



COSSIMULAÇÃO DE SISTEMAS ELÉTRICOS E DE COMUNICAÇÕES NO CONTEXTO DE REDES ELÉTRICAS INTELIGENTES

André Nudel Albagli

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Elétrica.

Orientadores: Djalma Mosqueira Falcão
José Ferreira de Rezende

Rio de Janeiro
Junho de 2016

COSSIMULAÇÃO DE SISTEMAS ELÉTRICOS E DE COMUNICAÇÕES NO
CONTEXTO DE REDES ELÉTRICAS INTELIGENTES

André Nudel Albagli

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ
COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE)
DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR
EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. Djalma Mosqueira Falcão, D.Sc.

Prof. José Ferreira de Rezende, Dr.

Prof. Lisandro Lovisolo, D.Sc.

Prof. Moises Vidal Ribeiro, D.Sc.

Prof. Célio Vinicius Neves de Albuquerque, Ph.D.

Prof. Glauco Nery Taranto, Ph.D.

RIO DE JANEIRO, RJ – BRASIL
JUNHO DE 2016

Albagli, André Nudel

Cossimulação de Sistemas Elétricos e de Comunicações
no Contexto de Redes Elétricas Inteligentes/André Nudel
Albagli. – Rio de Janeiro: UFRJ/COPPE, 2016.

XVIII, 149 p.: il.; 29,7cm.

Orientadores: Djalma Mosqueira Falcão

José Ferreira de Rezende

Tese (doutorado) – UFRJ/COPPE/Programa de
Engenharia Elétrica, 2016.

Referências Bibliográficas: p. 106 – 116.

1. Redes Elétricas Inteligentes. 2. cossimulação. 3.
Sistemas Distribuídos. I. Falcão, Djalma Mosqueira
et al. II. Universidade Federal do Rio de Janeiro, COPPE,
Programa de Engenharia Elétrica. III. Título.

*Aos meus pais Ruth e Chelomo
Albagli*

Agradecimentos

Gostaria de agradecer a Deus, a meus pais que sempre me apoiaram e estimularam, a minha esposa Cláudia que com paciência sempre me incentivou na continuidade dos estudos e à Petrobras, pela oportunidade que concedeu para realização da tese.

Aos colegas e professores do Laboratório de Rede de Computadores da Universidade de São Paulo, Wilson Ruggiero, Akio Nogueira, João, pelo apoio e pela disponibilidade para esclarecer as eventuais dúvidas no âmbito da modelagem computacional e programação.

Ao colega Cleber Okida, do LARC, pelas inúmeras noites dando apoio na resolução de problemas.

Aos colegas da Petrobras, Tirso Araújo e Zenas, pela inestimável contribuição no âmbito do desenvolvimento das diversas interfaces de programa e pelas discussões de fim de semana, que muito contribuíram para a elaboração desta tese.

Ao colega Gervásio pelo auxílio em disponibilizar servidor para realização das simulações.

Aos professores Djalma Falcão e José Rezende, pela orientação do trabalho e pelos diversos debates em relação aos cenários e topologias a serem avaliadas.

Aos professores Moises Vidal e Lisandro Lovisolo pela participação na banca examinadora e pelos comentários realizados na etapa do exame de qualificação, que muito contribuíram para a melhoria da escrita da tese e pelas sugestões dadas ao trabalho.

Aos professores Glauco e Célio Vinicius, pela participação nessa banca.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

COSSIMULAÇÃO DE SISTEMAS ELÉTRICOS E DE COMUNICAÇÕES NO CONTEXTO DE REDES ELÉTRICAS INTELIGENTES

André Nudel Albagli

Junho/2016

Orientadores: Djalma Mosqueira Falcão
José Ferreira de Rezende

Programa: Engenharia Elétrica

As redes elétricas inteligentes surgiram dentro de um contexto de dificuldades pelos quais os países passavam em função de falta de energia, seja por motivos de demanda crescente, falha na rede elétrica ou por fenômenos climáticos. Com o avanço das tecnologias de comunicação e informação, permitiu-se uma mudança de paradigma na automação das redes elétricas, sobretudo nas redes de distribuição até a residência do consumidor. Novas oportunidades de mercado foram abertas, porém entender como as tecnologias interagem com a rede ainda é um aspecto sob investigação. A tese propõe um estudo de caso de cossimulação da reconfiguração de uma rede de distribuição radial aplicada ao contexto de redes elétricas inteligentes. É desenvolvida uma heurística de controle para reconfiguração da rede elétrica utilizando-se o conceito de sistema distribuído. São avaliados os impactos de três topologias de rede de telecomunicações sobre os tempos de reconfiguração, assim como a latência e a perda de pacotes. É avaliada também a influência de tráfego de medidores inteligentes sobre o desempenho do algoritmo de reconfiguração. Para elaboração deste estudo foi criado um ecossistema para simulação integrada de três diferentes camadas: modelo elétrico, representado pela ferramenta Simulink/Matlab; modelo de rede de telecomunicações, representado pelo simulador Omnet++ e um programa, baseado em sistema multiagente, desenvolvido na linguagem Java. Esses três simuladores são integrados através da arquitetura de serviços denominada HLA (*High Level Architecture*), que é capaz de compartilhar, de forma sincronizada, objetos e atributos entre os federados.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

CO-SIMULATION OF ELECTRIC SYSTEMS AND COMMUNICATION IN
THE CONTEXT OF SMART GRIDS

André Nudel Albagli

June/2016

Advisors: Djalma Mosqueira Falcão

José Ferreira de Rezende

Department: Electrical Engineering

Smart grids appeared in a scenario of difficulties that many countries were facing related to power outages either caused by increasing demand, failures or climate phenomena. With the advance in information and communication technologies, a new paradigm was introduced in electric power systems automation, mainly in electric distribution network and consumer residences. New market opportunities were opened, but to understand how these technologies interact with electric power systems is still under investigation. This thesis proposes a case study of a radial distribution network reconfiguration scenario in the context of smart grids using co-simulation. An heuristic, based on multiagent system, is designed, to control the switches in order to restore power. Three telecommunications network topologies are evaluated and the impact of latency and packet loss on reconfiguration process is verified. In addition, the performance of reconfiguration algorithm is analyzed when background data traffic of smart meters is added. In order to support this study, an ecosystem for smart grid co-simulation is implemented by three different layers: electric model, developed in Simulink/Matlab power system simulator tool; a telecommunication network model, developed in Omnet++ simulator and an application, developed in Java language using multiagent framework. These three simulators are integrated by an architecture called High Level Architecture (HLA), which is able to share objects and attributes among federates in a synchronized way.

Sumário

Lista de Figuras	xii
Lista de Tabelas	xiv
Lista de Abreviaturas	xvi
1 Introdução	1
1.1 Rede Elétrica Inteligente	2
1.2 Contextualização	3
1.3 Objetivos da tese	6
1.4 Contribuições da tese	7
1.5 Divisão do Trabalho	7
2 Revisão bibliográfica	8
2.1 Introdução	8
2.2 Simulação coordenada com HLA	9
2.3 Simulação coordenada baseada em desenvolvimento de código integrado	12
2.4 Simulação coordenada híbrida	15
2.5 Simulação coordenada em tempo real	18
2.6 Resumo	19
3 Simulação de Sistemas Sistribuídos	21
3.1 Introdução	21
3.2 Histórico	22
3.3 Tipos de simulação	23
3.4 Arquitetura HLA	23
3.4.1 Conjunto de regras	24
3.4.2 Modelo de objetos	25
3.4.2.1 Estrutura do Modelo de Objetos	26
3.4.3 Especificação de Serviços	27
3.4.4 Sequência de interações entre Federação e RTI	28
3.4.5 Integração com RTI	29

3.5	Sincronização dos simuladores	30
3.5.1	Gerenciamento do tempo	30
3.5.2	Procedimento para avanço de tempo	31
3.6	Resumo	32
4	Desenvolvimento da Integração	33
4.1	Introdução	33
4.2	Metodologia de integração	34
4.3	Ontologia	37
4.3.1	Ontologia da rede de telecomunicações	38
4.3.2	Ontologia da rede elétrica	38
4.3.3	Ontologia das mensagens do sistema multiagente	39
4.4	Criação do modelo de objetos da federação	41
4.4.1	Representação das classes e atributos	42
4.5	Integração dos federados	43
4.5.1	Integração com o simulador Matlab	44
4.5.1.1	Diagrama de classes	45
4.5.1.2	Diagrama de atividades	47
4.5.1.3	Diagrama de componentes	47
4.5.2	Integração com sistema multiagente	48
4.5.2.1	Diagrama de classe	50
4.5.2.2	Diagrama de sequência	52
4.5.2.3	Diagrama de atividades	54
4.5.3	Integração com simulador Omnet++	55
4.5.3.1	Diagrama de classes	55
4.5.3.2	Diagrama de atividades	57
4.5.3.3	Diagrama de componentes	58
4.5.3.4	Sincronismo do Omnet++	58
4.6	Resumo	59
5	Estudos de Caso	60
5.1	Introdução	60
5.2	Algoritmo de reconfiguração	61
5.3	Validação do pseudocódigo	66
5.4	Cenário de avaliação	68
5.4.1	Capacidade das redes em malha e ponto-multiponto	71
5.5	Métricas de avaliação	72
5.6	Simulação do Matlab	73
5.7	Cenário composto com rede em malha sem-fio	75
5.7.1	Modelo de propagação	78

5.7.2	Protocolo de roteamento	79
5.7.3	Resultados da simulação	80
5.8	Cenário com rede ponto-multiponto	82
5.8.1	Modelo de propagação	84
5.8.2	Cálculo do raio de cobertura	84
5.8.3	Capacidade da rede ponto-multiponto	85
5.8.4	Resultados da simulação	86
5.9	Cenário composto com rede LTE	87
5.9.1	Modelo computacional	89
5.9.2	Modelo de propagação	90
5.9.2.1	Modelo de propagação do 3GPP	90
5.9.2.2	Modelo SUI	91
5.9.3	Cálculo da sensibilidade dos rádios	93
5.9.4	Determinação do ganho da antena do receptor	94
5.9.5	Resultados	96
5.9.5.1	Cenário com modelo de propagação SUI	96
5.10	Conclusão	98
6	Conclusões e Próximos Passos	102
	Referências Bibliográficas	106
A	Conceitos Gerais de Redes Elétricas Inteligentes	117
A.1	Redes elétricas inteligentes	117
A.1.1	Benefícios	118
A.1.2	Histórico e padrões	118
A.1.3	Arquitetura conceitual	122
A.1.3.1	Infraestrutura Tecnológica	124
B	Sistemas Multiagentes	127
B.1	Conceito de Controle Autônomo	127
B.2	Conceito de Sistemas Multiagentes	128
B.2.1	Benefícios	130
B.2.2	Aplicabilidade à redes elétricas inteligentes	131
B.3	Modelos de aplicações distribuídas	132
B.4	FIPA	134
B.4.1	Mensagens FIPA	137
B.4.1.1	Ontologia	139
B.5	Softwares para desenvolvimento	139
B.5.1	JADE	140

C	Tabelas do Arquivo Padrão de Modelos	141
C.1	Tabela de objetos	141
C.2	Tabela de interações	143
C.3	Tabela de atributos	143
C.4	Tabela de parâmetros	146
C.5	Tabela de sincronismo	146
C.6	Tabela de dados	147
C.7	Tabela de dados discretos	148

Lista de Figuras

1.1	Oferta de serviços de cidades inteligentes	5
2.1	Integração JADE e OPNET via RTI	11
2.2	Estrutura VPNET	13
2.3	Arquitetura da simulação híbrida	16
2.4	Arquitetura da simulação em tempo real	18
3.1	Arquitetura RTI e federados	24
3.2	Passos para criação dos federados e RTI	29
3.3	Interface entre RTI e a aplicação	29
3.4	Passos para o gerenciamento de tempo	32
4.1	Encapsulamento dos métodos Java	34
4.2	Modelo de nó da rede	36
4.3	Ontologia do nó da rede	38
4.4	Ontologia dos elementos que compõe uma rede elétrica	39
4.5	Ontologia do conteúdo das mensagens dos agentes	41
4.6	Diagrama de classes dos objetos	42
4.7	Diagrama de classes de interação dos objetos	43
4.8	Visão parcial do modelo elétrico e blocos de integração RTI	46
4.9	Diagrama de classes utilizadas pelo Matlab	47
4.10	Diagrama de atividades executadas pelo Simulink	47
4.11	Diagrama de componentes utilizado para integrar com Simulink	48
4.12	Estrutura da mensagem XML	49
4.13	Diagrama de blocos das interfaces entre o RTI e os agentes	50
4.14	Diagrama de classes do sistema multiagente	52
4.15	Diagrama de sequência para envio de mensagem para um agente	52
4.16	Diagrama de sequência de recebimento de mensagem por um agente	53
4.17	Diagrama de componentes do sistema multiagente	54
4.18	Diagrama de atividades do sistema multiagente	54
4.19	Diagrama de classes do Omnet++	56
4.20	Diagrama de integração dos simuladores	57

4.21	Diagrama de atividades das classes Omnet++	57
4.22	Diagrama de componentes das classes Omnet++	58
5.1	Modelo de rede de distribuição 33 barras modificado	61
5.2	Topologia dos agentes	63
5.3	Fluxo das mensagens trocadas entre agentes	64
5.4	Macrofluxo do algoritmo distribuído	66
5.5	Topologia da rede de telecomunicacões para teste do algoritmo	68
5.6	Topologia dos coletores de tráfego dos medidores inteligentes	70
5.7	Cargas P e Q amostradas a cada 4 segundos	74
5.8	Modelo de cargas P e Q	74
5.9	Topologia da rede de telecomunicações sobre uma grade	77
5.10	Nível de potência recebido em função da distância	79
5.11	Resultado dos tempos de reconfiguração da rede de distribuição de energia elétrica	81
5.12	Arquitetutura da rede de telecomunicações ponto-multiponto	83
5.13	Potência do sinal recebido por um nó da rede PMP	85
5.14	Resultado das simulações com o tráfego dos coletores	87
5.15	OFDMA	88
5.16	Arquitetura básica do LTE	89
5.17	Topologia física das rádio-bases LTE implementadas no Omnet++	90
5.18	Comparação dos modelos de propagação SUI e 3GPP	93
5.19	Padrão de irradiação no plano horizontal	95
5.20	Tempos de reconfiguração da rede elétrica com tecnologia LTE	98
5.21	Comparação dos tempos de reconfiguração	99
A.1	Modelo padrão de referência do IEEE	119
A.2	Arquitetura conceitual elaborada pelo NIST	122
A.3	Modelo conceitual de arquitetura de telecomunicações	123
B.1	Modelo de comunicação direta	133
B.2	Modelo de arquitetura híbrida	134
B.3	Arquitetura FIPA	136

Lista de Tabelas

2.1	Quadro comparativo das estratégias	20
3.1	Tabela de conceitos de tempo	30
4.1	Tabela de campos da mensagem XML	49
5.1	Tabela de parâmetros do modelo IEEE 33 barras	62
5.2	Tabela de aspectos sócio-econômicos da região da Ilha do Governador	69
5.3	Tabela de volume de tráfego gerado pelos coletores em bytes	70
5.4	Tabela de taxa de envio de mensagem em bytes/seg	71
5.5	Tabela de parâmetros dos rádios em malha	75
5.6	Tabela de resultados das simulações	80
5.7	Tabela de tempo de reconfiguração sem o tráfego dos coletores	81
5.8	Tabela de parâmetros utilizados no modelo ponto multiponto	85
5.9	Tabela dos resultados da simulações com tráfego dos coletores	86
5.10	Tabela de resultados das simulações da rede PMP	87
5.11	Tabela de parâmetros do modelo SUI	92
5.12	Tabela de parâmetros utilizados no modelo LTE	92
5.13	Tabela de relação modulação e sinal-ruído	94
5.14	Tabela de simulação LTE com tráfego dos coletores	97
5.15	Tabela de simulação LTE sem coletores de tráfego	97
5.16	Tabela comparativa sem coletores de dados	99
5.17	Percentual de finalização das simulações considerando-se tráfego dos coletores de dados	100
B.1	Tabela de formato da mensagem	135
B.2	Tabela de mensagens FIPA	139
C.1	Tabela de objetos do modelo padrão	142
C.2	Tabela de interações do modelo padrão	143
C.3	Tabela de atributos do modelo padrão	145
C.4	Tabela de parâmetros	146

C.5	Tabela de pontos de sincronismo	147
C.6	Tabela de dados simples	148
C.7	Tabela de dados discretos	149

Lista de Abreviaturas

ACC	Agent Communication Channel, p. 135
ACL	Agent Communication Language, p. 128, 136
ALSP	Aggregation Level Simulation Protocol, p. 22
AMI	Advanced Metering Infrastructure, p. 120
AMS	Agent Management System, p. 135, 136
API	Application Program Interface, p. 139
BDI	Belief-desire-intention, p. 129
CIM	Common Information Model, p. 38
CQI	Channel Quality Indicator, p. 93
DARPA	Defense Advanced Research Project Agency, p. 22
DDM	Data Distribution Management, p. 10
DF	Directory Facilitator, p. 136
DIS	Distributed Interactive Simulation, p. 22
DNP-3	Distributed Network Protocol, p. 14
EISA	Energy Independent Security Act, p. 117
ENodeB	Enhanced NodeB, p. 88
EPC	Evolved Packet Core, p. 88
EPOCHS	Electric Power and Communication Synchronizing Simulator, p. 9, 15
EPRI	Electric Power Research Institute, p. 38, 118
FDD	Frequency Division Duplex, p. 87

FIPA-ACL	Agent Communication Language, p. 40
FIPA-CCL	Constrain Choice Language, p. 133
FIPA-KIF	Knowledge Interchange Format, p. 40, 49, 133, 138
FIPA-RDF	Resource Description Framework, p. 40, 49, 133
FIPA-SL	Semantic Language, p. 40, 49, 133, 138
FIPA	Foundation for Intelligent Physical Agents, p. 33, 49, 130, 133, 135, 136, 138, 139
GECO	Global Event-Driven Co-Simulation, p. 14
GPS	Global Positioning System, p. 18
HLA	High Level Architecture, p. 9–11, 21–25, 33
IEC	International Electrotechnical Committee, p. 38, 42, 118
IEEE	International Electronic and Electric Engineers, p. 14–17, 22, 33, 34, 118, 122–124, 139
IoT	Internet of Things, p. 4
JADE	Java Agent Development Framework, p. 6, 33, 46, 50, 140
LTE	Long Term Evolution, p. 86
M2M	Machine to Machine, p. 4
MAS	Multi-agent Systems, p. 138
MDA	Model Driven Architecture, p. 103
MTLD	Menor Tempo Lógico Disponível, p. 31
MTP	Message Transport Protocol, p. 48
MTS	Message Transport System, p. 136
NED	Network Description Language, p. 34
NIST	National Institute for Standard Technology, p. 117–119, 121, 122
OFDMA	Orthogonal Frequency Division Multiplexing Access, p. 86
OMT	Object Model Template, p. 25, 37

OPGW	Optical Ground Wire, p. 125
OSI	Open Systems Interconnection, p. 38, 55, 58, 68, 143
PDC	Phasor Data Concentrator, p. 18
PLC	Power Line Communications, p. 124
PMP	Ponto-multiponto, p. 81
PMU	Phasor Measurement Unit, p. 125
PSFL	Positive Sequence Load Flow, p. 14
RTI	Runtime Infrastructure, p. 9–11, 23–25, 27–31, 44–46, 48–50, 142
SC-FDMA	Single Carrier Frequency Division Access, p. 86
SCADA	Supervisory Control and Data Acquisition, p. 14
SEP	Sistemas Elétricos de Potência, p. 2
SITL	System in the Loop, p. 18
TAR	Time Advance Request, p. 31
TCP	Transport Control Protocol, p. 24
TDD	Time Division Duplex, p. 87
TENA	Test and Training Architecture, p. 22
TIC	Tecnologia da Informação e Comunicação, p. 8
TTI	Transmit Time Interval, p. 87
UDP	User Datagram Protocol, p. 24
UML	Unified Modeling Language, p. 42, 43
Ue	User Equipment, p. 88
VTB	Virtual Test Bed, p. 12

Capítulo 1

Introdução

O aumento da demanda por energia, principalmente em países considerados grandes consumidores como Estados Unidos e China, tem requisitado novos investimentos em formas de geração e distribuição a fim de suportar esse crescimento da demanda. No entanto, a expansão e a manutenção da infraestrutura requer grandes investimentos e um controle adequado dos indicadores de qualidade de serviço prestado, além da garantia da segurança. Segundo o Departamento de Energia dos EUA [1], estima-se que no mundo serão investidos, na próxima década, mais de US\$ 1 trilhão na ampliação do fornecimento de energia para melhoria da rede elétrica.

É evidente que o planejamento da demanda tem grande importância para a sociedade e para a cadeia produtiva de uma economia. O uso racional e otimizado de diferentes formas de energia contribui para a melhoria do sistema, além de permitir um crescimento sustentado da rede elétrica.

Os últimos anos do século XX e início do século XXI foram marcados por fortes manifestações de diversos países pela adoção de fontes de energias alternativas que não agredem o meio ambiente. Dessa forma, fontes de geração eólica, fotovoltaica, geotérmica, biomassa e das marés que por terem baixa emissão de CO₂ na atmosfera são aderentes ao conceito de sustentabilidade. O uso dessas tecnologias permite preservar o meio ambiente e aumentar a oferta de energia elétrica.

Em 2009, a Comissão Européia [2] estabeleceu um plano de ação para a promoção do uso de fontes de energia alternativas. O documento destaca a grande importância econômica, social e tecnológica que sua adoção pode trazer para a região e estabelece a política de incentivos para implantação e desenvolvimento de novas fontes de energia alternativa. Dentro desse plano também foram definidas as metas a serem alcançadas por cada um dos países membros. Por exemplo, as metas estabelecidas para participação de diferentes fontes de energia limpa em países como Finlândia e Suécia são 38% e 49% no ano de 2020, respectivamente.

O Brasil também possui em sua matriz energética um grande percentual de fontes renováveis. Entre elas, a maior fonte de geração é a hidroelétrica com 60%

de participação total na produção de energia [3]. No entanto, essa fonte representa um custo elevado, relacionado ao meio ambiente, e que tem sido um dos fortes empecilhos para construção de novas hidroelétricas no país. O apelo por novas fontes alternativas e a grande extensão territorial conduzirão nos próximos anos a busca por melhorias no sistema de geração e distribuição de energia .

Em 2003 e 2009, *blackouts* ocorridos nos EUA e no Brasil [4] afetaram milhares de consumidores, causando enormes prejuízos para a economia. O restabelecimento do fornecimento demorou horas e demonstrou a fragilidade desse sistema em se recuperar das falhas ocorridas. Diante desse quadro fica evidente que a modernização da rede elétrica é necessária para garantir não só a sua disponibilidade como também melhorar a sua qualidade e eficiência através da redução da quantidade de interrupções e a manutenção dos níveis de tensão dentro da faixa adequada.

Outros fatores externos e internos como aumento do preço do barril de petróleo associado à crise financeira mundial e à crise hídrica dos últimos anos afetaram profundamente o mercado de geração de energia, elevando significativamente os custos aqui no Brasil.

1.1 Rede Elétrica Inteligente

Em função das crises de energia ocorridas em anos anteriores, o presidente dos Estados Unidos decidiu, em 2008, ampliar os investimentos no setor com a implantação de uma “rede elétrica inteligente”, que consiste na criação de um plano para reduzir a dependência do petróleo, criar empregos e ajudar os Estados Unidos a avançar na tecnologia de geração de energia limpa [5]. O objetivo é a modernização dos sistemas elétricos de potência (SEPs) através de investimentos e avanços tecnológicos. A meta traçada pelo governo é ambiciosa e prevê investir US\$ 11 bilhões para transformar a rede e aumentar a disponibilidade, com a instalação de 5000 Km de linhas de transmissão, e a instalação de mais de 40 milhões de medidores inteligentes de energia residenciais. Prevê-se um novo modelo de geração distribuída no país onde o consumidor poderá ser também um produtor e vendedor do excedente de energia para as distribuidoras. Pretende-se ampliar a capacidade de geração de energia limpa e, como meta, a sua participação na matriz energética do país para 25% até 2025. Esse plano de modernização planeja colocar no mercado mais de um milhão de veículos elétricos híbridos. No longo prazo, o objetivo é reduzir a importação de petróleo à metade e a emissão de CO₂ em 14% em 2020 e 83% em 2050 em relação a 2005. Essa iniciativa trará maior eficiência para o SEP através da minimização das interrupções de energia e introdução das fontes alternativas na rede de distribuição. Essa ação incentivará a criação de novas tecnologias, proporcionando maior competitividade na geração de energia, novos

serviços e equipamentos, e na criação de um novo mercado de trabalho.

A criação de uma “rede elétrica inteligente” oferece um ambiente rico para o uso de tecnologias da informação e telecomunicações e representa grandes desafios para a automação. Alguns aspectos são fundamentais: robustez da rede de telecomunicações, tempo de resposta e atuação adequados dos sistemas de controle de tal forma que, em casos de falhas no SEP ou ainda que na ocorrência de picos de demanda, a rede elétrica possa ser rapidamente reconfigurada, mitigando possíveis impactos econômicos e nos consumidores em geral. Portanto, é necessário que se disponha de unidades de controle inteligentes associadas às SEPs, sistemas de controle sofisticados e redes de telecomunicações bem dimensionados.

Outro aspecto relevante é que o desenvolvimento da infraestrutura de telecomunicações para a rede elétrica deve ser economicamente viável e não onerar o consumidor, além do razoável. Os investimentos, a médio prazo, poderão ser compensados por uma gestão mais eficiente da rede elétrica através da redução das perdas técnicas e não técnicas. Por isso, os planejamentos da rede de telecomunicações e dos sistemas de automação devem ser corretamente elaborados.

Existem dois tipos de arquiteturas de gerência: a centralizada, que consiste em um sistema de computação central, cujo papel é a coleta de todos os dados da rede elétrica e a baseada em sistemas distribuídos [6], capazes de tomar decisões com conhecimento parcial do problema.

Um sistema distribuído, quando associado a algoritmos baseados em inteligência artificial e heurísticas viabilizam diversas aplicações industriais e comerciais como, por exemplo, restauração da energia elétrica após a ocorrência de falhas [7–9].

O cenário citado é apenas um exemplo de aplicação dentro do conceito de redes elétricas inteligentes. Outras aplicações, como gerenciamento da demanda de usuários em microredes [10], gerenciamento inteligente em redes elétricas virtuais [11], gerenciamento inteligente em redes elétricas com geração distribuída [12] e gerenciamento da demanda de energia de veículos elétricos [13], ilustram potenciais desenvolvimentos capazes de tornar a SEP mais eficiente.

1.2 Contextualização

As redes elétricas inteligentes fazem parte de um conceito maior denominado sistema físico-cibernético que consiste na integração de componentes físicos com sistemas computacionais. Sistemas físicos podem ser de qualquer natureza como, por exemplo, químico, mecânico, energia e telecomunicações. Sistemas computacionais estão muito associados à ideia de computação embarcada, com sensores conectados aos sistemas físicos coletando dados e atuando sobre eles. Além das redes elétricas, pode-se citar algumas outros sistemas físicos como: carros com inteligência

embarcada, sistema de monitoração humana, automação predial e iluminação pública. Todos esses sistemas têm potencial econômico, social e para inovação. No entanto, para que se tornem uma realidade, os seguintes desafios necessitam ser vencidos [14]:

- Criação de uma infraestrutura de telecomunicações adequada para suportar o tráfego gerado por milhões ou bilhões de elementos;
- Desenvolvimento de uma inteligência distribuída para controlar todos os elementos;
- Garantia da segurança, integridade e disponibilidade dos dados.

