte (UERN) e Universidade Federal Rural do Semiárido (UFERSA).

Orientadora: Prof[a]. Dr[a]. Karla Darlene Nepomuceno Ramos;
Coorientador: Prof. Dr. Felipe Denis Mendonça de Oliveira

Natal
2017

S237d Santos, C. I. L.
Dependabilidade Aplicada a Protocolo Hierárquico Baseado no LEACH/ Credson Isaac Lopes dos Santos. – 2017.
66 p.
Dissertação (Mestrado) – Universidade do Estado do Rio Grande do Norte e Universidade Federal Rural do Semiárido, Programa de Pós-Graduação em Ciência da Computação, Natal, RN, 2017.
Orientadora: Profª. Drª. Karla Darlene Nepomuceno Ramos; Coorientador: Prof. Dr. Felipe Denis Mendonça de Oliveira

1. Redes de Sensores Sem Fio (RSSF) 2. Redes de Sensores Sem Fio Industriais (RSSFI) 3. Dependabilidade 4. Network Simulator3 (NS-3) 5. FTE-LEACH I. Dependabilidade Aplicada a Protocolo Hierárquico Baseado no LEACH
II. Profª. Drª. Karla Darlene Nepomuceno Ramos

Ficha catalográfica gerada pelo Sistema Integrado de Bibliotecas e Diretoria de Informatização (DINF) - UERN, com os dados fornecidos pelo autor

Credson Isaac Lopes dos Santos

Dependabilidade Aplicada a Protocolo Hierárquico Baseado no LEACH

Dissertação apresentada como requisito para obtenção do título de Mestre em Ciência da Computação, pelo Programa de Pós-graduação em Ciência da Computação (PPgCC), da Universidade do Estado do Rio Grande do Norte (UERN) e Universidade Federal Rural do Semiárido (UFERSA).

Aprovado em 28 de Agosto de 2017.

BANCA EXAMINADORA:

Prof[a]. Dr[a]. Karla Darlene Nepomuceno Ramos
Orientadora

Prof. Dr. Felipe Denis Mendonça de Oliveira
Coorientador

Prof. Dr[a]. Claudia Maria Fernandes Araújo Ribeiro – IFRN
Avaliadora Externa

Prof. Dr. Luiz Felipe de Queiroz Silveira – UFRN
Avaliador Externo

Prof. Dr. Rodrigo Soares Semente – UFERSA
Avaliador Externo

Este trabalho é dedicado às pessoas mais importantes em minha vida: Ao Senhor criador. Meu grande amor, Hitiara Shirley, por estar ao meu lado em todas as horas e em importantes momentos de minha vida. Aos meus queridos filhos Júlio César, Ivan Vinícius e Isa Beatriz. Minha mãe, pelos conselhos indispensáveis e pelo incentivo a busca científica. Meu pai, pela batalha de todos os dias. Sempre Juntos!

AGRADECIMENTOS

Inicio os agradecimentos primeiramente ao Senhor Deus, por sua magnífica ação em minha vida. Agradeço a paciência e o direcionamento da orientadora, Dr^{a}. Karla Darlene, e do coorientador Dr. Felipe Denis. Ao imprescindível parceiro de pesquisa, José Ewerton, pelo companheirismo e conhecimento desprendidos nesta pesquisa, ao Dr Marcelo Henrique Ramalho Nobre, por contribuir com esclarecimentos sobre sua pesquisa a qual veio a engrandecer este trabalho . Ao Programa de Pós Graduação em Ciências da Computação UERN/UFERSA, pelo acolhimento, e ao seu corpo docente, pelos seus ensinamentos. Ao Instituto Federal de Ciência e tecnologia do RN, pelo apoio no ambiente profissional. A minha esposa Hitiara Shirley, a qual com seu amor tem estado ao meu lado em todas as conquistas, bem como nas horas de aflição. Aos meus pais, instrumentos de Deus em minha vida. As minhas irmãs, pelo estimulo e apoio. Aos meus queridos filhos que, nas horas de ausência, tentaram entender o momento. E a todos que, de maneira direta ou indireta, contribuíram para este trabalho, sintam-se todos agradecidos.

RESUMO

Esta pesquisa visa inserir recursos para se alcançar mais qualidade e confiança (dependabilidade) no serviço fornecido pelas Redes de Sensores sem Fio Industriais (RSSFI). Para tanto, foram pesquisados e identificados protocolos de roteamento que oferecem suporte à dependabilidade, dentre eles o protocolo FTE-LEACH, que é baseado no LEACH. Com o objetivo de avaliar o impacto da inserção de técnicas de dependabilidade no protocolo FTE-LEACH, foi desenvolvido um módulo de simulação no *Network Simulator 3 (NS-3)* visando aplicar simulações com ferramentas próprias para ambientes de redes. De acordo com a taxonomia de dependabilidade, foram implementadas as técnicas de prevenção, tolerância, remoção e previsão de falhas, permitindo maior confiabilidade e segurança às RSSFI. Os resultados da pesquisa indicam que a inserção de técnicas de dependabilidade não comprometem o desempenho do protocolo FTE-LEACH, e que ainda pode ser aperfeiçoado, por meio de técnicas que visem a diminuição do tempo de configuração para o protocolo citado.

Palavras-chaves: Redes de Sensores Sem Fio (RSSF), Redes de Sensores Sem Fio Industrial (RSSFI), Dependabilidade, *Network Simulator* 3 (NS-3), FTE-LEACH.

ABSTRACT

This research aims to insert resources to achieve more quality and reliability (dependability) in the service provided by the Industrial Wireless Sensor Networks (IWSN). For that, routing protocols that support dependability were investigated and identified, among them the FTE-LEACH protocol, which is based on LEACH. In order to evaluate the impact of the insertion of dependability techniques in the FTE-LEACH protocol, a simulation module was developed in Network Simulator 3 (NS-3) to implement simulations with its own tools for network environments. According to the dependability taxonomy, the techniques of prevention, tolerance, removal and prediction of failures were implemented, allowing greater reliability and security to IWSN. The results of the research indicate that the insertion of dependability techniques does not compromise the performance of the FTE-LEACH protocol, and that it can still be improved by means of techniques aimed at reducing the set-up time for the cited protocol.

Key-words: Wireless Sensor Networks (WSN), Industrial Wireless Sensor Networks (IWSN), Energy Efficiency, Network Simulator 3 (NS-3), FTE-LEACH.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ACK	*Acknowleghment)*
ADV1	*Advertice 1*
ADV2	*Advertice 2*
ADV3	*Advertice 3*
ADV4	*Advertice 4*
ADV5	*Advertice 5*
ADV6	*Advertice 6*
BS	*Base Station*
CH	*Cluter Head*
CM	*Cluster Member*
CSMA-CA	*Carrier Sense Multiple Access with Collision Avoidance*
DDL	*Device Description Language*
DVFS	*Dynamic Voltage and Frequency Scaling*
DVS	*Dynamic Voltage Scaling*
ED	*Energy Detection*
EDDL	*Electronic Device Description Language*
FFD	*Full Function Device*
FL-LEACH	*Fuzzy Logic LEACH*
FTE-LEACH	*Fault-tolerant and Energy-efficient LEACH*
GPS	*Global Positioning System*
GTS	*Guaranteed Time Slots*
HART	*Highway Addressable Remote Transducer*
IB-LEACH	*Intra-Balanced LEACH*
ID	*Identificador*

IEEE	*Institute of Electrical and Electronics Engineers*
IoT	*Internet of Things*
ISM	*Industrial Scientific Medical*
IWSN	*Industrial Wireless Sensors Network*
LEACH	*Low Energy Adaptive Clustering Hierarchy*
LLN	*Low-power and Lossy Network*
LQI	*Link Quality Indication*
LR-WPAN	*Low Rate Wireless Personal Area Network*
MAC	*Media Access Control*
MR-LEACH	*Multi-hop Routing LEACH*
MS-LEACH	*More Secure LEACH*
NS-3	*Network Simulator 3*
PHY	*Physical*
RSSF	Rede de Sensores Sem Fio
RSSFI	Rede de Sensores Sem Fio Industrial
RSSI	*Received Signal Straith Indicator*
SLPM	*System Level Power Management*
SO	Sistema Operacional
SPM	*Static Power Manegement*
TDMA	*Time Division Machine Access*
VCH	*Vice-CH*
WSN	*Wireless Sensors Network*
BS	*Base Station - Estação-base*
CH	*Cluster Head*
CM	*Cluster Member*
CSMA-CA	*Carrier sense multiple access with collision avoidance*

FFD	*Full Function Device*
LEACH	*Low Energy Adaptive Clustering Hierarchy*
FTE-LEACH	*Fault-tolerant and Energy-efficient LEACH*
GPS	*Global Positioning System*
HTTP	*Hypertext Transfer Protocol*
ICMP	*Internet Control Message Protocol*
IdC	*Internet das Coisas*
IEEE	*Institute of Electrical and Electronics Engineers*
IETF	*Internet Engineering Task Force*
IoT	*Internet of Things*
IP	*Internet Protocol*
IPv6	*Internet Protocol version 6*
LLN	*Low-power and Lossy Network*
LOWPAN	*Low power Wireless Personal Area Network*
LR-WPAN	*Low-Rate Wireless Personal Area Network*
MAC	*Medium Access Control*
MTBF	*Tempo médio entre defeitos*
MTTF	*Tempo médio de funcionamento até a ocorrência de um defeito*
MTTR	*Tempo médio até o sistema reparar um defeito*
MTU	*Maximum Transmission Unit*
NMR	*Redundância modular múltipla*
NS3	*Network Simulation 3*
OSI	*Open Systems Interconection*
RFC	*Request for Comments*
RFD	*Reduced Function Devices*
RFID	Radio Frequency Identification

ROLL	*Routing Over Low-power and Lossy Network*
RSSF	*Redes de Sensores Sem Fio*
SBRC	*Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*
TDMA	*Acesso Múltiplo por Divisão de Tempo*
UDP	*User Datagram Protocol*
VCH	*Vice Cluster Head*
WHART	*WirelessHART*
WPAN	*Wireless Personal Area Network*
WSAN	*Wireless Sensor and Actuator Network*
WSN	*Wireless Sensor Network*
WSN	*Wireless Sensor Network*
dBm	*Decibel Miliwalt*

LISTA DE SÍMBOLOS

μ	Micro
$\in$	Pertence

SUMÁRIO

1 INTRODUÇÃO

O uso da comunicação sem fio para comunicar elementos como sensores e atuadores é chamado de "Redes de Sensores Sem Fio"(RSSF). As RSSF, no contexto da internet, já são uma realidade. Atualmente, é possível conectar eletrodomésticos, *smartphone*, veículos, dentre outros equipamentos munidos de sensores e atuadores, seja em ambientes públicos ou residenciais. O resultado disto é um novo paradigma, frequentemente citado como "Internet das coisas"(IdC) ou em inglês *Internet of Things* (IoT) (AL-FUQAHA et al., 2015). A Internet das Coisas é considerada uma nova revolução tecnológica, impulsionando pesquisas em diversas áreas, notadamente na área de Redes de Sensores sem Fio, resultando no surgimento de novos protocolos e normatizações de comunicação.

Na indústria, há um ambiente crítico de comunicação que teve a automação industrial como pivô do seu desenvolvimento (SILVA et al., 2013). Nas redes industriais trafegam dados de automação, controle e sensoriamento que são úteis nas mais diversas aplicações. Estas redes cobrem áreas de difícil acesso, as quais são contidas de equipamentos, gases e fluidos. Os riscos neste ambiente podem gerar prejuízos consideráveis de ordem econômica, de meio ambiente, além de riscos aos operadores. Trata-se de um ambiente no qual predominam os meios de comunicação guiados, como os cabos de cobre e fibra óptica (MACEDO; SILVA; GUEDES, 2013).

Apesar do seu alto custo de implantação, as redes cabeadas garantem os requisitos de segurança exigidos pela indústria, devendo isto aos seus protocolos já consolidados. Porém, verifica-se um aumento significativo do uso das RSSF em ambientes industriais. Convém saber que qualquer dispositivo que execute um protocolo de camada de enlace segundo (KUROSE; ROSS, 2013) é conhecido como "nó de rede". Nas RSSF, os sensores e atuadores são chamados de "nós-sensores", normalmente embarcados em pequenos dispositivos munidos de memória e processamento limitados, equipados por rádio e alimentados por baterias (NAKAMURA; LOUREIRO; FRERY, 2007). Dentre os benefícios do uso das RSSF estão o tempo de implementação, custo de instalação e o acesso a áreas de risco nas quais o sistema cabeado seria inviável. Porém, para que as RSSF possam se consolidar no ambiente das redes industriais, é necessário garantir os requisitos de dependabilidade (AVIZIENIS et al., 2004) demandados pela indústria, além de enfrentar os próprios desafios inerentes à comunicação sem fio.

Na busca de atender às demandas de dependabilidade, várias pesquisas estão em curso gerando vários protocolos com estas características. O protocolo FTE-LEACH descrito em (OLIVEIRA, 2015) visa incorporar características de tolerância a falhas e melhorar a eficiência energética de seu predecessor, o protocolo LEACH (HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2002). Ambos são protocolos de roteamento hierárquicos baseados em

cluster.

Outras tecnologias contribuem para o alcance de dependabilidade, como por exemplo, a tecnologia *WirelessHART* (CARLSON et al., 2012) que tem como base o já consolidado padrão HART, bastante utilizado em redes cabeadas industriais. Em (NOBRE et al., 2010) o autor utiliza o padrão *WirelessHART* para implementar técnicas de escalonamento de mensagens utilizando a tecnologia de múltiplo acesso por divisão de tempo, o TDMA, como forma de evitar perda de dados e melhorar a eficiência da comunicação sem fio. Para tanto o autor validou sua implementação criando sua camada de enlace *WirelessHART* no *Network simulator (NS-3)* (NSNAM, 2015), contribuição que será utilizada como base para este trabalho nas camadas inferiores.

Considerando as características do FTE-LEACH e do *WirelessHART* no aspecto de utilizarem o mesmo protocolo de comunicação sem fio IEEE 802.15.4, esta pesquisa utiliza as duas tecnologias para inserir dependabilidade a protocolo de roteamento hierárquico, para as Redes de Sensores Sem Fio Industriais (RSSFI).

1.1 Motivação

Alguns protocolos de roteamento estão surgindo para suprir a demanda das redes industriais de sensores sem fio, e segundo *Winter* (WINTER, 2012) a dependabilidade é um fator preponderante para garantir a resiliência destes protocolos, os quais devem ser testados ao máximo antes de serem postos no mercado e na indústria. Para tanto, a simulação em ambientes controlados é de fundamental importância, visando adaptar cenários e testar as ideias que surgem na academia.

Os simuladores discretos de rede propõem em seus módulos adaptar o máximo de cenários possíveis, inclusive disponibilizando modelos de e propagação de sinais. O NS-3 *Network Simulator* 3 possui uma gama de ferramentas, das quais os pesquisadores podem utilizar-se de técnicas de programação para criar os módulos de simulação e emitir resultados. O protocolo FTE-LEACH é um dos protocolos que surge baseado no protocolo LEACH e que ainda não foi testado em ambientes de simulação específicos para redes. O padrão *WirelessHART* contribui com suas técnicas no uso do TDMA e com seu módulo de simulação NS-3 servindo como base para o encapsulamento do módulo FTE-LEACH, visando obter resultados condizentes com a realidade.

1.2 Objetivo

Prevendo-se a iminente adoção em larga escala da comunicação sem fio no ambiente industrial e dos requisitos de dependabilidade intrínsecos nas normas e recomendações para o uso das RSSFI, o objetivo deste trabalho é incrementar técnicas de dependabilidade ao protocolo FTE-LEACH e adicionar as garantias de comunicação proposta por (NOBRE, 2015), já implementadas

no *WirelessHART* em NS-3, além de validar o módulo de simulação FTE-LEACH para o NS-3 desenvolvido no grupo de pesquisa GSET- Grupo de Sistemas Embarcados e de Tempo Real.

1.3 Estrutura da Dissertação

O documento está estruturado da seguinte forma: no Capitulo 2 é apresentado um detalhamento do roteamento em redes de sensores sem fio no ambiente industrial, bem como dos protocolos IEEE 802.15.4; neste capítulo também é feita uma descrição do roteamento em RSSF, um estudo sobre o protocolo LEACH e algumas de suas derivações, dentre estas o FTE-LEACH, ao qual é dedicada uma subseção traçando suas principais características e modos de operação. Há também uma subseção para o padrão *WirelessHART* o qual serve de base para o encapsulamento do FTE-LEACH nesta pesquisa. No Capítulo 3, é feito o levantamento dos critérios de dependabilidade e sua taxonomia, como também é levantada as principais técnicas para cumprir os critérios de dependabilidade em RSSF e as principais falhas recorrentes em RSSF. No Capitulo 4, é detalhado a implementação do FTE-LEACH no simulador NS-3 e o uso das técnicas para incrementar a dependabilidade no roteamento FTE-LEACH. Ainda neste capítulo é explanado sobre o simulador NS-3 e sua importância como ferramenta de simulação de rede. No Capítulo 5, são demonstrados os resultados obtidos durante as simulações. Ao passo que no sexto capítulo, apresentamos as conclusões com contribuições e trabalhos futuros para esta pesquisa.

2 ROTEAMENTO EM REDES DE SENSORES SEM FIO (RSSF)

Este capítulo trata da fundamentação teórica das redes de sensores sem fio, incluindo tecnologias e protocolos nos quais se dá a base de conhecimento desta pesquisa.

2.1 Redes de Sensores Sem Fio (RSSF)

As Redes de Sensores Sem Fio (RSSF) ou do inglês *Wireless Sensor Networks – (WSN)*, são redes com capacidade de conectar dispositivos com funções de sensoriamento e controle. Tais dispositivos, quando conectados a uma RSSF, são chamados de "nós"(do inglês *node*) (OLIVEIRA, 2015). Os nós, possuem características próprias de sistemas embarcados e em sua grande maioria, apresentam baixo custo, consumo de energia, consumo de memória e processamento. As aplicações destas redes estão presentes nas mais variadas áreas e o seu alto crescimento decorre da popularização dos elementos comunicáveis presentes na internet das coisas, tais como: *Radio Frequency Identification (RFID), smartphones, Global Positioning System (GPS)* e múltiplos objetos inteligentes (HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2000). Uma grande variedade de sensores pode equipar e coletar dados nestes nós: sensores como medidores de temperatura; luminosidade; umidade; poluição; acelerômetros; pressão; ruído e posição geográfica, entre outros.