Redes elétricas inteligentes estarão futuramente integradas a três novos conceitos de comunicação entre máquinas: *machine to machine* (M2M), cuja tecnologia poderá se incorporar à automação residencial [15] que proporcionará capilaridade da rede de telecomunicações para os dispositivos elétricos e eletrônicos; *internet of Things* (IoT), cujo papel é igual ao do M2M porém expandido para novas capacidades [16] e cidades inteligentes, cujo conceito engloba todos os anteriores através da integração de todas essas tecnologias.

O conceito de M2M consiste no uso de tecnologia de equipamentos conectados à rede que trocam informações e realizam ações através de sistemas autônomicos com o objetivo de coordenar processos. O mercado para essa tecnologia está em expansão e estima-se que alcance bilhões de dispositivos conectados às redes de telecomunicações nos próximos anos. Entre as várias aplicações possíveis, encontram-se a automação remota, robótica, telemetria, medidores eletrônicos e, mais recentemente, cidades inteligentes [17].

Embora o conceito da tecnologia IoT seja muito parecido com o de M2M, a sua utilização é mais ampla [18]. O objetivo é interligar todos os objetos de uso diário nas residências, prédios, cidades e indústrias ou dispositivos eletrônicos em rede, de forma ubíqua. A IoT fornecerá serviços diferenciados suportados por um nível de inteligência e com diferentes tipos de rede como 3G, 4G, RFID, PLC, Zigbee (IEEE 802.15.4) e rádios de espectro não licenciado (900 MHz e 2.4 GHz). Com o poder de processamento cada vez maior, menor consumo e tamanho, associado a diversos sensores, torna-se possível embarcar essa tecnologia em qualquer tipo de objeto que se deseja controlar. No entanto, controlar uma quantidade de objetos que se estima da ordem de bilhões é uma tarefa cuja tendência aponta para uma descentralização do gerenciamento, isto é, os mecanismos de tomada de decisão estarão mais próximos dos aparelhos do que em *datacenters*, por exemplo. Isso traz um enorme ganho de escala e permite reduzir fluxos de informação para um único ponto centralizado. Conseqüentemente, a demanda por tráfego de rede torna-se menor. Por outro lado, os protocolos de comunicação de dados precisam ser

adequados para suportar a diversidade de serviços, garantindo a qualidade de serviço, assim como o endereçamento desses elementos.

Com o avanço tecnológico dos últimos anos e em função do adensamento populacional nas grandes cidades, governos e sociedades começaram a pensar formas de melhoria da qualidade de vida e do meio-ambiente em que vivem, considerando os conceitos de sustentabilidade e economia “verde”. Nesse contexto, as cidades inteligentes surgem como uma forma alternativa para melhorar a gestão dos diversos problemas utilizando-se tecnologias inovadoras de comunicação de dados e de sistemas, permitindo que o cidadão possa interagir com governos e provedores de serviços, conforme ilustrado na Figura 1.1.



Figura 1.1: Oferta de serviços de cidades inteligentes

As redes elétricas inteligentes representam uma pequena parte das possíveis aplicações. Para as empresas do setor de energia, existem três alternativas para implementação de uma infraestrutura de telecomunicações:

- Utilização de infraestrutura de terceiros, como por exemplo operadoras móveis com tecnologias 3G e 4G;
- Implantar infraestrutura própria;
- Implantar infraestrutura compartilhada com outras empresas.

A decisão por qualquer uma das opções ou uma combinação entre elas requer análises técnica e econômica sobre o comportamento das aplicações compartilhadas em diferentes cenários e situações.

Essa análise requer conhecimento multidisciplinar e o uso de ferramentas de simulação integradas de tal forma que seja possível criar um cenário no qual os elementos de domínios diferentes possam ser manipulados e interajam entre si.

Para que seja possível avaliar um ambiente tão heterogêneo, a abordagem que mais tem sido utilizada é o desenvolvimento de modelagem computacional de sistemas e simulação integrada. Cada simulador descreve um modelo de objeto utilizando linguagem de domínio específico para criar instâncias de equipamentos de telecomunicações como roteadores, rádios e *switches*, componentes de um SEP, tais como cargas trifásicas, chaves, religadores, disjuntores, transformadores, linhas de distribuição, etc.

1.3 Objetivos da tese

Nessa tese busca-se elaborar uma arquitetura composta por diferentes domínios e integrá-los de tal forma que seja possível avaliar os impactos de um domínio (rede de telecomunicações e aplicação sobre a SEP).

A tese desenvolveu uma metodologia para selecionar a arquitetura mais adequada para realização de cossimulação envolvendo os domínios elétrico, telecomunicações e aplicação distribuída. Entre as possíveis soluções, optou-se pelo *High Level Architecture* (HLA), que é um modelo padronizado pelo IEEE 1516-2010 [19]. A solução proposta integra o simulador de rede de telecomunicações Omnet++, o simulador de rede elétrica, Simulink *SymPowerSystems* e uma biblioteca Java de sistema multiagente, utilizada para modelar um sistema distribuído.

A seleção dos simuladores seguiu os critérios de flexibilidade de uso, disponibilidade e capacidade de integração com linguagem de programação. Para cada simulador, foi desenvolvida interface de comunicação com o módulo que compõe a arquitetura HLA. Cada interface é constituída de um conjunto de métodos que recebem e transmitem os dados de cada simulador e gerenciam o avanço de tempo.

O Matlab [20] é simulador para análise de sistemas dinâmicos com uma biblioteca de modelos elétricos e eletrônicos e com uma interface capaz de dispor de funções de entrada e saída de dados desses modelos. Foram desenvolvidas funções no Simulink responsáveis pelo envio de objetos e atributos dos modelos e das interações entre os objetos.

O simulador Omnet++ é uma ferramenta de código aberto, razoavelmente documentada, extremamente modular, o que permite desenvolver modelos de forma hierárquica, além de ser capaz de executar paralelamente as simulações. Para integrá-lo com a arquitetura de cossimulação selecionada, é desenvolvido um conjunto de classes e interfaces para acesso ao modelo de rede de telecomunicações assim como trocar mensagens entre o simulador e a arquitetura de integração.

A biblioteca de multiagentes (JADE) [21], é também de código aberto, seguindo uma arquitetura padrão do IEEE-FIPA [22], bastante utilizada em sistemas distribuídos especialistas e capaz de se integrar a qualquer ambiente de programação.

1.4 Contribuições da tese

É proposta uma metodologia para integração dos simuladores baseada na arquitetura HLA e a criação de uma estrutura de modelos de objetos utilizados pelos simuladores e pela aplicação. Por isso, foram criadas ontologias referentes ao modelo elétrico e ao nó da rede de telecomunicações. A partir dessas estruturas, foram acrescentados atributos dos objetos, que contém os as variáveis elétricas e da rede de telecomunicações e são enviados para os simuladores que tratam esses dados.

É desenvolvido um conjunto de agentes que incorporam uma lógica distribuída para resolução do cenário proposto nesta tese. A interface com o sistema multiagente permite reproduzir com fidelidade as mensagens HTTP trocadas entre os agentes e simuladas na rede de telecomunicações. Além disso é desenvolvido uma ontologia de comandos que fazem parte do conteúdo da mensagem trocada pelo agente e que complementam a lógica distribuída.

Selecionou-se um cenário de falha em uma rede de distribuição de energia elétrica de 33 barras e criado um algoritmo distribuído, utilizando-se agentes distribuídos para resolver o problema de reconfiguração da rede de distribuição de energia elétrica. Três topologias diferentes de rede de telecomunicações (rede em malha sem-fio, ponto-multiponto e LTE) são avaliadas sob a ótica de duas métricas para avaliação do desempenho. As métricas consistem na avaliação do tempo de reconfiguração da rede de distribuição de energia elétrica e o percentual de conclusão do algoritmo de reconfiguração mediante a presença de outro tráfego de dados.

A contribuição foi publicada pelos autores André N. Albagli, Djalma M. Falcão e José F. de Rezende no jornal *Electric Power Systems Research* em Jan/2016, volume 130, sob o título *Smart grid framework co-simulation using HLA architecture* [23].

1.5 Divisão do Trabalho

A tese está organizada da seguinte forma: O Capítulo 2 trata da revisão bibliográfica sobre as principais formas de integração de simuladores. O Capítulo 3 apresenta as padronizações e arquiteturas de simulação coordenada e modelagem de sistemas. O Capítulo 4 apresenta o desenvolvimento de código e as classes e objetos utilizados para realização da integração dos simuladores. O Capítulo 5 apresenta um estudo de caso de simulação baseado nas três topologias de redes de telecomunicações. O Capítulo 6 apresenta as conclusões e próximos passos. O Apêndice A apresenta os conceitos de redes elétricas inteligentes. O Apêndice B apresenta o conceito de sistemas multiagentes e o Apêndice C apresenta tabelas e estruturas de objetos trocados entre os simuladores.

Capítulo 2

Revisão bibliográfica

2.1 Introdução

Nos últimos dez anos, as redes de telecomunicações tiveram uma rápida evolução com a incorporação de novas tecnologias de informação e telecomunicações (TICs) que viabilizam a transmissão de dados de automação da última milha das redes elétricas. Isso traz novos desafios em termos de qualidade, disponibilidade e vulnerabilidade que afetam também a arquitetura das redes de telecomunicações. Espera-se que o conceito de redes elétricas inteligentes permita melhorias na rede elétrica, tornando-a mais eficiente e robusta. A tendência é que a união das TICs com a rede elétrica proporcione uma gama de novos serviços de controle e automação que beneficiarão as empresas e os consumidores.

Os elementos tradicionais da rede elétrica como medidores, transformadores, chaves e cargas, por exemplo, serão fontes geradoras de dados e informação. Além desses elementos, outros estão previstos, como sensores na rede de distribuição, medidores bidirecionais de energia, painéis fotovoltaicos e aerogeradores que geram quantidades diferentes de dados que serão armazenados em sistemas distribuídos.

Redes de telecomunicações podem introduzir efeitos indesejados, tais como latência, perdas de pacote e *jitter*. Portanto, entender como esses fatores refletem na automação de um SEP são importantes para o seu planejamento.

Como não existe comercialmente um simulador que permita a modelagem de redes elétricas, de telecomunicações e o desenvolvimento de sistemas em um mesmo ambiente, a forma mais apropriada de realizá-la é através de cossimulação, isto é, a integração de simuladores que executem de forma simultânea e sincronizada. No entanto, estabelecer o sincronismo de tempo é um problema, já que modelos computacionais de redes elétricas são simulados no domínio do tempo, enquanto os modelos de redes de telecomunicações são, em geral, baseados em eventos discretos.

A dinâmica das redes elétricas é regida por um conjunto de equações diferenciais

cuja solução, quando aplicada a grandes modelos, somente pode ser obtida por métodos numéricos. As equações são discretizadas no tempo e a cada intervalo de tempo, novos valores dos estados são calculados.

A dinâmica da rede de telecomunicações baseia-se na geração de eventos, ou seja, a mudança de estado de um modelo depende da ocorrência de um evento, quando o registro do tempo é realizado. Portanto, para discretizá-lo no tempo é preciso que os intervalos sejam suficientemente pequenos para que os registros das ocorrências estejam muito próximos dos instantes amostrados. Grandes intervalos de tempo podem resultar em erros consideráveis no resultado da simulação.

Quatro abordagens de cossimulação são abordadas em recentes pesquisas [24]:

1. Utilização do *High Level Architecture* (HLA) - É uma arquitetura composta de serviços responsáveis pela troca de dados e sincronismo entre os simuladores;
2. Desenvolvimento de código integrado - Consiste no desenvolvimento da aplicação e da rede de telecomunicações dentro de um único simulador;
3. Híbrida - Consiste em ter pelo menos um simulador executando em tempo real integrado com outros simuladores com avanço de passo;
4. Simulação em tempo real - Consiste no avanço de passo sincronizado com um relógio do *hardware*. Isso tem se mostrado uma tarefa complexa pelo fato que máquinas diferentes requerem que o sincronismo dos relógios seja feito de forma bastante precisa entre eles. Cada avanço do passo de simulação requer a geração das variáveis internas e externas dentro do intervalo de tempo correto.

2.2 Simulação coordenada com HLA

Em [25], os autores apresentam uma das primeiras pesquisas que trata da integração de simuladores, na qual é apresentada a arquitetura denominada *Electric Power and Communication Synchronizing Simulator* (EPOCHS) composta pelos seguintes programas: simulador de rede de telecomunicações (NS-2), simulador de transientes eletromagnéticos (PSCAD), simulador de transientes eletromecânicos (PSLF), *High Level Architecture* (HLA) e uma aplicação de controle.

Na arquitetura proposta, a aplicação comunica-se com o *Runtime Infrastructure* (RTI), que é a classe principal que compõe o HLA, através de um *gateway*. A integração do NS-2 é realizada através da extensão do código para que mensagens TCP e UDP sejam enviadas para o RTI. O sincronismo ocorre através de pontos de sincronização definidos em cada simulador, isto é, enquanto um simulador avança o seu passo, o outro fica aguardando até que os dois pontos de sincronismo coincidam. Essa estratégia pode representar um problema se eventos ocorrerem entre dois pontos

de sincronização. O dado precisa ser armazenado em memória para ser processado no próximo ponto de sincronismo, fazendo com que essas pequenas diferenças se propaguem durante a simulação e conduzam a resultados incorretos. Logo, nesse caso, os intervalos de sincronismo precisarão ser suficientemente pequenos quando comparados com a escala de tempo dos eventos do simulador NS-2. O trabalho, embora pioneiro no assunto, não aprofundou a investigação sobre a implementação nem abordou questões relacionadas à infraestrutura de telecomunicações.

Em [26], os autores propõem a integração dos simuladores PSCAD/EMTDC e OPNET Modeler através do barramento de serviços RTI. Os simuladores são utilizados em máquinas distintas e uma funcionalidade do RTI, denominada *Data Distribution Management* (DDM), é a responsável pelo gerenciamento da troca de dados entre os simuladores. O gerenciamento do tempo é do tipo “pare-continue”, isto é, enquanto um simulador avança no tempo o outro suspende a própria execução até que os tempos estejam sincronizados.

É proposto um cenário composto por uma modelo de teste de rede elétrica IEEE 14 barras [27], simulando uma rede de distribuição de uma cidade, veículos elétricos e uma rede de telecomunicações sem-fio, utilizada para obter informações sobre os níveis de tensão e corrente nas baterias dos carros. Um módulo é desenvolvido para gerenciar as mensagens de solicitação de carga, descarga e controle de energia conectado a medidores fasoriais. O transitório da rede elétrica é analisado considerando-se os diferentes tipos de tecnologia de rede sem-fio (Wi-Fi IEEE 802.11g e Wimax IEEE 802.16e) e os tempos de recuperação durante o processo de descarga das baterias quando ocorre uma falha. O resultado aponta para um menor transitório da rede elétrica quando é adotado padrão IEEE 802.11g. Apesar dos resultados promissores da integração dos simuladores, não foram apresentados detalhes sobre a implementação.

Em [28], os autores apresentam uma arquitetura composta do sistema multiagente baseado no ambiente de desenvolvimento JADE, o simulador de rede de telecomunicações OPNET Modeler e arquitetura HLA. Na solução ilustrada na Figura 2.1, os agentes são integrados ao RTI através de interfaces desenvolvidas assim como o simulador OPNET Modeler.

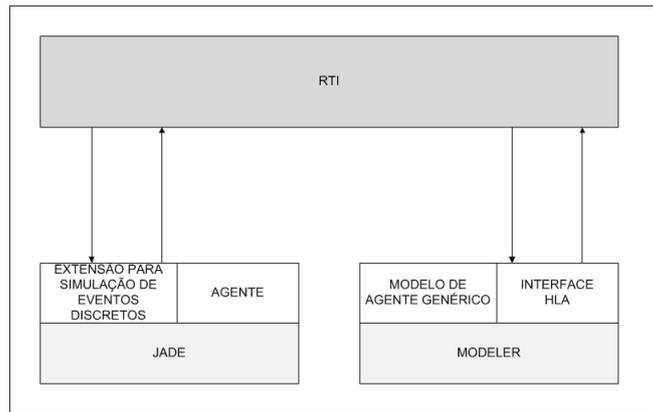


Figura 2.1: Integração JADE e OPNET via RTI

O agente é estendido com funcionalidades de simulação de eventos discretos que se integram ao simulador de rede de telecomunicações. Na Figura 2.1, o bloco “Modelo de Agente Genérico” representa um agente executando uma aplicação cuja função é representar o modelo de um nó da rede de telecomunicações tem apenas o papel de representar a fonte e o sorvedouro de tráfego da rede. Essa camada de aplicação comunica-se com a pilha de protocolos das camadas inferiores que já existem no simulador. A ferramenta OPNET Modeler disponibiliza interfaces de desenvolvimento para o HLA. Essa interface é adaptada para receber e enviar mensagens dos diversos agentes. O bloco “Agente” representa os agentes que simulam a camada de aplicação na qual a lógica de controle está implementada.

Os autores propuseram uma prova de conceito para testar o tempo de resolução de falha na rede elétrica. A arquitetura é constituída de rede de telecomunicações e um conjunto de agentes. Cada agente está associado a uma chave seccionadora da rede elétrica cujo modelo utilizado é o *New England IEEE 39 barras* [29]. Dois cenários foram estudados:

- Arquitetura de multiagentes com modelo concentrado: Um agente coordena a comunicação com outros agentes;
- Arquitetura de multiagentes baseada no modelo ponto-a-ponto: Todos agentes se comunicam diretamente sem a interferência de um terceiro.

Foram feitas várias simulações para ambos os cenários com tráfego de mensagens de tamanhos específicos. Em cada cenário foi adicionado um outro tipo de tráfego de dados na rede de telecomunicações, e avaliou-se o impacto no tempo de resposta. A topologia ponto-a-ponto mostrou-se a mais adequada quando submetida a tráfego compartilhado.

2.3 Simulação coordenada baseada em desenvolvimento de código integrado

Esse tópico aborda pesquisas realizadas com desenvolvimento de interfaces que integram simuladores utilizando-se das próprias estruturas internas do código, filas e linguagens de programação, de tal forma que o resultado combinado é um sistema computacional fortemente acoplado.

Um sistema fortemente acoplado tem dependência direta com uma classe concreta de objetos, ou seja, estão totalmente interligados. Em um sistema fracamente acoplado a dependência não existe ou é feita através de interfaces. Os artigos estudados abordam a estratégia de desenvolver códigos embarcados em simuladores ou fortemente dependente deles.

Em [30], é apresentado um modelo de simulação coordenada a partir de uma arquitetura constituída de três camadas. A primeira camada, denominada *Virtual Test Bed* (VTB) [31] é um simulador de código aberto desenvolvido para prototipação, em larga escala, de sistemas dinâmicos com ênfase em eletrônica de potência. A segunda camada, representada pelo simulador OPNET Modeler, é responsável pelos modelos de rede de telecomunicações. A terceira camada, um módulo é desenvolvido no VTB constituído de programas autônomos com características de sistema distribuído. Cada módulo autônomo foi implementado para se comunicar apenas com seus pares vizinhos a fim de minimizar o tráfego de comunicação. A união dessas camadas permite a análise, na escala de tempo, dos possíveis impactos do desempenho de uma rede de telecomunicação sobre o modelo da rede elétrica.

No VTB, é criado outro módulo com a função de orquestrar a troca de dados entre os simuladores e manter a sincronização. O mecanismo de sincronismo é desenvolvido a partir de uma referência global de tempo. Essa referência permite que os dados de cada simulador sejam amostrados a cada intervalo de tempo predefinido e as mensagens sejam enviadas dos módulos autônomos para o simulador Modeler.

No OPNET Modeler, os agentes são representados pelos nós da rede de telecomunicações e seguem a mesma disposição física dos componentes do modelo da rede elétrica. O código que reflete a implementação da lógica de controle comunica-se com esses nós que, por sua vez, enviam as mensagens através do modelo de rede de telecomunicações [30].

A estrutura, ilustrada na Figura 2.2, conforme os próprios autores destacam, apresenta desvantagens como a complexidade de desenvolvimento do software coordenador e o longo tempo de simulação.

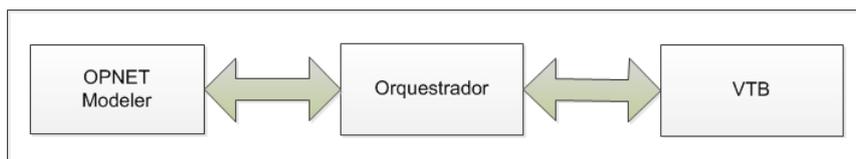


Figura 2.2: Estrutura VPNET

Os autores elaboraram uma prova de conceito constituído por um modelo de rede elétrica para navios movidos com motores elétricos integrado a uma rede de telecomunicações, composta por roteadores interligados, seguindo a topologia física do modelo da rede elétrica. O resultado dessa simulação é a análise do transitório da rede elétrica no domínio do tempo, levando-se em consideração os efeitos da latência dos roteadores e do roteamento.

Em [32], os autores criaram uma estrutura análoga ao modelo do VTB composta por dois simuladores: Modelica[33] e NS-2. Modelica é um simulador de código aberto, desenvolvido por um consórcio de centros de pesquisa, que utiliza uma linguagem de programação que permite desenvolver sistemas complexos como mecânico, elétrico e eletrônico. É uma ferramenta muito flexível, pois é escalável através de adição de módulos hierarquizados, permitindo criar modelos mais complexos.

O modelo de troca de mensagens adotado é diferente de outros trabalhos já publicados e consiste na utilização de um mecanismo de comunicação entre processos denominado *named pipes* ou filas, cujo princípio baseia-se na leitura e escrita dos dados em arquivos.

O sincronismo é gerenciado por um módulo desenvolvido e integrado ao simulador NS-2. Dessa forma, o NS-2 comanda o processo de leitura e escrita dos dados para o Modelica. O sincronismo ocorre na forma de “inicia-para”, isto é, enquanto o NS-2 está rodando, Modelica permanece parado, aguardando a liberação. O mesmo ocorre ao contrário. A solução apresentada não permite sincronizar a partir do Modelica, o que representa um ponto de falha. Nele não é apresentado nenhum cenário de estudo, o que torna difícil a comprovação da ideia proposta.

Em [34], os autores propõem o mesmo conceito de desenvolvimento de código integrador dentro de um dos simuladores. É utilizado o simulador de sistemas de potência PSCAD/EMTDC [35] que não possui nenhuma integração com outros simuladores. Tudo é feito no PSCAD, inclusive a simulação do comportamento de uma rede de telecomunicações. A abordagem é complexa pois cria vários módulos internos que simulam características como latência e perda de pacotes.

Dois módulos desenvolvidos representam os “motores” da integração. O primeiro módulo é responsável pelo envio e recebimento de mensagens e gerenciamento e sincronismo dos dados. Ele incorpora a camada de aplicação que é constituída

por um mecanismo de controle de perda de pacotes, baseado em uma função de distribuição de probabilidade, e um protocolo que, no caso do estudo, é o *Distributed Network Protocol* (DNP-3) [36], adotado em processos de automação de redes elétricas utilizando sistemas SCADA [37]. O segundo módulo tem a função de receber as mensagens e encaminhá-las, conforme um perfil de atraso e descarte de pacotes, para o seu nó de destino, simulando um elemento de roteamento de rede de telecomunicações.

No artigo, é apresentada uma análise de um cenário composto pelo sistema de teste IEEE 13 barras [38] com uma unidade geradora eólica com potência geradora de 0,66 MW, conectado a um alimentador e bateria com conversor bidirecional. A bateria acumula toda a energia gerada. Compartilham o modelo os dois módulos descritos. O cenário é submetido às variações do vento e, conseqüentemente, às variações de potência no modelo do gerador eólico.

É demonstrado através de simulações que, em função da latência da rede de telecomunicações, as tensões e frequências do gerador sofrem alterações momentâneas quando o modelo é submetido a essas variações de velocidade do vento.

Percebe-se a complexidade da arquitetura já que o modelo do nó da rede de telecomunicações necessita de desenvolvimento de código para simular perda de pacote, latência e roteamento. Além disso, a rede de telecomunicações não é escalável e não há possibilidade de inclusão de outros tráfegos concorrentes com o de automação.

Em [39], é desenvolvida uma arquitetura de cossimulação, na qual é criado um controlador e escalonador de eventos denominado *Global Event-Driven Co-Simulation* (GECO). O escalonador é responsável pela referência global de tempo e pelo gerenciamento da fila de eventos. A fila recebe os eventos dos simuladores e ordena em função da etiqueta de tempo associada. Somente um evento é processado por vez. Durante a execução, caso exista um evento de interação, os simuladores são capazes de suspender o processamento e retornar o controle para o gerenciador global. Esse mecanismo garante então a adequada sincronização entre os simuladores.

Os autores propõe uma arquitetura de cossimulação com o simulador para cálculo de fluxo de carga PSFL, da empresa GE, e o simulador de rede de telecomunicações NS-2. No PSFL, é desenvolvido um módulo que recebe os dados do NS-2 e simultaneamente atualiza os atributos dos modelos elétricos. Por outro lado, no NS-2 é desenvolvida uma nova classe de objetos responsável pelo controle de eventos do PSFL. Quando a execução se inicia, uma série de iterações do PSFL são prealocadas na fila de eventos globais. Quando há necessidade de outra rodada de iteração, o PSFL é liberado através de um comando. Além dessa classe, outras foram

criadas com o objetivo de simular sistemas autônomos que definem a estratégia de controle. A comunicação entre entre esses sistemas autônomos é feita através do NS-2, utilizando-se a pilha de protocolos TCP e UDP.

Em [39], é proposta uma prova de conceito que consiste em um esquema de proteção de linhas de transmissão utilizando-se o sistema *New England* IEEE 39 barras com acionamento de relés de impedância em três zonas de proteção. A operação é baseada na medição de tensão e corrente que o relé percebe. Quando um curto-circuito acontece, o relé “sente” uma variação na impedância medida. Nesse cenário, 68 agentes são responsáveis pelo algoritmo de recomposição das linhas de transmissão. O esquema de proteção desta rede é desenvolvido em máquinas de estado.

Ainda em [39], os autores realizam uma comparação entre a arquitetura EPOCHS e o modelo proposto no artigo. Foi demonstrado, via simulação, que as duas arquiteturas de cossimulação reproduziam o mesmo transitório da rede elétrica no caso que a arquitetura EPOCHS utiliza intervalo de passo pequeno. Quando o intervalo de tempo é ampliado as diferenças entre os resultados apresentados eram maiores.

Em termos de escalabilidade, segundo os próprios autores, os módulos desenvolvidos podem tornar-se um gargalo quando cenários maiores forem estudados, o que pode requerer outras abordagens como simulação paralela.

2.4 Simulação coordenada híbrida

Em [40], é apresentada uma arquitetura híbrida, baseada na arquitetura HLA, denominada INSPIRE, composta pelos seguintes simuladores: OPNET Modeler, Digsilent e aplicação desenvolvida em Java que emula uma subestação baseada no modelos de dados da norma IEC 61850 [41]. A cossimulação permite a avaliação de transitórios do SEP observando-se os impactos gerados pelos tempos de propagação de uma rede de telecomunicações e pelas aplicações desenvolvidas. O cenário avaliado é um sistema de monitoração, controle e proteção de sistemas de transmissão.

O simulador Digsilent é utilizado para modelagem de complexos SEPs, capaz de exportar dados em tempo-real e resolve as equações diferenciais em intervalos de tempo discretos e não dispõe de interface para sincronismo. Para resolver o problema, os autores desenvolvem APIs para acesso aos dados e para a integração com o RTI.

A rede de telecomunicações, modelada para cada subestação, é composta de *switches*, para atendimento à rede local, que interconecta vários modelos computacionais de controladoras, e um roteador interligado ao *backbone* da rede

de telecomunicações através de enlaces de baixa velocidade (64 kbps).

Na camada de aplicação, é implementado o protocolo *Manufacturing Message Specification* (MMS), da norma IEC 61850 [41], que é normalmente utilizado em automação de subestações. O protocolo contém mensagens que são usadas na proteção e controle na subestação.

É feita uma análise de cenários sobre o modelo de sistema de teste *New England* IEEE 39 barras, modificado para os estudos em questão, e desenvolvido o modelo de dados da norma IEC 61850 nas ferramentas OPNET Modeler e no sistema desenvolvido em Java que simula a subestação.

As simulações demonstram que há impacto da latência da rede de telecomunicações quando uma falha na rede elétrica é detectada e um novo despacho de energia é realizado. Outra análise apresenta os efeitos da falha da rede de telecomunicações e inclusão de um tráfego adicional de dados nos enlaces da rede de comunicação. Os resultados mostram que uma latência adicional pode trazer consequências mais graves no sistema de proteção da rede elétrica, embora o cenário de utilização dos enlaces e as respectivas taxas de interconexão a 64 kbps tenham contribuído fortemente para que os efeitos fossem observados.

Em [42], os autores apresentam o modelo de cossimulação híbrida do cenário apresentado em [40], porém com maior detalhamento. O ambiente é composto por diversas camadas de simulação, divididas em modelos computacionais da rede elétrica, da rede de telecomunicações, da subestação e do centro de controle, conforme ilustrado na Figura 2.3.

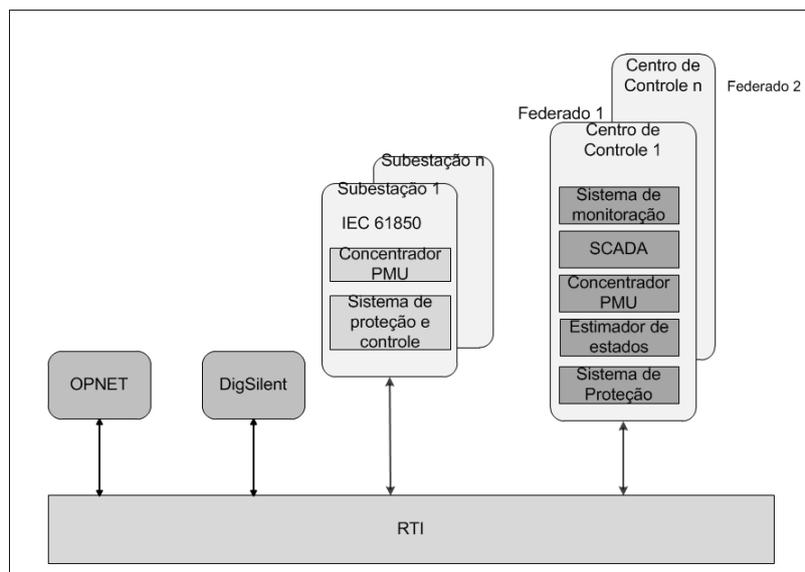


Figura 2.3: Arquitetura da simulação híbrida

A arquitetura é composta pelos mesmos simuladores e vários federados, os quais são desenvolvidos na linguagem Java. Entre eles, o modelo computacional

de subestação utiliza a estrutura de dados da norma IEC 61850 [41] que é composta por vários subsistemas. Esse modelo é responsável por representar todas as funções da subestação e coletar os dados gerados pela subestação. Toda interação entre modelos computacionais da subestação e os atributos dos modelos, como tensões e correntes, são publicados no RTI. Após a publicação para o RTI, os modelos computacionais dos centros de controle e das subestações coletam esses dados e trocam informações seguindo a norma IEC 61850, através do RTI. A arquitetura de rede de telecomunicações é baseada no modelo de infraestrutura de subestação da norma IEC 61850.

O centro de controle, que é todo desenvolvido em linguagem de programação Java, é composto por uma série de módulos que simulam o gerenciamento de uma subestação, dentre os quais: sistema SCADA, sistema de monitoração, controle e proteção e estimador de estados.

O objetivo do artigo é a avaliação do desempenho de cenários o mais próximo possível da realidade, isto é, naquele em que os parâmetros de duração adotados fossem reais quando a rede elétrica fosse submetida a uma situação de *stress*. Para se alcançar a precisão desejada, os autores levaram em consideração os seguintes parâmetros: tempos de propagação da informação na rede de telecomunicações, tempo de execução das aplicações, tempo de espera para envio das informações, tempo de comutação de equipamentos primários da rede elétrica e tempo de evolução da dinâmica do modelo do sistema de potência. Esses tempos estão incluídos nos modelos, a cada passo de simulação, cujo valor é também enviado através do RTI. Segundo os autores, uma vez que estejam bem definidos, o resultado da simulação torna-se muito próximo do real.

É proposto um cenário, como prova de conceito, que consiste no controle da proteção de um modelo de uma rede de transmissão *New England IEEE 39* barras com 10 geradores, acrescido de três controladores de fluxo de potência e uma fonte de corrente contínua em alta tensão.

A subestação possui um nó de controle, uma switch na rede de telecomunicações local e um roteador conectado a uma taxa de 1,55 Mbps. A aplicação roda sobre o protocolo UDP e as mensagens possuem tamanho fixo de 240 bytes. Em um certo momento, uma carga da rede de transmissão é desconectada e o processo de controle entra em operação. A partir desse ponto, é realizada a análise do transitório da rede elétrica.

O tempo de propagação da rede de telecomunicações foram desprezados quando comparados com os de comutação de *taps* dos transformadores, assim como os tempos de execução das aplicações. O tempo de espera para envio de dados é da ordem de segundos devido ao tempo de comutação dos *taps* dos transformadores.

2.5 Simulação coordenada em tempo real

Em [43], os autores apresentam uma arquitetura aplicada ao cenário de controle e monitoração de grandes áreas. A cossimulação é composta pelos programas: OPNET Modeler, Matlab/Simulink e um *software* de controle e gerência.

Os sistemas responsáveis por gestão e monitoração são compostos por medidores fasoriais, *Phasor Data Concentrators* (PDC) e uma aplicação de controle que possui um algoritmo próprio para tomada de decisão baseado nos dados fasoriais gerados. Os medidores fasoriais são equipamentos que têm como referência de sincronismo o *Global Positioning System* (GPS), portanto os dados gerados são bastante precisos. Os dados dos medidores fasoriais são transferidos, através da rede de telecomunicações, para o PDC que recebe os dados e os organiza por ordem de chegada e com o registro do tempo.

O sincronismo é realizado através da interligação com *Global Positioning System* (GPS). Para uma simulação de modelo de rede elétrica em tempo real é utilizado um *hardware*, chamado OPAL-RT [44], capaz de ler o modelo gerado pelo Simulink e criar um conjunto de blocos que simula medidores fasoriais, conforme ilustrado na Figura 2.4. Esses blocos têm a função de gerar dados e enviá-los via TCP ou UDP para o PDC. O simulador OPNET Modeler dispõe de um módulo *System in the Loop* SITL que permite a interação de dados com o mundo externo, através de interfaces, como se fosse uma rede real de telecomunicações. Por fim, o concentrador é desenvolvido sobre a ferramenta de código aberto OpenPDC [45].

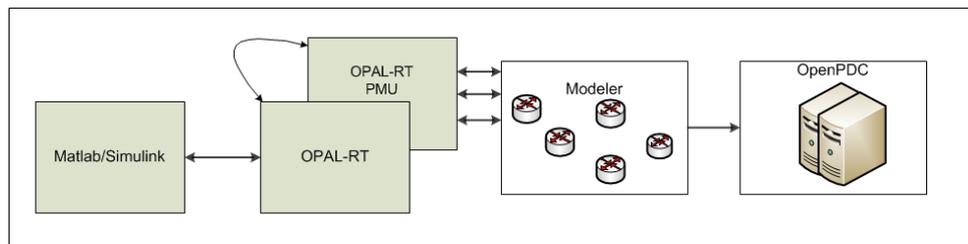


Figura 2.4: Arquitetura da simulação em tempo real

Os autores apresentaram um cenário para prova de conceito, apenas para ilustrar a potencialidade da arquitetura e a forma como os parâmetros puderam ser obtidos. Foram modelados cinco medidores fasoriais e cinco roteadores, alocados em subredes das subestações e uma rede elétrica. O resultado é apenas uma comparação das latências encontradas para cada medidor fasorial quando os dados são transmitidos pela rede de telecomunicações. Além disso, não se observou referência aos potenciais impactos ocasionados pelas latências ou perdas de pacote que poderiam ocasionar na rede elétrica.

2.6 Resumo

Para se integrar diferentes ambientes de simulação foram estudadas quatro estratégias. A primeira, adota uma arquitetura distribuída baseada no HLA. A principal característica da solução é permitir um baixo acoplamento entre simuladores, o que significa maior independência entre eles.

Uma segunda abordagem adotou um desenvolvimento bem mais complexo no qual o escalonador e gerenciador compõem uma parte do simulador. Isso dificulta ainda mais, por particularizar a integração, além de forçar um acoplamento rígido. O simulador acaba tendo dois papéis, sendo que um deles pode dificultar a escalabilidade, além de não flexibilizá-la.

Uma terceira estratégia, utilizada mais recentemente, é a criação de um escalonador e gerenciador de fila de eventos. A integração ocorre através das filas de gerência de cada ambiente o que demanda maior complexidade e acoplamento entre as ferramentas. Nesse caso, a grande vantagem é a precisão dos resultados na escala de tempo. Por outro lado, os autores destacam que mesmo nesta arquitetura, o escalonador pode ser um gargalo para escalabilidade.

Uma quarta e última estratégia é composta de simuladores baseados em tempo real e sincronizados via GPS. São utilizadas ferramentas comerciais que possuem alta precisão e reproduzem a rede em *hardware* apropriado para esse tipo de cenário. É uma solução bastante adequada para análise de transitórios, porém também restrita quanto à escalabilidade de uma rede elétrica. Fica evidente que a precisão dos resultados depende do correto sincronismo das aplicações. O uso do GPS assegura que os simuladores estejam sincronizados e oferece mais precisão nos resultados das simulações, porém o seu uso é mais relevante quando se trata de avaliar impactos nos transitórios da rede elétrica.

O desenvolvimento da fila global de eventos parece ser uma boa estratégia para resultado mais preciso. No entanto, os autores fazem uma ressalva quanto à escalabilidade da solução. Observam também por meio de comparação de simulações da arquitetura EPOCHS e da GECO que a forma e os tempos dos transitórios são muito próximos o que justifica que arquitetura HLA também pode apresentar resultados muito próximos se adotados passos de simulação pequenos.

Com base na Tabela 2.1 foi selecionada a arquitetura HLA pelas seguintes razões:

- Oferece serviço de sincronismo;
- O desenvolvimento da integração é menos complexo do que o desenvolvimento de código embarcado já que não é necessário ter conhecimento de todo o esquema de escalonamento de eventos do simulador. Através de chamadas dos métodos do código do HLA [19] é possível enviar ou receber dados;

- Para a prova de conceito proposta, a escala de tempo não é um fator crítico quando comparada com os tempos utilizados nos simuladores;
- A arquitetura é padronizada, o que permite integrar-se outros simuladores que, no futuro, poderão já estar aderentes;
- A arquitetura permite um fraco acoplamento entre simuladores, ou seja, não é preciso desenvolver código em um simulador que faça referência a outro.

Estratégia	Sincronismo	Vantagem	Desvantagem
HLA	Tempo lógico	Baixo acoplamento entre Simuladores	Requer pequenos avanço de passo
HLA	Tempo real	Baixo acoplamento Maior precisão	Maior complexidade
Desenvolvimento de código	gerenciamento próprio	Totalmente integrado	Forte acoplamento Difícil manutenção
Fila global de eventos	gerenciamento próprio	Totalmente integrado Sem problema de sincronismo	Alto acoplamento Escalabilidade
Tempo real	GPS Relógio de máquina	Precisão dos resultados	Escalabilidade

Tabela 2.1: Quadro comparativo das estratégias

Capítulo 3

Simulação de Sistemas Sistribuídos

3.1 Introdução

O objetivo deste capítulo é apresentar os conceitos sobre simulação distribuída e o HLA necessários para o desenvolvimento da arquitetura adotada nesta tese. A simulação distribuída é uma tecnologia que permite o compartilhamento da execução de um programa por diversos núcleos de uma máquina. Essa divisão de tarefas pode ser realizada de duas formas: paralelismo, através da geração de um código que quebre as tarefas em subtarefas independentes, de tal forma que o resultado final seja a combinação delas; distribuição de sistemas com objetivo de integrar diversos simuladores em um único ambiente.

A simulação distribuída permite reduzir o acoplamento entre subsistemas, tornando-os mais independentes. Nesse caso, cada simulador possui modelos computacionais especificamente desenvolvidos para cada domínio de conhecimento. Isso significa dividir um modelo em partes menores e explorar o que há de específico e de melhor em cada simulador. O resultado final é uma solução composta, a partir da qual é possível analisar o comportamento físico de todo um processo.

As simulações são classificadas em [46]:

- **Analítica:** Utiliza modelos detalhados e discretizados no tempo capaz de reproduzir eventos em ordem cronológica de fenômenos físicos, como, por exemplo, redes elétricas, modelos mecânicos e sistemas de telecomunicações. O objetivo é a coleta de dados estatísticos para análise dos resultados.
- **Virtual:** Utiliza simulação de cenários com possibilidade de interação e manipulação do ambiente virtual. É largamente adotada pelas forças armadas para virtualização de campos de batalha e treinamento militar pois permite que pessoas possam participar desse ambiente. Dentro dos cenários, é possível a inclusão de *hardwares* para testes e validação, permitindo mais rapidez nos resultados e com menores custos.

3.2 Histórico

O estudo sobre simulação distribuída incia-se na década de 80 quando se buscava por alternativas mais eficientes e baratas para realização de simulações de guerra, cenários de campos de batalha e testes de tecnologia, proporcionando uma “imersão” total no cenário avaliado. O aspecto mais importante da integração era a criação de um mecanismo de sincronismo que era necessário para garantir a ordem cronológica dos eventos distribuídos pelos simuladores.

Inicialmente, o *Defense Advanced Research Project Agency* (DARPA) apresentou a arquitetura SIMNET [46], destinada a integração de ambientes virtuais para treinamento militar. Em função do bem sucedido desenvolvimento, outra arquitetura, denominada *Distributed Interactive Simulation* (DIS), foi desenvolvida para permitir a simulação de exercícios de guerra das três forças armadas, separadas geograficamente e interligadas por uma rede de telecomunicações.

Após anos 90, surgiu outra arquitetura, mais avançada, denominada *Aggregation Level Simulation Protocol* (ALSP) [47], que foi desenvolvida para as forças armadas e incorporava as características das arquiteturas predecessoras. O conceito de sua arquitetura era baseado em modelos caracterizados pelos seus atributos. Essas arquiteturas seguiam a mesma organização hierárquica em linguagem de programação orientada a objetos. Cada simulador possuía um mapeamento entre os seus modelos internos e aqueles que eram adotados pelo grupo de simuladores. Essa arquitetura foi posteriormente substituída por outra, denominada *High level Architecture* (HLA) [19].

Outra arquitetura proposta, denominada *Test and Training Architecture* (TENA) [48], foi desenvolvida e seu modelo distribuído permitia a integração e simulação de diversos sistemas com base no conceito de virtualização, interação e construção de ambiente. A TENA consiste na utilização de modelos reais, ambientes virtuais (emulando entidades físicas reais) e um mundo puramente sintético onde um grande número de elementos, baseado em modelos complexos, interage entre si. Embora fosse desenvolvida pelo DARPA, é livremente distribuída e ainda encontra aplicação no ambiente militar.

Posteriormente, já em fins dos anos 90, surgiu o HLA, arquitetura adotada pelo departamento de defesa dos Estados Unidos. Nesse período, o DARPA recomendou que as novas versões de simuladores estivessem preparadas para integrar-se através dessa arquitetura. Do ponto de vista tecnológico, o HLA permite criar simulações analíticas [46] e virtuais.

No ano 2000, o IEEE padronizou o HLA, cuja norma ficou conhecida por IEEE 1516-2000, sendo revisada em 2010 (IEEE 1516-2010). Esse padrão chamou a atenção dos centros de pesquisa, especialmente com o objetivo de utilizar essa

arquitetura em cenários de redes elétricas inteligentes.

3.3 Tipos de simulação

As simulações dividem-se em dois tipos: discreta e contínua [46]. Na simulação contínua, por exemplo, os modelos são regidos por equações diferenciais que descrevem a trajetória em função do tempo e os estados do sistema são atualizados continuamente.

Já em uma simulação discreta, o sistema somente muda de estado em instantes discretos ao longo da execução e ainda se divide em duas outras categorias: orientada a eventos e avanço de passo. O primeiro é caracterizado quando o sistema somente altera os seus estados na ocorrência de um evento. Quando isso ocorre, o evento recebe uma marcação de tempo associada. Nessa classificação enquadra-se o simulador de telecomunicações. A simulação por avanço de passo ocorre quando o tempo é subdividido em intervalos e a simulação é executada a cada passo. Nessa classificação os simuladores Matlab e Modelica como exemplos que se enquadram.

3.4 Arquitetura HLA

O HLA é um conjunto de especificações de serviços que permite a integração de aplicações através da troca de dados de forma sincronizada. A arquitetura é composta por sistemas independentes, denominados federados, e pelo agrupamento de serviços denominado *Runtime Infrastructure* (RTI), conforme ilustrado na Figura 3.1. O grupo de federados é denominado de federação.

Entre o federado e o RTI são enviadas ou recebidas entidades como classes de objetos e um conjunto de atributos desses objetos, classes de interações e um conjunto de parâmetros dessas interações. As classes de objetos são selecionadas a fim de criar uma organização de uma estrutura de dados compartilhada pelos federados. Os atributos identificam uma parte do objeto que são compartilhados pelo RTI através da execução de serviços que são identificados como atualizar atributos e, para os federados que recebem, refletir atributos. A interação representa uma classe de tomada de ação de um federado que terá efeito ou impacto sobre outro federado e o parâmetro representa a informação que é associada à classe de interação. Assim como as classes de objetos são compartilhadas, o mesmo acontece com as interações e seus parâmetros. Cada federado envia a interação através do serviço disponibilizado pelo RTI para outro federado que recebe os parâmetros e dá o tratamento.

Para compartilhar os dados entre as aplicações, o HLA utiliza um mecanismo para o envio e recebimento de mensagens baseado no paradigma de publicar-assinar. Para se assegurar que uma arquitetura de sistemas distribuídos opere de uma forma

autônoma, desacoplada e tolerante a falhas de comunicação, adota-se a estratégia de enviar mensagens ou “publicá-las”, de forma assíncrona, sem saber o sistema destino. Por outro lado, sistemas que tenham interesse em receber essa mensagem devem “assinar”. Quem gerencia essas mensagens é uma camada de orquestração que as recebe e notifica para quem assinou. As mensagens são colocadas em uma fila e removidas quando são enviadas por quem assinou.

Caso os federados estejam instalados em plataformas geograficamente distribuídas, o RTI faz o papel de interconexão entre eles, sendo que a comunicação pode ser feita pela Internet ou em uma rede local. Nesses casos, o RTI permite que se configure os protocolos de transporte TCP ou UDP.

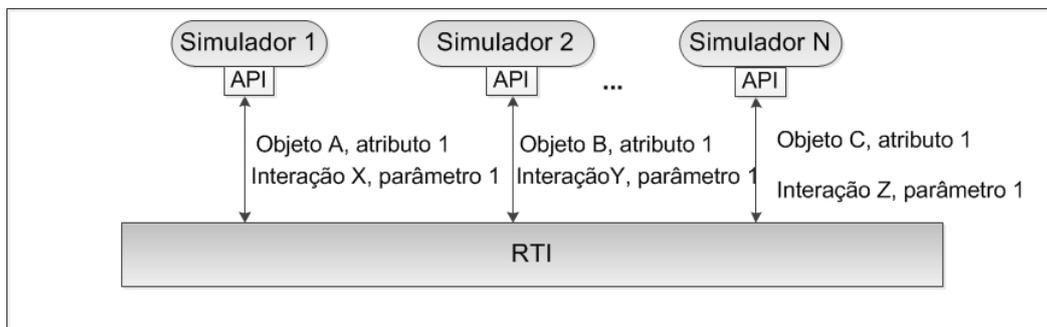


Figura 3.1: Arquitetura RTI e federados

O modelo de arquitetura estabelece três componentes principais: O HLA com um conjunto de regras, a especificação da interface e a sintaxe do modelo que descreve os objetos utilizados pelos federados.

3.4.1 Conjunto de regras

Um conjunto de regras foi criado para estabelecer papéis e responsabilidades que se aplicam ao federado e à federação. Esse conjunto define as seguintes regras:

1. A federação deve ter um modelo de objetos que são instanciados e seguem o modelo padrão de objetos definido na norma IEEE 1516-2010 [49];
2. Na federação, a representação dos objetos instanciados deve estar no federado e não no RTI;
3. Na federação, toda troca de objetos somente pode ocorrer através do barramento RTI;
4. Na federação, toda interação entre federados somente pode ocorrer via barramento RTI através dos serviços disponibilizados;

5. Na federação, durante a simulação, o RTI permite que federados controlem diferentes atributos de um mesmo objeto compartilhado. Esse controle pode ser concedido ou removido a qualquer instante da simulação. No entanto, somente um federado pode ter o controle por vez;
6. O federado deve ter um conjunto de modelos de objetos simulados que são instanciados e especificados no modelo de objetos. Esse conjunto define as classes, atributos e as interações que ocorrem entre os federados;
7. O federado deve ser capaz de atualizar qualquer atributo de classes de objetos instanciadas, assim como interagir de acordo com o modelo de objetos definido para o federado;
8. O federado deve ser capaz de transferir/aceitar a propriedade ou a posse dos atributos dos objetos, isto é, o controle de um atributo é transferido de um federado para outro;
9. O federado deve ser capaz de alterar as condições de atualização dos atributos dos objetos instanciados, como por exemplo, a alteração a taxa de amostragem de um atributo de uma determinada classe;
10. O federado deve ser capaz de gerenciar o tempo de simulação local de tal forma que permita a troca de dados com outros federados de forma sincronizada.

3.4.2 Modelo de objetos

A norma define uma representação para um conjunto de classes de objetos, atributos, formatos e características relacionadas às trocas de dados entre os federados [49]. Essa representação é escrita em no arquivo *Object Model template* (OMT) e especifica a estrutura de dados trocados entre os federados, permitindo o reuso de seus componentes.

Os objetos do HLA possuem certa semelhança com o conceito de programação orientada a objetos. Em termos de programação, um objeto é um componente que representa uma entidade física e que contém variáveis que representam o estado interno do objeto implementado através de um conjunto de propriedades denominadas atributos. No OMT, os objetos e os atributos são lidos e seus valores são enviados através do RTI. Por exemplo, pode-se definir um objeto denominado “transformador” e seus atributos como “potência máxima”, “tensão” no primário, “corrente” no primário.

O conceito de classe de objetos tem estreita semelhança com a orientação a objeto. Uma classe, por definição, representa um conjunto de objetos que têm propriedades em comum e podem realizar as mesmas ações. As classes de objeto

do modelo também possuem o mecanismo de herança. A herança permite que objetos compartilhem atributos e métodos e adotem um relacionamento do tipo “é um”, permitindo representar uma generalização ou especialização entre duas classes. As subclasses além de possuírem os mesmos atributos da classe superior a que pertencem, podem também ter novos atributos. Para exemplificar, se a classe “geração distribuída” possui duas subclasses “painel fotovoltaico” e “turbina eólica”, ambas herdarão atributo “potência ativa” da classe geração distribuída.

3.4.2.1 Estrutura do Modelo de Objetos

O OMT é um arquivo escrito na sintaxe XML e é constituído por um grupo de Tabelas que representa os objetos, suas relações e seus atributos. As seguintes Tabelas são utilizadas no modelo e representadas na estrutura de dados:

- a) **Tabela de identificação do modelo:** Contém informações como versão, data de modificação, descrição, etc;
- b) **Tabela de classes:** Contém classes e as subclasses na forma hierárquica. Cada classe tem associado um parâmetro de controle do tipo publica/assina;
- c) **Tabela de Interações:** Contém um conjunto de ações que um federado pode tomar e impactar em outros federados. Essas ações obedecem também a uma estrutura hierárquica com classes e subclasses e têm um parâmetro de controle do tipo publica/assina;
- d) **Tabela de atributos:** Contém um conjunto de informações de atributos das classes e o tipo de dado associado, podendo ser *char*, *string*, *int*, *float* ou outro customizável. Os atributos são classificados pela política de atualização como: estática, periódica ou condicional. Atributos possuem parâmetros que possibilitam a transferência de delegação ou receber delegação, isto é, se a posse de um objeto pode ser transferida para outro federado. Dependendo do grau de confiabilidade da rede de telecomunicações, os atributos podem ser enviados através do protocolo TCP ou UDP;
- e) **Tabela de parâmetros:** Contém as classes de interação entre federados e valores de atributos, podendo ser do tipo *int*, *float*, *string* e customizáveis;
- f) **Tabela de dimensão:** Especifica dimensões para filtragem dos atributos e interações. Geralmente é utilizado em associação com o mecanismo de gerenciamento de distribuição de dados que tem o papel de limitador de entrega de dados de certas classes para os federados;

- g) **Tabela de representação do tempo:** O RTI permite definir duas representações de tempo: etiqueta de tempo (*timestamp*) ou *lookahead*. Ambos podem ser definidos no formato inteiro 64 bits ou ponto flutuante 64 bits;
- h) **Tabela de sincronização:** O RTI disponibiliza uma funcionalidade para que os federados se sincronizem, isto é, todos os federados devem alcançar um determinado ponto da execução a partir do qual todos estejam sincronizados. Esses pontos de sincronismo são comunicados através do RTI;
- i) **Tabela de tipos de transporte:** O RTI disponibiliza dois tipos de transporte de dados para envio de interação e de atributos:
 - *HLAreliable* → Baseado no protocolo TCP
 - *HLAbesteffort* → Baseado no protocolo UDP

O primeiro oferece garantia de entrega das mensagens. É mais adequado em ambientes geograficamente distribuídos e conectados através da rede de longa distância. O segundo tipo não oferece garantia de entrega e pode ser utilizado quando os federados se encontram em uma rede local;

- j) **Tabela de taxa de atualização:** O RTI disponibiliza um mecanismo de amostragem dos atributos cuja unidade é dada em Hz(Hertz). Um federado pode, quando executa a assinatura de um atributo, especificar a taxa de atualização;
- k) **Tabela de configuração:** Contém uma série de parâmetros que podem afetar tanto federados como a federação e alguns deles podem ser alterados em tempo de execução;
- l) **Tabela de tipos de dados:** É uma estrutura de dados que contém os vários tipos de dados já predefinidos;
- m) **Tabela de informações:** É uma simples tabela onde se pode acrescentar informações gerais sobre o modelo utilizado.

3.4.3 Especificação de Serviços

O RTI disponibiliza um conjunto de serviços [19], dividido em sete grupos:

- a) **Gerenciamento da federação:** Representa o conjunto de serviços que coordena o ciclo de vida da federação, de pontos de sincronismo, entrada e saída de federados da federação;

- b) **Gerenciamento de declaração:** Permite que federados publiquem ou assinem quais objetos vão ser enviados ou lidos através do RTI;
- c) **Gerenciamento de tempo:** Define um conjunto de serviços para operar o sincronismo entre os federados e pela manutenção de um relógio virtual comum a todos e suporta o controle de eventos discretos e a garantia da ordem de chegada desses eventos ao RTI;
- d) **Serviços de suporte:** Define um grupo de serviços gerais para o gerenciamento da federação.
- e) **Gerenciamento de objetos:** Define um conjunto de serviços de registro de objetos e descoberta desses, atualização e remoção de instâncias de objetos, assim como envio e recebimento de interações;
- f) **Gerenciamento de delegação dos objetos:** Define um conjunto de serviços que são utilizados para manipulação de objetos dentro da federação, tal como a transferência de delegação entre os federados. É possível delegar para outro federado a responsabilidade de remover um determinado objeto;
- g) **Gerenciamento de distribuição de dados:** Define um conjunto de serviços que gerencia a distribuição de dados entre os federados, permitindo que haja um controle de tráfego mais eficiente. Em cenários com muitos federados, é possível restringir o envio de classes de objetos e interações com o objetivo de reduzir o tráfego da rede de telecomunicações que interliga os diversos federados que se encontram espalhados geograficamente.