As RSSF estão ganhando mercado em aplicações industriais, ambientes que no passado era inimaginável absorver tal tecnologia. O conservadorismo deste ambiente se deve às características críticas dos dados que trafegam em suas redes. A automação industrial exerce funções de monitoramento e controle que impactam diretamente no funcionamento de uma planta industrial, por este motivo o meio de comunicação cabeado sempre dominou este ambiente. Porém o alto custo de implantação e a dificuldade de instalação em áreas de difícil acesso das redes cabeadas estão beneficiando as RSSF, as quais em ambientes industriais são conhecidas como RSSFI "Redes de Sensores Sem Fio Industriais"(MACEDO; SILVA; GUEDES, 2013).

2.2 Desafios das RSSFI

Na indústria os requisitos de confiabilidade requerem normas especificas para operar com redes sem fio. As normas para estas redes visam a garantia da qualidade da comunicação em ambientes industriais.

Há um esforço para se normatizar os requisitos de qualidade para as RSSFI. As recomendações do documento NAMUR-NE 124, norteiam características de interoperabilidade, disponibilidade, confiabilidade, tempo real, segurança de dados, dentre outras. Capacidades

como auto-configuração, auto-organização da rede e garantia de confiabilidade dos caminhos são descritas em (MACEDO; SILVA; GUEDES, 2013).

Na indústria verificam-se aplicações que exigem dezenas e até centenas de nós na mesma RSSFI. Neste ambiente a densidade de nós sensores em uma área geográfica é importante e visa garantir redundância de hardware em áreas inóspitas onde a troca de dispositivos é dificultada. Nas RSSFI a autonomia energética de seus nós determina o tempo de vida útil da rede; portanto, a carga energética dos nós da rede é um recurso a ser preservado e bem gerenciado visando prolongar ao máximo a vida útil.

Muitos nós em uma RSSFI podem se localizar em distâncias consideráveis do nó central dependendo da topologia e, consequentemente, aumentar gasto energético com o incremento de potência de seus transmissores. Somando-se a isto as RSSFI sofrem forte influência do ambiente onde estão inseridas, como temperatura, umidade, interferências eletromagnéticas e exposição a sinais de comunicação de múltiplas faixas de frequências. Isto influencia de maneira negativa na comunicação sem fio. Vários padrões de protocolos estão surgindo no sentido de minimizar estes problemas.

2.3 Topologia das RSSF

As RSSF podem operar em diferentes tipos de topologias, como estrela, árvore, malha ou topologias hibridas, através conexões ponto a ponto ou ponto multiponto. As topologias podem ser formadas a partir de pelo menos dois tipos de nós sensores:

- Dispositivo de função completa (FFD) – Os FFD - *Full Function Device* são nós com maior capacidade de processamento e armazenamento na rede e, além de realizar atividades de sensoriamento, podem receber atribuições adicionais como, por exemplo, vir a ser um roteador na rede, neste caso um "Coordenador PAN *Personal Area Network*.

- Dispositivo de função reduzida (RFD) - Os RFD *Reduced Function Devices* possuem apenas funções básicas de sensoriamento e transmissão de dados, não podendo se comunicar direto com outro RFD e não podendo lhes serem atribuídas, funções de controle na rede ou de roteamento. Dispositivos RFD possuem uma estrutura de memória e processamento limitada, sendo necessário a otimização nos gastos destes recursos.

A função de cada tipo de nó pode ser melhor compreendida quando observada a sua posição na topologia. Pelo menos quatro tipos de topologias são empregadas em redes RSSF (FALUDI, 2010), Figura 1

- A topologia em ponto a ponto: é a comunicação entre apenas dois nós. Um dos nós assume a função de coordenador da comunicação e o outro é gerenciado, podendo ser do tipo RFD

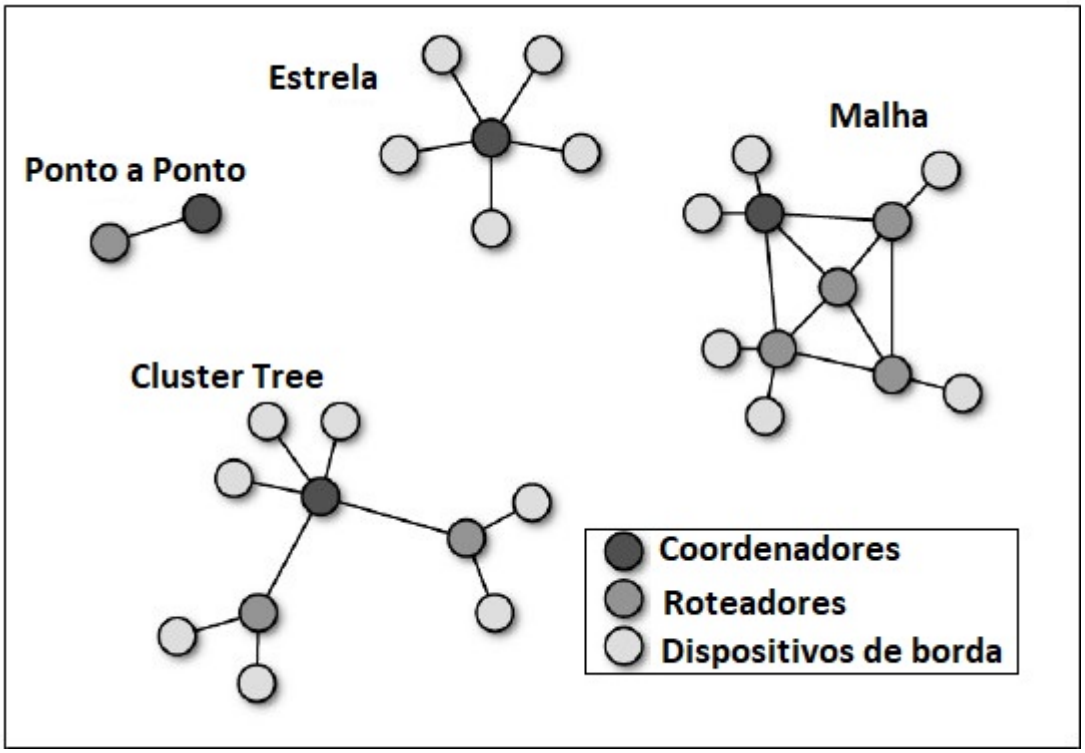
ou FDD. Uma das principais funções desta coordenação é o gerenciamento de acesso ao meio físico.

- A topologia estrela: há um nó coordenador do tipo FDD, que centraliza as conexões de toda a topologia e se torna um nó crítico na rede. Todos os dados necessariamente é roteado através do nó central e redirecionado ao destino. Novamente o nó coordenador assume a função de gerenciar o acesso ao meio físico e os demais nós desta estrutura podem ser do tipo RFD ou FDD.

- A Topologia em malha: podem ser acrescentados nós FFD com função de roteamento na estrutura, entre o nó coordenador raiz ou PAN e os nós RFD. Estes nós com função de roteamento podem gerar múltiplos caminhos entre outros nós roteadores até o nó coordenador PAN.

- Topologia em árvore de *cluster*: este tipo de topologia possui um nó coordenador e nós de roteamento intermediário (sub-coordenadores), podendo cada nó FFD comunicar-se com aglomerados de nós de função reduzida RFD e formar conjuntos conhecido como *cluster*, de maneira hierárquica onde o nó coordenador assume a posição de raiz desta árvore.

Figura 1 – Topologias possíveis



Fonte: Adaptada (FALUDI, 2010).

O padrão IEEE 802.15.4 ou *LR-WPAN* é um protocolo bastante utilizado para dar suporte a estas topologias na camada física e sub-camada MAC do modelo OSI *(Open Systems Interconection)* e que será detalhada a seguir.

2.4 IEEE 802.15.4

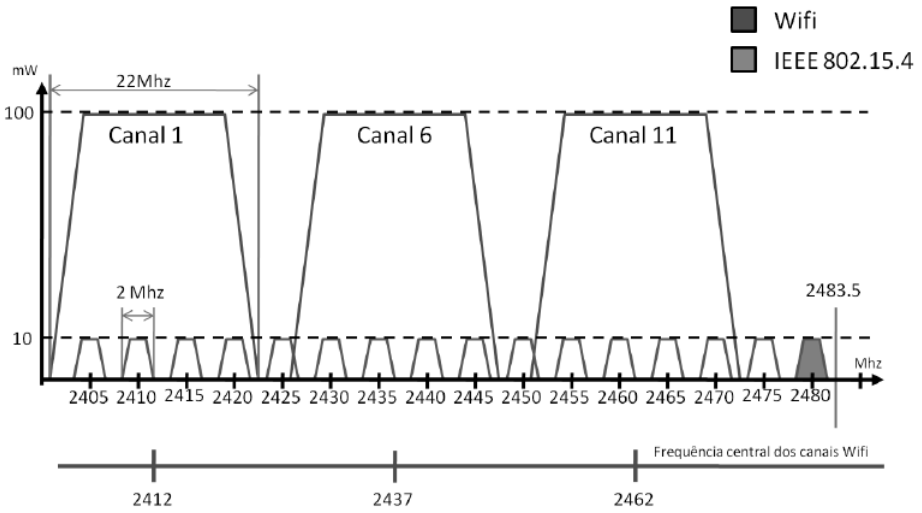
Nas RSSF, é comum o uso de sensores e atuadores em ambientes nos quais a potência de transmissão e o consumo de energia devem ser reduzidos. As redes do tipo LLN (do inglês *Low-power and lossy networks*) são redes com características de baixo consumo de energia e perda de dados, além de limitadas quanto a taxa de transferência e potência de sinal. O grupo de trabalho IEEE 802.15.4 (2015, 2006) foi criado para definir o padrão de comunicação nas camadas físicas e sub-camada MAC do modelo OSI *(Open Systems Interconection)* das redes LLN e este padrão vem se consolidando com sua tecnologia de baixa perda e de fácil acesso econômico. A primeira versão do padrão IEEE 802.15.4 foi relatada em 2003 no grupo de trabalho do *Institute of Electrical and Electronics Engineers* (IEEE) (2015, 2006), com revisões nos documentos IEEE 802.15.4-2006, IEEE 802.15.4-2012, IEEE 802.15.4-2015. Mais recentemente, em 2016, foram publicados as versões IEEE 802.15.4g, visando atender áreas no campo da medicina, e a versão IEEE 802.15.4n, para baixíssimas potências.

Visando atender as demandas crescentes da sociedade, o padrão está em constante evolução. O documento IEEE 802.15.4 de 2006 especifica limites de até 250 Kbps de banda operando a 2.4 GHz de frequência, 40 Kbps a 916 MHz e 20 Kbps a 858 MHz. Este padrão ainda possui variações 802.15.4a até 802.15.4n, ampliando o estudo nas camadas físicas e de enlace para quantidade de canais chegando a 27 canais. Para acrescentar suporte a mobilidade e comunicações em real-time, os rádios que trabalham com o protocolo Xbee (FALUDI, 2010) atendem as especificações do padrão IEEE 802.15.4 com taxa de 250 kbps, alcance na faixa de 100m *out-door* podendo ir além para alguns tipos de cenários e 30m *in-door* com obstáculos.

A frequência 2.4 GHz está sendo utilizada praticamente em todos os continentes e também por outros protocolos, por ser uma faixa não licenciada na maioria dos países. A tecnologia *wifi* IEEE 802.11 utiliza esta faixa em grande escala, porém com potências de operação bem mais elevadas se comparado com IEEE 802.15.4. Outra comparação interessante é quanto a largura de canais alocados no espectro de frequência. O protocolo wifi trabalha com canais de largura de 22 Mhz, enquanto o IEEE 802.15.4 com largura de 2 Mhz de sinalização e 3Mhz de banda de guarda. Não apresenta, portanto, sobreposição de canais. O IEEE 802.15.4 por usar canais não sobrepostos, menor potência e utilizar o método *Direct Sequence Spread Spectrum*-(DSSS) de sinalização, é visto nos aparelhos wifi como um ruído branco. Uma comparação do wifi IEEE 802.11 com o protocolos IEEE 802.15.4 é retratada na Figura 2.

O IEEE 802.15.4, quando operando na faixa de 2.4 GHz, possui capacidade para até 102 octetos de *bits* em sua área de dados. Com a área de dados limitada, o padrão IEEE 802.15.4 fica impossibilitado de comunicação direta com a internet, por não suportar o *Maximum Trasmission Unit* (MTU) de 1280 octetos de *bits* do datagrama Ipv6 em sua área de dados de acordo com a RFC 2460 (DEERING STEVE E HINDEN,). O grupo IEEE 802.15.4 não define as camadas superiores que tratam roteamento ou demais camadas. Portanto, o protocolo IP em nenhuma de suas versões não pode ser implantado diretamente em seu encapsulamento. Na

Figura 2 – Canais wifi IEEE 802.11 x IEEE 802.15.4



Fonte: (NOBRE, 2015)

indústria vem surgindo outros protocolos que buscam cumprir os requisitos de roteamento e garantam dependabilidade e eficiência energética. Dentre estes protocolos temos o LEACH (HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2000) e suas variações como é o caso do FTE-LEACH (OLIVEIRA, 2015).

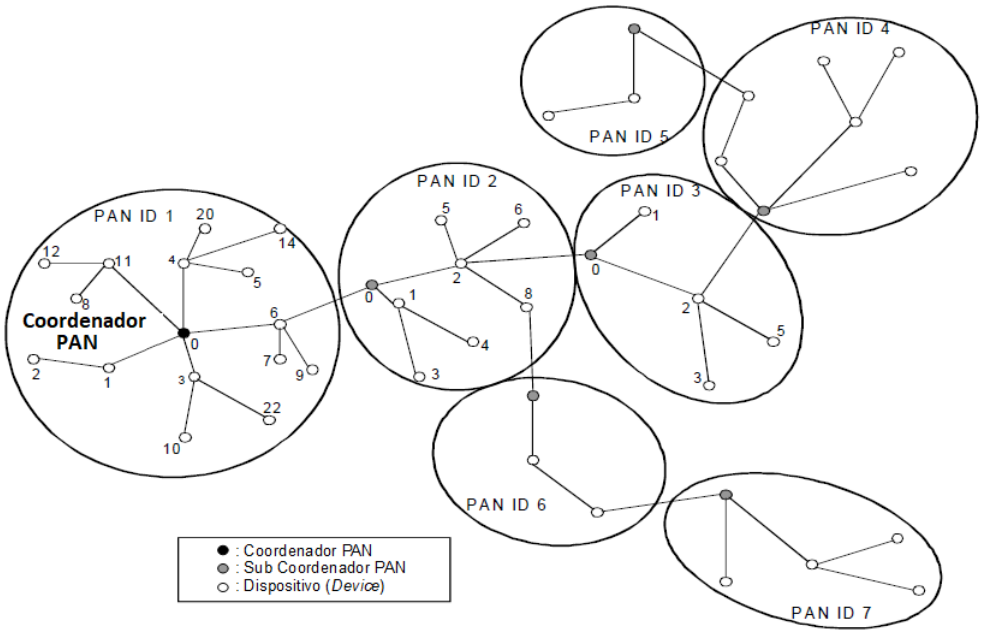
As redes IEEE 802.15.4 possuem três tipos de nós ou dispositivos (IEEE, 2006):

- Coordenador PAN: Com maiores recursos de processamento e armazenamento, possui maiores atribuições na rede, podendo criar redes em árvore e comutar pacotes entre elas. Apenas um coordenador PAN é atribuído para cada rede e deve ser um nó do tipo FFD (*Full Function Device)*) onde possui requisitos de hardware proporcional às suas funções, geralmente possuindo alimentação energética contínua, está representado na figura 3 como "*first* Coordenador PAN".

- Sub coordenador PAN: possui uma coordenação setorizada e é geralmente usado em redes em *cluster* ou *cluster-tree* para coordenar um *cluster*, devendo fazer comunicação entre os nós sensores do *cluster* e o coordenador PAN ou "*first PAN* ". Deve ser um nó do tipo FFD *(Full Function Device)* onde possui requisitos de hardware proporcional às suas funções. Na figura 3 está representado como coordenador PAN dentro de cada *cluster*.

- Nó Sensor: Podendo ser um dispositivo do tipo FDD ou RFD, possui função de sensoriamento e produz dados de leitura para alimentar as aplicações da rede. Na figura 3 está representado como *Device*, podendo haver vários em cada *cluster*.

Figura 3 – Topologia IEEE 802.15.4



Fonte: Adaptada (IEEE, 2006).

2.5 Roteamento em RSSF

Há vários protocolos de roteamento em ambientes de RSSF para LLN (WINTER, 2012), dependendo da aplicabilidade do protocolo, pode-se utilizar recursos variados para traçar o melhor caminho entre um nó de origem e um nó de destino. Os protocolos de roteamento podem ser classificados como a seguir (SHARMA; ALAM, 2012):

- Protocolos de roteamento pró-ativos: agem constantemente atualizando suas tabelas de roteamento, trocando mensagens entre seus nós. Por este motivo necessitam de recursos de energia, memória e processamento para armazenamento e consulta de tabelas de roteamento. Tais recursos crescem à medida que cresce a quantidade de nós envolvidos e de acordo com a topologia da rede.

- Protocolos de roteamento reativos: este tipo de protocolo só atualiza seus dados, mediante uma consulta de rota realizada por algum nó. Neste momento é realizada troca de mensagens entre os nós até que todos os nós envolvidos tracem a rota entre o nó de origem e o nó de destino. Em geral este tipo de roteamento não necessita ou necessita apenas parcialmente do armazenamento de tabelas de roteamento.

- Protocolos de roteamento híbridos: os protocolos híbridos assumem características de protocolos proativos e características reativas. Geralmente empregados em redes hierárquicas em formato de arvore, os nós estão separados em zonas. Cada zona possui um nível de hierarquia no qual as zonas mais próximas da raiz exercem o papel de protocolo reativo e as mais distantes agem de maneira proativa.

2.6 Protocolo LEACH

No intuito de prolongar a vida útil das redes de sensores sem fio, o *Low-Energy Adaptive Clustering Hierarchy-(LEACH)* foi proposto por (HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2000) visando otimizar a dissipação total de energia nas RSSF de baixa potência e também propondo distribuir uniformemente a carga de energia entre os sensores na rede (HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2000), além de também visar escalabilidade e robustez para as RSSF cada vez mais dinâmicas.

O protocolo LEACH implementa uma topologia hierárquica baseada em *clusters*, sendo cada *cluster* possuidor de dispositivos de sensoriamento denominados "nós sensores", sendo um desses eleito como coordenador *Cluster-Head* (CH) e age como um sub-coordenador PAN. Outro dispositivo importante nesta topologia hierárquica é a *Base Station* (BS) responsável pelas funções de um coordenador PAN ou *first* PAN; por fim, os demais nós são os *Cluster-Members*-(CM), nesta estrutura os nós são dispositivos do tipo FFD.