3.4.4 Sequência de interações entre Federação e RTI

Para um federado participar da federação é necessário seguir alguns passos para estabelecer a conexão entre a federação e o RTI, ilustrados na sequência lógica da Figura 3.2. Cada passo é executado pelos métodos desenvolvidos nas linguagens de programação Java, C++ e Matlab, que fazem chamadas de serviços ao barramento RTI. Existem vários outros métodos que o padrão IEEE 1516-2010 define, disponibilizando outros recursos para programas mais complexos. O gerenciamento da federação é responsável pelas funções relacionadas ao controle, criação, remoção do federado e a finalização da federação.

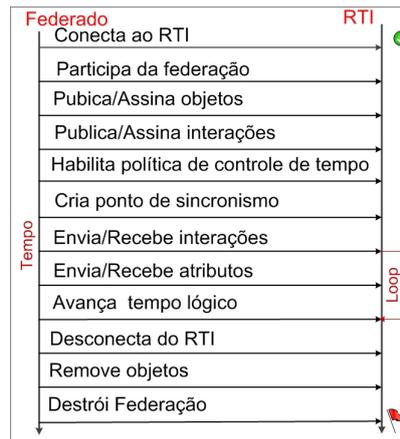


Figura 3.2: Passos para criação dos federados e RTI

Uma vez inicializados, os federados continuam a execução de suas atividades e ao mesmo tempo estão aptos a receber e enviar classes de interações e atualizações de atributos das classes de objetos. Quando a simulação é finalizada, o federado desconecta-se da federação e remove os seus objetos. A federação pode ser destruída, caso não exista mais nenhum federado anexado ao barramento. Durante a simulação, os federados podem suspender, temporariamente, a execução dos federados. A continuidade da execução está vinculada ao alcance do ponto de sincronismo por todos os federados.

3.4.5 Integração com RTI

A chamada de serviços ocorre através de interfaces programáveis desenvolvidas na linguagem Java ou C++, conforme descrito no padrão IEEE 1516-2010. As aplicações instanciam duas classes do RTI: *RTIFederate*, que agrega os serviços citados na seção 3.4.3, e a classe *FederateAmbassador*, que agrega os métodos que recebem mensagens de interação e atributos dos objetos. Toda comunicação com os simuladores ocorre através das chamadas aos métodos (métodos de retorno) da classe *FederateAmbassador*, conforme ilustrado na Figura 3.3.

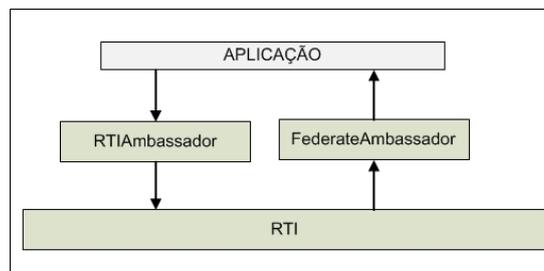


Figura 3.3: Interface entre RTI e a aplicação

3.5 Sincronização dos simuladores

Compreender como o tempo é representado em uma simulação é muito importante para assegurar que os dados sejam corretamente disponibilizados aos participantes da federação no mesmo instante de tempo [46]. Para melhor compreensão sobre o tema, algumas definições básicas sobre tempo são apresentadas, descritas na Tabela 3.1.

Definição	Conceito
Relógio de sistema	É o tempo atual do relógio.
Tempo lógico	É o tempo adotado pelo simulador, individualmente, ou pela federação. A unidade será aquela que for a mais adequada ao cenário simulado podendo ser microsegundos, segundos ou horas.
Tempo real	É o intervalo de tempo executado no simulador, cuja duração é a mesma em um cenário real.
Relógio sincronizado	Indica a situação na qual o relógio de sistema é igual ao tempo lógico do simulador.
Tempo reduzido	Indica que o tempo no federado avança mais rápido que o relógio do sistema.
Tempo expandido	Indica que o tempo no federado avança mais lentamente do que o tempo de relógio do sistema.

Tabela 3.1: Tabela de conceitos de tempo

3.5.1 Gerenciamento do tempo

Federados podem enviar dados fora de ordem cronológica, se assim forem configurados dessa forma. Esse comportamento pode levar a anomalias nos resultados das simulações, tais como a causa de um evento aparecer depois do seu efeito. A cada execução, os resultados podem ser totalmente diferentes e, portanto, a reprodução do modelo não é garantida. Por isso, o sincronismo é necessário para que se possa reproduzir as relações na escala de tempo.

O RTI é responsável por garantir a interoperabilidade entre os federados através de um grupo de serviços que permite que cada simulador use diferentes mecanismos de gerenciamento de tempo. Isto significa que cada federado não necessita ter conhecimento da forma da gestão do tempo em outros federados.

Dois mecanismos de gerência são disponíveis no HLA:

- **Baseado em eventos discretos:** Cada federado processa os seus eventos e avança no tempo a cada ocorrência;

- **Baseado em avanço de passo:** O avanço de tempo é feito em passos discretos e somente ocorre depois que todas as atividades de simulação tenham sido executadas até o instante corrente;

É possível que em uma federação alguns federados tenham abordagens distintas para o gerenciamento do tempo. O importante é garantir a causalidade temporal, assim como assegurar o avanço do tempo em cada federado. O RTI pode garantir essa causalidade, desde que dois serviços sejam chamados: habilitação da regulação do tempo e habilitação da restrição do tempo. O primeiro serviço define que os eventos sejam enviados em ordem cronológica e previne que outros federados avancem o tempo de simulação antes que o RTI libere para o próximo passo [50]. O segundo serviço permite que o federado receba os eventos na ordem cronológica.

3.5.2 Procedimento para avanço de tempo

O RTI trata dois tipos de serviço de ordenação de mensagens: **ordem de chegada**, que são inseridas em uma fila para posterior processamento e **ordem cronológica** que são mensagens estritamente ordenadas no tempo. O avanço de tempo de simulação somente ocorre quando o federado invoca o método *Time Advance Request* (TAR). Quando todas as mensagens são enviadas, o RTI envia uma mensagem de retorno, informando que o avanço foi concedido com sucesso.

O tempo máximo que um federado pode avançar é denominado **Maior Tempo Lógico Disponível** (MTLD). O MTLD garante a causalidade dos eventos e representa o instante de tempo que um federado pode avançar com segurança. No mecanismo conservativo, cujos eventos são processados na ordem cronológica, o tempo é utilizado para limitar o avanço e garantir que as mensagens cheguem na ordem certa e é expresso por:

$$MTLD = \min_n \{ TL_n + lookahead_n \} \quad (3.1)$$

em que TL_n é o tempo lógico do n -ésimo federado e *lookahead* é uma restrição imposta em cada federado. Essa restrição define o intervalo de tempo adicional que um federado pode avançar, ou seja, nenhum federado poderá enviar mensagens que tenham etiqueta de tempo inferior a MTLD, evitando que as mesmas estejam fora de ordem.

Um federado que é regulado no tempo somente entrega as mensagens na ordem cronológica quando a etiqueta de tempo for maior ou igual ao MTLD. O RTI se encarrega de colocar o objeto e atributos em um *buffer*. Uma vez que o tempo MTLD seja alcançado, o RTI libera o federado, informando-o através da chamada ao método de retorno *timeAdvanceGrant()* da classe *RTIAmbassador*. Simultaneamente o RTI

envia os atributos das classes de objetos e os parâmetros das classes de interação para os federados que “assinaram” pelos objetos, conforme ilustrado na Figura 3.4.

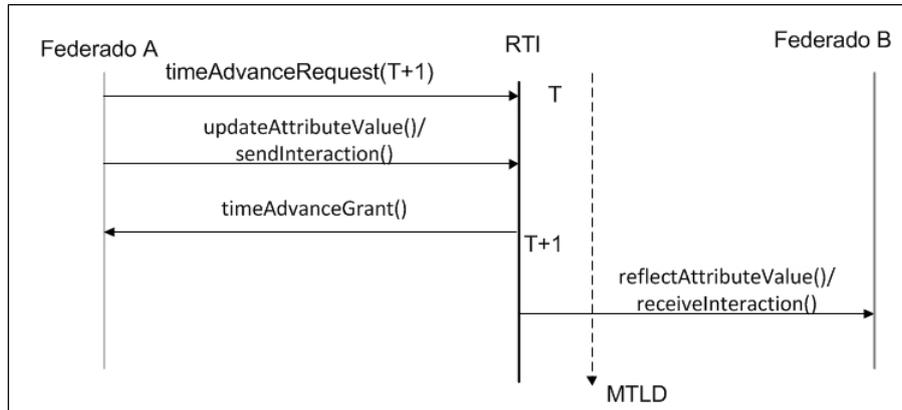


Figura 3.4: Passos para o gerenciamento de tempo

Não existe uma fórmula definida para atribuir um valor ao *lookahead*, pois depende muito do cenário que se está simulando e dos modelos simulados. Existem algumas boas práticas [51] que podem ser adotadas. Em simulações baseadas em avanço de passo fixo, o *lookahead* deve ser da mesma ordem de grandeza do intervalo de passo do federado. Por exemplo, em se tratando de simulação de uma rede de telecomunicações, o ideal é que o avanço de passo seja da ordem de grandeza do passo do federado.

3.6 Resumo

Foram apresentadas as bases da simulação distribuída. Foi introduzido o método de simulação coordenada com HLA e os mecanismos para integração e sincronismo.

Na técnica de avanço de passo fixo, há um aspecto quanto à possível imprecisão dos resultados, se o intervalo for grande. A ocorrência de eventos entre duas amostragens necessita que a notificação do envio de uma mensagem seja feita no próximo passo, podendo ocorrer a propagação do erro. Logo, para mitigar esse risco, faz-se necessário o uso de intervalos pequenos, comparáveis com a ordem de grandeza dos passos dos simuladores. Há então um claro compromisso entre precisão dos resultados e tempo de simulação. Isso torna-se mais evidente no caso de simulação com transitório muito rápido, em que se deseja analisar os impactos sobre a rede elétrica em pequenos intervalos de tempo.

Capítulo 4

Desenvolvimento da Integração

4.1 Introdução

O objetivo deste capítulo é apresentar o desenvolvimento da integração dos simuladores com o RTI composto de três programas: inteligência distribuída, baseado na arquitetura de múltiplos agentes denominada *Java Agent Development Framework* (JADE), o simulador de redes de telecomunicações Omnet++ e o simulador de redes elétricas Matlab Simulink. A união dos três permite criar cenários através dos quais é possível estudar a influência que as redes de telecomunicações e a aplicação exercem sobre as redes elétricas, assim como determinar a latência da rede com a aplicação e possíveis impactos de tráfegos de dados concorrentes sobre a rede elétrica.

A arquitetura de simulação coordenada será montada utilizando-se o HLA, pelo fato de ser um padrão do IEEE e proporcionar um baixo acoplamento entre simuladores. O HLA dispõe de algumas soluções comerciais, tais como das empresas *Pitch Technologies* [52], *VT MAK* [53], *SimWare* [54], *RAYTHEON* [55] e as soluções de código aberto, tais como *OpenRTI* [56], *GERTICO* [57], *OpenHLA* [58], *poRTIco* [59] e *CERTI* [60]. Entre as de código aberto, foi selecionada a *software poRTIco* pelo fato de ser a mais bem documentada e por ser aderente à norma do IEEE 1516-2010. O núcleo do código é desenvolvido em Java e possui interfaces escritas na linguagem C++ que encapsulam as chamadas dos métodos em Java, conforme ilustrado na Figura 4.1. O *software poRTIco* é uma arquitetura desenvolvida na Universidade da Austrália que tem sido utilizado em pesquisas sobre integração de sistemas.

A biblioteca JADE foi selecionada por ser aderente ao padrão IEEE FIPA (Apêndice B) e por estar em constante evolução. O padrão representa um grupo de especificações definidas para proporcionar a interoperabilidade com outras tecnologias de agentes, sendo largamente utilizada pelas universidades em estudos

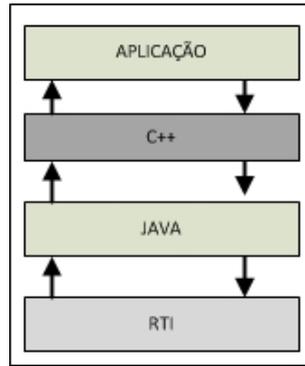


Figura 4.1: Encapsulamento dos métodos Java

que envolvem sistemas distribuídos.

O simulador Omnet++ possui uma ampla biblioteca de modelos de componentes como roteadores, *switches*, rádios, fibra, computadores, pilhas de protocolos do modelo OSI (UDP, TCP, IPV6, ethernet, MPLS, OSPF, BGP, etc), da camada de enlace de redes sem-fio e cabeadas (ethernet, ppp, IEEE 802.11, etc). A arquitetura do simulador permite o desenvolvimento de modelos hierárquicos, isto é, os objetos de rede podem ser hierarquias de outras classes formando uma estrutura bem organizada, facilitando o seu reuso. Além disso, é possível operar em modo gráfico para criar a arquitetura de rede utilizando-se a sintaxe *Network Description Language* (NED) [61] que, através de comandos, permite criar hierarquias de modelos ou novos modelos. O núcleo do simulador é desenvolvido na linguagem C++, de forma modular. As classes de objetos podem ser estendidas pelo usuário para aperfeiçoamento ou para criação de novos modelos, integrando-se a outros programas desenvolvidos em linguagens, tais como Java e C#, e com base de dados para geração de relatórios. A versão mais recente permite simulação em tempo real.

O Matlab Simulink é uma ferramenta de simulação e análise de sistemas dinâmicos, dentre os quais destaca-se o SEP. As simulações podem ser executadas em tempo real, desde que o código seja portátil para um *hardware* capaz de executar em tempo-real. Os métodos numéricos definem o tipo de avanço de passo que pode ser fixo ou variável. A escolha depende da dinâmica e do tamanho do modelo simulado.

4.2 Metodologia de integração

Para a criação de um modelo de integração, o IEEE [62] elaborou uma metodologia que define as melhores práticas para desenvolvimento de um ambiente de simulação distribuída. Essas práticas descrevem, em linhas gerais, um processo para a construção de uma federação, a partir dos seguintes passos:

1. **Definir os objetivos do ambiente de simulação:** Listar um conjunto de

necessidades e gerar documentação detalhada sobre os objetivos da simulação. Neste contexto, a simulação tem como objetivo reproduzir um cenário de uma rede de distribuição de energia elétrica quando uma falta ocorre em um ramal da rede. Os agentes do sistema distribuído lêem continuamente os dados de potência ativa e reativa das cargas, através de sensores, e em caso de falha executam um algoritmo para reconfiguração da rede de distribuição de energia elétrica. Os agentes trocam mensagens que seguem o padrão IEEE-FIPA, por intermédio do RTI e simulados na rede de telecomunicações. O protocolo utilizado para comunicação entre os agentes é o HTTP. O fato de não se adotar o HTTPS é por não ser escopo desta tese avaliar a segurança da rede de telecomunicações e impactos sobre a rede elétrica. A partir do conteúdo da mensagem do protocolo HTTP é calculado o tamanho exato da mensagem que servirá como parâmetro para simulação da rede de telecomunicações.

2. **Desenvolver análise conceitual:** Criar uma representação dos domínios a serem simulados e desenvolver o cenário a ser estudado. O desenvolvimento é composto por três fases:

- Desenvolvimento do cenário a ser estudado, incluindo-se as entidades que representam a federação, a localização de objetos, a descrição funcional das capacidades, os comportamentos e os relacionamentos entre os objetos;
- Desenvolvimento do modelo conceitual da federação, que é uma representação em alto nível de abstração, que independe dos simuladores adotados para representá-los. Serve como meio para transformar os objetivos da simulação em algo mais real, tal como a implementação do código;
- Desenvolvimento dos requisitos da federação composto pela definição do cenário a ser estudado, pela forma do gerenciamento de tempo (tempo-real x tempo lógico), pelos requisitos de *hardware* e *software* necessários para rodar todos os simuladores e o RTI.

O cenário a ser estudado é de uma rede de distribuição de energia elétrica composta por 33 barras com cargas e chaves normalmente abertas e fechadas. Foi definido por premissa que essa rede deve estar localizada em uma grande área suburbana da cidade. Todos os componentes do modelo da rede de distribuição de energia elétrica citada possuem capacidade de se comunicar e interagir com outros componentes, além de ter autonomia na gestão dos dados coletados. Entende-se que estes componentes podem ser interpretados cada um

como um nó de uma rede distribuída composto pelas camadas representadas na Figura 4.2.

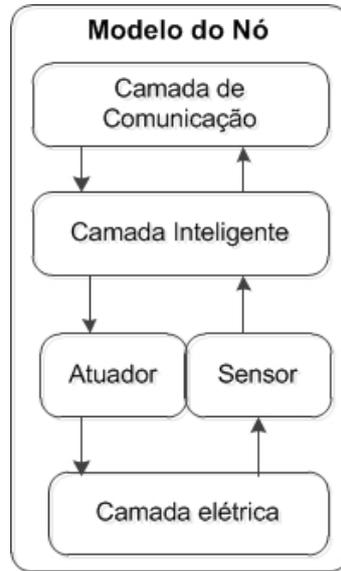


Figura 4.2: Modelo de nó da rede

Cada agente tem papel e comportamento distinto, isto é, cada nó da rede elétrica tem uma ação a ser executada dentro do algoritmo de reconfiguração da rede de distribuição de energia elétrica;

3. **Projetar o ambiente de simulação** - Definir os simuladores que participarão do ambiente, suas funcionalidades e a arquitetura a ser adotada;
4. **Desenvolver o ambiente de simulação** - Desenvolver o modelo de troca de dados entre os federados e adequar as aplicações envolvidas na federação. Para se criar esse modelo será utilizado o conceito de ontologia que é capaz de organizar objetos de forma hierarquicamente estruturada;
5. **Integrar e testar ambiente de simulação** - Planejar e executar a simulação testando todos os federados, antes da simulação propriamente dita, e realizando as devidas correções, até que todos os problemas tenham sido resolvidos;
6. **Executar a simulação** - Executar a simulação e coleta os dados;
7. **Analisar dados e avaliar os resultados** - Verificar os resultados da simulação e elaborar relatório qualitativo.

Os passos de 1 a 4 são descritos neste capítulo enquanto os passos 5 e 7 são detalhados no capítulo 5.

4.3 Ontologia

Como os simuladores já foram estabelecidos, o primeiro passo é a criação da lista de objetos com os respectivos atributos e as interações entre eles que serão utilizadas pelos federados. Esses objetos são organizados em uma estrutura hierárquica e, dependendo da quantidade de elementos e atributos, podem tornar o processo complexo. Os objetos são elementos que fazem parte de cada domínio de informação em cada federado. É proposto um formalismo que antecede a criação do arquivo padrão. Esse formalismo permite a organização em uma tabela, agrupando objetos que guardam relação entre si e os seus respectivos atributos. Dessa forma fica mais fácil, em uma segunda etapa, transportar esses objetos para o arquivo *Object Model Template* (OMT). Essa forma de estruturação do domínio do conhecimento é denominada de ontologia.

Uma modelagem baseada em ontologias permite estruturar um conceito de forma a facilitar a sua representação. Existem diversas definições para ontologia [63], entre as quais destacam-se duas pelo fato de representarem de forma sintética o que as outras descrevem:

- a) Ontologia é uma especificação explícita de uma conceituação [64]. É a criação, de um modelo abstrato de algum fenômeno, através da identificação de conceitos relevantes e suas restrições;
- b) A ontologia de domínio expressa conceituações que são específicas para um domínio [65].

A ontologia permite a estruturação de uma base de conhecimento para acesso aos objetos e suas relações, de tal forma que seja possível recuperar informações de forma fácil.

A ontologia de um domínio de uma área do conhecimento define tipos de objetos, suas propriedades e suas relações com suas instâncias, como por exemplo, “é um” ou “é parte de”. Forma-se, então, uma hierarquia de conceitos representada por uma classe principal e as subclasses. Para se criar uma ontologia, não existe uma metodologia definida, mas um conjunto de recomendações que orientam o desenvolvimento. Uma das estratégias é reutilizar ontologias já existentes, evitando nomes e convenções que sejam somente adotados por um único usuário [66].

Propõe-se criar modelos de ontologias para as camadas de aplicação, rede de telecomunicações e rede elétrica. Todos os modelos apresentados neste estudo foram criados na ferramenta *Protégé*, desenvolvida pela Universidade de Stanford [67].

Como os federados Simulink e Omnet++ são de domínios de conhecimentos diferentes, são construídas ontologias para cada um deles e estruturadas no arquivo de objetos OMT.

4.3.1 Ontologia da rede de telecomunicações

Um nó da rede de telecomunicações é constituído por blocos básicos que seguem a estrutura do modelo OSI, como camada de aplicação, rede e acesso ao meio físico. Para se utilizar esses blocos no âmbito da simulação coordenada e no contexto estudado, a rede de telecomunicações é apenas utilizada como meio de transporte das mensagens originadas na camada de aplicação. Portanto, sob esta ótica, propõe-se modelar o nó, conforme ilustrado na Figura 4.3, caracterizando-o basicamente pelo tipo de aplicação envolvida e por uma identificação do nó que será utilizada pela aplicação.



Figura 4.3: Ontologia do nó da rede

O campo **TipodaMensagem** identifica que a mensagem segue o padrão FIPA que é utilizado pelo JADE (vide apêndice B). Os nomes das aplicações propostas, tais como **LeituradeMedidores** ou **GerenciamentodaDemanda** podem ser utilizados para identificar a aplicação que está sendo atendida pelo agente em um dado instante de tempo.

4.3.2 Ontologia da rede elétrica

A criação da ontologia de rede elétrica baseou-se no reuso do modelo de objetos das normas IEC 61970 e 61968 [68] como ponto de partida para o desenvolvimento. Ambas foram elaboradas pela organização *Electric Power Research Institute* (EPRI). O objetivo da IEC é garantir a interoperabilidade entre aplicações de empresas do mercado de energia e estabelecer uma definição comum para os componentes em SEPs e é largamente adotado pela indústria e pelas concessionárias de energia. As IEC são objetos de pesquisa, no contexto de redes elétricas inteligentes, por padronizar modelos de dados e protocolos de comunicação.

Os objetos das normas são criados a partir de uma estrutura chamada de *Common Information Model* (CIM). O CIM é uma forma abstrata de representar

elementos gerenciáveis em um ambiente de tecnologia da informação como classes de objetos, seus atributos e as relações entre eles.

Para não ficar restrito ao cenário estudado, a ontologia foi estruturada de uma forma genérica a partir das classes de objetos das normas e foi estendida para se adequar ao ambiente de redes elétricas inteligentes. O objetivo não é utilizar todo o modelo, mas apenas complementar parte de uma estrutura existente das normas com novos elementos, embora não utilizados na simulação, mas que podem ser utilizados em outras pesquisas. Destaca-se na ontologia a inclusão de modelos de aparelhos inteligentes, baterias, fontes alternativas de energia, capacitorEmPoste, conforme ilustrado na Figura 4.4



Figura 4.4: Ontologia dos elementos que compõe uma rede elétrica

4.3.3 Ontologia das mensagens do sistema multiagente

Uma ontologia padrão aplicada à sistemas multiagentes não existe porque depende do cenário estudado e das tarefas que cada agente executa. A mensagem que é trocada pelos agentes é formada por vários campos, conforme ilustrado na Tabela B.2, do apêndice B, dentre os quais destacam-se: conteúdo, ontologia e linguagem. O conteúdo define as ações que serão executadas. Apesar do padrão FIPA informar que

o conteúdo é geralmente constituído por expressões lógicas que seguem a semântica FIPA-SL (*semantic language*), o seu formato é livre. No entanto, optou-se pela construção de uma lista simplificada de comandos que são trocados pelos agentes para que executem operações sobre o modelo da rede elétrica. Esses comandos permitem que um agente possa solicitar, por exemplo, o status de um religador, comandar abertura ou fechamento de uma chave ou solicitar a outro agente, que representa um segmento de uma linha de distribuição, o valor total das cargas associadas.

A linguagem indica a sintaxe usada para expressar o conteúdo que todos os agentes devem compreender. Embora o padrão IEEE FIPA-ACL (*Agent Communication Language*) não obrigue a adoção de nenhum tipo de linguagem para expressar o conteúdo, algumas linguagens como FIPA-SL (*Semantic Language*), FIPA-KIF (*Knowledge Interchange Format*), FIPA-(RDF *Resource Description Framework*), citadas na norma, são capazes de expressar proposições lógicas complexas. Entre elas, a FIPA-SL é a mais recomendada e adotada pelas universidades por ser padronizada também.

O JADE possui três formas de comunicação [69]. A primeira é constituída de troca de mensagens no formato de *strings* e é a mais simples, pois não representa relações. A segunda, via serialização de objetos Java, muito utilizada quando os agentes estão todos no mesmo ambiente de desenvolvimento. A terceira, utiliza a transmissão de ontologias dos objetos de tal forma que as mensagens enviadas sejam codificadas ou decodificadas por outros agentes no formato padrão de ontologias FIPA-SL.

A FIPA-ACL estabelece uma regra de formação do conteúdo da mensagem que é uma classificação de todos possíveis elementos que apareçam em um domínio de discurso. Os elementos importantes são:

- a) **Atributo** - Representado por expressões que explicam algo sobre o *status* de um componente do domínio, como por exemplo, verdadeiro ou falso;
- b) **Conceito** - Representado por expressões de estruturas complexas definidas em um campo que devem aparecer em conjunto com seus atributos;
- c) **Ação** - Representado por uma ação que um agente pode desempenhar.

Como na prova de conceito não se adotou a regra da linguagem semântica, aplicou-se essas classificações à ontologia proposta, conforme ilustrado na Figura 4.5. O conteúdo da mensagem é constituído de um grupo de ações que representa os comandos a serem executados por outros agentes, como por exemplo **LeAtributos.corrente** ou simplesmente **FechaChave(CH(001))**. A mensagem é constituída de atributos, que representam um *status* de um

elemento, como por exemplo (**Chave0001(estáFechada)**). O conceito é uma representação da entidade, como por exemplo, (((**Subestação001**)(**Falha**)(**Trafo0001(Temperatura(Alta))**))), na qual informa que em uma subestação uma falha ocorreu no transformador e foi ocasionado pela alta temperatura do óleo. O campo da mensagem denominado “ontologia” contém o nome da estrutura da linguagem de domínio específico. Como se trata de uma ontologia de rede elétrica, adotou-se a mesma ontologia da norma IEC 61970, denominada **RecursodeSistemadePotência**.