- *Base Station* ou (BS): é o nó raiz da hierarquia e faz a interação da RSSF com o meio externo, sendo responsável inclusive de conectar RSSF com as redes de protocolos da estrutura OSI na terceira camada.
- *Cluster Head* ou (CH): assume a função de coordenador de um *Cluster*, realizando agregação de dados e roteamento dos CM's para a BS.
- *Cluster Member* ou (CM): assume a função de sensoriamento remoto, é pertencente a um *cluster* e está associado a um CH coordenador.

A proposta de prolongar a vida útil da rede possui duas etapas: uma etapa de configuração e uma etapa de comunicação. Na etapa de configuração ocorre a formação dos *clusters* onde é eleito um CH que será responsável por rotear os dados do CM na etapa de comunicação. O CH é um nó crítico na rede e, quando inativo, os CM só poderão direcionar os nós para BS na próxima rodada. Na etapa de comunicação os dados de sensoriamento produzidos nos CM são redirecionados para a BS através de sua comunicação com o CH. Os CH não realizam coletas de dados por seus sensores. Todos os nós da rede são do tipo FDD. Uma rodada completa inclui uma etapa de configuração e uma etapa de comunicação. Uma rodada termina quando o último nó CH envia seus dados para a BS. Ao término de uma rodada inicia-se outra, em cuja etapa de configuração são escolhidos outros CHs baseados na energia residual através do cálculo T(n). Após esta etapa, os demais nós sensores escolhem o CH mais próximo para participar de seu *cluster*. Este rodizio de CH e de CMs leva a rede a uma uniformidade no consumo de energia dos nós.

O protocolo LEACH apresenta algumas desvantagens:

- Cada CH, quando em inatividade, torna-se um ponto crítico na rede, pela ausência de redundância gera perdas de comunicação para os CM pertencentes ao seu *cluster*.
- O uso de *broadcast* em concomitância ao protocolo CSMA-CA gera congestionamentos e perdas de pacotes por colisão, quando da necessidade de pacotes de confirmação.
- Quando o nó passa pela etapa de configuração sem a comunicação com a BS, este nó só participa da rede se alcançado pela BS na próxima rodada. O que gera falha para os nós mais distantes da BS.

2.7 Derivações do Protocolo LEACH

O protocolo *LEACH* é bastante usado no meio acadêmico e há variações que tentam garantir diversas funcionalidades visando aperfeiçoá-lo. Deste mesmo modo é significativo o número de estudos em artigos detalhando o incremento de suas funcionalidades. Em (SINGH; KUMAR; SINGH, 2017) o autor faz um significativo paralelo entre diversas variações do *LEACH*. Podemos classificar estas variações pelo cenário empregado em cada protocolo:

- LEACH centralizado: Neste cenário o protocolo posiciona a BS no centro ou próximo ao centro do cenário monitorado. Neste tipo de cenário, há um certo equilíbrio no gasto energético dos nós sensores. A distância entre a BS e o nó mais distante é menor do que no próximo cenário.
- LEACH setorizado: A BS está posicionada em uma das arestas ou vértice do quadrante a ser monitorado. Neste cenário a distância da BS ao último nó pode ser maior do que no cenário centralizado, o que acarreta um consumo maior de energia. Este cenário é bastante empregado em locais de difícil acesso onde a alimentação energética encontra-se a alguns metros ou dezenas de metros do sensor mais próximo.

  Outra classificação dos derivados do *LEACH* é quanto a quantidade de saltos de roteamento que um pacote realiza ente o nó CM até a BS. Podendo ser do tipo:

- múltiplos saltos ou *multi-hop* : quando os dados produzidos por um nó sensor passa por múltiplos níveis de roteamento até o seu desencapsulamento na BS. Este modelo proporciona uma escalabilidade maior, porém sua complexidade gera um desgaste energético maior.
- Salto único *single-hop*: quando todos os CH possuem comunicação direta com a BS. Não indicado a redes cuja escalabilidade seja preponderante; este cenário tem se mostrado mais eficiente em redes de até 200 nós.

O estudo recente (SINGH; KUMAR; SINGH, 2017) norteia áreas que são tratadas como ponto alvo de novas pesquisas como parâmetros de QoS, tolerância a falhas e segurança. Estes três

pontos possuem extrema ligação com dependabilidade. O estudo aponta que em protocolos baseados em *cluster*, a perda do CH causa maior impacto na rede por afetar diretamente os nós membros. Protocolos que utilizam o método de *re-clustering* como o LEACH-FT foram desenvolvidos para aumentar a confiabilidade da rede e a tolerância a falhas, porém o consumo excessivo de energia também é um fator preocupante. Segundo (SINGH; KUMAR; SINGH, 2017) na gestão de tolerância a falhas, os principais desafios são a detecção e a recuperação de falhas. Algumas tabelas de comparação podem ser observada no estudo (SINGH; KUMAR; SINGH, 2017), porém para esta pesquisa os derivados do LEACH para salto único são de grande contribuição.

Foi visto no estudo (SINGH; KUMAR; SINGH, 2017) que um dos protocolos que melhor trabalhou com eficiência energética foi o Intra-Balanced LEACH protocol for wirelles sensor networks (IBLEACH) (SALIM; OSAMY; KHEDR, 2014). Usa o balanceamento de carga para a função do roteamento, deixando somente a função de coordenação para o CH e, com isso, obteve grandes resultados com a vida útil da rede. Porém o fator dependabilidade não foi o foco deste protocolo. O V-LEACH implementa a ideia de um vice-CH, não obstante ainda utiliza técnicas de *broadcast* em CSMA-CA, o que gera ainda uma grande competição por canal de comunicação em sua fase de configuração.

2.8 Protocolo FTE-LEACH

Esta pesquisa dedica-se a um cenário setorizado em relação a BS, com protocolo derivado do *LEACH* de salto único e com requisitos de dependabilidade FTE-LEACH.

O *FTE-LEACH* ou Protocolo Energeticamente Eficiente e Tolerante a Falhas (OLIVEIRA, 2015) é um protocolo baseado no *LEACH* (HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2000) e opera em modelo híbrido de roteamento. Assim como o *LEACH*, possui uma topologia baseada em *cluster* com incremento de algumas funcionalidades visando melhorar a eficiência energética bem como os aspectos de tolerância a falhas.

Em sua topologia os *cluster* são formados por pelo menos 3 tipos de dispositivos, além do coordenador PAN conhecido neste caso como a *Base-Station* (BS).

A BS possui alimentação energética constante, está conectada a uma rede IP podendo obter interação direta com as camadas superiores de modelo OSI, recebe informação diretamente dos *Cluster-Head* (CH) e assume, portanto, função de coordenador PAN.

O *Cluster-Head* (CH) é um nó do tipo FDD que assume a função de um sub-coordenador PAN e coordena o seu *Cluster*, esse nó é responsável pelo roteamento dos dados provenientes dos nós sensores do tipo *Custer-Member*(CM) com a BS. Para cada *cluster* da estrutura *FTE-LEACH* existe apenas um CH. Os CHs também produzem informações de sensoriamento que são repassadas para a BS, sendo esta uma das contribuições do *FTE-LEACH* em relação ao protocolo *LEACH*.

Apesar de cada *cluster* possuir apenas um CH, existe também a figura do *Vice Cluster Head* (VCH), nó sensor do tipo FDD que em determinado momento pode assumir a função de CH em um *cluster* já formado. O VCH também possui a função de produzir dados de sensoriamento que são repassados para o CH quando este se encontra em atividade. A inclusão da figura do VCH garante um incremento na dependabilidade e cria funcionalidade de tolerância à falha, através da redundância em nível de *hardware*. A inclusão do VCH nesta topologia foi mais uma contribuição do protocolo *FTE-LEACH* (OLIVEIRA, 2015), em relação ao seu predecessor *LEACH* (HEINZELMAN; CHANDRAKASAN; BALAKRISHNAN, 2000).

Cluster Member (CM) são nós sensores do tipo FFD responsáveis apenas por produzir informações de sensoriamento, e como membro de um *cluster*, repassa seus dados ao CH. Nós do tipo CM não trocam informações entre si, mas obedecem à estrutura hierárquica do *cluster* enviando dados somente para o CH. Na figura xx possui um exemplo da topologia *FTE-LEACH*.

O funcionamento do protocolo *FTE-LEACH* propõe no documento (OLIVEIRA, 2015) algumas melhorias para o protocolo *LEACH*, que visam implementar mecanismos adicionais de tolerância a falha e contribuir com uma melhoria em sua eficiência energética. Para dar inicio à criação da rede de sensoriamento baseada em *FTE-LEACH* é necessário o cadastramento de todos os dispositivos nós sensores que participarão da rede. Estes dispositivos devem estar equipados com baterias, instrumentos de sensoriamento produtores de dados, rádio TX-RX compatível com o protocolo IEEE-802.15.4 e o mínimo de recursos de processamento e armazenamento. O FTE-LEACH possui duas fases: fase de configuração e fase de comunicação.

- Fase de configuração: na fase de configuração a BS através de trocas de mensagens reconhece todos os nós da rede e elege uma quantidade (K-ótimo) de CH. Cada CH será o líder de um *cluster*. Os nós CM escolhem através do melhor sinal o CH que irá se associar naquela rodada. O CH elege um de seus nós CM para ser seu VCH, gerando uma redundância intra-*cluster* para a fase de comunicação. Quanto ao método de acesso ao meio o FTE-LEACH utiliza o *Carrier sense multiple access with collision avoidance*(CSMA-CA) como modelo de acesso ao meio para o envio de pacotes *broadcast* e*unicast* da fase de configuração. Neste momento temos os *cluster* formados e inicia-se a fase de comunicação.

- Fase de Comunicação: na fase de comunicação cada CM possui a função de coletar dados e envia-los ao CH de seu *cluster*, este envio é feito através de TDMA em seu *time-slot* de utilização para transmitir os pacotes de dados sensoreados sem o risco de colisões. Um estudo comparativo em RSSF sobre o uso do TDMA foi desenvolvido por (KUMAR; CHAUHAN, 2011). Cada CH além de produzir dados, possui a função de roteamento dos dados para a BS em um só salto. O CH também monitora e realiza agregação de dados antes de enviar seus dados a BS, com isto diminuindo o envio de pacotes. O VCH possui a função de assumir a identidade do CH quando notada sua ausência na rede; neste momento,

o VCH torna-se CH na rede e assume todas suas características até o final da rodada.

2.9 WirelessHART

O *WirelessHART* (CARLSON et al., 2012) é um dos padrões atuais de comunicação sem fio na indústria e está descrito no documento IEC 62591 (**??**). Teve como seu predecessor o padrão *HART* utilizado em larga escala nas redes cabeadas no ambiente industrial. O padrão *HART* surgiu na década de 1980 e durante esse tempo foi agregando uma gama de funcionalidades, destas incluem transmissões de dados proativas, notificação de eventos, transferência de dados em blocos, segurança e diagnósticos avançados. Na versão sete do padrão *HART* foi incluída a sua variação de comunicação sem fio, chamada de *WirelessHART*. O *WirelessHART* agrega as funcionalidade do padrão IEEE 802.15.4, opera na frequência 2.4GHZ e utiliza rádio DSSS, já compatível com o padrão e o FRSS utilizado para chaveamento de canais, pacotes a pacote. Em sua topologia*WirelessHART* figuram:

- Os dispositivos de campo *(fiel-device)*: dispositivos simples que atuam como sensores ou atuadores;
- Roteadores: que realizam a função de roteamento de pacotes da rede sem fio ao adaptador de conexão à rede IP;
- Adaptadores: conecta o dispositivo *HART* legado (cabeada) à rede sem fio;
- Dispositivos portáteis: são dispositivos móveis utilizados por usuários;
- Ponto de acesso: conecta o dispositivo de campo ao *gateway*.
- *Gateway*: Funciona como intermediário com a aplicação.
- *Network* Manager: Gerencia o escalonamento por divisão de tempo na rede.

O padrão *WirelessHART* também utiliza TDMA como modelo de acesso ao meio físico e, com isso, evita erros de comunicação por colisão de pacotes, gerenciando o tempo de transmissão e recepção de dados e seu canal de frequência utilizado. Cada unidade de tempo é conhecida como *time-slot*, unidade que deve ser fixa e síncrona. O *WirelessHART* é um padrão que pode atuar desde a camada física até a camada de interação com o usuário, porém a partir da camada de rede pode ser adaptada com diferentes protocolos, dentre eles o de roteamento. Em (NOBRE; SILVA; GUEDES, 2015) foi implementado o módulo *WirelessHART* para o simulador NS-3, visando a implementação de escalonamento de mensagens sobre o protocolo IEEE 802.15.4 e utilizando o TDMA como modelo de acesso. No documento, o autor detalha como é dividido o uso do tempo em cada *time-slot*. Somado, um conjunto fixo de time-slot forma um *superframe*. Neste caso em um sistema de comunicação um *superframe* repete-se ao longo do tempo durante todo o processo deste ciclo comunicativo, porque em uma comunicação TDMA, os *superframes* podem possuir

diferentes tamanhos, porém o seu funcionamento em paralelo deve evitar sobreposições de canais no mesmo *time-slot*. Um nó participante da rede pode operar em mais de um *superframe*.

Cada *superframe*, pode ser atrelado a diferentes etapas da comunicação ou a diferentes tipos de dados sensoreados na rede. Os diferentes *superframes* devem possuir uma sequência harmônica em seus tamanhos variados, medido em quantidade de *time-slot* e deve seguir a expressão: $(axb)^n$, onde a e b são constantes e n um numeral natural qualquer. A possibilidade de manipular os tamanhos dos diferentes *superframes* possibilita a otimização do tempo em cada etapa de configuração do FTE-LEACH e evita um alto número de *time-slot* ocioso.

O processo de escalonamento descrito em (NOBRE; SILVA; GUEDES, 2015) foi validado utilizando diferentes métodos de propagação de sinais, exemplo *low distance*, *long distance* e modelo de *friis*. Já o modelo de erro adotado foi *Gilbert/Elliot*: Modelo nativo do NS-3. O *WirelessHART*, por deixar a camada de roteamento aberta aos utilizadores, permite que seu uso seja atribuído para diferentes fins e neste trabalho será utilizado o módulo desenvolvido por (NOBRE, 2015) para encapsular o roteamento do *FTE-LEACH* visando ampliar as características de dependabilidade e eficiência energética .

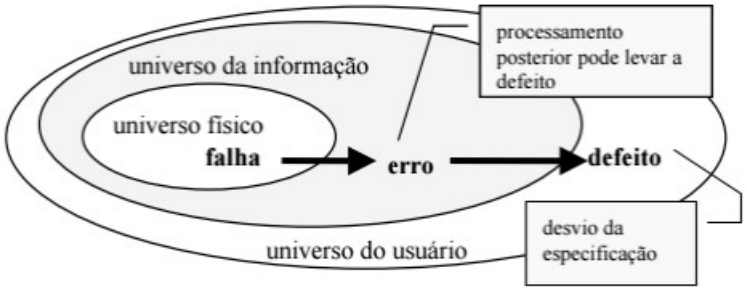
3 DEPENDABILIDADE

Este capítulo tem o objetivo elucidar os conceitos de dependabilidade necessários a esta pesquisa e entender a taxonomia envolvida e normatizar a nomenclatura utilizada para este texto.

3.1 Falha, Erro e Defeito:

Como premissa deste estudo é de suma importância saber diferenciar os termos "defeito", "erro"e "falha". Segundo Tayse em (WEBER, 2003), “falhas estão associadas ao universo físico, erros ao universo da informação e defeitos ao universo do usuário”. Falha (*fault*) é a origem do problema, seu processamento posterior pode causar um erro, e quando este erro quando é percebido em nível de usuário, mostra-se como um desvio da especificação em forma de defeito (*failure*). Uma falha pode não ser percebida a priori pelo usuário e pode ser incapaz de causar um erro, porém todo erro resulta de uma falha (PRADHAN, 1996). A autora Tayse modela estes conceitos como na Figura 4 (WEBER, 2002).

Figura 4 – Modelo de 3 universos: falha, erro e defeito



Fonte:(WEBER, 2003).

De acordo com o tempo de vida a falha podem ser classificadas em:

- Transitória: sua duração é limitada, causada em geral por mau funcionamento temporário ou em consequência de alguma interferência externa. Em geral de difícil diagnóstico.
- Intermitente: falha de curta duração, ocorre de maneira repetida em um curto intervalo de tempo. Sua detecção também é dificultada pela temporariedade.
- Permanente: Falha de duração continuada.

As falhas também podem ser classificadas de acordo com a fase em que são introduzidas:

- Projeto: Ocorre na fase do projeto de um sistema computacional.
- Operacional: Ocorre em tempo operacional do sistema.

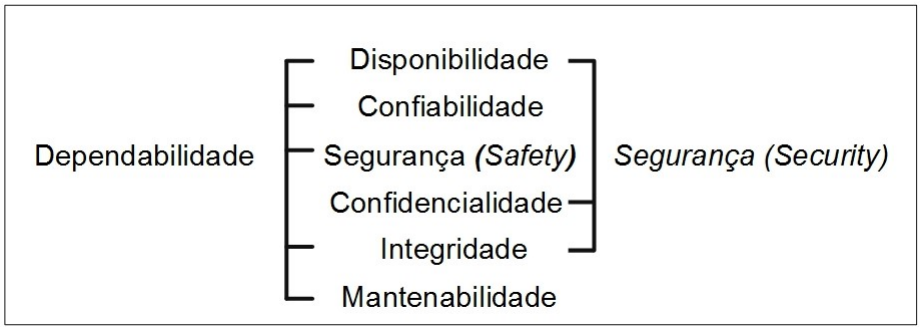
Para evitar-se ao máximo que falhas venham causar erros com possíveis defeitos impactando o usuário do sistema ou mesmo a outros sistemas os diversos tratamentos sobre este assunto são classificados na taxonomia de (AVIZIENIS et al., 2004).

3.2 Taxonomia:

O conceito de dependabilidade é uma tradução literal do inglês *dependability*, que significa a qualidade e confiabilidade que um serviço computacional possui (WEBER, 2002). Para entender como a dependabilidade pode ser alcançada em redes de sensores sem fio é importante entender os atributos que levam um sistema computacional ao nível de dependabilidade. Importante salientar que a dependabilidade não é um índice que possa ser medido ou quantificado diretamente. Segundo (WEBER, 2003) todos os atributos de dependabilidade correspondem a medidas numéricas. A nomenclatura conceitual de dependabilidade e seus atributos foi descrita por (PRADHAN, 1996) que os listou da seguinte forma: confiabilidade, disponibilidade, segurança de funcionamento (*safety*), segurança (*security*), mantenabilidade, testabilidade e comprometimento do desempenho (*performability*). Nesta nomenclatura a segurança (*security*) se tornava simplista para o estudo da segurança da informação, onde possui seus pilares nos atributos de disponibilidade,integridade e confidencialidade.

Visando normatizar esta nomenclatura, (AVIZIENIS et al., 2004) propõe a taxonomia que separa bem os atributos de segurança, em conformidade com o entendimento de dependabilidade e segurança (*security*). Na Figura05 é ilustrada a separação tal qual o autor define.

Figura 5 – Atributos de Dependabilidade x Segurança (*security*)



Fonte: Adaptada (AVIZIENIS et al., 2004).

Nesta nomenclatura o autor trata os seguintes atributos de dependabilidade:

- Disponibilidade: estar operacional em tempo determinado; alternância de períodos de funcionamento e reparo.
- Confiabilidade: atender especificação dentro de condições de tempo e funcionamento.
- Segurança de Funcionamento (*safety*): estar operacional e com funcionamento correto ou quando não, não causar impacto em outros sistemas ou pessoas.
- Integridade: a garantia de ausência de alterações impróprias do sistema.
- Mantenabilidade: condições para eficácia das atividades de manutenção.

Já no que diz respeito a segurança (*security*) soma-se ao conceito de disponibilidade e integridade o termo:

- Confidencialidade: a não divulgação não autorizada da informação.

Na segurança (*security*) os três atributos devem estar presentes concomitantemente. Cabe acrescentar ainda que a disponibilidade no contexto da segurança (*security*) é apenas para ações autorizadas, enquanto que a integridade para ações não autorizadas.

Visando melhorar os índices de cada atributo de dependabilidade foram desenvolvidas algumas técnicas. Na próxima secção estão descritas as técnicas para se atingir a dependabilidade.

3.3 Técnicas para atingir a Dependabilidade

Para se atingir níveis de dependabilidade e segurança aceitáveis foram desenvolvidas diversas técnicas que atuam sobre um ou mais atributos de dependabilidade. Este documento possui o foco nas técnicas aplicáveis em comunicações não orientadas a conexão, onde utiliza o protocolo UDP em sua camada de transporte do modelo OSI, não tratando, por exemplo, esquemas de correção de erro ou recuperação de falhas suportadas pelas comunicações orientadas a conexão realizadas na camada de transporte. O autor *AVIZIENIS* (AVIZIENIS et al., 2004) descreve algumas destas técnicas:

- Prevenção de falhas: Significa usar mecanismos com o intuito de evitar a ocorrência ou a introdução de falhas.
- A tolerância a falhas: Significa que no caso da ocorrência de falhas, esta não venha a comprometer a funcionalidade do serviço.

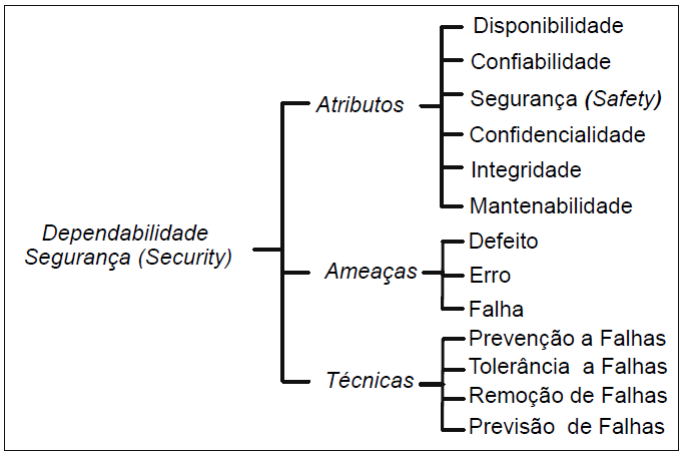
  O conceito de tolerância a falhas, sistemas redundantes, alta disponibilidade são sinônimos no meio acadêmico. A redundância é a principal forma de aplicação de tolerância a falhas (WEBER, 2003). Existem várias formas de redundância que podem ser exploradas:

  - Redundância de Informação: quando a informação é duplicada propositalmente para suprir uma possível perda de dados.
  - Redundância temporal: quando em tempos distintos determinada tarefa é realizada com o mesmo fim.
  - Redundância de hardware: quando se há uma duplicidade de equipamentos que possa suprir a falha de um equipamento primário.
  - Redundância de software: quando a duplicidade de aplicações pode suprir a falha de um determinado aplicativo.

- Remoção de falhas: significa reduzir a quantidade e/ou a gravidade de falhas existentes. Possui o intuito de minimizar a probabilidade que um erro torne a ocorrer.
- Previsão de falhas: significa estimar a quantidade, a incidência futura e as prováveis consequências de falhas.

Na figura abaixo é diagramada a taxonomia dos conceitos acima mencionados.

Figura 6 – Atributos, Ameaças e Técnicas



Fonte: Adaptada (AVIZIENIS et al., 2004).

Não é objetivo deste documento o estudo de questões de segurança do tipo (*security*) que não estejam em concomitância com as exigências para se atingir a dependabilidade aceitável às RSSFI. Para maior aprofundamento em *security* nas RSSF o documento (SEMENTE, 2015) apresenta um estudo sobre o assunto.

No capítulo 4, é descrito como foram implementadas as técnicas de dependabilidade, como contribuição desta pesquisa para alavancar o nível de dependabilidade do protocolo FTE-LEACH no *Network Simulator 3* (NS-3).

4 IMPLEMENTAÇÃO DO *FTE-LEACH* EM NS-3 COM TÉCNICAS DE DEPENDABILIDADE

O *FTE-LEACH* implementa algumas técnicas de dependabilidade descritas no capítulo 3. Esta pesquisa ampliou o estudo e incrementou técnicas que visam atender a criticidade de cenários de monitoramento que necessitam da informação monitorada com o mínimo de perda possível:

- Prevenção de falhas e disponibilidade: Visando uma maior vida útil da rede, o FTE-LEACH utiliza o sensoriamento dirigido a eventos, onde cada nó sensor do tipo CM só envia seus dados para seu *Cluster-Head*, quando em um evento sua leitura ultrapassar o limiar de alteração mínimo em relação a última leitura válida. Isto tem o intuito de diminuir a quantidade de informações de mesmo conteúdo. As leituras de dados cujo teor não ultrapassar o limiar mínimo (que pode ser definido pela aplicação), torna esse dado duplicado e é descartado pelo nó sensor. Quando um nó CM não possuir dados a serem transmitidos em seu *time-slot*, é enviada uma mensagem do tipo *keep-alive* para o CH informando sua participação na rede. Quando determinado CM não está em seu *time-slot* de transmissão dentro do TDMA, ele entra em modo *sleep* e só será acordado na próxima rodada de configuração e leitura. Esta contribuição além de evitar tráfego desnecessário na rede, colabora com o aumento da vida útil de todos os nós.

- Tolerância à falhas: Quanto à tolerância a falhas, uma das técnicas já empregada por (OLIVEIRA, 2015) é a redundância de hardware que foi obtida pelo aumento da densidade de nós por área geográfica. A auto-configuração da rede se torna um desafio quando aumenta-se a quantidade de nós por m^2 na rede. Este problema foi solucionado com a troca de mensagens de configuração realizada em TDMA, onde cada nó possui seu *time-slot* de operação com o canal reservado. Isto também uma prevenção de falhas obtida por se evitar erros através da colisão de pacotes.

  Redundância de *Hardware*: o uso de um *Vice-Cluster-Head* (VCH) é uma forma de redundância de *hardware* implementada pelo *FTE-LEACH* e garante que na ausência do CH, o VCH assuma sua função de CH a partir de seu *time-slot* de transmissão. O VCH é o único nó sensor que aguarda confirmação de recebimento de mensagem por parte do CH, caso esta mensagem não obtenha sucesso o VCH assume que o *Cluster-Head* está inoperante para suas funções de roteamento; neste momento, o VCH assume o endereçamento do *Cluster-Head* e passa a receber os dados dos *cluster-member* nos *slot* posteriores. Esta redundância de hardware diminui o número de pacotes perdidos na fase de monitoramento por parte dos nós CM.

- Técnicas como recuperação de falhas e previsão de falhas serão descritas durante a implementação da solução logo abaixo na seção módulo FTE-LEACH.

No desenvolvimento do protocolo *FTE-LEACH* descrito em (OLIVEIRA, 2015) foi utilizada a ferramenta MATLAB (MATHWORKS, 2005) para simulação e validação em seu primeiro momento. Com esta ferramenta o autor produziu resultados que garantem sua aplicabilidade. Nesta pesquisa prevendo uma evolução do *FTE-LEACH*, há a necessidade da utilização de uma ferramenta de simulação específica para ambientes de redes cuja a prática é comum na criação de novos protocolos de rede (NOBRE, 2015) . No ambiente acadêmico e como detalhado no estudo (NOBRE, 2015) o *Network Simulator 3* (NSNAM, 2015), é uma ferramenta recomendada para validação de protocolos de rede.

4.1 Network Simulation 3 (NS-3)

As etapas de simulação e testes são relevantes para a criação de qualquer tipo de protocolo de comunicação. O NS-3 é um simulador baseado em eventos discretos. Com o foco na pesquisa e no uso educacional. É um software livre desenvolvido para o GNU-Linux e licenciado pela *General Public License version 2*-(GPLv2) onde é incentivada sua distribuição e colaboração para o seu desenvolvimento. O NS-3 não é considerada a versão 3 do popular simulador NS-2, que também funciona baseado em eventos discretos. O NS-2 é projetado para implementar simulações com *scripts* em linguagem OTcl que é uma extensão da linguagem Tcl, voltada para a orientação a objetos, e ainda manterá seu grupo de trabalho. Já o NS-3 possui sua base de *scripts* na linguagem C++ e *python*. Os *scripts* criados para simulações em NS-2 não são compatíveis com o NS-3. O grupo de desenvolvedores do NS-3 está reescrevendo muitos dos módulos já implementados em NS-2 e por utilizar a linguagem orientada a objetos do C++ e pela popularização do uso do *python* o NS-3 vem ganhando destaque no meio acadêmico.

A criação de um novo módulo de simulação em NS-3 não é tarefa das mais simplistas. A falta de módulos de protocolos de camadas inferiores da arquitetura OSI condiciona a criação de praticamente toda a pilha de protocolos até o protocolo estudado. Para esta pesquisa, foi utilizado o módulo *WirelessHART* (NOBRE; SILVA; GUEDES, 2015) nas camadas inferiores do modelo OSI, para encapsular o FTE-LEACH.

O módulo *WirelessHART* desenvolvido por (NOBRE; SILVA; GUEDES, 2015) já utiliza o protocolo IEEE 802.15.4 e implementa o TDMA como principal técnica de modo de acesso, indo ao encontro do aumento da dependabilidade promovida por esta pesquisa para o FTE-LEACH. Este módulo, utiliza em seus testes o modelo de erro de Gilbert/Elliot (SIENA, 2016) em cinco casos de variação, este trabalho separa uma subseção para descrever este modelo de erro o qual também é utilizado nesta pesquisa para garantir resultados próximos ao cenário real de uma planta industrial cujo abalos sísmicos venham a comprometer seu funcionamento.

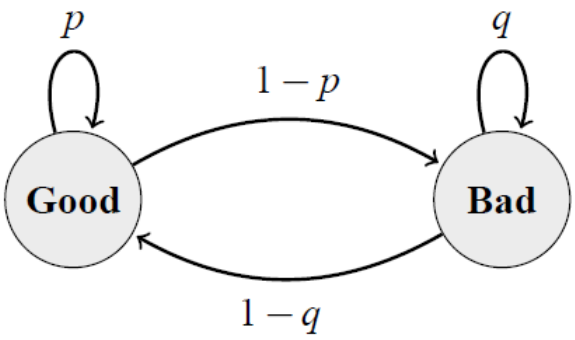
Modelo de Erro

Com a finalidade de testar as possibilidades e aproximar o FTE-LEACH de condições reais de uso, é necessária a adoção de um modelo de erro que venha aferir as condições que um cenário real possa apresentar.

O NS-3 através de sua classe *ErrorModel* oferece algumas possibilidades de modelo de erro para injeção de interferências em redes sem fio. Para melhor comparação dos resultados foi escolhido o mesmo modelo de erro adotado em (MACEDO; SILVA; GUEDES, 2013) Gilbert/Elliot (SIENA, 2016), modelo aceito na academia. Trata-se de uma sub-cadeia de Marcov de dois estados. Uma cadeia de *Markov* (GRIGOLETTI, 2011) leva em consideração apenas o estado presente da simulação, não levando em consideração eventos futuros ou passados em determinado espaço temporal. O modelo pode transitar entre dois estados, sendo atribuídas variáveis a cada estado, que determina a probabilidade de ocorrer a migração para um evento mais próximo de ocorrência de erro ou não.

Nesta situação o estado inicial é o "good", porém a probabilidade de permanência neste estado é dada por Pp e Pq para a probabilidade de estado de erro "bad"como mostrado na Figura 7.

Figura 7 – Gilbert/Elliot



Fonte:(NOBRE et al., 2010).

Para este modelo foram implementados três casos de uso dos cinco apresentados por (NOBRE; SILVA; GUEDES, 2015) e definidos por meio das variáveis de estado Pp e Pq, que definem a probabilidade de um estado *Good* ou *Bad* durante a simulação discreta utilizando os valores da Tabela 1. O efeito rajada citado na tabela é em decorrência do estado estar com um modelo pessimista quanto ao erro Pq retornar ao estado good gerando erros em rajadas.

Tabela 1 – Casos de Simulação

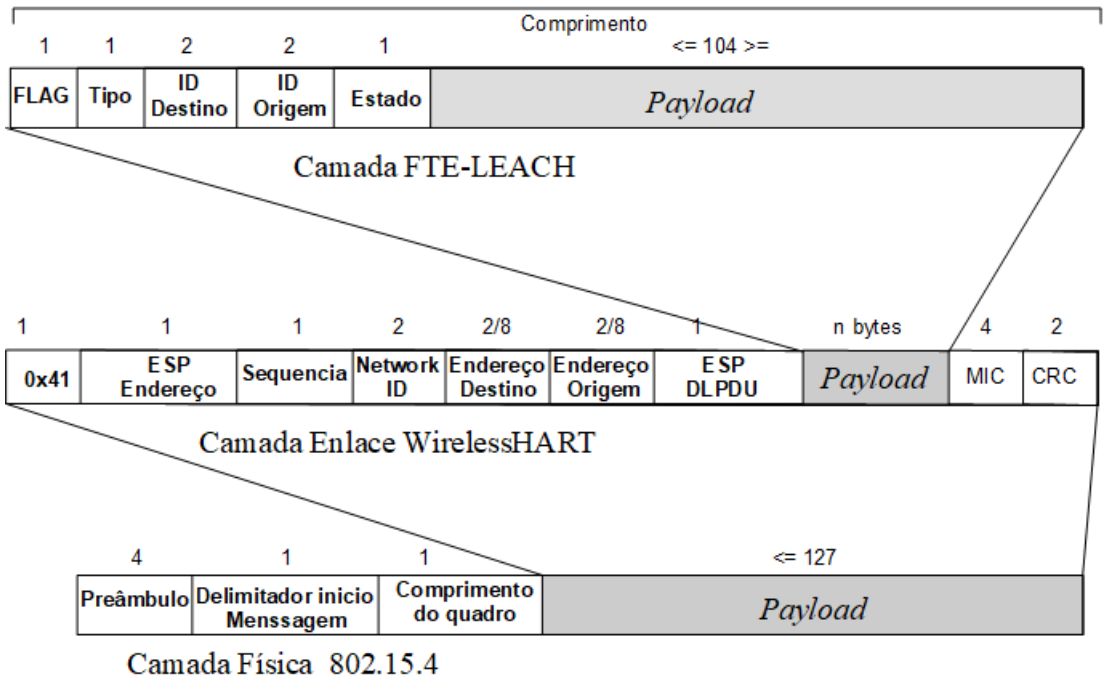
Cenário	Pp	Pq	Descrição
Caso 1	0,9999918	0,999184	Ambiente Externo (WANG; MOAYERI, 1995)
Caso 2	0,9999	0,998	Cenário de erro em Rajada (WILLIG et al., 2002)
Caso 3	0,999	0,98	Cenário de erro em Rajada (BHAGWAT et al., 1997)

Fonte: Adaptada (NOBRE; SILVA; GUEDES, 2015).

4.2 Módulo FTE-LEACH

Para a implementação do módulo *FTE-LEACH* no NS-3 foi utilizado como base de encapsulamento o módulo *WirelessHART* (NOBRE, 2015) de camada de enlace. Este, por sua vez utiliza o padrão IEEE 802.15.4 como protocolo de sub-camada MAC e camada física. Na figura 8 pode-se acompanhar o encapsulamento utilizado nesta implementação.

Figura 8 – Encapsulamento FTE-LEACH WirelessHART



O Pacote FTE-LEACH sugerido possui um cabeçalho fixo de sete *bytes*, o primeiro campo é o "FLAG", que possui a função especifica de identificar e distinguir o protocolo FTE-LEACH dos demais protocolos de roteamento em RSSFI, seu comprimento é de 8 *bits*. O segundo campo refere-se ao tipo de mensagem e determina qual a mensagem a ser enviada. As mensagens referentes a este campo podem variar, de acordo com a Figura 9. O terceiro e o quarto campo, refere-se ao identificador do nó de destino e de origem do pacote e possuem o tamanho de 16 *bits* cada, inclusive podendo seguir o padrão de endereçamento utilizado pela camada *6lowpan* (SHELBY ZACH; BORMANN, 2009) para integração com redes IPv6. O

quinto campo é o estado do nó e identifica o estado do nó de origem naquele momento. O estado de cada nó informa em qual situação o nó está no momento do envio, podendo variar entre: nó BS, nó não alocado, nó CM, nó CH, e nó VCH.

Características das mensagens FTE-LEACH:

- ADV (*Advertise*): possui 23 *bytes*. Este possui seis variantes: ADV01-6, todas estas utilizadas na etapa de configuração da rede.
- ACK (*Acknowlehment*): mensagens de 26 *bytes* de reconhecimento e controle de rede, é utilizado tanto na fase de configuração quanto na fase de comunicação.
- DATA Mensagens do tipo dados: habilita a camada superior à leitura do campo *payload* e seu tamanho pode variar entre 21 a 127 *bytes*, sendo utilizado apenas na etapa de comunicação.
- *Keep-Alive*: mensagens de 21 *bytes* informando que o nó está ativo é utilizado na fase de comunicação.

Figura 9 – Tipos de Mensagens Trocadas pelo FTE-LEACH

Menssagens do campo Tipo		Uso	Tipo de Resposta
ADV (*Advertise*)	01	No pooling da BS para reconhecimento da rede	ACK
	02	Para eleição do CH pela BS	Sem resposta
	03	No pooling do CH para reconhecimento da rede e formação do *cluster*	Sem resposta
	04	Pelo CM em direção ao CH para associação em seu *Cluster*	Sem resposta
	05	Usado pelo CH em para eleição do VCH	Sem resposta
	06	Usado pelo CH para alocação de slot para o CM	Sem resposta
ACK		Pacotes de resposta ao ADV 01	Sem resposta
KEEP-ALIVE		Pacote enviado quando o nó está em estado ocioso, para determinar em seu *time-slot* que o nó está operacional na rede	Quando enviado pelo VCH – recebe um ACK do CH.
DATA		Para envio de dados de sensoriamento	Quando enviado pelo VCH – recebe um ACK do CH.