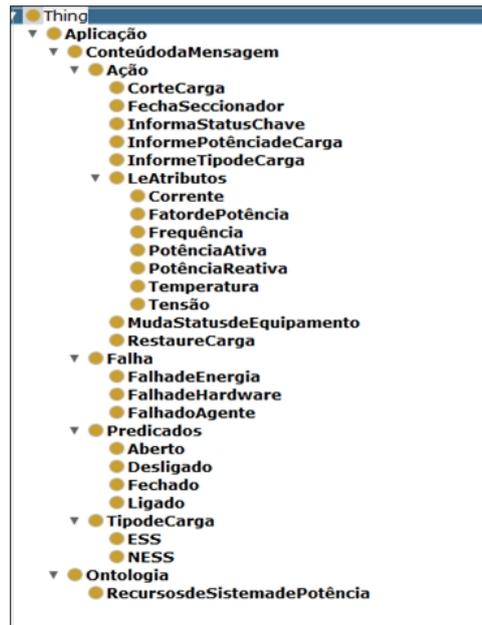


Figura 4.5: Ontologia do conteúdo das mensagens dos agentes

Alguns comandos enviados por agentes podem ser expressos de acordo com os seguintes exemplos de mensagens trocadas:

*RecursodeSistemadePotencia.ConsumidordeEnergia.CargaTrifasica***REQUEST**
(*CorteCarga*)

*RecursodeSistemadePotencia.ConsumidordeEnergia.CargaTrifasica***REQUEST**
(*TipodeCarga*)

Nesses conteúdos identificam-se os objetos envolvidos, o ato performativo (vide apêndice B) e o comando solicitado pelo agente.

4.4 Criação do modelo de objetos da federação

Embora o arquivo de modelos de objetos seja composto por quinze tabelas, para este estudo são utilizadas apenas sete que são fundamentais para qualquer desenvolvimento de modelo: tabela de classe de objetos, tabela de interações, tabela de atributos, tabela de parâmetros, tabela de representação do tempo, tabela de

sincronismo e tabela de dados simples e numerados. Para auxiliar na criação utilizou-se o programa editor de modelos **SimGe** [70]. As especificações de cada tabela e demais parâmetros da estrutura de dados podem ser vistos no apêndice C.

4.4.1 Representação das classes e atributos

Uma forma adequada de apresentação de uma estrutura de objetos é através da representação de diagrama de classes de objetos utilizando a notação UML. A UML é uma linguagem usada para representar sistemas que utiliza representação gráfica. Embora tenha certa semelhança com orientação a objetos, a representação tem como objetivo disponibilizá-los em uma interface pública acessível a todos os federados. Dessa forma, na descrição e na representação, as classes herdam os atributos definidos em classes hierarquicamente superiores.

A Figura 4.6 e a Figura 4.7 representam graficamente a notação das tabelas de classes e atributos propostos. A classe *HLARoot* representa a raiz de todas os objetos e a classe **RecursodeSistemadePotência** é a raiz de todos os objetos segundo a norma IEC 61970. As classes na cor cinza claro representam uma agregação de subclasses. As classes que estão em cinza escuro são propostas de acréscimo ao modelo existente da norma IEC 61970.

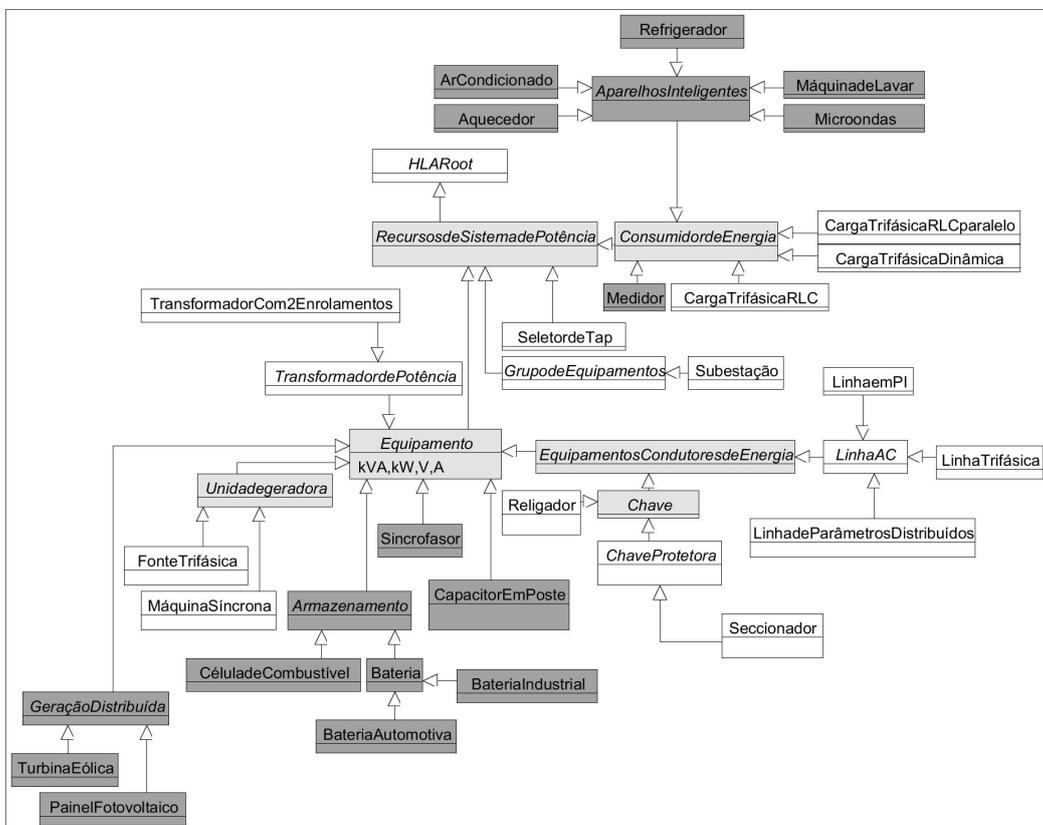


Figura 4.6: Diagrama de classes dos objetos

Na Figura 4.7, a classe *HLAInteractionRoot* representa a classe raiz ou a classe mãe de todas as classes de interação. Abaixo dela, seguem as classes abstratas **ControleDaSimulação** e **ProcessamentoDoAgente**. A primeira, representa uma classe de gerência responsável pelo início, fim ou parada das simulações e dividida em três subclasses associadas aos respectivos atributos lógicos. A segunda representa a classe de interações entre agentes. Subdivide-se em:

- a) **EnviaMsgFIPA** - O atributo é a mensagem a ser enviada entre os agentes através do RTI;
- b) **LeStatus** ou **AtribuiStatus** - Os atributos das classes permitem interações como abrir ou fechar elementos da rede elétrica.

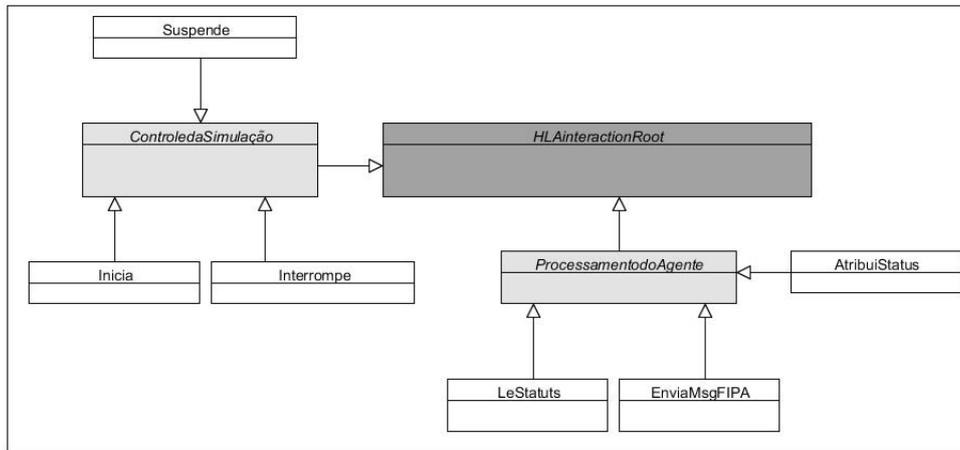


Figura 4.7: Diagrama de classes de interação dos objetos

4.5 Integração dos federados

Para demonstrar o desenvolvimento da cossimulação, utiliza-se a representação em diagramas da linguagem UML. A representação proporciona maior facilidade para compreensão do desenvolvimento da integração dos federados e permite uma forma de representação das classes de objetos e métodos utilizados nas interfaces dos sistemas com o RTI.

A UML é uma notação gráfica para especificar sistemas desenvolvidos com orientação a objetos e permite a exibição de diversas visões de um mesmo sistema. Essas visões dividem-se em dois grandes grupos:

- a) **Estruturais**: Tratam do aspecto estrutural do sistema e das classes que a compõe. Fornecem uma visão estática do sistema, ou seja, aspectos das estruturas que são fixas e abrangem classes, interfaces, colaborações,

componentes. Na versão UML 2.5, a visão estruturada é composta por doze diagramas.

- b) **Comportamentais:** Representam a dinâmica do sistema, ou seja, como o sistema age ou reage e o relacionamento com o ambiente. A visão é representada por onze diagramas de visualização.

São adotados os seguintes diagramas de visualização para representar o desenvolvimento realizado para integração do simuladores:

- a) **Diagrama de componentes (Estático):** Representa os aspectos físicos de um sistema. O componente é uma parte substituível e reutilizável do sistema;
- b) **Diagrama de classes (Estático):** Representa a estrutura das classes utilizadas, seus atributos e relacionamentos com outras classes e dependências. É o diagrama mais utilizado para representação de um sistema;
- c) **Diagrama de atividades (Comportamental):** Representa um fluxo de atividades executadas pelos programas e contém objetos, fluxos de controle e ações;
- d) **Diagrama de sequência (comportamental):** Representa uma sequência de mensagens trocadas entre objetos ao longo de uma linha de tempo. O decorrer do tempo é observado no diagrama no sentido vertical, de cima para baixo. As mensagens são simbolizadas por setas entre os objetos.

4.5.1 Integração com o simulador Matlab

Para o Simulink integrar-se ao RTI, é preciso configurá-lo a fim de que o mesmo possa carregar a biblioteca *portico.jar* [59], assim como informar o diretório de localização das classes que foram desenvolvidas para serem utilizadas. Essas classes são obrigatoriamente carregadas na inicialização do simulador.

Para se criar os mecanismos de integração com o RTI é utilizada uma função do simulador conhecida como “funções S”, capaz de estender as capacidades do ambiente com várias entradas e saídas. O bloco pode ser programado nas linguagens Matlab, C, C++, Fortran, além da chamada de métodos de classes de objetos na linguagem Java. A função utiliza uma sintaxe especial que permite a interação com o núcleo do ambiente. A função que é escrita em Matlab pode ter diferentes níveis de complexidade e de estrutura. Para ser reconhecida pelo simulador é necessário definir alguns parâmetros como número de entradas e saídas, tamanho das estruturas internas, estados das variáveis, entre outros.

A execução ocorre em etapas. A primeira é a inicialização de suas estruturas internas. A segunda, refere-se ao laço de execução que realiza o avanço de passo.

Durante essa fase, o Simulink executa cada bloco pertencente ao modelo chamando funções que calculam os estados das variáveis e as saídas em cada instante de tempo.

A função divide-se em cinco categorias entre as quais destaca-se a *Level-2* [71], cuja característica principal é o acesso à maior quantidade de interfaces sofisticadas e funcionalidades avançadas do sistema. O tipo *Level-2* permite criar blocos com múltiplas portas de entradas e saídas e capaz de tratar qualquer sinal produzido pelo Simulink. A função é constituída de um conjunto de métodos que acessam o núcleo do simulador que inicializam e calculam as saídas do bloco. As funções extraem os atributos do modelo e são repassados para os métodos, escritos na linguagem Java, que por sua vez, reenviam para RTI. O mesmo fluxo ocorre inversamente quando interações são enviadas para o Simulink.

4.5.1.1 Diagrama de classes

Foram criadas três funções escritas em *Level-2*:

a) **Portico** - Responsável pelas seguintes atividades:

- Inicialização do barramento RTI;
- Criação das classes de objetos;
- Controle do avanço de passo;
- Criação de atributos entre outras funções inerentes ao controle do barramento.

O módulo recebe como entrada o avanço de passo do simulador, que pode ser definido como fixo ou variável, dependendo do método numérico adotado. O programa se encarrega de ler o próprio arquivo de modelo, que contém informações que descrevem os sistemas, subsistemas, portas e linhas. O objetivo da leitura é identificar, através dos nomes dos elementos, aqueles que terão os seus atributos publicados para o RTI.

b) **EnviaAtributosParaRTI**: Função responsável por publicar os atributos de tensão e corrente de barra para o RTI. O parâmetro passado para esta função é o número de entradas.

A essa função conectam-se portas que recebem os atributos que são enviados para o RTI. A identificação ocorre através da associação do nome do atributo + “_RTI”, de tal forma que a função saberá identificar quais variáveis serão enviadas.

c) **RecebeInteração**: Função que recebe mensagem de interação através do RTI e consiste de apenas uma única saída que pode ser do tipo *boolean*, *float* ou outro qualquer definido no modelo de objetos.

A Figura 4.8 apresenta os modelos criados no ambiente Simulink, no qual as três funções descritas são utilizadas.

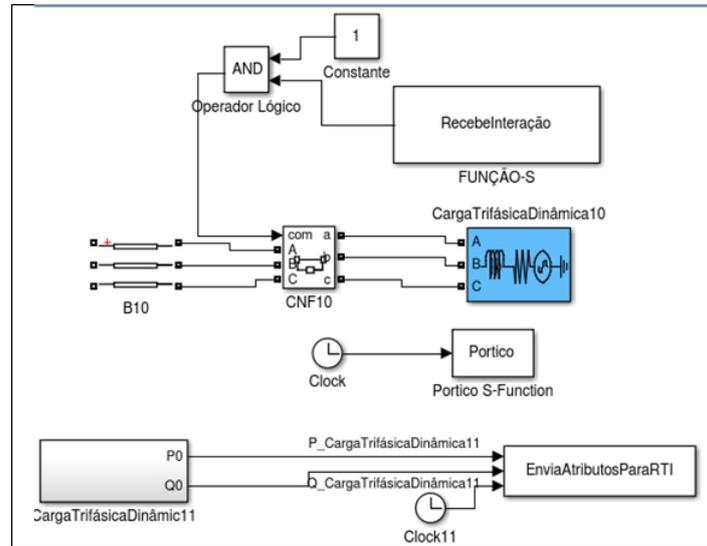


Figura 4.8: Visão parcial do modelo elétrico e blocos de integração RTI

O simulador comunica-se de duas formas com o RTI: através de atualização de atributos (método java **AtualizaValoresDeAtributos**) e através de interação (método java **RecebeInteração**). Os atributos correspondem àquelas que foram definidos no arquivo padrão de objetos e cada uma delas possui um tipo associado, que pode ser um *float*, *string*, *boolean*, ou outro qualquer que esteja declarado no modelo.

O método que recebe interação é responsável por receber e enviar os parâmetros pertencentes às classes de interação. Estas classes de objetos, como já foi relatado no capítulo 3, são enviadas na ordem cronológica.

A Figura 4.9 ilustra o diagrama das classes em Java e suas interdependências. As interfaces *RTIfederate* e *RTIambassador* foram estendidas nas classes *MatlabFederado* e *MatlabFederadoambassador*, respectivamente. Ambas dispõem dos métodos de inicialização e controle do RTI, assim como as chamadas de retorno para receber as interações dos agentes JADE.

A classe *ProcessaMDL* interpreta o arquivo do Simulink, que contém o modelo em estudo, para extrair os atributos que serão publicados para o barramento.

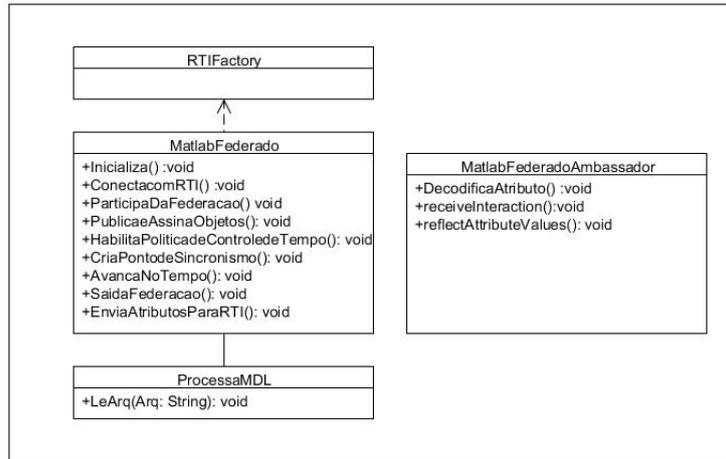


Figura 4.9: Diagrama de classes utilizadas pelo Matlab

4.5.1.2 Diagrama de atividades

O diagrama da Figura 4.10 ilustra atividades executadas para inicialização do RTI pelas classes relacionadas.

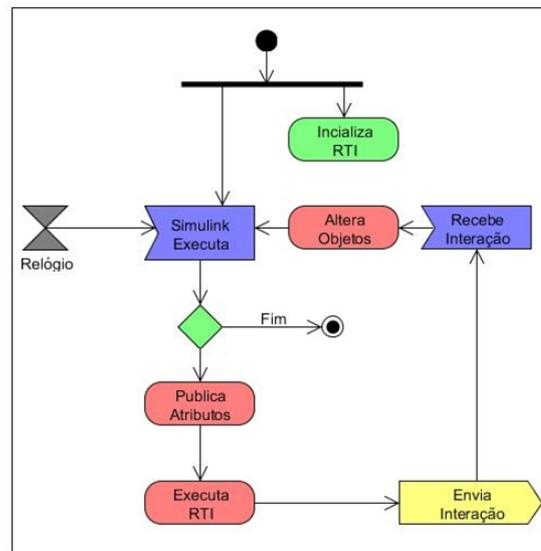


Figura 4.10: Diagrama de atividades executadas pelo Simulink

4.5.1.3 Diagrama de componentes

No diagrama da Figura 4.11 estão representadas as principais classes em Java que fazem interface com o RTI e as funções do Simulink responsáveis por chamar os métodos pertencentes às classes citadas.

O programa **Portico.m** é responsável por instanciar a classe **MatlabFederado** e a chamada de seus métodos enquanto que **RecebeInteracao.m** e

EnviaAtributosParaRTI.m são responsáveis por receber as interações de parâmetros e por enviar os atributos das classes de objetos publicadas respectivamente. O módulo recebe interações de parâmetros do tipo *boolean* e é capaz de controlar, por exemplo, abrir ou fechar chaves ou comandar qualquer outro componente eletrônico que faça parte do modelo elétrico estudado.

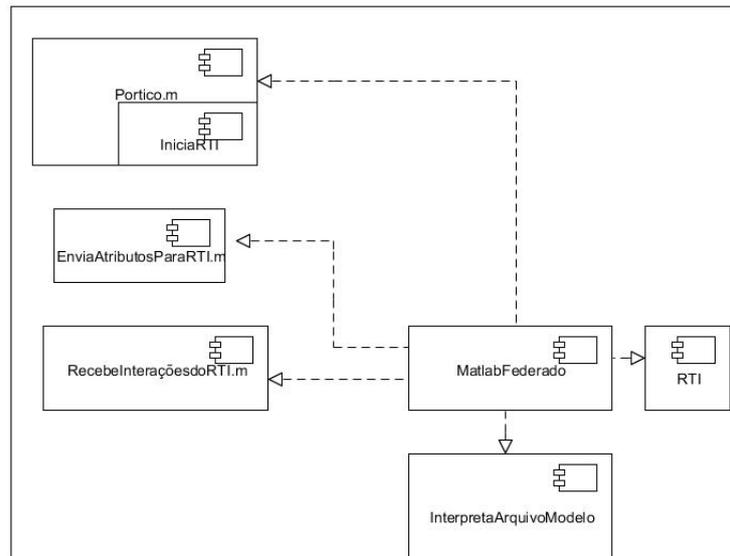


Figura 4.11: Diagrama de componentes utilizado para integrar com Simulink

4.5.2 Integração com sistema multiagente

Conforme descrito no apêndice B, cada agente é um sistema autônomo que se comunica através do *Manage Transport Protocol* (MTP). O MTP, definido pela FIPA, é uma camada de serviço na qual o protocolo HTTP é implementado e as mensagens são codificadas no padrão FIPA (vide Apêndice B). Do ponto de vista de simulação de redes de telecomunicações, o essencial é o tamanho das mensagens trocadas e a quantidade de requisições de comandos trocados entre os agentes. Entre os três disponíveis, o protocolo HTTP é o mais simples e leve de ser utilizado quando comparado como o HTTPS ou o IIOP e, por esse motivo, foi selecionado para implementação em cada agente.

Os agentes comunicam-se através do RTI através de um artifício transformando-se o conteúdo da mensagem HTTP em uma mensagem no formato XML cuja estrutura é ilustrada na Figura 4.12.

```

?xml version="1.0"<root>
<codigoFIPA>codigo</codigoFIPA>
<Remetente>Remetente</Remetente>
<Destinatario>Destinatario</Destinatario>
<Linguagem>Linguagem</Linguagem>
<Ontologia>Ontologia</Ontologia>
<Conteudo>Conteudo</Conteudo>
<Tamanho>Tamanho</Tamanho>
<Direcao>Direcao</Direcao>
</root>

```

Figura 4.12: Estrutura da mensagem XML

Os campos da mensagem XML, descritos na tabela 4.1 são utilizados pelo simulador Omnet++ para envio do pacote. O simulador utiliza apenas três campos: nó de origem, nó de destino e tamanho da mensagem.

Campo	Descrição
Código FIPA	Representa um código numérico que identifica o ato performativo (<i>INFORM</i> , <i>REQUEST</i> , <i>QUERY-IF</i> , etc)
Remetente	Nome do agente que está enviando a mensagem
Destinatario	Nome do agente receptor ou destinatário da mensagem
Linguagem	Tipo de linguagem adotada. Pode ser FIPA-SL, FIPA-KIF, FIPA-RDF. Como não se está adotando nenhum critério de uso de linguagem semântica, apenas atribuiu-se como FIPA
Ontologia	Representa a ontologia adotada
Conteudo	Representa o conteúdo da mensagem
Tamanho	Representa todo o tamanho da mensagem HTTP
Direção	Representa uma variável <i>booleana</i> . Se for verdadeira, é uma mensagem resposta enviada pelo agente; se for falsa, é uma mensagem recebida pelo agente. O sentido serve para informar aos agentes que, ao receberem uma mensagem do RTI, identifiquem o tipo de mensagem que será enviada

Tabela 4.1: Tabela de campos da mensagem XML

Duas classes de objetos foram desenvolvidas a fim de permitir o envio e o recebimento simultâneo destas mensagens. Essas classes são extensões do sistema multiagente e são responsáveis por traduzir as mensagens encaminhadas pelo

RTI em direção aos agentes e no sentido contrário. Para enviar para o RTI, a classe **gatewayRTI-Agente** é responsável pela extração das informações da mensagem XML e transformá-la em uma mensagem no formato FIPA. A classe **gatewayAgente-RTI** possui um conjunto de métodos que recebe a mensagem no formato de mensagem FIPA, extrai as informações necessárias e envia para o RTI na sintaxe XML. A Figura 4.13 ilustra a interface entre os agentes e RTI.

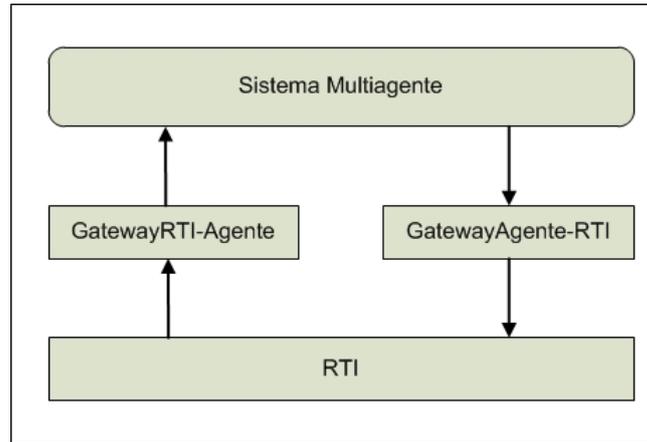


Figura 4.13: Diagrama de blocos das interfaces entre o RTI e os agentes

4.5.2.1 Diagrama de classe

O diagrama de classes, ilustrado na Figura 4.14, representa todas as classes e subclasses utilizadas para integrar o JADE ao RTI, composto pelas seguintes classes:

- a) **gatewayRTI-Agente**: É uma classe que estende a classe *agent*, que pertence ao JADE, e faz o papel de *gateway* entre o mundo dos agentes e o RTI;
- b) **DecodificaMsgdoRTI** : É uma classe que contém um conjunto de métodos chamada pelo agente **gatewayRTI-Agente** responsável por fazer a interpretação da mensagem no formato XML e enviar diretamente para o agente destino;
- c) **CodificaMsgparaRTI**: É uma classe que contém um conjunto de métodos chamados por todos os agentes da arquitetura e é responsável por converter a mensagem para a sintaxe XML;
- d) **gatewayAgente-RTI**: É uma classe que estende a classe *agent* do JADE e é responsável pelo envio da mensagem para o RTI na sintaxe XML;
- e) **PreparaMsg**: É uma classe instanciada por cada agente responsável por montar todo o cabeçalho HTTP que simula o envio da mensagem do agente para rede de telecomunicações. Uma vez que a mensagem é montada o

seu tamanho é calculado e colocado no campo “TAMANHO” da mensagem XML. Essa informação é utilizada pelo simulador Omnet++ para calcular a fragmentação dos pacotes que são enviados pelos nós;

- f) **EnviaParaAgente**: É uma classe instanciada pelo agente **gatewayRTI-Agente** que é responsável por coletar as mensagens enviadas pelo RTI para os agentes, inserir na fila de mensagens recebidas e sinalizar através de uma *flag*;
- g) **EnviaParaRTI**: É uma classe instanciada pelo agente **gatewayAgente-RTI** que é responsável por enviar as mensagens para o RTI que, por sua vez, envia para o simulador Omnet++;
- h) **RTI**: É uma classe *thread* composta de métodos que inicializam o RTI;
- i) **Agent**: É uma superclasse comum a todos agentes, disponibilizando métodos para executar tarefas entre as quais: envio de mensagens; suporte a todo o ciclo de vida de um agente, incluindo-se iniciar, suspender e finalizar; planejamento e execução de múltiplas atividades concorrentes;
- j) **EscreveMsgparaAgentes**: É uma classe que estende a classe *CyclicBehaviour* que por sua vez está definida dentro da classe **GatewayRTI-Agente**;
- k) **LeMensagem**: É uma classe que estende a classe *CyclicBehaviour*, responsável pelo recebimento das mensagens instanciada na classe **gatewayAgente-RTI**;
- l) **RTIAmbassador**: É uma classe pertencente à arquitetura HLA composta de vários métodos que respondem assincronamente às requisições dos federados. Os métodos representam um mecanismo para a federação notificar um federado que um determinado objeto está disponível para leitura;
- m) **Agentes**: Representa um conjunto de classes de agentes, cujos nomes são os mesmos dos componentes da rede elétrica (B1, B2, L1, L2, etc). Cada agente possui um ou mais comportamentos que executam a lógica planejada para cada um destes elementos através da troca de mensagens.