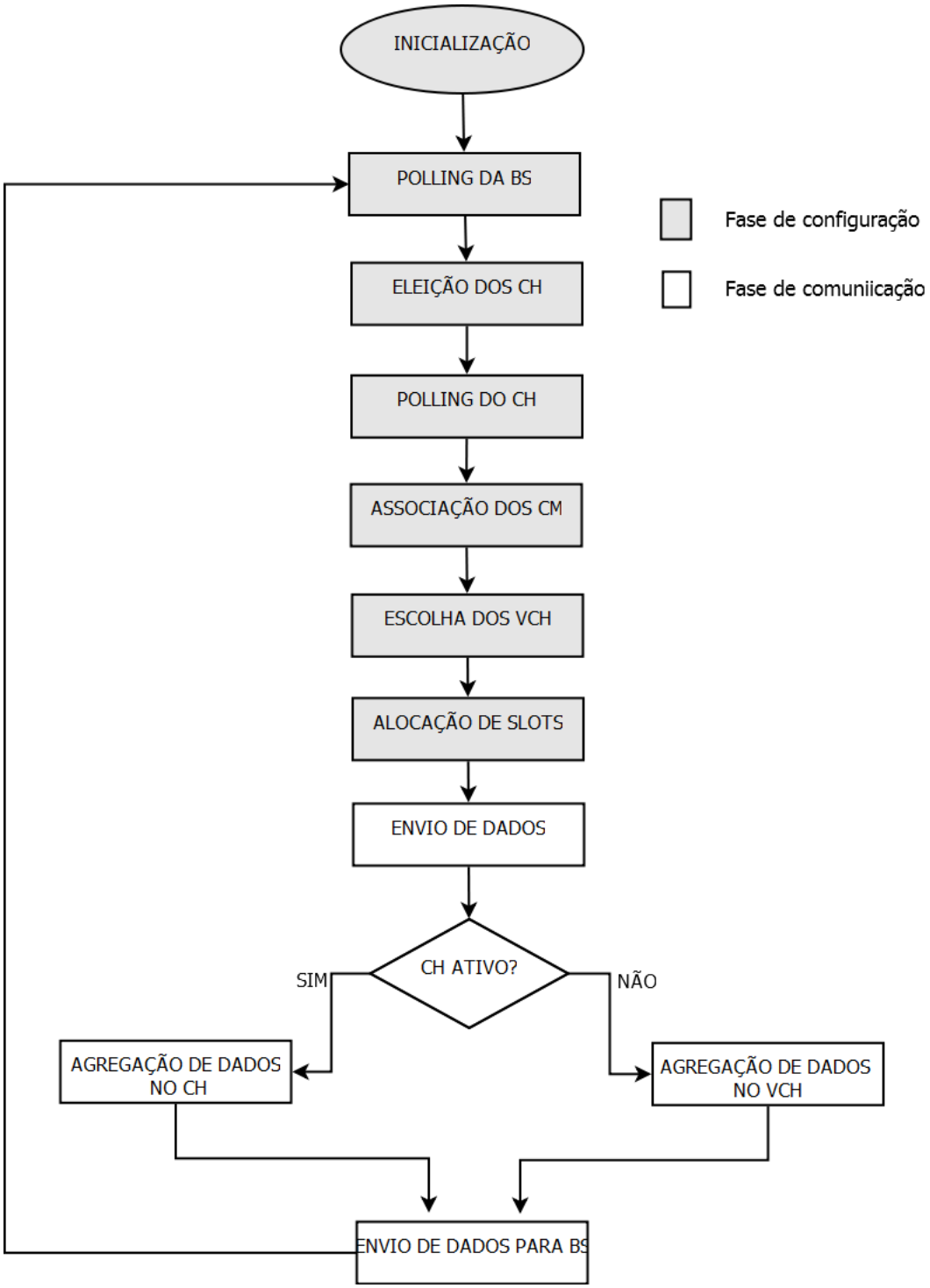
No caso do pacote de agregação do tipo DATA, esse tamanho varia para a simulação entre 21 e 123 *bytes*, entretanto, de acordo com o tipo de aplicação é possível usar até os 127 *bytes* permitidos pelo padrão IEEE 802.15.4, ou mesmo não utilizá-lo e desconsiderar a fusão de dados.

Este documento divide as duas fases do protocolo *FTE-LEACH*, Configuração e Comunicação, em etapas de implementação.

Fluxograma

O desenvolvimento do módulo FTE-LEACH divide as fases de configuração e comunicação em etapas (Figura 10). Os nós que formam a rede habilitam-se a cada troca de mensagens e etapa executada para saber quais as próximas ações, segue em sequencia o fluxograma de acompanhamento do protocolo .

Figura 10 – Fluxograma FTE-LEACH



Etapa de Configuração

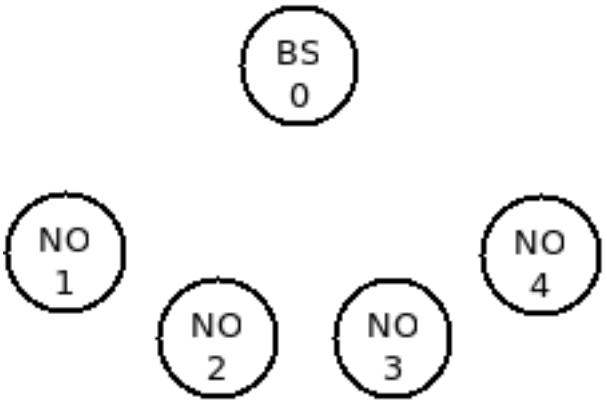
Antes de iniciar o processo de ativação e configuração da rede é importante registrar que todos os nós participantes devem ser previamente cadastrados. Todos os nós deverão estar setados no canal 0, em estado de escuta *idleRX* visando recebimento de pacote ADV01 para sincronização de TDMA.

Para efeito didático, é realizada a demonstração de um cenário com apenas 5 nós, incluindo a BS. Todos os nós são conhecedores dos endereços MAC dos demais nós da rede. A BS geralmente está localizada em uma das arestas ou vértices do quadrante a ser monitorado. Os primeiros nós estão a uma distância entre 5 a 20 metros da BS. Com a exceção da BS, os demais nós estão no modo *idleRx* esperando uma comunicação inicial da BS.

Inicialização

A etapa de inicialização é onde os nós são distribuídos no cenário de monitoramento. Esta etapa inicia com os nós tendo conhecimento sobre os demais dispositivos pré- cadastrados na BS. Todos os nós possuem a lista de endereços dos demais nós pertencentes à rede, por meio do seu endereço *Media Access Control*(MAC). Nessa etapa também todos os nós são ligados e encontram-se em status RX escutando possíveis mensagens advindas da BS para poder sincronizar o TDMA. Neste momento dá-se inicio à fase de configuração do cenário ilustrativo da Figura 11. Para efeitos didáticos usaremos um cenário com 5 nós e apenas um *cluster* irá ser formado.

Figura 11 – Cenário Didático



A tabela abaixo representa os endereços de MAC conhecido antes da etapa de reconhecimento.

Figura 12 – Tabela MAC

ID	MAC	Estado
0	02-08-00:00:00:00:00:00:00:01	RX
1	02-08-00:00:00:00:00:00:00:02	RX
2	02-08-00:00:00:00:00:00:00:03	RX
3	02-08-00:00:00:00:00:00:00:04	RX
4	02-08-00:00:00:00:00:00:00:05	RX

Reconhecimento

A segunda etapa, denominada fase de reconhecimento ou *Polling* da BS, corresponde a BS fazer o reconhecimento de todos os nós na rede. No módulo FTE-LEACH criado e mantendo o foco na dependabilidade, esta etapa já se inicia com o uso do TDMA como modelo de acesso ao meio, evitando assim a disputa por canal de transmissão. A BS inicia este processo com base no *time-slot* alocado para cada nó, levando em consideração a ordem alfanumérica do endereço MAC de cada dispositivo. O FTE-LEACH, descrito em (OLIVEIRA, 2015) os nós que não responderem à mensagem enviada pela BS não participariam da rodada, causando assim uma falha intermitente do nó para a rede. Nesta implementação o FTE-LEACH em NS-3, a BS envia um pacote do tipo ADV1 para os demais nós e recebe uma resposta do tipo ACK com a informação da energia residual do nó. Entretanto, nem todos os nós precisam enviar este pacote, aqueles que não obtiverem um valor T(n) aceitável para candidatar-se como CH não respondem a BS, visando preservar sua energia, evitar a ocupação do canal e diminuir o tempo desta fase de configuração. O cálculo do valor de T(n) é descrito em (OLIVEIRA, 2015) e transcrito na fórmula:

$$T_{(n)} = \begin{cases} \frac{P}{1-P(r \pmod{\frac{1}{P}})} \times \left(\frac{E_i}{E_0}\right) \times \left(\frac{1}{1-((f \times l) \times r)}\right) & \text{se} \quad n \in G \\ 0 & \text{caso contrário} \end{cases} \tag{1}$$

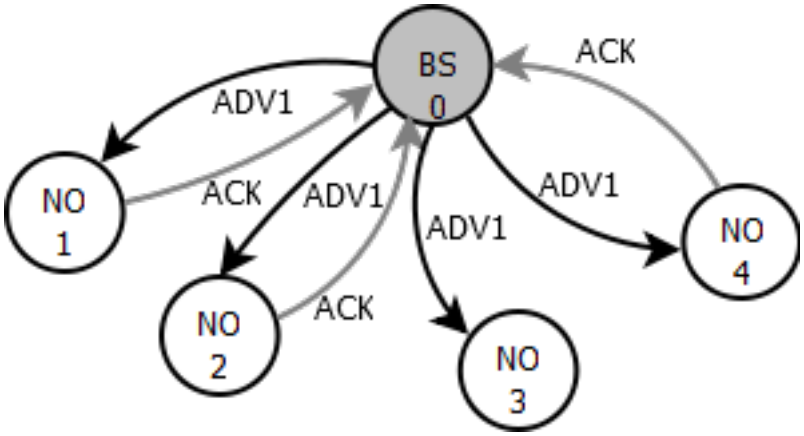
Em que:

- n é o nó candidato a CH;
- $T_{(n)}$ é o limiar do nó;
- P é o percentual desejável de CH de acordo com os estudos apresentados por Heinzelman, Chandrakasan e Balakrishnan (2002);
- r refere-se a rodada do protocolo;

- $E_{(i)}$ Energia atual do nó;
- $E_{(0)}$ Energia inicial do nó;
- f é o fator de dissipação energética do nó;
- l referente a tava de *bits* transmitidos na rodada atual;

Esse processo no FTE-LEACH em NS-3, não impede um nó de associar-se a rede como um CM nas fases seguintes, ele apenas funciona como um mecanismo de prevenção de falhas para a rede, impedindo que um nó sem condições de tornar-se CH venha a ser escolhido. Existe também a possibilidade de o nó enviar o ACK, mas não chegar até a BS, ou mesmo nem receber o ADV1 da BS por estar fora de seu alcance. Nestes casos também este dispositivo ainda pode vir a fazer parte da rede como um VCH ou CM nas etapas seguintes. A Figura 13 demonstra em um cenário de 5 nós o *polling* realizado pela BS com o nó 3 não respondendo a BS por um dos motivos anteriormente citados.

Figura 13 – Polling de Reconhecimento



Cada nó na rede neste momento está com seu status de não alocado, ao receber a mensagem do tipo ADV01 da BS é emitida uma resposta pelo nó por meio de uma mensagem ACK, visando o seu reconhecimento inicial na rede. Este procedimento ocorre com todos os nós da rede.

Cada *slot* de tempo no TDMA implementado pelo *WirelessHART* em NS-3 tem duração de 10 milissegundos (ms), o ADV01 e a resposta ACK são trocados no mesmo *time-slot* conforme descrito na implementação do *WirellesHART* (NOBRE; SILVA; GUEDES, 2015). Neste momento alguns nós podem deixar de receber o pacote ADV01 e não ser reconhecido pela BS ou mesmo a BS deixar de receber um ACK. Isso pode ocorrer por pelo menos três motivos. Erros de propagação de sinal, atenuação do sinal, ou pelo mecanismo de eficiência energética realizado pelo FTE-LEACH. No FTE-LEACH um nó calcula a cada rodada o seu

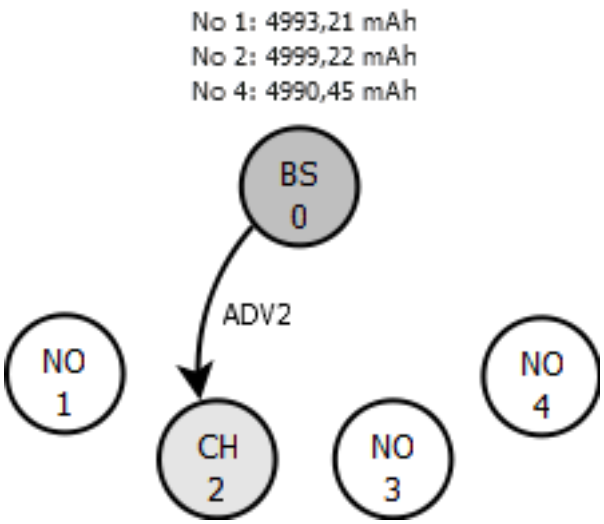
limiar energético visando candidatar-se ou não a CH . Caso seu limiar esteja acima de um percentual desejável a ser eleito CH, o nó envia um ACK com a sua energia residual para BS. Caso contrário, visando preservar suas reservas de energia o nó não envia o ACK para a BS. A BS só considera aptos a se tornar um CH, os nós que respondem o ADV01 com um ACK. Outra contribuição desta implementação é a possibilidade do nó que ainda não foi reconhecido nesta etapa fazer parte da rede como um CM nas etapas seguintes do protocolo. Na Figura 13 verifica-se que o nó 3 não envia o ACK de retorno à BS evitando-se assim seu reconhecimento.

Eleição do CH

Fase 02 eleição do CH: a BS ao receber o ACK dos nós aptos a ser CH, realiza o cálculo do K-ótimo para a quantidade de CHs na rede e inicia o processo de eleição. Nesta fase de eleição dos CHs do FTE-LEACH em NS-3 foi realizado o processo de escolha do CH centralizado; neste sentido, é a BS quem deverá escolher quais serão os CHs na rodada atual por meio da informação da energia residual, informação esta recebida pela comunicação dos nós que responderam com ACK a etapa de *polling* da BS. Esta alteração possui as seguintes vantagens de dependabilidade:

- Não são inseridos novos pacotes na rede;
- A BS tem o controle sobre o percentual adequado de CH, evitando que vários nós se candidatem na rodada atual.
- A BS escolhe nós com melhores condições energéticas e qualidade de comunicação.
- Nós potencialmente distantes não podem ser candidatos a CH, mesmo com energia suficiente para isto evitando, falhas futuras. Esta restrição é regulada pela BS baseado no sinal *Received Signal Straith Indicator– (RSSI)* do ACK. A Figura 14 trata da escolha do CH no cenário de 5 nós.

Figura 14 – Eleição do CH

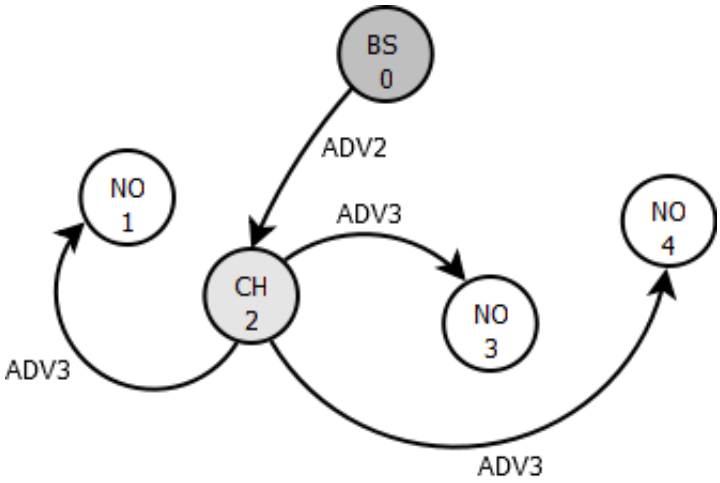


Neste momento já são repassada as informações dos canais que cada CH irá operar e o identificador TDMA para cada CH iniciar seu *polling* na rede.

Polling dos CHs

Quando os CHs são escolhidos inicia-se a etapa de *polling* dos CHs. Nesta etapa a BS aloca canais diferentes para comunicar-se com os CHs e para que eles utilizem na comunicação *intracluster* dentre os 16 canais fornecidos pelo padrão IEEE 802.15.4 (IEEE, 2006). Nesse processo quando o CH recebe um pacote de controle do tipo ADV2, enviado pela BS, automaticamente inicia um *polling* usando o *slot* TDMA para todos os nós na rede, menos a BS enviando um pacote do tipo ADV3. O percentual adequado de CHs usados no FTE-LEACH é de cinco por cento, de acordo com as especificações nos estudos de (OLIVEIRA, 2015). Portanto, como a BS conhece todos os nós que podem estar em funcionamento na rede, a duração do *polling* é controlada pela BS e avisa o momento em que um CH deve anunciar-se na rede Figura 15. Essas garantias foram criadas como forma de prevenção de falhas para reduzir as colisões de pacotes e perdas que existiriam se por acaso dois ou mais CHs realizassem o *polling* ao mesmo tempo.

Figura 15 – Polling do CH

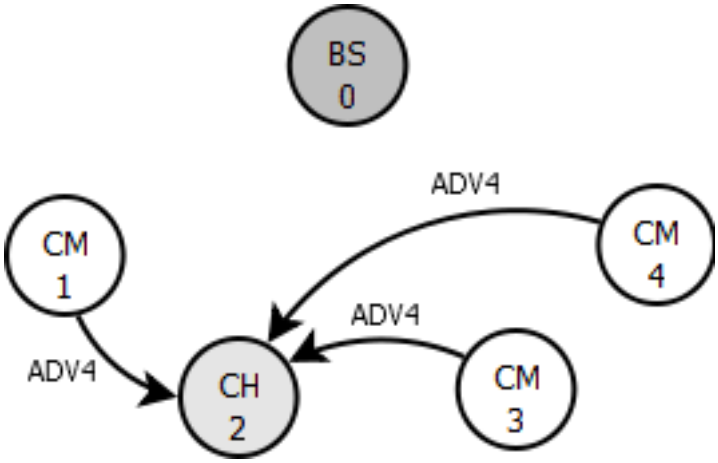


Associação dos CM

A etapa de associação também em TDMA leva o tempo necessário para que todos os nós possam associar-se a um CH. Cada nó que não foi eleito como CH, ao receber o anúncio dos CHs, salva o identificador (ID) do nó e o indicador da intensidade do sinal recebido *Received Signal Straith Indicator (RSSI)*. Essa etapa segue os mesmos procedimentos da proposta FTE-LEACH em (OLIVEIRA, 2015), com o CM associando-se ao CH, que oferece melhores condições para

comunicação, ou seja, aquele que exija o menor esforço (consumo) para enviar seus pacotes de dados. A Figura 16 demonstra a associação dos nós ao CH2 enviando a mensagem do tipo ADV4 definida como mensagem de associação nas definições do protocolo.

Figura 16 – Associação dos CM



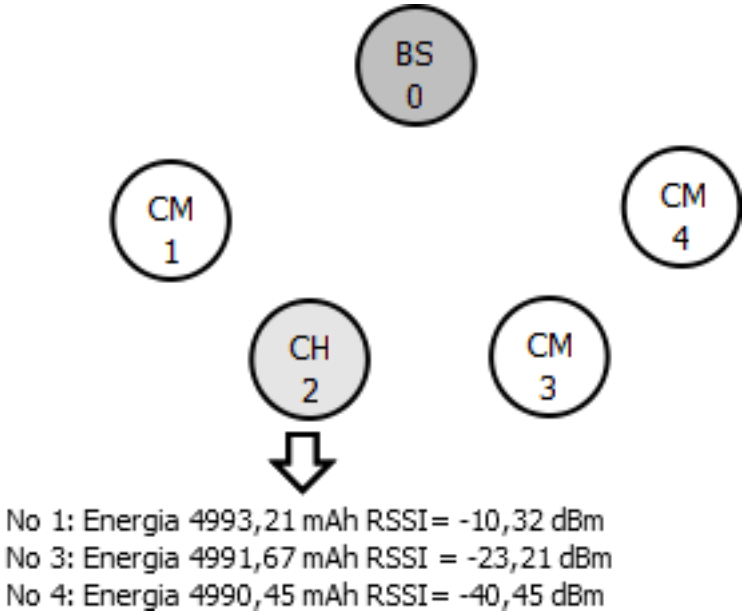
Escolha do VCH

Similar à escolha dos CHs por parte dos CMs é feita a escolha do VCH no FTE-LEACH por parte dos CH. Cada CH ao receber a informação de associação dos CM no pacote ADV4, armazena duas informações necessárias para esta tomada de decisão: o RSSI da comunicação com o nó e a informação do nó através do pacote ADV4. Com isto, pode-se estar habilitado ou não para submeter-se como VCH. Estar habilitado significa possuir energia suficiente para assumir a posição de CH caso este venha a falhar. Quando nenhum nó dentre os CMs associados não oferecer condições, então a escolha é totalmente baseada naquele de melhor RSSI, possivelmente mais próximo ao CH, e na sua capacidade de obter uma boa comunicação com a BS e os demais nós do *cluster*. A Figura 17 ilustra o processo de escolha do VCH ocorrendo no CH 2

Alocação de Time-Slot

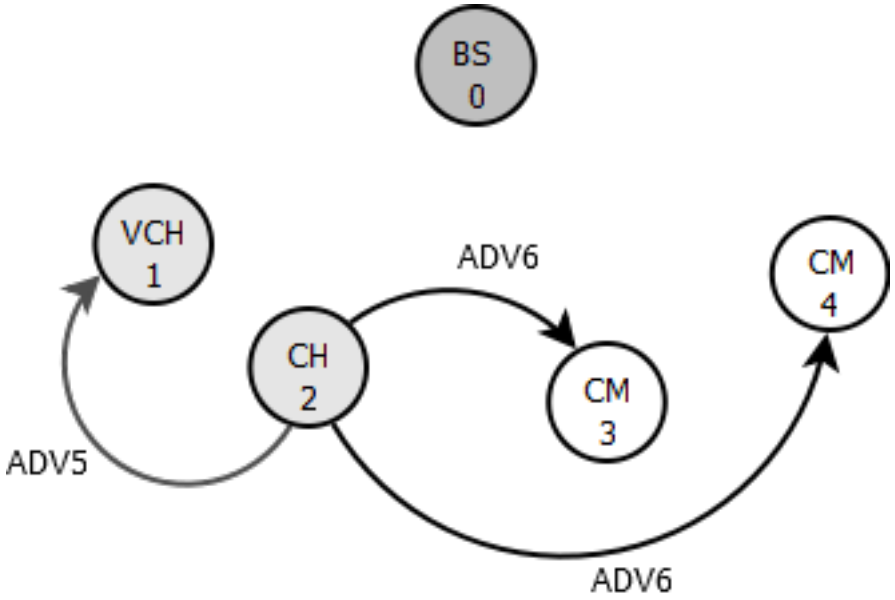
O processo de alocação de *slots* é um tratamento com características de dependabilidade onde se evita a concorrência de canais e se estipula o *time-slot* TDMA de cada nó na rede. Este procedimento previne falhas e garante o fluxo da comunicação. Para garantir que os nós não enviem em paralelo seus pacotes de dados ao mesmo destino e ao mesmo tempo, evitando colisões e perdas de pacotes. Essa etapa leva o tempo necessário para que todos os dispositivos recebam estas informações. Nesta situação é assumido o tempo do pior caso, onde todos os nós escolhem associar-se ao mesmo CH e, desta forma, o tempo para que o CH aloque *slot* para

Figura 17 – Escolha do VCH



que cada CM consiga enviar seus pacotes de dados usando o TDMA é o tempo para enviar o maior *superframe* da rede. A figura 18 ilustra o processo de alocação de *slots* do protocolo FTE-LEACH.

Figura 18 – Alocação Slot



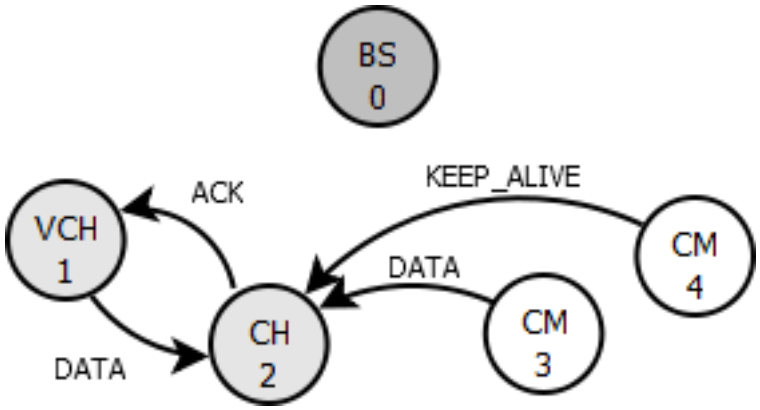
Etapa de Comunicação

Com todos os *clusters* formados e todos os nós associados a um *time-slot* TDMA, inicia-se a fase de comunicação e sensoriamento.

Coleta e Envio de dados

Nesta etapa assim como no FTE-LEACH visto em (OLIVEIRA, 2015) foi adotado o envio de dados dirigido a eventos (event-driven) (AQUINO et al., 2008). Neste sentido, o envio de pacotes DATA varia dentro de um percentual de acordo com a aplicação. Quando não existem dados significativos a serem enviados, um nó envia um pacote do tipo *KEEP-ALIVE* informando estar ativo para a rede que assume que seus dados são os mesmos ou irrelevantes. Em particular, o processo do VCH é diferenciado, no caso dos CMs, eles apenas enviam um pacote do tipo DATA ou *KEEP ALIVE*, já o VCH envia também os mesmos tipos de pacotes, mas espera um ACK de confirmação por parte do CH. Esse pacote é que informa o VCH se ele deve assumir ou não a posição do CH e ocorre no *slot* de tempo designado para sua comunicação. Este mecanismo implementado é definido como uma técnica de dependabilidade de prevenção e recuperação de falha, evitando assim que todos os pacotes de uma rodada sejam perdidos. A figura 19 demonstra o comportamento da rede nesta etapa.

Figura 19 – Coleta e Envio de dados



Vale salientar que a etapa de envio de dados permite o paralelismo de mensagens quando em *clusters* e canais diferentes. Nesse caso, uma rodada de coleta de dados do CM e envio ao CH tem a duração do maior *cluster* na rede, embora para manter o sincronismo desta rede deva-se esperar pelo tamanho do *superframe* com os nós em estado *sleep*.

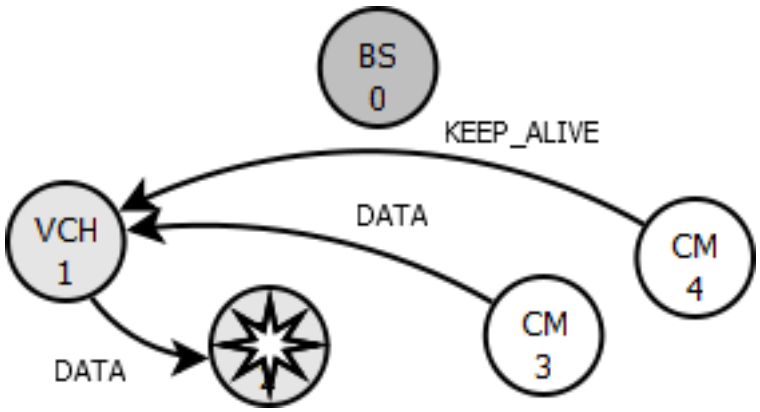
Agregação de dados no CH

Esta fase depende muito da aplicação e dos operadores da rede, pois a agregação ou não dos dados cabe ao usuário da rede decidir. Para tanto, FTE-LEACH permite a agregação de pacotes até o tamanho máximo especificado pelo padrão IEEE 802.15.4 que é de 127 *bytes* (IEEE, 2006). A agregação dos dados no protocolo ocorre no CH e na BS de acordo com as especificações já citadas. Para isso é necessário adotar um bom algoritmo e técnicas de agregação de dados, por exemplo, o algoritmo definido por (KALPAKIS; DASGUPTA; NAMJOSHI, 2003).

Tolerância a falhas do CH

FTE-LEACH adota o VCH não por acaso, como foi provado em (OLIVEIRA, 2015),com reduções consideráveis de perda de pacotes. O papel do VCH é assumir a posição do CH caso ele venha a falhar, personificando-se como CH e assumindo as suas funcionalidades até o final da rodada. Trata-se da técnica de dependabilidade de redundância de hardware descrita em (WEBER, 2003). A Figura 20 demonstra a falha do CH e os nós que ainda não enviaram seus dados (CM 3 e CM 4) transmitindo-os a partir de agora para o VCH.

Figura 20 – Tolerância a falhas do CH

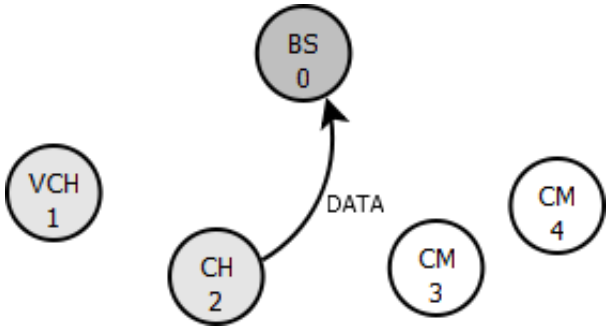


Envio de dados à BS

A última etapa antes de iniciar uma nova rodada para o FTE-LEACH é enviar os dados a BS. Neste momento os CH já possuem os dados dos CMs e precisam repassá-los a BS. Como tratam-se de *cluster* consideravelmente grandes é normal que cada CH envie mais de um pacote, mesmo fazendo agregação de dados. A etapa também utiliza TDMA e aloca *slots* para os CH poderem enviar dados no canal reservado pela BS anteriormente. Visando evitar que a BS receba pacotes simultâneos de CHs distintos, causando assim perdas de pacotes, foi definido que cada CH deve esperar o término de outro para poder começar a enviar seus pacotes. A solução é construída na fase de eleição dos CHs; nesse momento a BS informa ao nó quantos *superframes* ele deverá esperar para iniciar sua transmissão após a etapa de coleta de dados.

Dentre os CHs, aquele com piores condições energéticas na fase de escolha é considerado "privilegiado" da BS, porque ao receber todos os pacotes dos seus CM pode iniciar imediatamente o envio de dados para BS em um *superframe* do tamanho do maior *cluster* formado, É normatizado este superframe para todos os demais *clusters* e cada um possui o seu canal diferente podendo então acontecer comunicações simultâneas. Com isso, é possível eliminar a disputa pelo nó de

Figura 21 – Agregação de dados no CHs



destino e permitir que ocorra ou não a agregação de dados. A figura 21 demonstra um caso simples apenas de ilustração do CH enviando pacote do tipo DATA à BS, encerrando assim uma rodada do FTE-LEACH. Durante todo o *round* foram vistas várias implementações de técnicas para se atingir a dependabilidade na rede. Quando termina-se um round todo o processo se reinicia e pode-se perceber que o CH e o VCH irá mudar de acordo com seu gasto energético. Com isto a rotatividade dos CH na rede previne falhas de perda de nós por desigual consumo energético. Esta característica mantém a rede com níveis energéticos relativamente equivalentes, o que aumenta a vida útil da rede como um todo e mantém uma prevenção de falhas em relação à perda de nós prematura por desgaste energético.

Há a possibilidade de sobrepor camadas de monitoramento visando uma diminuição no tempo de atualização da rede e aumentando a tolerância a falhas por redundância de hardware. Esta possibilidade é possivel pelo fato de haver canais livres para ao menos 10 a 12 **clusters** dependendo da aplicação. No próximo capitulo são exibidos os resultados obtidos nesta pesquisa.

5 RESULTADOS

O cenário de simulação consiste de ambientes de monitoramento sísmico, com informações críticas, cujos dados de monitoramento e metodologia são descritos a seguir:

Metodologia

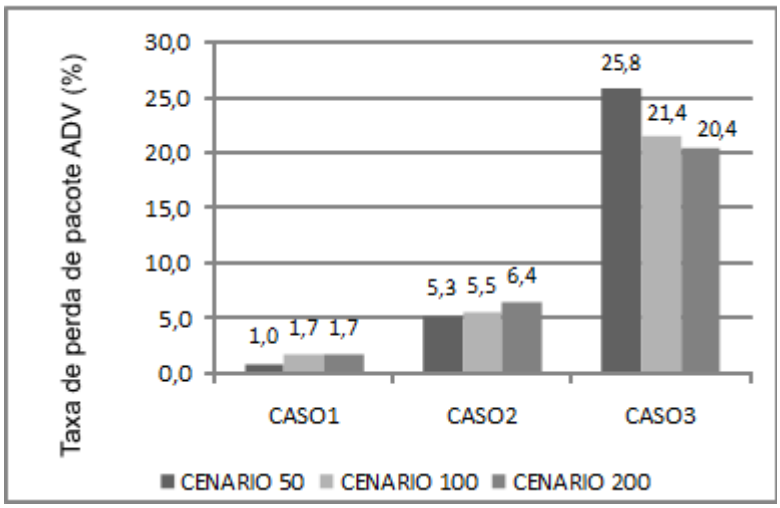
Para fins de análise do protocolo FTE-LEACH desenvolvido em NS-3, são utilizados os seguintes dados de simulação:

- Quanto à densidade de nós, foram testados cenários contendo 50, 100 e 200 nós; cenário com dimensões de 100 metros quadrados;
- Quanto ao posicionamento da BS, foi estabelecido o pior caso para testes, quando a BS está localizada em um dos vértices do plano cartesiano proporcionando distâncias máximas entre a BS e os nós mais distante no cenário;
- Quanto à distância minima dos nós para a BS, foi atribuída uma distância mínima de 10m da BS para o nó mais próximo desta;
- Quanto ao tamanho dos pacotes, foram utilizados tamanhos variados entre 21 e 127 bytes (quando da necessidade de agregação de dados);
- Quanto à bateria; foi escolhida a bateria com 5000 mAh em cada nó, ela possui uma voltagem de 3,3V de operação.
- Quanto ao Consumo da CPU, foi adotado de 3.3 mAh; Consumo referente transceptor (INSTRUMENTS, 2011)
- Quanto à quantidade de amostragens, foram realizados uma centena de simulações para cada caso de erro em cada cenário proposto. Totalizando novecentas simulações para a produção de resultados.
- Quanto ao modo de propagação escolhido foi adotado o *friis* (FRIIS, 1946), modelo mais adequado para cenários abertos.
- Quanto ao modelo de erro, foram utilizados os três casos do modelo Gilbert/Elliot já descritos no módulo de simulação e na tabela 1 seção 4.1.

Simulação para etapa de configuração.

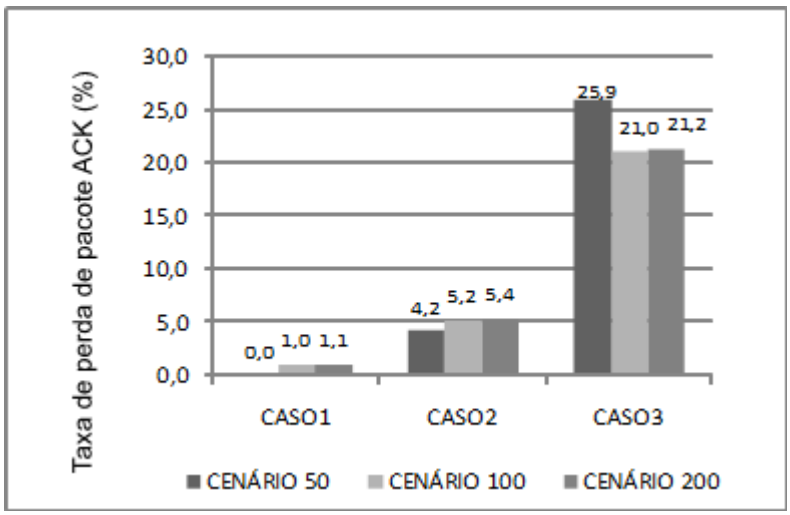
A Simulação para pacotes ADV e sua respectiva taxa de erro para os três casos e para os três cenários de densidade de nós é visualizada na Figura 22.

Figura 22 – Taxa de perda de pacotes ADV



Simulação para pacotes ACK e sua respectiva taxa de erro para os três casos e para os três cenários de densidade de nós é percebido na Figura 23.