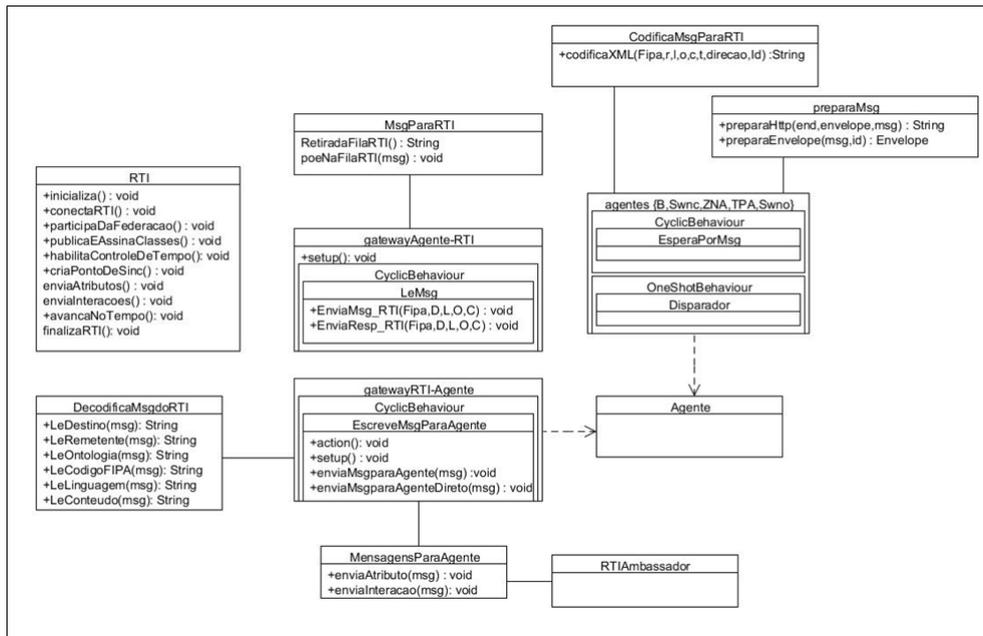


Figura 4.14: Diagrama de classes do sistema multiagente

4.5.2.2 Diagrama de seqüência

Para melhor retratar o fluxo de mensagens entre as classes e os agentes, utilizou-se a representação em diagrama de seqüência para demonstrar as seqüências lógicas do envio de mensagem e o recebimento por um agente, respectivamente, conforme ilustradas nas Figuras 4.15 e 4.16.

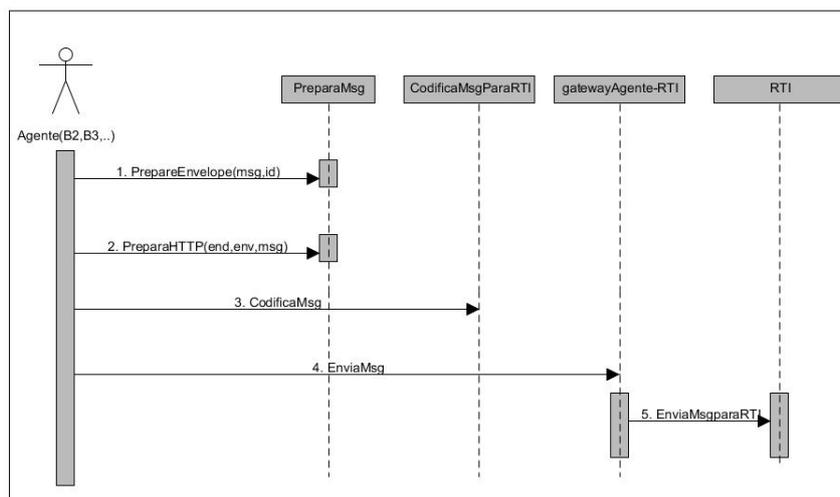


Figura 4.15: Diagrama de seqüência para envio de mensagem para um agente

O envio da mensagem do agente para o RTI requer a execução de uma seqüência que consiste nos seguintes passos:

- a) O Agente cria uma mensagem envelope cujo conteúdo segue a estrutura do padrão FIPA e que será encapsulada pelo protocolo HTTP;
- b) O Agente recebe a mensagem e a encapsula em uma mensagem HTTP POST. Esta mensagem é construída apenas para se determinar o real tamanho e repassá-lo para o simulador Omnet++;
- c) O Agente transforma a mensagem HTTP POST em uma mensagem cujo formato possui a sintaxe XML com informações do nó da rede de telecomunicações, o nó de destino e o tamanho total da mensagem;
- d) O Agente repassa a mensagem para o agente **gatewayAgente-RTI** que envia a mensagem XML para o RTI;
- e) O RTI notifica o simulador Omnet++ sobre a chegada de uma mensagem destinada a ele;

Quando a mensagem chega ao RTI o processo de recuperação ocorre ao inverso. A sequência para recuperação e envio da mensagem para o agente consiste nos seguintes passos:

- a) O RTI ao receber uma mensagem armazena em uma fila;
- b) O agente **GatewayRTI-Agente** monitora a fila e retira a mensagem. A mensagem que está codificada em XML é decodificada e, em sequência, envia a mensagem para o agente destino.

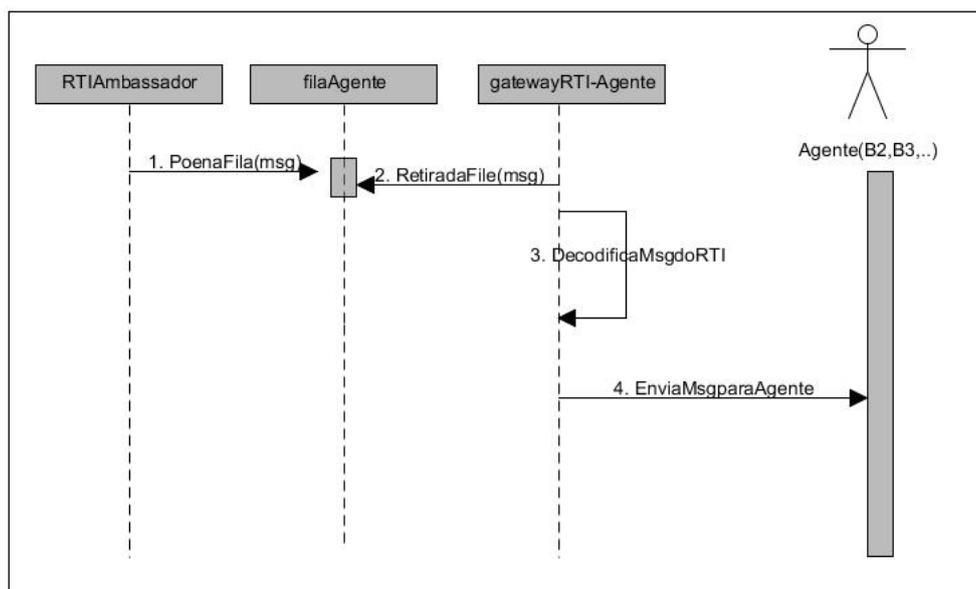


Figura 4.16: Diagrama de sequência de recebimento de mensagem por um agente

A Figura 4.17 representa o diagrama de componentes que compõem as classes utilizadas pelos agentes.

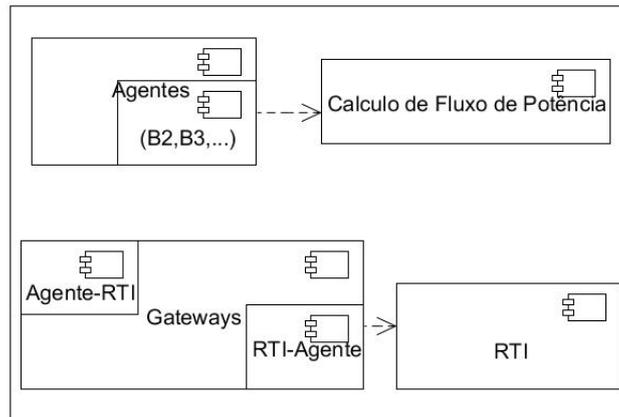


Figura 4.17: Diagrama de componentes do sistema multiagente

4.5.2.3 Diagrama de atividades

O diagrama de atividades, ilustrado na Figura 4.18, representa as ações executadas pelas classes. Dois fluxos são representados: o primeiro apresenta as ações para “iniciar” o barramento RTI e que são rigorosamente comuns a todos os federados. O segundo representa o fluxo executado pelas classes responsáveis por converter o cabeçalho HTTP da mensagem do agente em um formato de mensagem XML para o RTI e vice-versa. A ação “executa lógica de controle”, ilustrada na Figura 4.18, representa o comportamento do agente na execução de tarefas para tomadas de ações destinadas a alcançar um objetivo comum de todo o sistema, conforme descrito no apêndice B.

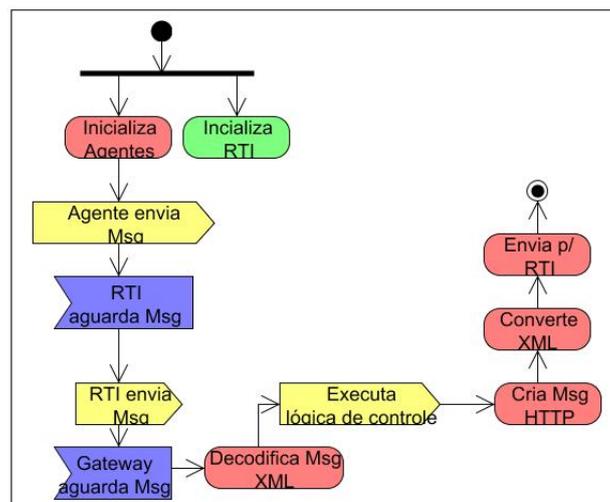


Figura 4.18: Diagrama de atividades do sistema multiagente

4.5.3 Integração com simulador Omnet++

O RTI não dispõe de recursos para tráfego de protocolo da camada OSI, mas apenas envio e recebimento de mensagens que representam os atributos ou interações entre os federados cujos tipos são *strings* e os pré-definidos pela norma IEEE 1516-2010. No modelo proposto, o simulador Omnet++ utiliza a pilha de protocolo de rede TCP/IP para simular a comunicação entre os diversos agentes. O simulador Omnet++ possui três modalidades de envio de mensagens:

- a) Envio da mensagem HTTP;
- b) Envio apenas do tamanho da mensagem, ou seja, o simulador não tem a necessidade de conhecer o conteúdo da mensagem da camada de aplicação. Essa abordagem é suficiente para o protocolo TCP fragmentar o pacote;
- c) Envio de uma estrutura de dados associada ao pacote que é passado de nó em nó até alcançar o nó de destino. Essa estrutura é construída para levar informações que possam ser relevantes para o estudo, como nome do nó de origem e destino, tamanho da mensagem.

Das três estratégias citadas, adotou-se a terceira. O nó de destino se encarrega de desanexar a estrutura e reenviá-la para o *gateway* do Omnet++, que por sua vez, reencaminha para o RTI.

4.5.3.1 Diagrama de classes

Para integrar o Omnet++ ao RTI é desenvolvida a classe **omnetGateway** que tem o papel de receber as mensagens enviadas pelo agente, decodificá-las e enviá-las para o Omnet++. O Omnet++ então decodifica a mensagem, abre uma conexão via TCP *socket* e envia para o nó de destino na rede de telecomunicações. Essa mensagem simula o comando POST, que significa uma requisição de envio de dados para o destino. Quando o pacote alcança o nó, a mensagem é recomposta para a sintaxe XML e enviada para o RTI. Simultaneamente, o nó de destino envia um código de *status* cujo conteúdo é “HTTP/1.1 200 Ok” para a origem. O RTI sinaliza que uma mensagem foi publicada e põe em uma fila de mensagens recebidas. O *gateway* dos agentes é responsável por retirar a mensagem da fila, decodificá-la e enviar a mensagem para o agente de destino.

Cada nó é composto por duas aplicações:

- a) **Tx**: Responsável por retirar a mensagem da fila e abrir uma solicitação de conexão com o nó de destino através de *socket* TCP. Após o estabelecimento da conexão, o tamanho do pacote é extraído da mensagem XML e é enviado em conjunto com uma estrutura de dados que inclui a própria mensagem XML.

- b) **Rx**: Responsável por receber a mensagem e reenviá-la para o *OmnetGateway*. Ao mesmo tempo, envia código resposta para o nó origem.

Após o estabelecimento da conexão, os pacotes são enviados pela rede de telecomunicações até o nó de destino e de lá retornam para o **gatewayOmnet**. Além das duas classes já citadas outras foram desenvolvidas e estão representadas no diagrama de classes da Figura 4.19.

- **Omnetfederado**: É a classe responsável pela inicialização do RTI assim como da publicação e assinatura das objetos, atributos e interações além de dispor de método de avanço de passo;
- **OmnetfederadoAmb**: É a classe constituída de chamadas de retorno e implementa métodos para receber interações e atributos;
- **cRTIScheduler**: É uma classe desenvolvida que implementa o método de atualização do avanço do tempo;
- **GatewayOmnet**: É uma classes desenvolvida que implementa a inicialização do RTI e se encarrega de receber as mensagens, decodificá-las e encaminhá-las para o nó da rede de telecomunicações. Quando o nó recebe o tamanho da mensagem HTTP e o nome do nó de destino, a aplicação **Tx** monta o quadro ethernet e envia o pacote.

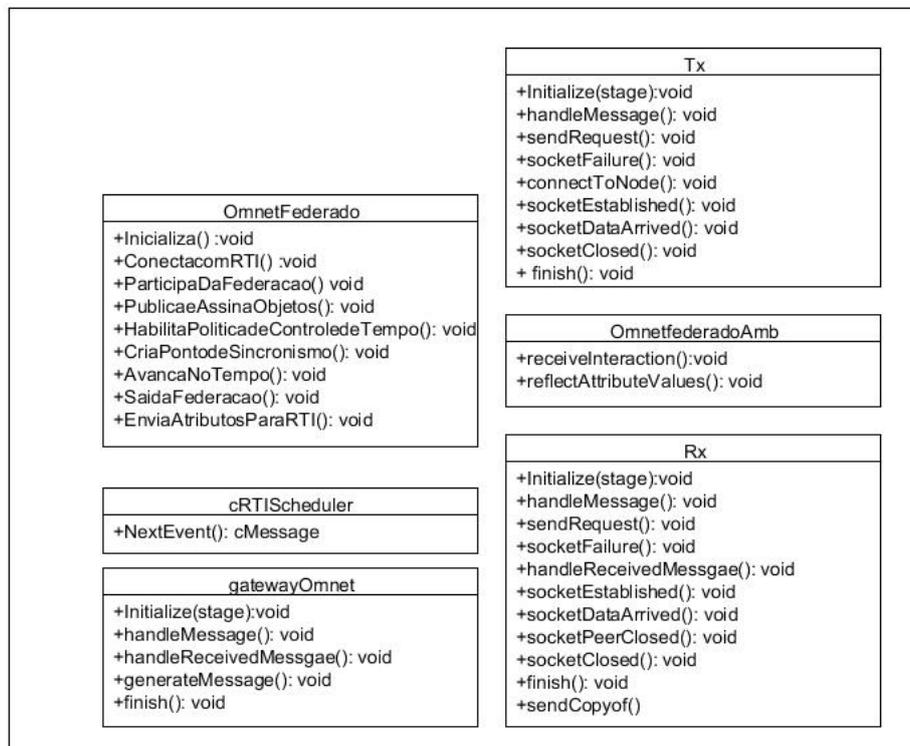


Figura 4.19: Diagrama de classes do Omnet++

O diagrama da Figura 4.20 ilustra a arquitetura integrada de todas as classes desenvolvidas para estabelecer a comunicação dos nós de rede do simulador e ao mesmo tempo uma visão dos fluxos entre os federados e o RTI.

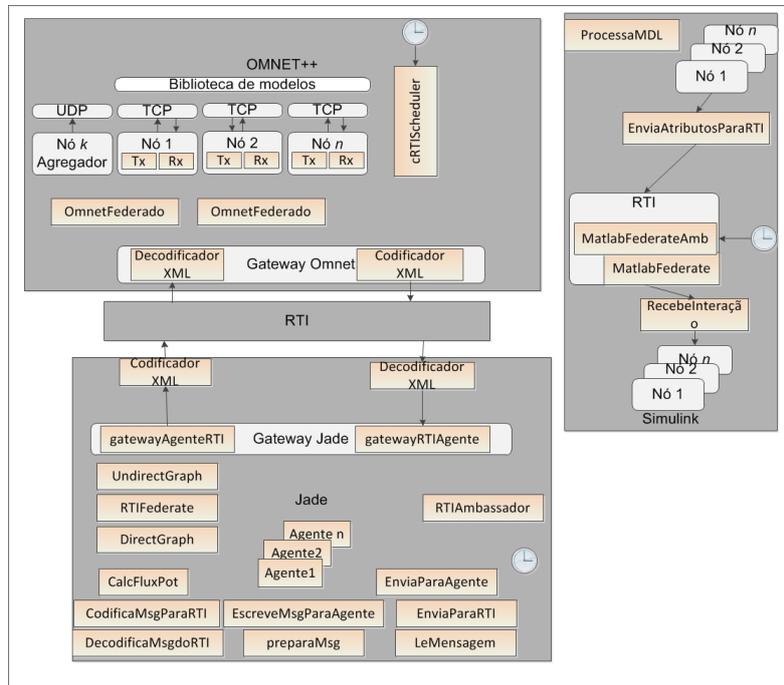


Figura 4.20: Diagrama de integração dos simuladores

4.5.3.2 Diagrama de atividades

O diagrama da Figura 4.21 representa as atividades executadas pelas classes criadas para o Omnet++.

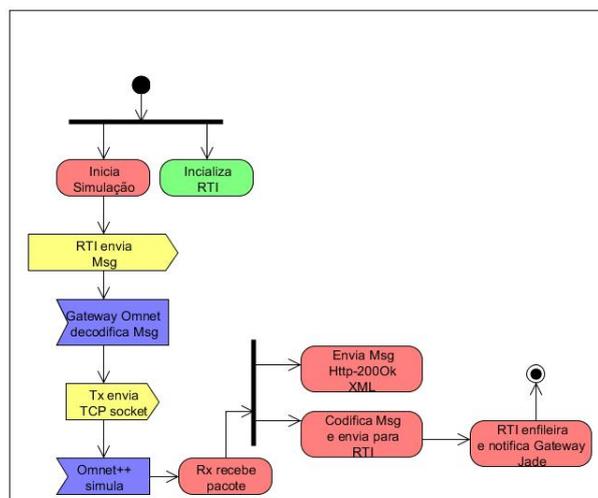


Figura 4.21: Diagrama de atividades das classes Omnet++

4.5.3.3 Diagrama de componentes

O diagrama ilustrado na Figura 4.22, é composto pelos componentes:

- a) Núcleo do simulador Omnet++;
- b) Arquivo de configuração omnetpp.ini e dos blocos que fazem parte do modelo, tais como Tx e Rx; Esses arquivos representam blocos com entradas e saídas para o simulador. O arquivo de topologia contém uma estrutura que contempla todos as descrições dos nós;
- c) Tx e Rx simulam a camada de aplicação do modelo OSI;
- d) O conjunto de serviços RTI;
- e) **OmnetFederado** responsável pela inicialização do federado;
- f) Arquivo de modelos que contém toda a estrutura de objetos da federação;
- g) Os métodos da classe **gatewayOmnet** se comunicam com os executáveis **Tx** e **Rx** para enviar e mensagens.

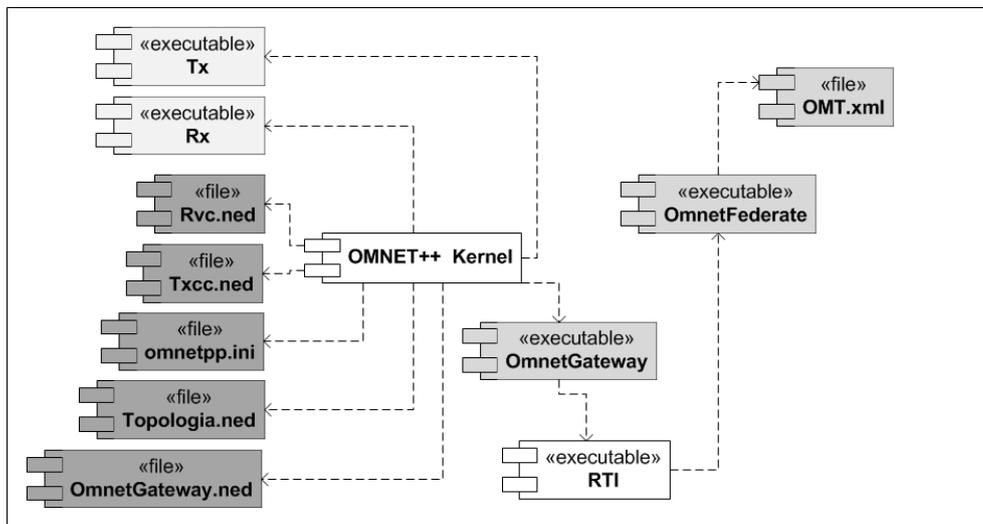


Figura 4.22: Diagrama de componentes das classes Omnet++

4.5.3.4 Sincronismo do Omnet++

O sincronismo é a peça chave para manter os eventos e os passos de cada simulador enquanto avança o tempo. O RTI dispõe de protocolos distintos para avançar o tempo lógico, seja para federados baseados em eventos discretos ou em intervalos de tempo fixo. Quando um federado executa a solicitação para o RTI, o mesmo se encarrega de calcular o próximo tempo lógico e liberar os simuladores para avançar

na execução. Como o Omnet++ é um simulador de eventos discretos e o Simulink baseado em avanço de passo, para compatibilizá-los utiliza-se a abordagem de avanço de tempo com um intervalo suficientemente pequeno. Para melhor esclarecer o funcionamento, o pseudocódigo1 ilustra o mecanismo de avanço de passo nessa condição.

Algoritmo 1: Pseudocódigo do avanço de passo

```

Avanço de passo: avanço de passo
while executa simulação do
  AVANCO_PENDENTE=true ;
  TimeAdvanceRequest(T+ lookahead) ;
  while AVANCO_PENDENTE do
    | Aguarde ;
  end
end

```

A variável T representa o tempo do simulador e o *lookahead* é o intervalo de tempo que o simulador pode avançar. O simulador dispõe de duas classes de escalonadores de eventos destinados a execução em tempo-real (*cRealTimeScheduler*) ou em tempo lógico (*cScheduler*). Para se estabelecer o sincronismo, cria-se uma subclasse da classe *cScheduler* e executa-se o algoritmo 2, implementado dentro do método *getNextEvent()*, cuja função é retornar o próximo evento da fila a ser processado.

Algoritmo 2: Pseudocódigo do sincronismo do Omnet++

```

Resultado: Próxima mensagem da fila
getNextEvent() while etiqueta de tempo da msg > Tempo da simulação do
  Avance o tempo de simulação(Intervalo) ;
  while AVANÇO=true do
    | Aguarde avanço de tempo ... ;
  end
end

```

4.6 Resumo

Foram apresentados os passos para criação de interfaces para cada um dos simuladores selecionados e para o sistema multiagente. A modelagem em UML norteou o desenvolvimento em linguagem java e C++. Foi apresentado um algoritmo geral para avanço de passo assim como a estratégia para realizá-lo no simulador Omnet++.

Capítulo 5

Estudos de Caso

5.1 Introdução

Entre as diversas aplicações de redes elétricas inteligentes, selecionou-se um cenário de reconfiguração de rede de distribuição de energia elétrica para aplicação do conceito de cossimulação.

As rede de distribuição de energia elétrica estão sujeitas a vários tipos de falhas, provocadas por tempestades, curto-circuito e eventuais acidentes. Nesses casos, a concessionária busca, ao receber uma reclamação, identificar a falha ocorrida, deslocar as equipes de manutenção, isolar o problema e restabelecer o fornecimento de energia para o maior número de clientes no menor espaço de tempo possível. A restauração da energia da rede elétrica de distribuição é um problema do tipo NP-completo [72], que trata de uma quantidade de estados discretos das chaves. O problema é formulado como multiobjetivo e com restrições. Todas as manobras objetivam atender às restrições de operação, entre as quais, destacam-se:

- Não ultrapassar os limites de corrente das linhas;
- Manter a radialidade da rede de distribuição;
- Manter os limites de queda de tensão nas linhas dentro dos limites;
- Efetuar o menor número de manobras;
- Manter a radialidade da rede após a manobra;
- Atender ao maior número de consumidores possível.

A falha da rede elétrica de distribuição representa um tipo de situação que tem sido amplamente estudado com várias abordagens de reconfiguração da rede de distribuição de energia elétrica, entre as quais programação matemática, meta-heurísticas, sistemas especialistas, algoritmos genéticos e sistemas multiagentes [73–75]. Algumas estratégias podem levar a um tempo de reconfiguração da rede energia

Linha	O	D	R(Ω)	XL(Ω)	P(kW)	Q(kVAr)	Comp.(m)
1	Gerador	2	0,0922	0,0477	100	60	500
2	2	3	0,493	0,2511	90	40	1000
3	3	4	0,366	0,1864	120	80	1000
4	4	5	0,3811	0,1941	60	30	1000
5	5	6	0,819	0,707	60	20	1000
6	6	7	0,1872	0,6188	200	100	1000
7	7	8	1,7114	1,2351	200	100	1000
8	8	9	1,03	0,74	60	20	1000
9	9	10	1,044	0,74	60	20	1000
10	10	11	0,1966	0,065	45	30	1000
11	11	12	0,3744	0,1238	60	35	1000
12	12	13	1,468	1,155	60	35	1000
13	13	14	0,5416	0,7129	120	80	1000
14	14	15	0,591	0,526	60	10	1000
15	15	16	0,7463	0,545	60	20	1000
16	16	17	1,289	1,721	60	20	1000
17	17	18	0,732	0,574	90	40	1000
18	2	19	0,164	0,1565	90	40	900
19	19	20	1,5042	1,3554	90	40	1000
20	20	21	0,4095	0,4784	90	40	1000
21	21	22	0,7089	0,9373	90	40	1000
22	3	23	0,4512	0,3083	90	50	2000
23	23	24	0,898	0,7091	420	200	2000
24	24	25	0,896	0,7011	420	200	2000
25	6	26	0,203	0,1034	60	25	1000
26	26	27	0,2842	0,1447	60	25	1000
27	27	28	1,059	0,9337	60	20	1000
28	28	29	0,8042	0,7006	120	70	1000
29	29	30	0,5075	0,2585	200	600	1000
30	30	31	0,9744	0,963	150	70	1000
31	31	32	0,3105	0,3619	210	100	1000
32	32	33	0,341	0,5302	60	40	1000
33*	21	8	2	2	0	0	3750
34*	9	15	2	2	0	0	6000
35*	12	22	2	2	0	0	6500
36*	18	33	0,5	0,5	0	0	2250
37*	25	29	0,5	0,5	0	0	2250

Tabela 5.1: Tabela de parâmetros do modelo IEEE 33 barras

de *proxy* é o que está associado à cada barra da rede de distribuição, ilustrado na Figura 5.2. Cada bloco representa um agente associado a um componente da rede de distribuição e desempenha um papel dentro da arquitetura. O sistema é distribuído por 32 agentes associados às barras (a barra 1 representa a fonte geradora e não possui agente), um agente orquestrador (**AO**), um agente gerador de topologias (**ATP**), 37 agentes associados às chaves, sendo 32 normalmente fechadas (**CNF**) e 5 normalmente abertas (**CNA**), e 32 agentes associados às cargas (**L_i**). A topologia pode ser facilmente escalável bastando replicar cada um dos tipos de agentes.

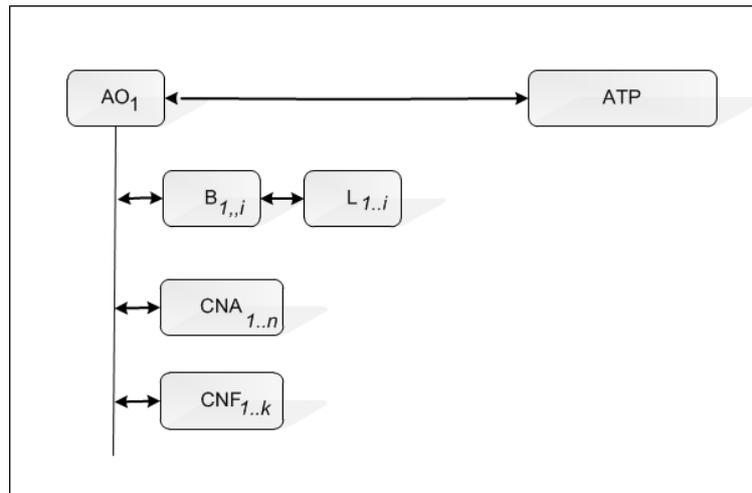


Figura 5.2: Topologia dos agentes

O agente da carga **L_i** é responsável pela coleta de dados das cargas cujos valores P(kW) e Q(kVAr) são enviados pelo Simulink através do RTI. O algoritmo considera que todas as cargas são classificadas como essenciais, isto é, não há cargas mais prioritárias na rede.