Figura 23 – Taxa de perda de pacotes ACK

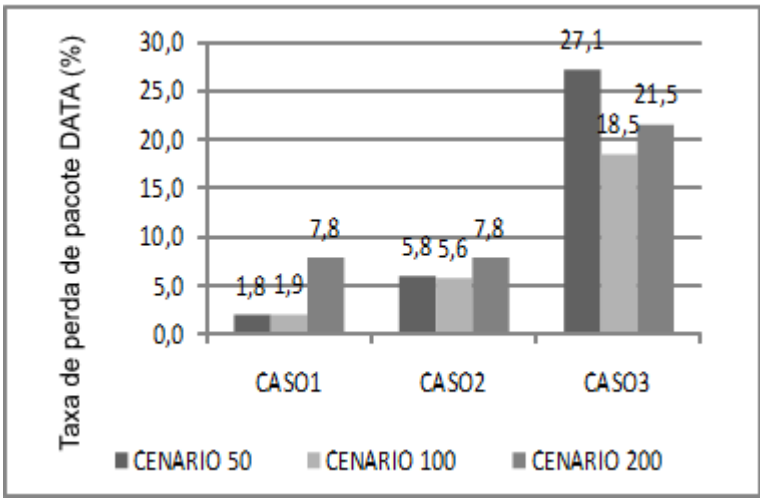


Observa-se no gráfico da Figura 22 que a taxa de perda em todos os cenários para o caso 1 é inferior a 2 por cento. Já para o caso 2 todos os cenários apresentam taxas próximas dos 5 por cento e uma piora significativa para o cenário de 200 nós, o mesmo ocorre em mesmas proporções para o pacote ADV exibido na Figura 23. Cabe salientar que estas taxas podem ainda ser melhoradas caso a eficiência energética não seja levada em consideração por exemplo em um cenário de alimentação energética continua.

Simulação para etapa de Comunicação.

O tratamento do TDMA e das separações de canais na fase de comunicação também geram resultados expressivos para o envio dos pacotes de dados na rede. A Figura 24 demonstra a taxa de perda dos pacotes de dados.

Figura 24 – Taxa de perda de pacotes de dados



Verifica-se a similaridade nas taxas de perda para todas as etapas de operação do FTE-LEACH visto que a 24 do pacote DATA só opera na camada de comunicação e possui pacotes maiores ou iguais a 23 bytes .

O intuito de toda comunicação é a transmissão de dados úteis para tomadas de decisões. O teste abaixo está relacionado as taxas de vazão de dados que chegam a BS. A figura 25 representa a taxa de transferência efetiva de um pacote enviado desde sua coleta no CM até o seu destino final à BS. Cabe salientar que o dado produzido por um CM apenas, como neste caso, sofre atraso de espera na agregação de dados realizado pelo CH. O pacote quando enviado a partir de um CH nos primeiros *time-slot* do *super-frame* sofre um atraso superior aos pacotes enviados em *time-slot* posteriores.

Analisando os dados produzidos na Figura 25, verifica-se a necessidade de calcular a taxa de transferência efetiva para o conjunto de dados coletados pelos CM e agregados pelo CH, desde sua coleta até o seu destino final quando o dado chaga a BS. O resultado é exibido em todos os cenários e casos de erro na Figura 26.

Figura 25 – Throughput

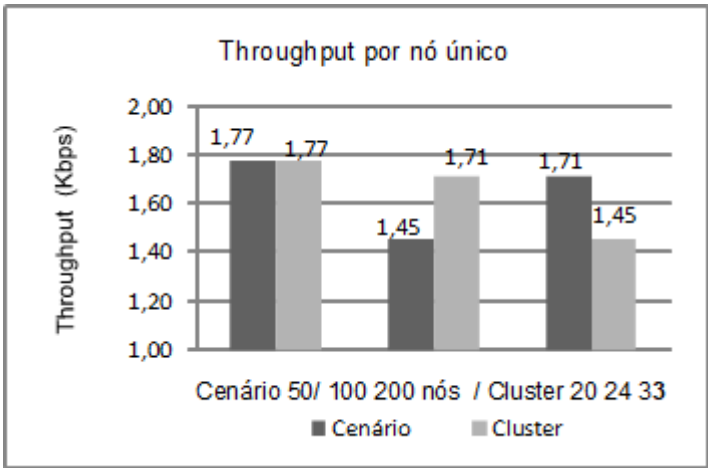
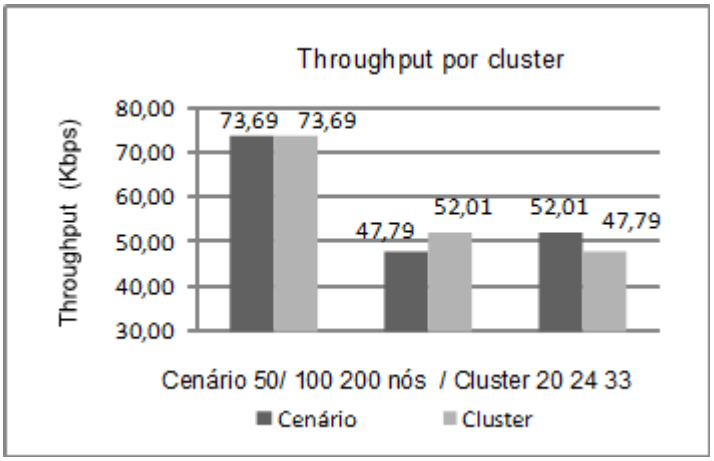


Figura 26 – Throughput



Plot de Cenários utilizados com formação dos respectivos clusters

Para efeito de visualização na Figura 27, foi plotado o cenário de 100 nós para a etapa de configuração incluído-se a BS. Cenário de 100 nós com a BS em destaque na coordenada (0,0) no canto inferior esquerdo, o cenário destaca uma área de 100m por 100m de abrangência com o intuito de simular uma área de mineração propicia a abalos sísmicos. Os nós são distribuídos aleatoriamente no cenário onde a distância da BS ao nó mais próximo é de 20 metros. Para inicio da etapa de configuração todos os nós são previamente cadastrados na aplicação de monitoramento e estes são replicados a todos os nós, estando então todos os nós conhecedores dos endereços físicos MAC de cada um dos demais nós na rede.

Na Figura 28 os *Chuster Heads* foram eleitos e neste momento os CM tendem a escolher o CH mais próximo, mas por algum fator de erro quando o CM não recebe o *polling* do CH a escolha visa o 2º CH mais próximo. O nó CM mais próximo ao CH, tende a ser escolhido nesta primeira rodada como VCH.

Figura 27 – Cenário inicial de 100 nós

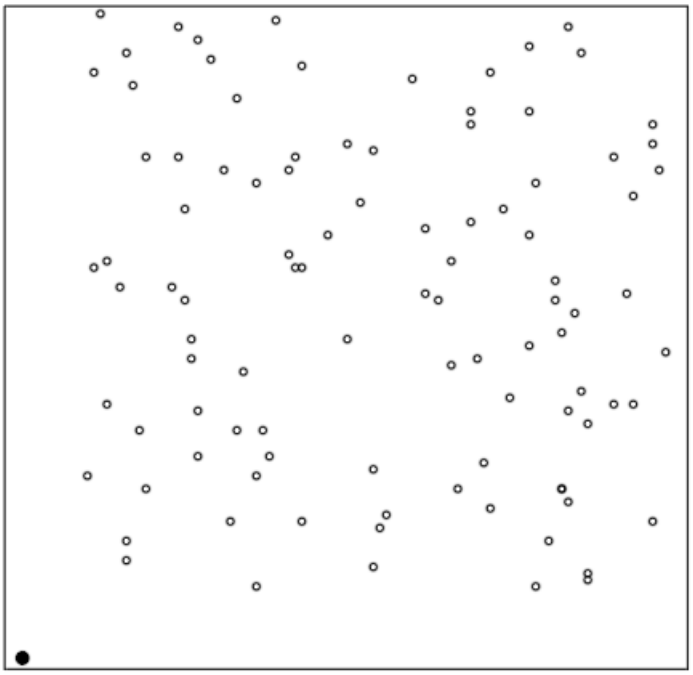
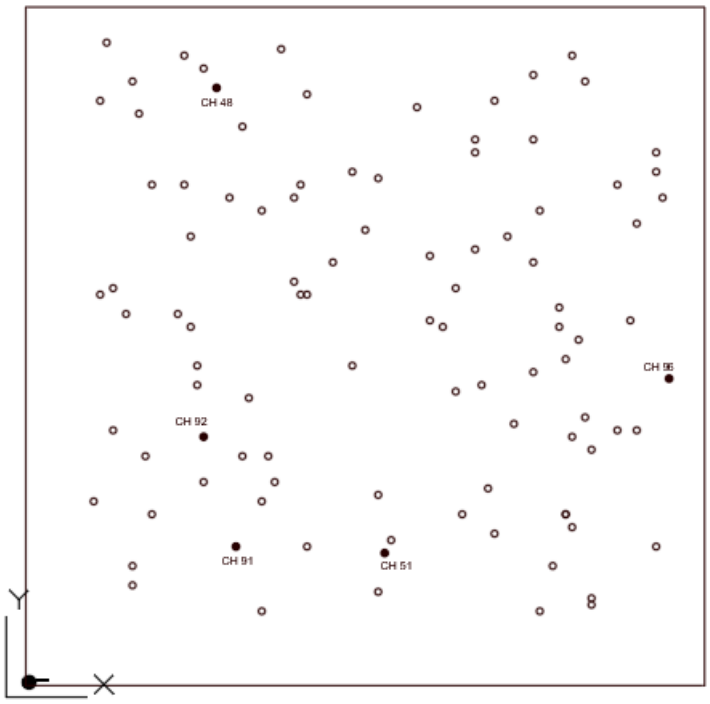
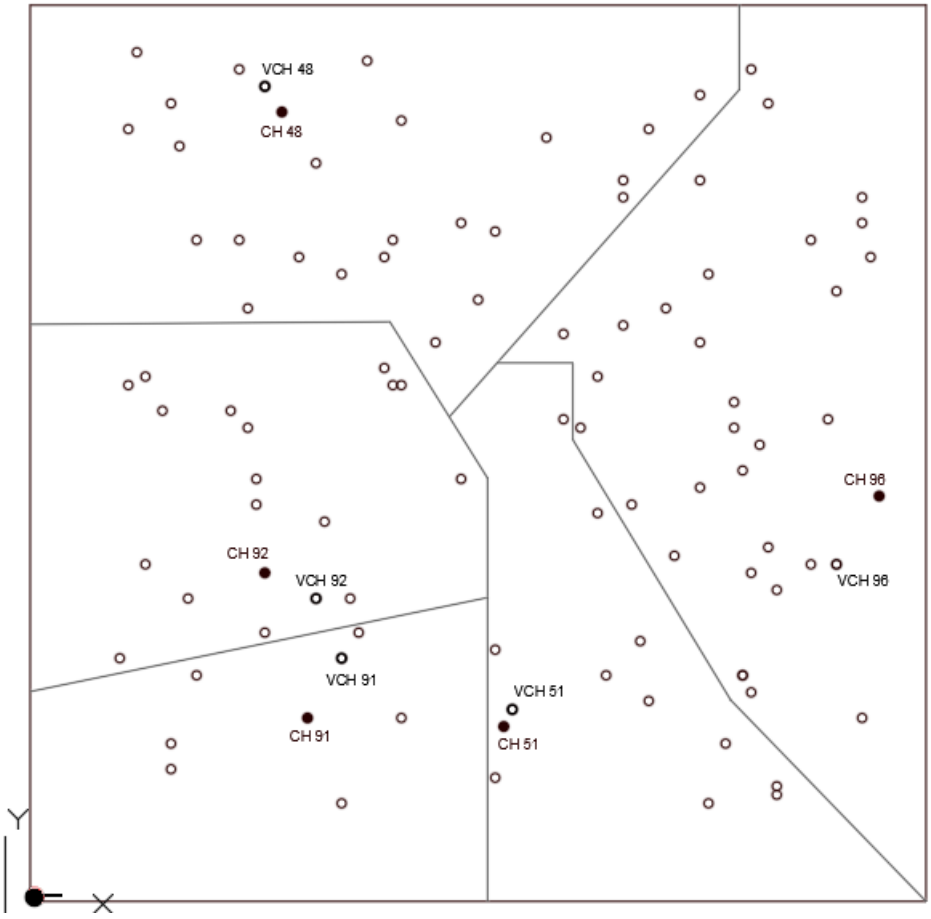


Figura 28 – Cenário de 100 nós com eleição dos CH e VCH



Na Figura 29, observa-se o resultado das associações de cada CM para seus respectivos *cluster*, e completando a fase de configuração, o posicionamento do VCH de cada *cluster* alocado pelo CH. No algoritmo FTE-LEACH após o último CH transmitir seus dados à BS, a etapa de comunicação é finalizada para esta rodada, dando-se início a próxima etapa de configuração.

Figura 29 – Cenário de 100 nós após etapa de configuração



A repetição do ciclo de rodadas de configuração e transmissão de dados, garante que a rede tenha sua vida útil equilibrada em sus nós, onde seus recursos energéticos esvanecem de maneira equilibrada.

O gráfico da Figura 30 mostra a comparação da média dos pacotes de dados recebidos pela BS na proposta FTE-LEACH em NS-3, comparando com o FTE-LEACH da versão (OLIVEIRA, 2015). No gráfico pode-se analisar que há um ganho mínimo na etapa de comunicação para o modelo proposto, verificando-se uma certa similaridade do modelo matemático com o modelo real. Com o ganho discreto, pode-se concluir que o modelo empregado para as demais variáveis onde há maiores mudanças no algoritmo resultará em ganhos expressivos.

Para esta etapa de resultados também foi realizado simulações como já mencionado, com cenário de 50 nós e 200 nós, quanto ao cenários de 50 nós Figura 31 é utilizando a mesma área de abrangência, onde apesar de obter-se uma menor densidade de nós há ganhos quanto ao *throughput* na ordem de 35 por cento, se comparado com o cenário de 100 nós 27.

Figura 30 – Comparativo do FTE-LEACH proposto e simulado em NS-3 com o FTE-LEACH versão anterior modelado em MARLAB

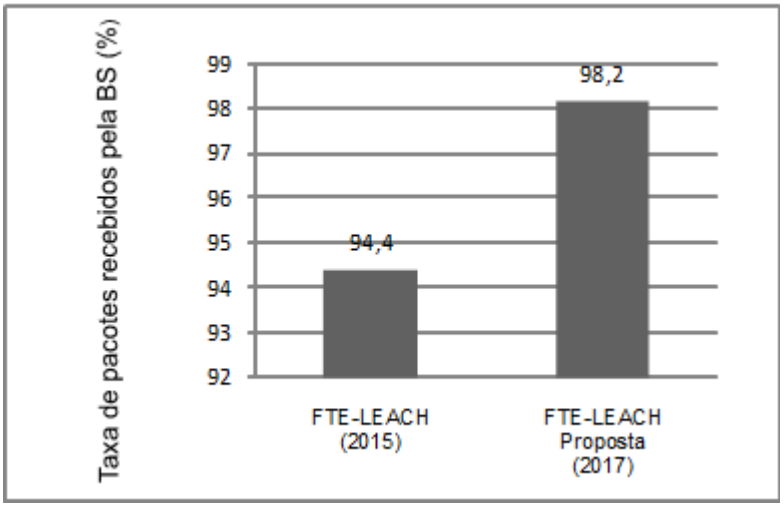
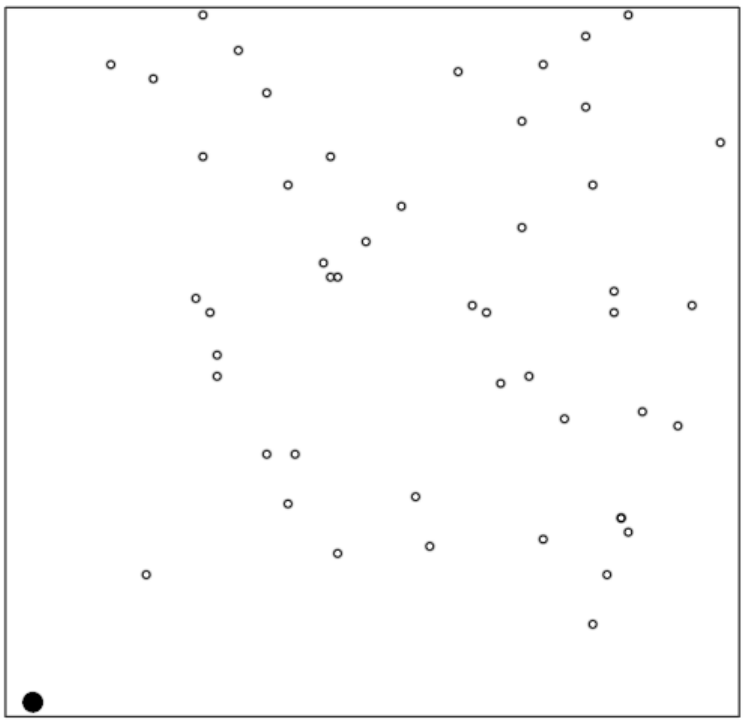


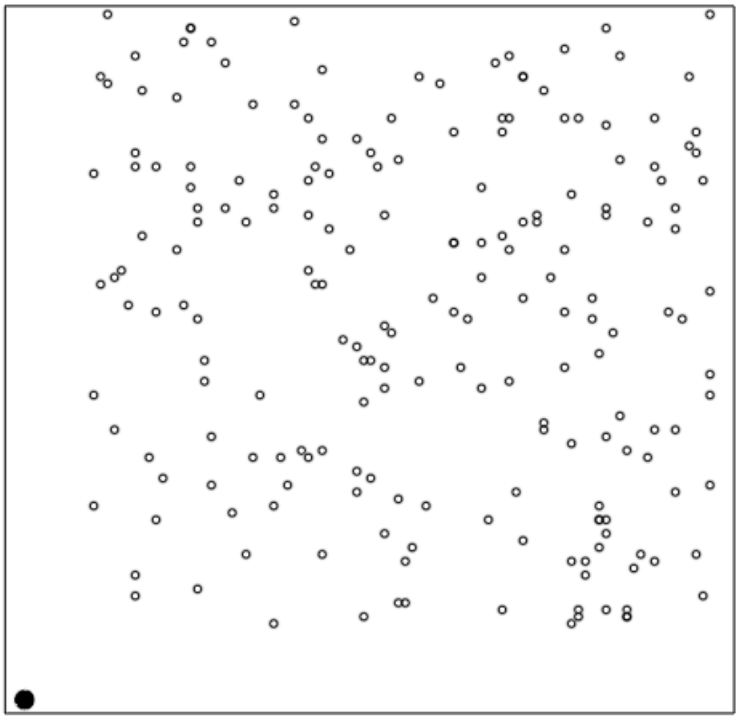
Figura 31 – Cenário de 50



Ainda em efeito comparativo o cenário com 200 nós possui uma densidade de nós 4 vezes maior gerando uma redundância de hardware significativa, quanto a perda de vasão em relação aos cenário de 50 nós é de 28 por cento. O cenário de 200 nós é ilustrado na Figura 32.