O agente associado à barra **B_i** é responsável por gerenciar a ampacidade e requisitar todos os valores de todas as cargas pertencentes às quais estão conectadas. O agente orquestrador **AO₁** é responsável por gerenciar e agrupar todos os dados recebidos de todos os barramentos que estão sob a sua responsabilidade, implementa a heurística principal de controle e interage com os que estão abaixo na hierarquia e acima também. O agente **ATP** é responsável por gerar o resultado do cálculo do fluxo de potência e por fornecer ao agente orquestrador transformações em subgrafos, assim como determinar as suas respectivas folhas que serão utilizadas pelo agente **AO₁** para análise de corte da carga. As chaves normalmente abertas e fechadas (**CNA_n** e **CNF_k**) têm o papel de interagir com o agente **AO₁** para receber comandos de abrir ou fechar e informar o seu *status*.

Os agentes se comunicam trocando mensagens por intermédio do RTI através da execução de chamada de envio de mensagem. A mensagem é enviada primeiramente para o simulador de rede de telecomunicações. O pacote é tratado

na interface entre o RTI e o simulador. O pacote trafega pela rede e, se não houver perda ou necessidade de retransmissão, chega ao seu destino. O nó recebe a mensagem, envia uma informação de resposta (HTTP 200 Ok) para a origem e, simultaneamente, devolve a mensagem para o RTI, codificando-a. Uma vez que a mensagem é recebida, o RTI notifica o agente que faz o papel de *gateway* com o sistema multiagente. Esse agente recebe a mensagem, decodifica e encaminha a mensagem para seu destino. O fluxo de troca das mensagens está ilustrado na Figura 5.3

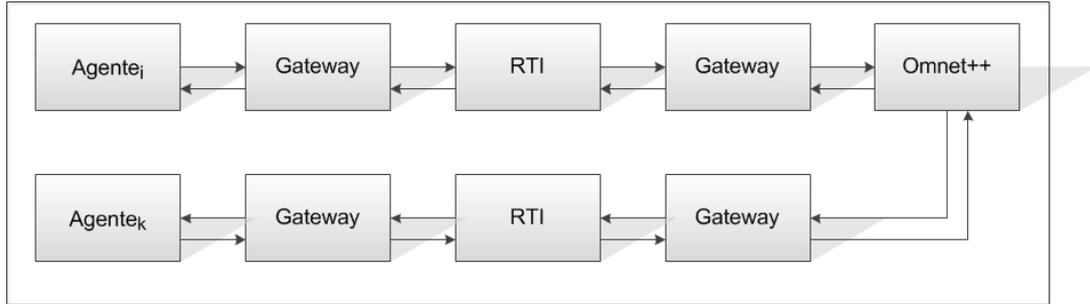


Figura 5.3: Fluxo das mensagens trocadas entre agentes

A abordagem da restauração da energia é tratada como um problema multi-objetivo com restrições descritas pelas equações (5.1) e (5.2).

$$\text{Min } Sw_n \quad (5.1)$$

$$\text{Max } \sum_{j \in L} S_j \quad (5.2)$$

Sujeitas a:

$$|I_k| \leq |I_k^{max}|, \quad k \in L \quad (5.3)$$

$$|V_k^{min}| \leq |V_k| \leq |V_k^{max}|, \quad k \in L \quad (5.4)$$

com as seguintes restrições:

- a) Sw_n representa o número de operações de abertura e fechamento das chaves;
- b) L representa o conjunto de cargas da solução encontrada;
- c) $|I_k^{max}|$ é a corrente máxima do barramento;
- d) $|V_k|$ é a tensão no barramento k ;
- e) S_j é a carga total associada ao j -ésimo barramento, medida em kVA;

Os mecanismos de restrição e as equações são implementados no agente **AO₁**, responsável pela tomada de decisão sobre a seleção das chaves que serão abertas

e fechadas. Dado que os agentes são elementos autônomos e com papéis e responsabilidades, é proposto um algoritmo distribuído, ilustrado no pseudocódigo 3 no qual estão representadas as ações tomadas para se chegar a uma solução.

Uma abordagem utilizada para se determinar os possíveis caminhos entre o segmento que foi isolado e a fonte de energia é através da representação da rede de distribuição na forma de um grafo, no qual as chaves são representadas pelas arestas e os trechos entre chaves representados pelos nós. O agente **ATP** quando solicitado, cria dois tipos de grafos: não direcionado, para a descoberta de caminhos que reconectem o segmento à jusante do barramento isolado pela falha; grafo direcionado, que lista os ramos da rede a fim de auxiliar o agente AO_1 a calcular a corrente em cada barra e comparar com os respectivos limites. Os caminhos descobertos são obtidos através do algoritmo de busca em profundidade (DFS) [78] e possuem apenas uma única chave normalmente aberta. Portanto, apenas uma parte do espaço de soluções é determinado. Através do grafo direcionado determina-se também as suas folhas, necessárias para que o agente AO_1 analise os valores das cargas e determine o corte delas ou não atendendo-se às restrições impostas.

As tensões nas barras são obtidas através do cálculo do fluxo de potência a cada solução de topologia encontrada. O resultado é uma lista de tensões por unidade (p.u) e ângulo dos fasores das barras. Se todas as linhas estiverem com os valores de tensão entre $0.95p.u \leq V \leq 1.05p.u$ então as configurações de chaves abertas e fechadas são aceitas como soluções encontrada, desde que todas as outras restrições tenham sido atendidas.

As correntes nas barras, limitadas pela ampacidade de cada cabo são calculadas a cada configuração encontrada, mesmo quando as cargas ou as folhas do grafo são cortadas. Pelo grafo, são obtidos os caminhos desde a extremidade do ramo até a fonte geradora e a corrente em cada segmento da rede é calculada e comparada com o limite máximo. Se esse limite é ultrapassado, então aquela configuração deve ser descartada pois não atenderá à restrição imposta na equação (5.3).

O agente AO_1 impõe o corte de cargas quando nenhum caminho mínimo é encontrado. O agente reinicializa a lista de caminhos possíveis e solicita ao agente **ATP** a lista de folhas do grafo. De posse dos dados, o algoritmo ordena e remove o de menor potência aparente, objetivando atender ao maior número de consumidores. Após o corte, o agente solicita nova determinação de cálculo de fluxo de potência. Se as tensões se mantiverem fora da faixa, o agente decide pelo segundo caminho mais próximo e nova rodada de cálculos é elaborada até que todas as restrições tenham sido atendidas. A Figura 5.4 oferece um outro tipo de visão sobre o macrofluxo das ações tomadas pelos agentes.

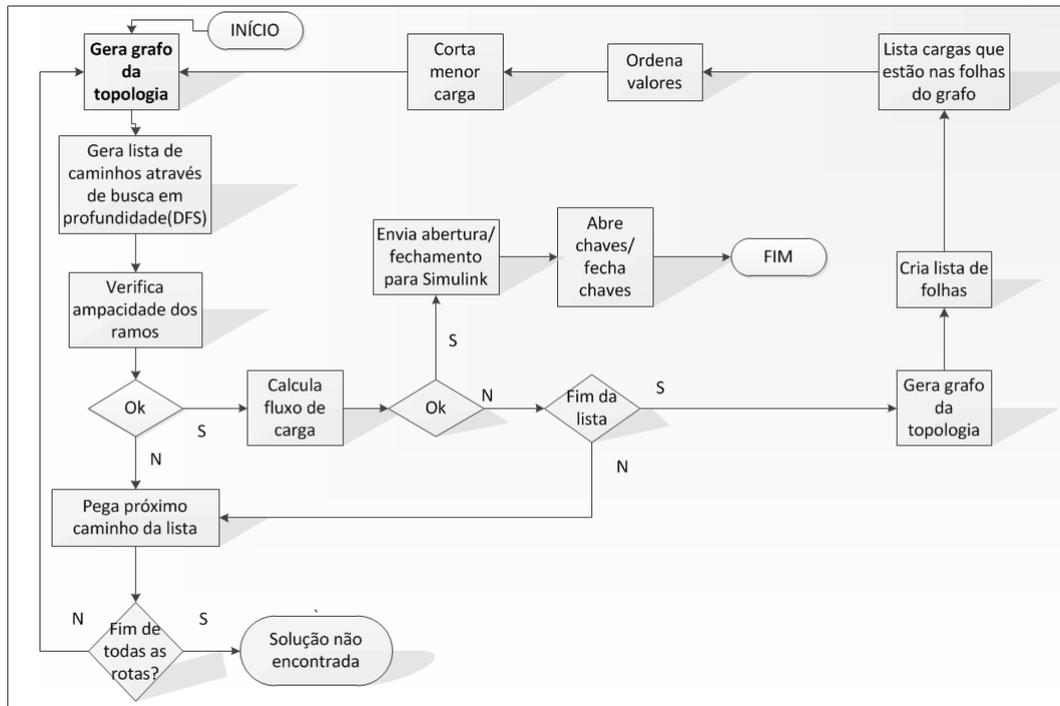


Figura 5.4: Macrofluxo do algoritmo distribuído

No cenário avaliado, assume-se que o barramento B_5 sofreu um curto circuito. As chaves CNA_4 e CNA_5 abrem, isolando o segmento assim como as barras B_6 a B_{18} . O agente B_5 recebe as notificações das chaves conectadas, reconhece a falha e envia requisição ao agente orquestrador para restauração da energia. O algoritmo converge para a seguinte solução:

- *Fechar Chaves* $\rightarrow CNA_{35}$
- *Abrir Chaves* $\rightarrow CNF_{12} - CNF_{17}, CNF_{22} - CNF_{24}, CNF_{27} - CNF_{32}$

5.3 Validação do pseudocódigo

O algoritmo proposto foi validado sobre uma rede hipotética constituída de máquinas que simulam o comportamento dos agentes e a pilha de protocolos TCP/IP e interconectadas através da *switch* nível 2, conforme ilustrado na Figura 5.5. Nessa topologia, as máquinas são interligadas por enlaces ethernet.

Algoritmo 3: Pseudocódigo dos agentes

Resultado: Lista de chaves abertas e fechadas

Carrega topologia;

B_{falha} solicita recuperar energia \leftarrow AO₁ ;

AO₁ informa $B_{falha} \rightarrow$ ATP ;

while algoritmo não converge **do**

 ATP cria grafo sem B_{falha} ;

 ATP informa ramos sem energia \rightarrow AO₁ ;

 AO₁ solicita caminhos possíveis para recuperar energia \rightarrow ATP ;

 ATP calcula rotas possíveis ;

 ATP informa rotas a jusante da falha \rightarrow AO₁ ;

for $k \leftarrow 1$ **to** N° de Barras **do**

 AO₁ $\rightarrow B_k \rightarrow L_k$;

 Barra _{k} \rightarrow AO₁ ;

end

if Todas cargas recebidas **then** **for** $k \leftarrow 1$ **to** No. de Barras **do**

 AO₁ solicita ampacidade \rightarrow Barra _{n} ;

end

else Espere pelo dado;

 AO₁ solicita lista de ramos da rota \rightarrow ATP;

 ATP envia lista de ramos \rightarrow AO₁ ;

 AO₁ calcula ampacidade dos ramos ;

if ampacidade excedida **then**

 Passa para próxima rota ;

else

 AO₁ solicita cálculo de fluxo de carga \rightarrow ATP;

end

 ATP solicita valor de todas as cargas $L_k \rightarrow$ AO₁ ;

 ATP solicita rota a ser analisada \rightarrow AO₁ ;

if $0.95p.u \leq$ limites de tensão $\leq 1.05p.u$ **then**

 ATP envia msg que passou \rightarrow AO₁;

 AO₁ solicita abertura e fechamento de chaves ;

FIM do algoritmo ;

else

 ATP envia msg que não passou \rightarrow AO₁ ;

 AO₁ passa para próxima rota ;

if AO₁::Todas rotas utilizadas **then**

 AO₁ solicita corte de carga \rightarrow ATP;

 ATP propõe lista de folhas do grafo \rightarrow AO₁;

 AO₁ ordena carga por menor valor ;

 AO₁ solicita fluxo de potência com carga cortada \rightarrow ATP;

else

 AO₁ passa para a próxima rota ;

end

end

end

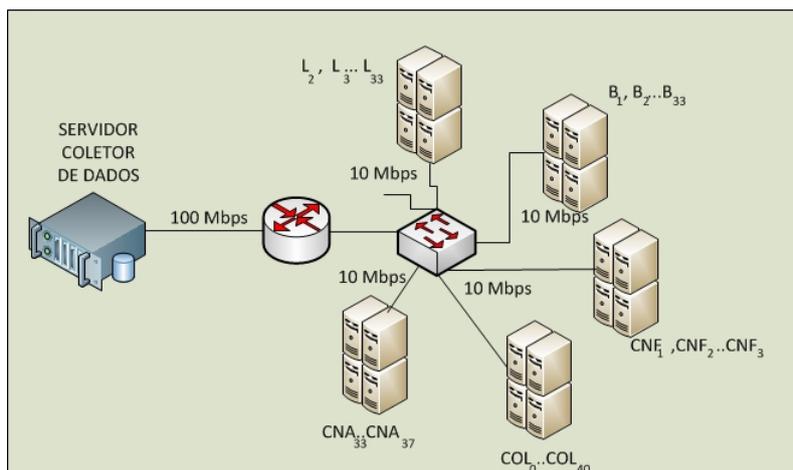


Figura 5.5: Topologia da rede de telecomunicações para teste do algoritmo

O cenário foi executado 100 vezes com sementes distintas. O percentual de conclusão é de 100% com tempo médio de reconfiguração da rede de distribuição em 4 segundos. Dessa forma, verifica-se que a aplicação da reconfiguração da rede de distribuição de energia elétrica consegue passar por todas as camadas de comunicação do modelo OSI, simuladas no Omnet++ e receber dados do simulador Simulink através do RTI.

5.4 Cenário de avaliação

Em função dos parâmetros de comprimento de cabos utilizados na Tabela 5.1, a rede de distribuição apresenta uma área aproximada de 40 km² para acomodar a distribuição física. Foi feito um levantamento para identificar uma região do Estado do Rio de Janeiro que tivesse as características físicas da dimensão aproximada do problema. Para esse estudo, adota-se como premissa que a localização da rede de distribuição é em uma região do subúrbio do Estado. Por definição uma área suburbana é composta por muitas residências, com prédios baixos, com pouca vegetação e áreas abertas. A partir dos dados publicados pela GeoRio [79], uma fundação da Secretaria de Obras do Estado do Rio de Janeiro, é possível obter informações sócio-econômicas das regiões administrativas da cidade do Rio de Janeiro. Entre as várias informações disponíveis destaca-se a quantidade de domicílios particulares permanentes que possuem medidores de energia elétrica. O bairro selecionado para estudo que mais se aproxima desse conceito e da área estimada é a Ilha do Governador, que apresenta as características sócio-econômicas para o ano de 2010, ilustradas na Tabela 5.2. Essas características são úteis para determinar a quantidade de coletores de dados que agregam tráfego originado dos medidores inteligentes.

Localidade	Ilha do Governador
Área(km ²)	40,8
Habitantes	212.574
Densidade populacional(ha/km ²)	5598,2
Número de domicílios	71.743
Número de domicílios com medidor	69.574
Número de prédios	26.688
% de apartamentos	38,4
Número de estabelecimento de indústria/comércio/serviço	6072

Tabela 5.2: Tabela de aspectos sócio-econômicos da região da Ilha do Governador

São propostas três topologias de redes de telecomunicações sem-fio: rede em malha, rede ponto-multi-ponto e LTE. A adoção de redes sem-fio é uma estratégia adequada para a última milha, pois proporciona maior cobertura de consumidores e sensores e maior flexibilidade na expansão para atendimento à demanda. Os cenários são avaliados de duas formas: a primeira, apenas com o tráfego gerado pelo algoritmo de reconfiguração da rede de distribuição de energia elétrica; a segunda, acrescentando-se um tráfego concorrente e avaliando o possível impacto ocasionado adotando-se duas métricas para avaliação da cossimulação.

O segundo tráfego, que roda de forma concorrente, baseia-se no uso dos medidores inteligentes, que coletam dados do consumidor e enviam para concessionária. A função básica é a medição da potência consumida, porém, futuramente, com o avanço das aplicações, outros dados poderão ser enviados, como fator de potência, dados de consumo dos aparelhos eletrodomésticos e gestão da demanda de equipamentos. A topologia proposta é constituída de medidores inteligentes que enviam dados para os coletores de tráfego que estão fixados no topo dos postes a uma altura de 10 metros que por sua vez repassam para a rede de telecomunicações, conforme ilustrado na Figura 5.6.

Assume-se que os enlaces de comunicação entre os medidores e coletores e entre coletores e a rede em estudo não sofrem interferência ou por operarem em canais, de frequências diferentes, ou por utilizar outra tecnologia de interconexão. Não é escopo do estudo qualquer avaliação de tráfego entre os medidores e o agregador. O tráfego total da rede considerado será a soma do próprio tráfego de automação agregado com o tráfego dos medidores inteligentes.

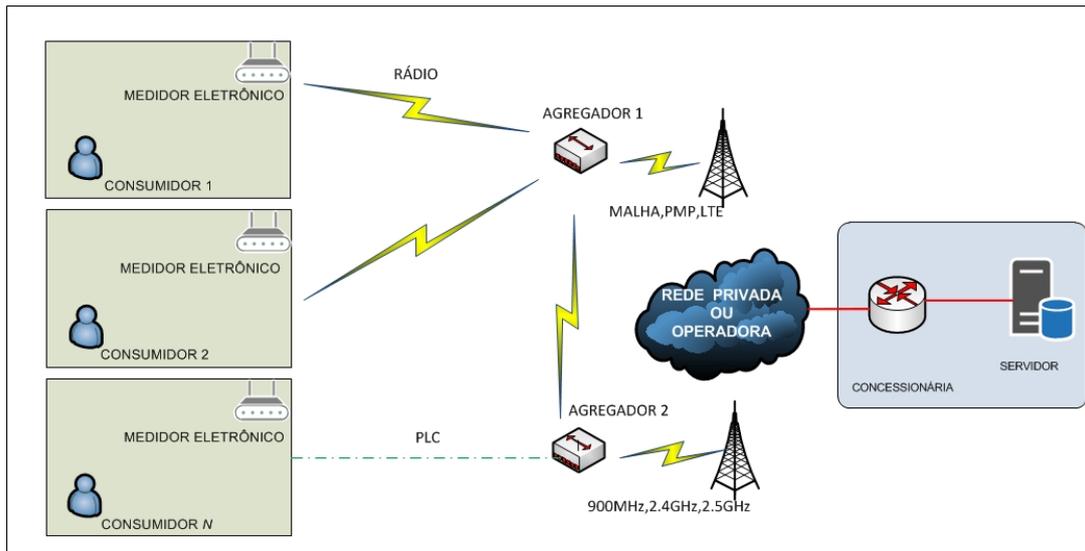


Figura 5.6: Topologia dos coletores de tráfego dos medidores inteligentes

Em termos práticos, o tamanho do pacote enviado por um medidor inteligente pode variar pois depende da solução adotada pelo fabricante. Portanto, foram definidos quatro tamanhos para análise, baseando-se nos artigos [80–82]: 50, 100, 200 e 300 bytes. Definiu-se que o envio dos dados de medição é realizado a cada intervalo de 5, 10, 15 e 30 minutos para avaliar se o volume de tráfego gerado exerce sobre a automação dos religadores. Adotou-se o valor de densidade de medidores por km^2 como parâmetro para cálculo do número de concentradores necessários para atender a toda região. Dividindo-se a soma dos estabelecimentos e das residências pela área total chega-se ao valor de 1854 medidores/ km^2 . O valor está muito próximo do parâmetro adotado pelo estudo realizado pelo órgão normatizador NIST [83], que estabelece uma densidade de 2000 medidores/ km^2 . Assumindo-se como premissa que um agregador possa atender uma área de 1 km^2 , então obtém-se a quantidade de 41 coletores de tráfego para cobrir toda a região estudada.

Cada agregador irá gerar a quantidade de tráfego, em bytes, ao longo do dia, representada na Tabela 5.3. Os valores correspondem à coleta de 1854 medidores inteligentes realizada ao longo de 24 horas.

Intervalo das mens. (min)	50 bytes	100 bytes	200 bytes	300 bytes
5	26.698.588	53.395.200	106.790.400	160.185.600
10	13.349.294	26.697.600	53.395.200	80.092.800
15	8.899.529	17.798.400	35.596.800	53.395.200
30	4.449.765	8.899.200	17.798.400	26.697.600

Tabela 5.3: Tabela de volume de tráfego gerado pelos coletores em bytes

Esse volume é transformado em uma fonte de tráfego constante e utilizado

pelo agregador que envia dados para um servidor de banco de dados localizado na concessionária. As taxas, para cada tamanho de pacote, são ilustradas na Tabela 5.4.

Intervalo das mensagens (min)	Tamanho do pacote(Bytes)			
	50	100	200	300
5	309	618	1236	1854
10	155	309	618	927
15	103	206	412	618
30	52	103	206	309

Tabela 5.4: Tabela de taxa de envio de mensagem em bytes/seg

Os coletores de dados estão dispostos fisicamente na topologia das redes ponto-multiponto e LTE de forma aleatória, porém dentro do raio de cobertura das respectivas estações rádio-bases. No caso da topologia de rede em malha, os nós coletores são também nós de repetição que compõem a malha.

As fontes geradoras de tráfego dos medidores e coletores utilizam o protocolo UDP. Cabe ressaltar que, em todos os cenários estudados, os agentes utilizam o protocolo TCP Reno, que utiliza o mecanismo de *fast recovery* para reenvio de pacotes em caso de não recebimento de confirmação (ACK). Quando um pacote é perdido ou o receptor envia três ACK's duplicados, o TCP inicia esse mecanismo reduzindo a janela de congestionamento à metade. Quando o tempo de retransmissão expira, o TCP implementado no Omnet++ chama um método da aplicação do usuário que é responsável por dar o tratamento adequado. No caso em estudo, quando isso acontece, o método inicia uma nova conexão com o nó de destino e reinicia o processo de transmissão da mensagem.

5.4.1 Capacidade das redes em malha e ponto-multiponto

As redes sem-fio oferecem um desafio que é a determinação do dimensionamento da capacidade de tráfego. Estudos apresentam cálculo teóricos para determinar a capacidade nominal de redes *ad-hoc* [84, 85] adotando-se um modelo determinístico no qual a transmissão de um nó é bem sucedida baseando-se em um modelo teórico cuja área de alcance seja determinada por um círculo de área unitária. Para uma rede cujos nós são aleatoriamente distribuídos, a capacidade de transmissão do nó, assumindo que não haja colisão, é expressa pela equação (5.5), dado que R é taxa do canal em bits/seg e n representa o número de nós da rede.

$$\Theta \left(\frac{R}{\sqrt{n \log n}} \right) \text{ [bits/seg]} \quad (5.5)$$

Os autores ainda esclarecem que quando uma rede em malha possui m nós com o papel de apenas repetidores o resultado é alterado para a equação (5.6).

$$\Theta \left(\frac{R(n+m)}{n\sqrt{(n+m)}\log(n+m)} \right) \text{ [bits/seg]} \quad (5.6)$$

A abordagem não leva em consideração aspectos físicos relacionados à colisão de pacotes, que contribuem para redução significativa da capacidade nem a interferência causada por nós localizados dentro da área de cobertura. Na prática, a capacidade é também afetada pelo número de saltos até o seu destino, pela atenuação por propagação e os fenômenos físicos como desvanecimento por multipercuro.

Em [86], os autores apresentam uma metodologia para calcular a capacidade máxima de uma rede em malha, baseando-se no conceito de domínio de colisão. Este conceito é definido como sendo a representação de um grupo de enlaces cujo enlace k , para poder transmitir com sucesso, deve ter todos os outros enlaces inativos. Nesse mesmo artigo, é demonstrado que a rede em malha, com uma topologia arbitrária a capacidade assintótica de cada nó é expressa por (5.7), assumindo-se as seguintes premissas:

- Existência de apenas um *gateway* para rede externa;
- Existência de esquema de justiça para todos os nós, isto é todos os fluxos de dados são tratados de forma igualitária;
- O tráfego é unidirecional dos nós para o *gateway*.

$$O \left(\frac{1}{n} \right) \quad (5.7)$$

em que n é o número de nós da rede. Todas as três redes propostas na prova de conceito contém dois tipos fluxos de tráfego: restrito à rede, isto é um fluxo inicia e termina dentro da rede que corresponde ao tráfego da reconfiguração das chaves e outro que é destinado à rede externa, correspondente ao tráfego dos medidores inteligentes.

5.5 Métricas de avaliação

Propõem-se duas métricas para avaliação dos três cenários propostos para a prova de conceito. A primeira delas é a determinação do tempo médio de reconfiguração da rede de distribuição de energia elétrica. Como o tempo é uma variável desconhecida, adotou-se o cálculo do intervalo de confiança para uma probabilidade $(1-\alpha)$, em que α é o nível de significância desejado. Esse intervalo informará com probabilidade de

$(1 - \alpha)$ o intervalo $t_1 < t < t_2$ em que o valor médio será encontrado. O intervalo de confiança é dado pela equação 5.8,

$$P \left(\bar{X} - t_{\alpha} \frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{X} + t_{\alpha} \frac{\sigma}{\sqrt{n}} \right) = (1 - \alpha) \quad (5.8)$$

em que P é a probabilidade, \bar{X} é a média amostral, μ é a média populacional e σ é o desvio-padrão da média amostral. Em todos os cenários avaliados o nível de significância utilizado foi de 0.05. Logo, para um intervalo de confiança P de 95% o tempo médio de reconfiguração estará entre $[\bar{X} - 1.96\sigma\sqrt{n}; \bar{X} + 1.96\sigma\sqrt{n}]$. Duas condições de tráfego de dados são avaliadas. O primeiro cenário prevê apenas o tráfego de dados da aplicação que reconfigura as chaves. O segundo cenário prevê que, além do tráfego de reconfiguração das chaves, um segundo tráfego composto de diversas fontes de tráfego dos medidores inteligentes é inserido. Não há filas de prioridades que separem esses dois tráfegos e ambos compartilham a mesma fila de acesso ao meio físico. Nessa condição, é avaliado se a variação do tráfego resulta em aumento do tempo médio de reconfiguração da rede de distribuição de energia elétrica.

A segunda métrica refere-se ao percentual de finalização das simulações sob as mesmas duas condições de tráfego citadas. Define-se como percentual de finalização o intervalo de tempo, inferior a 1000 segundos, que o algoritmo de reconfiguração consegue definir as chaves que serão abertas e fechadas. Se ao final desses 1000 segundos o algoritmo não tiver encontrado a solução será considerado como insucesso.

5.6 Simulação do Matlab

O tráfego dos 41 coletores de dados dos medidores é distribuído ao longo de um intervalo de 30 segundos, determinado experimentalmente, cujo valor é suficiente para que todas as tabelas de rotas sejam inicializadas em cada nó da rede de telecomunicações. O algoritmo de reconfiguração tem o seu início escalonado aleatoriamente no intervalo $30 < t_{\text{Simulação}} < 45$ segundos para todos os três cenários avaliados.

O simulador envia, através das interfaces desenvolvidas, os dados dos atributos de potência ativa e reativa de cada carga a cada passo de simulação. A variação das cargas ao longo do tempo reproduz o comportamento de variação da carga. A real variação da carga ocorre ao longo de um ciclo de 24 horas. No entanto, o intervalo de tempo para reconfiguração da rede de distribuição de energia elétrica é da ordem de grandeza de segundos. A rede de distribuição de energia elétrica

é utilizada do ponto de vista do regime permanente. Portanto, os dados gerados pelo Simulink são utilizados para cálculo da nova configuração a cada rodada do algoritmo de reconfiguração. Por isso, a amostragem dos valores das cargas é feita em longos intervalos de 4 segundos, cuja forma é ilustrada na Figura 5.7, para uma carga específica (P,Q).