Esta pesquisa pôs à prova vários conhecimentos adquiridos no ambiente acadêmico,

Figura 32 – Cenário de 200



onde a teoria pesquisada nas disciplinas deste programa de pós graduação, culminou com uma contribuição para as pesquisas atuais bem como às próximas pesquisas na areá das Redes de Sensores sem Fio Industriais, por meio do módulo do FTE-LEACH no NS-3 e dos resultados gerados a partir deste.

6 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

A partir dos resultados obtidos, pode-se destacar que o módulo FTE-LEACH proposto nesta pesquisa teve seu objetivo alcançado. A proposta de criar um módulo de simulação para o NS-3 e adicionar técnicas de dependabilidade a um protocolo derivado do LEACH foi alcançada e validada quando comparada com o FTE-LEACH modelado matematicamente em MATLAB por (OLIVEIRA, 2015). Verificou-se, por exemplo, uma similaridade entre comportamentos do modelo MATLAB com a simulação de NS-3 para testes realizados em condições semelhantes e onde não há alteração no algoritmo como no caso da Figura 30, mostrando que para as alterações realizadas no algoritmo visando menores perdas o comparativo seque a mesma tendência.

A metodologia empregada para obtenção dos resultados foi validada quando o comparativo entre as taxas de pacotes recebidos pelo modelo proposto foram similares ao modelado matematicamente em (OLIVEIRA, 2015). Todos os resultados foram obtidos por média de valores em 100 simulações para cada modelo de erro e mais 100 simulações para cada cenário proposto totalizando 900 simulações válidas para estes resultados.

Esta pesquisa aplicou técnicas de dependabilidade no uso do protocolo FTE-LEACH que não são nativos de seu funcionamento (OLIVEIRA, 2015), além da criação de um módulo de simulação para o FTE-LEACH no NS-3. Isso dá a oportunidade de comparar o funcionamento deste protocolo de roteamento com os demais protocolos emergentes, visando preencher as lacunas de dependabilidade e de testabilidade das RSSFI.

Foram realizadas simulações com o módulo de roteamento criado para o FTE-LEACH implementando a auto-configuração e auto-organização de sua topologia. Foi utilizado o modelo de propagação *friis* (FRIIS, 1946) nativo do NS-3 (NSNAM, 2015) e aplicado o modelo de erro de *Gilbert/Elliot* (EBERT; WILLIG et al., 1999) já implementado no módulo *WirelessHART* por (NOBRE, 2015) com características condizentes ao cenário proposto.

A prevenção de falhas foi implementada ao eliminar o uso da tecnologia CSMA-CA (TANENBAUM, 2003) nas etapas de configuração do FTE-LEACH (OLIVEIRA, 2015), substituindo pela tecnologia TDMA em todas as suas etapas, com isso eliminando a probabilidade de colisões de pacotes e diminuindo a possibilidade de falsos positivos quanto a atividade de um nó sensor na rede.

A previsão e prevenção de falha foi adicionada pelo o uso dos canais de comunicação do FTE-LEACH, que foi alterado, não mais sendo necessário o uso da teoria de grafos de 4 cores da versão (OLIVEIRA, 2015), proporcionando um aumento na quantidade de canais utilizáveis de acordo com o nível de interferência.

A recuperação de falha foi implementada com o mecanismo de *polling* dos *Cluster Head*, onde foi adicionada a possibilidade de inclusão de nós que por algum motivo não tenham sido

reconhecidos pela BS no *polling* inicial, aumentando a disponibilidade de sensoriamento da rede por redundância de hardware de reconhecimento. Esta possibilidade ainda não existia no FTE-LEACH.

A previsão e prevenção a falhas também é empregada em todos os momentos que se evita o aumento de tráfego na rede; isto ocorre no monitoramento dirigido a eventos, e na fase da escolha do CH quando se evita o envio de mensagens ACK para a BS além da fase de agregação de dados; estas contribuições estão a nível de simulação, pois o modelo original FTE-LEACH (OLIVEIRA, 2015) já previa estas funcionalidades, porém ainda não simulada.

O protocolo FTE-LEACH ainda possui margem para melhoramentos e a dependabilidade ainda pode ser envolvida nesta evolução, visando os ambientes cuja perda da informação seja um fator de extrema relevância como o caso da sismologia em ambientes de mineração ao céu aberto.

Trabalhos futuros

A dependabilidade em redes de sensores sem fio pode ainda obter grandes ganhos para o protocolo FTE-LEACH e suas variações. A comparação das técnicas utilizadas nesta pesquisa ainda devem ser realizadas com outros protocolos de roteamento em redes RSSFI. Para tanto é de fundamental importância que os módulos NS-3 de simulação para estes padrões sejam criados. Além das comparações com outros padrões algumas implementações no FTE-LEACH serão alvo de pesquisas futuras.

- É proposta do grupo de pesquisa GSET implementar escalonamento de canais, mensagens e *time-slot* de maneira autômata para diferentes tamanhos de *superframe*.
- Aplicação de mais de um VCH por *cluster*, seguindo a filosofia implementada por (SALIM; OSAMY; KHEDR, 2014) aplicando redundância ao mesmo.(DAVEY et al., 2006).
- Realizar simulações de comparação em outros modelos de propagação além do *friis*.
- Documentar o módulo criado, e iniciar o processo de introdução no sistema nativo do NS-3 visando sua distribuição para a comunidade de pesquisa.
- A diminuição do tempo de cada rodada pode ser obtida com o uso de difusão de pacotes substituindo o polling de envio.

No Apêndice A, é visto o trecho de código que é responsável pela eleição do VCH de cada cluster, e este só assume sua função quando há inoperância por parte do CH.

No Apêndice B, exibe-se o trecho de código responsável pela aplicação dos três casos de níveis de erro aplicado na simulação.

REFERÊNCIAS BIBLIOGRÁFICAS

2015, I. . Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low rate wireless personal area networks (wpans). *IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003)*, p. 1–320, Sept 2006. Citado na página 22.

AL-FUQAHA, A. et al. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys Tutorials*, v. 17, n. 4, p. 2347–2376, Fourthquarter 2015. ISSN 1553-877X. Citado na página 16.

AQUINO, A. L. et al. An in-network reduction algorithm for real-time wireless sensor network applications. In: *Proceedings of the 4th ACM Workshop on Wireless Multimedia Networking and Performance Modeling*. New York, NY, USA: ACM, 2008. (WMuNeP '08), p. 18–25. ISBN 978-1-60558-238-2. Disponível em: <http://doi.acm.org/10.1145/1454573.1454578>. Citado na página 48.

AVIZIENIS, A. et al. Basic concepts and taxonomy of dependable and secure computing. *IEEE transactions on dependable and secure computing*, IEEE, v. 1, n. 1, p. 11–33, 2004. Citado 4 vezes nas páginas 16, 32, 33 e 34.

BHAGWAT, P. et al. Using channel state dependent packet scheduling to improve tcp throughput over wireless lans. *Wireless Networks*, Springer-Verlag New York, Inc., v. 3, n. 1, p. 91–102, 1997. Citado na página 38.

CARLSON, D. et al. Iec 62591 wirelesshart® system engineering guide. *Revision 3.0 ed.: Emerson Process Management*, 2012. Citado 2 vezes nas páginas 17 e 29.

DAVEY, R. P. et al. Dwdm reach extension of a gpon to 135 km. *Journal of lightwave technology*, IEEE, v. 24, n. 1, p. 29–31, 2006. Citado na página 60.

DEERING STEVE E HINDEN, R. *RFC 2460: Protocolo de Internet*. Citado na página 22.

EBERT, J.-P.; WILLIG, A. et al. A gilbert-elliot bit error model and the efficient use in packet level simulation. Citeseer, 1999. Citado na página 59.

FALUDI, R. *Building wireless sensor networks: with ZigBee, XBee, arduino, and processing*. [S.l.]: "O'Reilly Media, Inc.", 2010. Citado 3 vezes nas páginas 20, 21 e 22.

FRIIS, H. T. A note on a simple transmission formula. *Proceedings of the IRE*, IEEE, v. 34, n. 5, p. 254–256, 1946. Citado 2 vezes nas páginas 51 e 59.

GRIGOLETTI, P. S. Cadeias de markov. *Recuperado em*, v. 19, n. 10, p. 2014, 2011. Citado na página 37.

HEINZELMAN, W. B.; CHANDRAKASAN, A. P.; BALAKRISHNAN, H. An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, v. 1, n. 4, p. 660–670, Oct 2002. ISSN 1536-1276. Citado 2 vezes nas páginas 16 e 42.

HEINZELMAN, W. R.; CHANDRAKASAN, A.; BALAKRISHNAN, H. Energy-efficient communication protocol for wireless microsensor networks. In: *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*. [S.l.: s.n.], 2000. p. 10 pp. vol.2–. Citado 5 vezes nas páginas 19, 23, 25, 27 e 28.

IEEE. Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low rate wireless personal area networks (wpans). *IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003)*, p. 1–320, Sept 2006. Citado 4 vezes nas páginas 23, 24, 45 e 48.

INSTRUMENTS, T. Cc2500 low-cost low-power 2.4 ghz rf transceiver. *Data Sheet*, 2011. Citado na página 51.

KALPAKIS, K.; DASGUPTA, K.; NAMJOSHI, P. Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks. *Computer Networks*, Elsevier, v. 42, n. 6, p. 697–716, 2003. Citado na página 48.

KUMAR, S.; CHAUHAN, S. A survey on scheduling algorithms for wireless sensor networks. *International Journal of Computer Applications*, International Journal of Computer Applications, 244 5 th Avenue,# 1526, New York, NY 10001, USA India, v. 20, n. 5, 2011. Citado na página 28.

KUROSE, J.; ROSS, K. *Computer networking: a top-down approach. Always Learning*. [S.l.]: Pearson, London, 2013. Citado na página 16.

MACEDO, D.; SILVA, I.; GUEDES, A. Uma ferramenta para análise de dependabilidade de processos industriais. In: *Simpósio Brasileiro de Automação Inteligente*. [S.l.: s.n.], 2013. Citado 4 vezes nas páginas 16, 19, 20 e 37.

MATHWORKS, I. *MATLAB: the language of technical computing. Desktop tools and development environment, version 7*. [S.l.]: MathWorks, 2005. v. 9. Citado na página 36.

NAKAMURA, E. F.; LOUREIRO, A. A.; FRERY, A. C. Information fusion for wireless sensor networks: Methods, models, and classifications. *ACM Computing Surveys (CSUR)*, ACM, v. 39, n. 3, p. 9, 2007. Citado na página 16.

NOBRE, M.; SILVA, I.; GUEDES, L. A. Performance evaluation of wirelesshart networks using a new network simulator 3 module. *Computers & Electrical Engineering*, Elsevier, v. 41, p. 325–341, 2015. Citado 6 vezes nas páginas 29, 30, 36, 37, 38 e 43.

NOBRE, M. et al. Towards a wirelesshart module for the ns-3 simulator. In: IEEE. *Emerging technologies and factory automation (ETFA), 2010 IEEE conference on*. [S.l.], 2010. p. 1–4. Citado 2 vezes nas páginas 17 e 37.

NOBRE, M. H. R. *Contribuições em Escalonamento e Analise de Desempenho de Redes WirelessHART*. Tese de Doutorado — Universidade Federal do Rio Grande do Norte, Novembro 2015. Citado 6 vezes nas páginas 17, 23, 30, 36, 38 e 59.

NSNAM. *ns-3 Consortium*. 2015. Disponível em: <https://www.nsnam.org/>. Citado 3 vezes nas páginas 17, 36 e 59.

OLIVEIRA, F. D. M. *FTE-LEACH: Um Protocolo Energeticamente Eficiente e Tolerante a Falhas Aplicado às Redes Industriais de Sensores sem Fio*. Tese de Doutorado — Universidade Federal do Rio Grande do Norte - UFRN, Agosto 2015. Citado 14 vezes nas páginas 16, 19, 23, 27, 28, 35, 36, 42, 45, 48, 49, 56, 59 e 60.

PRADHAN, D. K. *projeto de sistema de computador tolerante a falhas*. [S.l.]: Prentice-Hall, Inc., 1996. Citado 2 vezes nas páginas 31 e 32.

SALIM, A.; OSAMY, W.; KHEDR, A. M. Ibleach: intra-balanced leach protocol for wireless sensor networks. *Wireless networks*, Springer, v. 20, n. 6, p. 1515–1525, 2014. Citado 2 vezes nas páginas 27 e 60.

SEMENTE, R. S. Estudo e desenvolvimento de algoritmos criptográficos para redes de sensores sem fio, utilizando técnicas de programação genética. Universidade Federal do Rio Grande do Norte, 2015. Citado na página 34.

SHARMA, V.; ALAM, B. Unicaste routing protocols in mobile ad hoc networks: a survey. *International Journal of Computer Applications*, Foundation of Computer Science, v. 51, n. 14, 2012. Citado na página 24.

SHELBY ZACH; BORMANN, C. *6LoWPAN : the wireless embedded internet*. [S.l.: s.n.], 2009. 244 p. Citado na página 38.

SIENA, W. Modelo de perda de pacote para projeto e simulação de sistemas de controle em rede sem fio. 2016. Citado 2 vezes nas páginas 36 e 37.

SILVA, I. et al. Tecnologias emergentes para redes industriais sem fio: Wirelesshart vs isa100. 11a. *Rio Automação*, 2013. Citado na página 16.

SINGH, S. K.; KUMAR, P.; SINGH, J. P. A survey on successors of leach protocol. *IEEE Access*, IEEE, v. 5, p. 4298–4328, 2017. Citado 2 vezes nas páginas 26 e 27.

TANENBAUM, A. S. *Redes de computadoras*. [S.l.]: Pearson Educación, 2003. Citado na página 59.

WANG, H. S.; MOAYERI, N. Finite-state markov channel-a useful model for radio communication channels. *IEEE transactions on vehicular technology*, IEEE, v. 44, n. 1, p. 163–171, 1995. Citado na página 38.

WEBER, T. S. Um roteiro para exploração dos conceitos básicos de tolerância a falhas. *Relatório técnico, Instituto de Informática UFRGS*, 2002. Citado 2 vezes nas páginas 31 e 32.

WEBER, T. S. Tolerância a falhas: conceitos e exemplos. *Apostila do Programa de Pós-Graduação–Instituto de Informática-UFRGS. Porto Alegre*, 2003. Citado 4 vezes nas páginas 31, 32, 33 e 49.

WILLIG, A. et al. Measurements of a wireless link in an industrial environment using an ieee 802.11-compliant physical layer. *IEEE Transactions on Industrial Electronics*, IEEE, v. 49, n. 6, p. 1265–1282, 2002. Citado na página 38.

WINTER, T. Rpl: Ipv6 routing protocol for low-power and lossy networks. 2012. Citado 2 vezes nas páginas 17 e 24.

APÊNDICE A – ELEIÇÃO DO VCH

```
1  void
2  Step4 (NodeContainer nodes){
3  WHartNetDevice obj;
4  std::vector< uint32_t > ch_associado = obj.GetCHs();
5  std::vector< uint32_t > cm_associado = obj.GetCandidatosVCH();
6  std::vector< uint32_t > canalCH = obj.GetCanalCH();
7  std::vector< double > RSSI = obj.GetRSSI();
8  std::vector< uint32_t > VCHs, CHs;
9
10 uint32_t x = 1, cont = 0;
11 while (x <= (nodes.GetN() * 5)/ 100){
12 double rssi_aux[nodes.GetN()];
13 cont = 0;
14 for (uint32_t i = 1; i< ch_associado.size(); i++){
15 if (canalCH.at(i) == x){
16 rssi_aux[cont] = RSSI.at(i);
17 cont++;
18 }
19 }
20 double melhor_sinal = *std::max_element(rssi_aux,rssi_aux+cont);
21 for (uint32_t i = 1; i< ch_associado.size(); i++){
22 if ( (canalCH.at(i) == x) and (RSSI.at(i) == melhor_sinal) ) {
23 VCHs.push_back(cm_associado.at(i));
24 CHs.push_back(ch_associado.at(i));
25 }
26 }
27 x++;
28 }
29 }
```

APÊNDICE B – MODELO DE ERRO

```
1
2
3   void
4   InstallNetwork(NodeContainer nodes){
5
6   Ptr<WHartChannelList> channellist = CreateObject<WHartChannelList> ();
7   Ptr<WHartChannel> channel;
8   Ptr<FriisPropagationLossModel> log;
9   Ptr<GilbertElliotErrorModel> rem;
10  MobilityHelper mobility;
11  //Jie
12  //Caso 1
13  double Pg=0.9999918;
14  double Pb=0.999184;
15  /*
16
17  //Caso 2
18  double Pg=0.9999;
19  double Pb=0.998;
20  */
21
22  //Caso 3
23  //  double Pg=0.999;
24  //  double Pb=0.98;
25
26  for(uint32_t i = 1; i<=16;i++){
27  channel = CreateObject<WHartChannel> ();
28  log = CreateObject<FriisPropagationLossModel> ();
29  log->SetFrequency((2.400+(0.005*i))*1e9);
30  channel->SetPropagationLossModel (log);
31  rem = CreateObject<GilbertElliotErrorModel> ();
32  Ptr<UniformRandomVariable> URV = CreateObject<UniformRandomVariable> ();
33  rem->SetRandomVariable (URV);
34  rem->BuildDeviceMatrices(nodes.GetN());
35  rem->SetPg(Pg);
36  rem->SetPb(Pb);
37  channel->SetErrorModel(rem);
38  channellist->AddChannel(channel);
```

```
39    };
40    /* Depois da inicializacao dos CANAIS */
41    std::vector<Address> addressList;
42    for (uint32_t i = 0; i < nodes.GetN (); ++i){
43    CreateSimpleDevice(nodes.Get (i),i,channellist, nodes.GetN());
44    addressList.push_back(nodes.Get (i)->GetDevice(0)->GetAddress());
45    }
46    /* Depois da inicializacaoo dos DISPOSITIVOS */
47    for(uint32_t i = 0; i<16;i++){
48    channellist->GetChannel(i)->GetErrorModel()->GetObject<GilbertElliotErrorModel>
          ()->SetAddressList(addressList);
49    }
50    for(uint32_t i = 0; i < nodes.GetN (); i++){
51    for(uint32_t j = 0; j < nodes.GetN (); j++){
52    for(uint32_t z = 0; z < 16; z++){
53    channellist->GetChannel(z)->GetErrorModel()->GetObject<GilbertElliotErrorModel>
          ()->SetErrorMatrixValues(i,j,Pg,Pb);
54    channellist->GetChannel(z)->GetErrorModel()->GetObject<GilbertElliotErrorModel>
          ()->SetErrorMatrixValues(j,i,Pg,Pb);
55    }
56    }
57    }
58
59    }
```