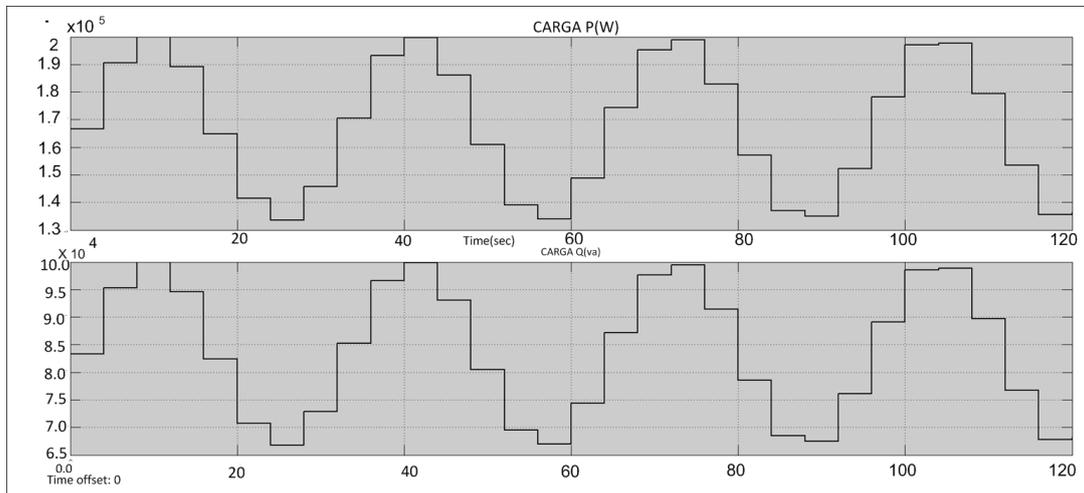


Figura 5.7: Cargas P e Q amostradas a cada 4 segundos

A cossimulação utiliza os dados da rede de distribuição de energia elétrica apenas como insumo do cálculo do fluxo de carga, ou seja, é uma análise estática. Para reproduzir uma carga dinâmica criou-se um bloco que contém duas fontes geradoras monofásicas, ilustrado na Figura 5.8, cujos valores reproduzem a forma da variação da carga ao longo da simulação. Os valores de pico coincidem com os valores máximos das cargas do modelo IEEE 33 barras modificado.

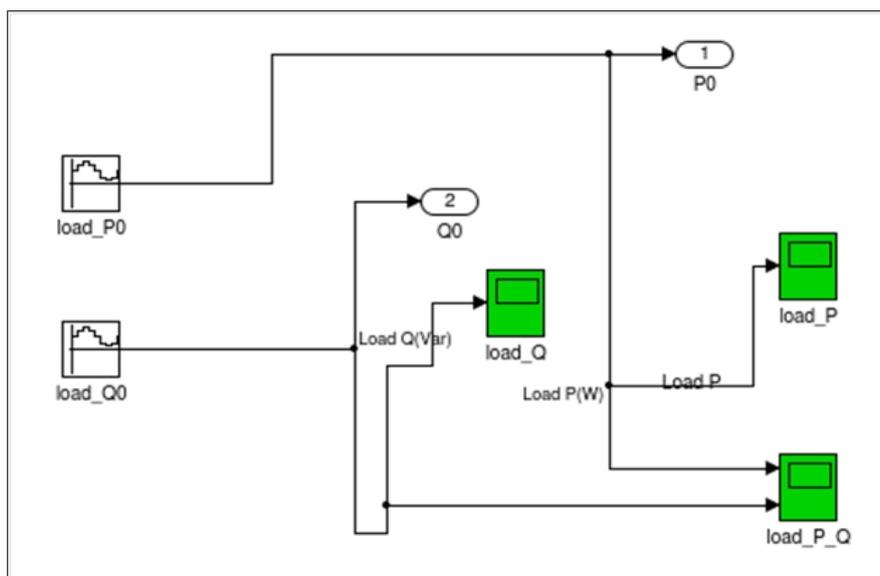


Figura 5.8: Modelo de cargas P e Q

5.7 Cenário composto com rede em malha sem-fio

Rede em malha é um tipo de arquitetura composta por rádios organizados topologicamente interconectados com outros de sua vizinhança de tal forma que o fluxo de dados ocorre em diversos saltos até alcançar o seu destino. Protocolos de roteamento determinam a melhor rota entre dois pontos e podem, dinamicamente, alterá-la em caso de falha ou quando há congestionamento. Os rádios podem ser compostos de um único canal, dois ou múltiplos. Diferentes topologias permitem oferecer capacidade e qualidade de serviço para aplicações avançadas como provedores de serviços ou segurança pública. Neste estudo, utiliza-se uma rede com apenas um canal e um nó sem-fio como *gateway* para uma rede de telecomunicações privada. Quanto mais pontos para escoamento de tráfego, maior a capacidade da rede assim como a sua confiabilidade.

Os nós da rede de telecomunicações ocupam a mesma posição física dos elementos que compõem a rede elétrica. Os rádios operam na faixa de espectro não licenciado em 2.4 GHz e são aderentes ao padrão IEEE 802.11g. Os principais parâmetros são listados na Tabela 5.5. Dado o curto alcance, a rede é complementada por um conjunto de nós com a função de repetição de sinal.

Parâmetros	Valor
Número de canais	1
Potência máxima	13 mW
Taxa fixa	6 Mbit/s
Sensibilidade	-90 dBm
Ruído Térmico	-110 dBm
α (expoente da perda de propagação)	2,7
Ganho da antena Tx/Rx	8 dBi omni
Limiar relação sinal/ruído+interferência (SNIR)	4 dB

Tabela 5.5: Tabela de parâmetros dos rádios em malha

É proposto que os nós da rede de telecomunicações sejam dispostos sobre uma grade com espaçamento de 250 metros para facilitar o desenvolvimento da arquitetura. Como o alcance dos rádios é curto, é necessário inserir nós com a função de repetição de sinal a fim de garantir a cobertura e a interconexão de todos os nós da rede de telecomunicações. A topologia física da rede em malha sem fio, é sobreposta aos nós da rede de distribuição de energia elétrica e é ilustrada na Figura 5.9. A localização física dos elementos da rede de distribuição de energia elétrica segue aproximadamente os comprimentos dos cabos que estão definidos na Tabela 5.1 e são dispostos linearmente a fim de facilitar o desenho. Nessa topologia,

as figuras geométricas representam o elemento da rede de distribuição de energia elétrica e os agentes associados a cada um. A rede é composta por agentes que gerenciam as barras (**Barra**), as cargas (**cargas**), as chaves normalmente fechadas (**CNF**), as chaves normalmente abertas (**CNA**), os repetidores de sinal (**R**) e o nó **gateway** que é o ponto central para onde o fluxo de dados dos coletores de dados é direcionado. A localização do nó **gateway** é aproximadamente o baricentro da rede a fim de minimizar o número de saltos para que os coletores de dados dos medidores inteligentes.

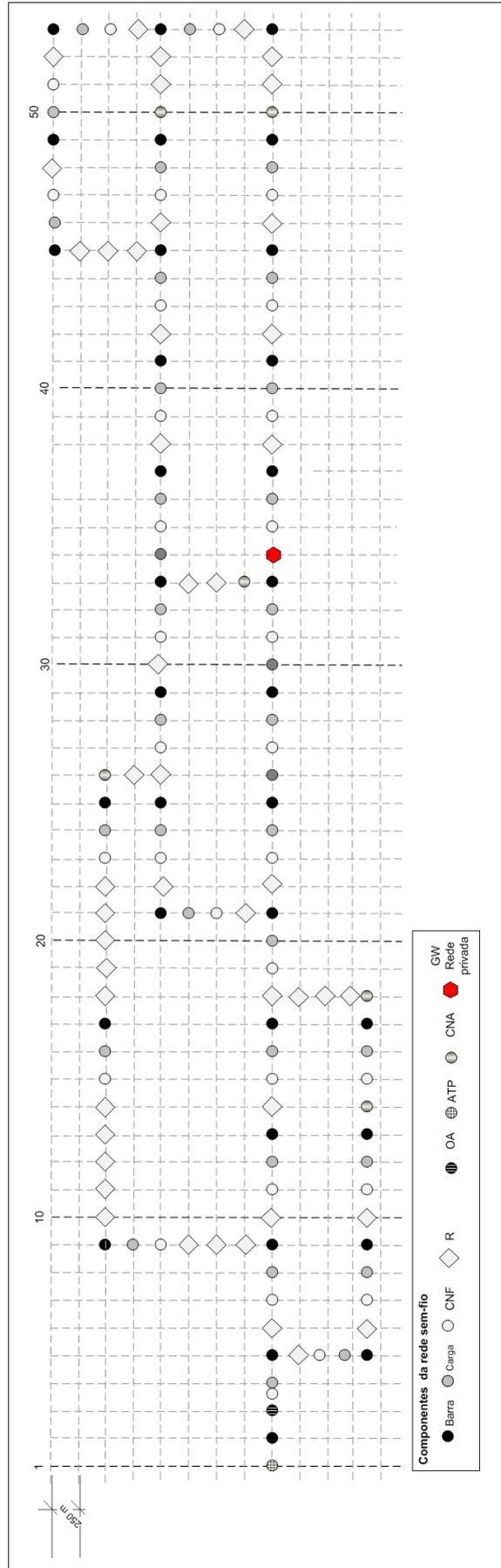


Figura 5.9: Topologia da rede de telecomunicações sobre uma grade

A rede possui ao todo 46 nós de repetição, sendo que 41 ao mesmo tempo simulam os coletores de dados dos medidores inteligentes. Na figura 5.9, o objeto na cor vermelha representa a localização física do nó de rede sem-fio que se interconecta à rede privada para trafegar os dados até o servidor, localizado na concessionária.

5.7.1 Modelo de propagação

O simulador Omnet++ modela um nó de rede sem-fio como sendo composto de blocos em que cada um representa uma camada do modelo OSI. A camada física é simulada calculando-se o nível de potência recebido, levando-se em consideração o modelo de propagação do sinal, ganhos das antenas, relação sinal-ruído e nível de interferência cocanal.

Os modelos de propagação do sinal dividem-se em determinístico, empírico e estocástico. O determinístico é baseado na teoria de propagação eletromagnética; o estocástico é baseado em dados estatísticos de medida em campo; o empírico é um modelo qualitativo, mas também faz uso de medidas em campo. Os modelos consideram as flutuações do sinal ocasionadas por reflexões, atenuações por propagação e por multi-percurso. Em cenários reais, medidas em campo são realizadas e o resultado é comparado com os modelos existentes e ajustes são feitos para calibração [87]. Neste estudo, serão adotados alguns modelos de propagação para efeito de análise dos resultados como o *freespace*, SUI, *logshadow* e 3GPP. Cada modelo é apropriado para diferentes tipos de ambientes, frequências e cenários.

Como as distâncias entre os nós são pequenas, adotou-se o modelo de perda de propagação denominado *log-normal*. A perda de propagação é um modelo físico, representado pela equação (5.9) [88], expresso em dB. A distância d é um ponto de referência à distância de 1 metro e α é o expoente que depende do ambiente de propagação. O expoente geralmente varia entre 2 e 4 [88]. Para ambientes urbanos e suburbanos adotam-se os valores de 4 e 2.7 [88], respectivamente.

$$PL(dB) = PL(d_0) + 10\alpha \log_{10}(d/d_0) \quad (5.9)$$

Substituindo-se os parâmetros da Tabela 5.5 e o valor do expoente de propagação na equação (5.9), resulta no gráfico da Figura 5.10, no qual ilustra o nível de potência recebida por um nó e o seu limiar de sensibilidade de -90 dBm. Através da diferença entre a sensibilidade do rádio e a perda de propagação representa a soma dos ganhos das antenas. Disso resulta que cada antena deve ter um ganho de 8 dBi.

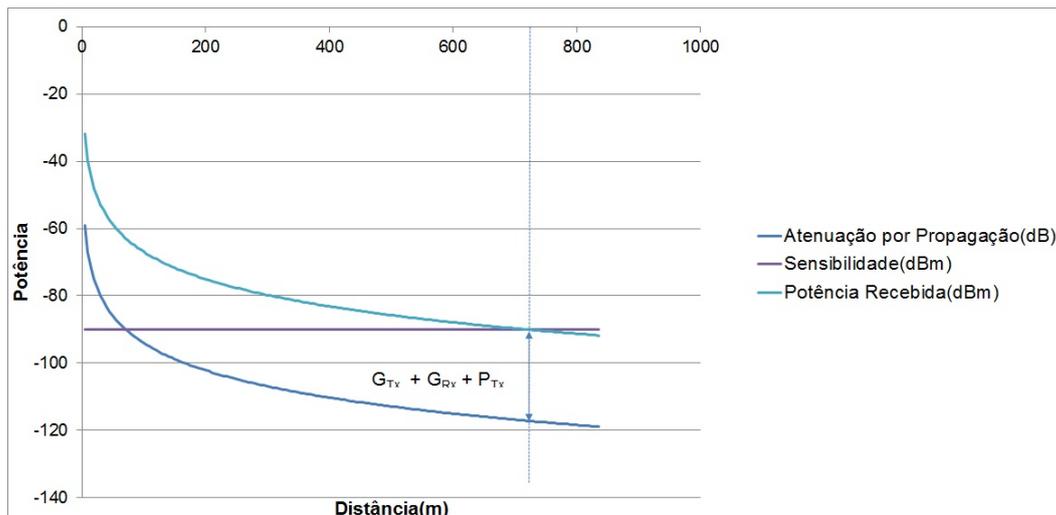


Figura 5.10: Nível de potência recebido em função da distância

5.7.2 Protocolo de roteamento

Existem três categorias de protocolo de roteamento: proativo, que periodicamente divulga as Tabelas de roteamento para os nós vizinhos; o reativo, que divulga a tabela somente no momento do envio do pacote e o híbrido, que combina as vantagens dos dois, aplicado às zonas de roteamento restritas a distâncias de um nó específico. O Omnet++ disponibiliza alguns protocolos das duas primeiras categorias.

Sabe-se que o protocolo proativo introduz menor latência por manter sempre a tabela de rotas atualizada mesmo que não exista pacote para envio. No entanto, como as mensagens de roteamento são difundidas para toda a rede periodicamente, isso pode se tornar ineficiente quando se trata de rede em grande escala.

Ao contrário, o protocolo reativo acrescenta uma latência à propagação dos dados pelo fato de divulgar as rotas apenas no envio dos pacotes.

O simulador Omnet++ disponibiliza alguns modelos de protocolos ativos e reativos para redes ad-hoc. Esses protocolos geralmente utilizados em mobilidade, serão testados com o tipo de rede proposta. Entre os reativos, pode-se citar o *Ad-hoc On-demand Distance Vector* (AODV), *Distance Vector Routing* (DSV) e *Dynamic Manet On-Demand* (DYMO), sendo que esse último é parecido com o AODV exceto que não divulga mensagens de *HELLO*. Em termos de protocolo proativo, o *Optimized Linkstate Protocol* (OLSR) e o *Destination Sequence Demand Vector* (DSDV) se encontram disponíveis também. Em [89], os autores realizam uma comparação de desempenho entre o DYMO, AODV, DSR e DSDV e demonstram que o DYMO é o que apresenta o melhor desempenho de entrega de pacotes e de menor latência. Isso se deve ao fato do protocolo acumular as rotas divulgadas ao longo do caminho do pacote, desde a fonte até o destino. Além disso, o acúmulo de informação das rotas dos vizinhos reduz a necessidade de se iniciar uma requisição

de rotas e, conseqüentemente, redução de *overhead*.

5.7.3 Resultados da simulação

Todos os protocolos disponíveis foram testados, porém somente dois tiveram algum sucesso de funcionamento e ambos são reativos: o DSR e o DYMO em condições específicas. Essas condições foram desenvolvidas a nível de aplicação, cujo nó correspondente, ao requisitar o valor de uma carga, aguarda o retorno da resposta para enviar a próxima solicitação.

Essa rede que contém uma grande quantidade de saltos com um único ponto de escoamento de tráfego e um único canal de transmissão é possível a ocorrência de problemas para envio de pacotes, já que o número de saltos é grande. Segundo a equação (5.7), considerando-se que o número de nós da rede é 149, o limite superior para capacidade de tráfego por nó é de 40 kbps, sem considerar o efeito das colisões de pacote e a latência. Esse seria o valor máximo da taxa que os coletores poderiam enviar.

Por outro lado, considerando-se uma rede *ad-hoc*, a expressão (5.6) levaria a um máximo de 345 kbps por nó. Das simulações executadas, observa-se que os nós podem alcançar taxas de 170 kbps, quando os nós estão muito próximos, a algumas poucas dezenas de bytes/seg para uma longa cadeia de nós. O resultado das simulações está ilustrado na Tabela 5.6.

Taxa (Bytes/seg)	Tempo de reconfig. (seg)	Intervalo de confiança (95%)	% Finalização
1854	300	± 20,72	100
1236	202,27	± 28,74	100
927	193,55	± 23,18	100
618	197,27	± 25,20	100
412	182,85	± 23,40	100
309	194,63	± 23,97	100
206	198,34	± 24,03	100
155	180,35	± 20,42	100
103	190,87	± 25,24	100
52	169,28	± 21,50	100

Tabela 5.6: Tabela de resultados das simulações

Como pode ser verificado na Figura 5.11, para taxas mais elevadas dos coletores de dados, a rede aumenta a sua latência devido à formação de gargalos no entorno do nó de interconexão com a rede privada. Em [90], os autores demonstram por meio de

simulações de uma topologia aleatória e uma topologia na forma de um reticulado com vários *gateways*, a capacidade do nó independe do protocolo de roteamento e decresce a uma taxa superior a $O(1/n)$, devido principalmente às interferências, colisões e ao gargalo existente no entorno dos *gateways*.

Apesar da camada de transporte retransmitir os pacotes, o número de saltos e colisões tornam a rede ineficiente. Em princípio, o tráfego dos coletores não contribuiu para perda de pacotes no tráfego de reconfiguração da rede.

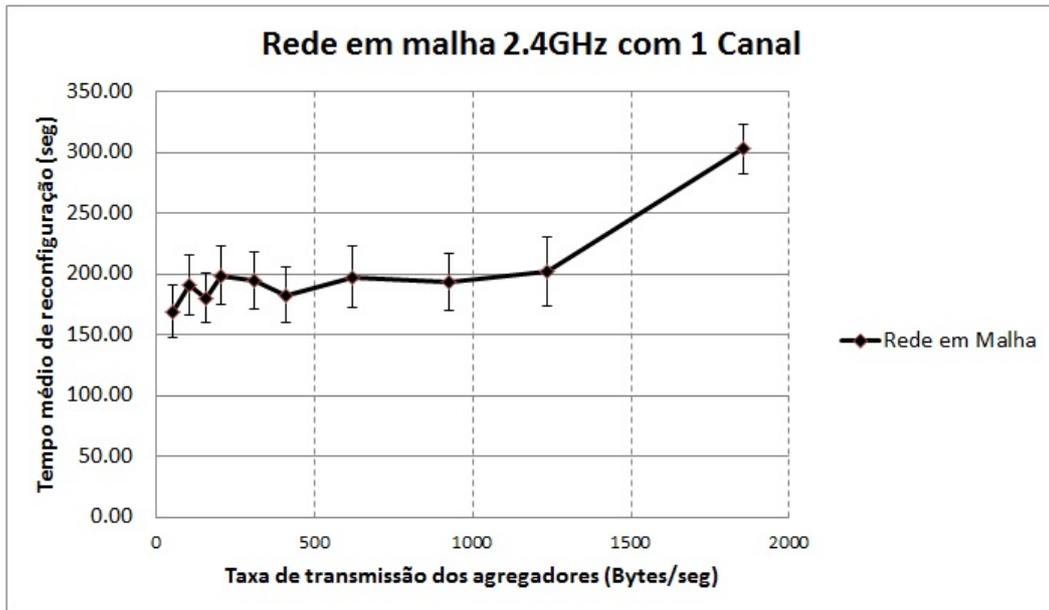


Figura 5.11: Resultado dos tempos de reconfiguração da rede de distribuição de energia elétrica

Retirando-se os coletores de dados, a simulação da rede chega aos resultados indicados na Tabela 5.7. Nessa Tabela, \bar{T} representa a média populacional, $T_{\text{máx}}$ representa o valor máximo alcançado por uma cossimulação, $T_{\text{mín}}$ representa o tempo mínimo alcançado por uma cossimulação, σ é o desvio-padrão da média amostral, IC é o intervalo de confiança e P_f é o percentual de cossimulações finalizadas com sucesso.

\bar{T}	108,18 seg
$T_{\text{máx}}$	170,24 seg
$T_{\text{mín}}$	69,91 seg
σ	19,09 seg
IC	3,76 seg
P_f	100%

Tabela 5.7: Tabela de tempo de reconfiguração sem o tráfego dos coletores

Comparando-se as Tabelas 5.6 e 5.7, é possível observar a influência do tráfego nos tempos médios de reconfiguração, o que indica que à medida que a intensidade de fluxo do tráfego dos medidores aumenta, maior é a competição pelo acesso ao meio e, conseqüentemente, ocorre a redução da capacidade de escoamento de tráfego de reconfiguração da rede.

5.8 Cenário com rede ponto-multiponto

A rede de telecomunicações ponto-multiponto (PMP) caracteriza-se por um grupo de rádio-bases com uma cobertura de sinal em todas as direções. O sistema é composto de sete estações, 41 coletores de dados, nós agentes com papéis de atuadores, sensores e com inteligência distribuída, conforme ilustrado na Figura 5.12. Os nós atuadores referem-se ao elemento da rede de distribuição de energia elétrica que é uma chave normalmente aberta (CNA_i) ou fechada (CNF_k). Os nós sensores referem-se aos elementos das cargas (L_i), cujos valores das potências ativa e reativa são medidos e repassados para o nó associado à barra (B_i). As cores delimitam a área de cobertura e indicam também os canais utilizados (4) não sobrepostos. Como a rede opera no modo infraestruturado, cada um dos receptores se associa a cada um dos sete pontos de acesso.

Os rádios utilizam o padrão IEEE 802.11b com as características listadas na Tabela 5.8 e operam em espectro não licenciado em 900 MHz. A escolha desse padrão deve-se ao fato que a biblioteca do simulador Omnet++ dispor de apenas alguns modelos de rádio (IEEE 802.11 a/b/g). Na prática existem alguns modelos cujas interfaces são parecidas com o IEEE 802.11 e operam em 900 MHz [91, 92].

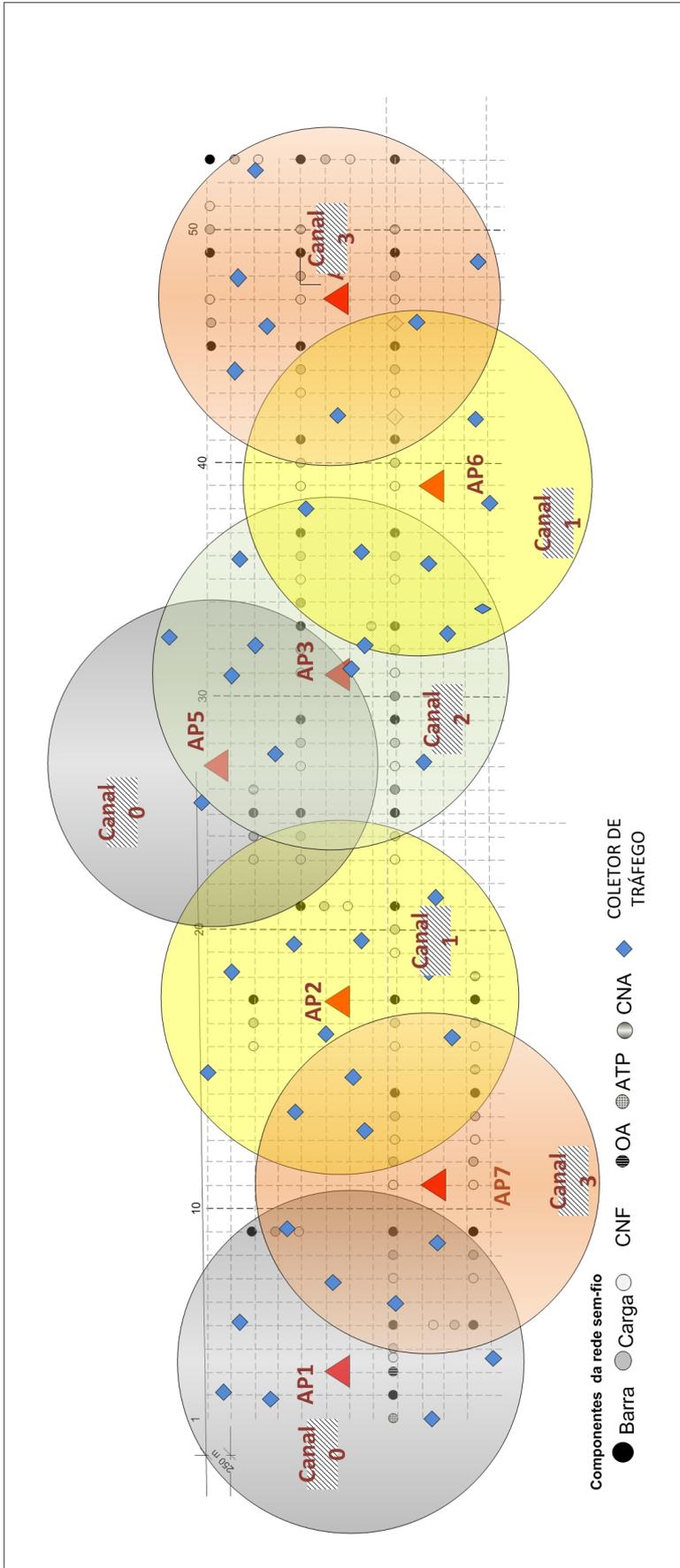


Figura 5.12: Arquitetura da rede de telecomunicações ponto-multiponto

5.8.1 Modelo de propagação

Nesse cenário adotou-se o modelo *log-shadowing*, representado pela equação (5.10). Considera-se que d_0 é uma distância referência, normalmente considerada com valor de 1 metro, cuja atenuação até esse ponto é a de espaço livre.

$$PL(dB) = PL(d_0) + 10\alpha \log_{10}(d/d_0) + \chi_\sigma \quad (5.10)$$

A parcela χ_σ representa as flutuações aleatórias no nível de potência do sinal provocado por objetos posicionados entre o transmissor e receptor. É modelada por uma função de distribuição $\mathcal{N}(0, \sigma)$. A variável α representa o expoente da perda de propagação e depende do meio e da frequência utilizada. Geralmente é estimada através de medições dos níveis de potência em campo. Para esse cenário adotou-se também o valor de 2.7.

5.8.2 Cálculo do raio de cobertura

O raio de cobertura é determinado pelo menor balanço de potência entre os sentidos dos enlaces rádio-base \leftrightarrow receptor, expresso pela equação (5.11).

$$P_r = P_t + G_t + G_r - PL \quad (5.11)$$

$$P_r \geq \text{Sensibilidade} \quad (5.12)$$

em que PL é expressa pela equação (5.10) A parcela P_t representa a potência transmitida, a parcela G_t é o ganho da antena transmissora, a parcela G_r é o ganho da antena receptora e PL é a perda de propagação do sinal descrita pelo modelo de propagação *log-shadow*. A cobertura de um rádio pode ser especificada para um determinado percentual, pois não é possível garantir 100% do tempo. O ambiente interfere na propagação por meio de reflexão ou multipercursos. Para se determinar o raio de cobertura de uma estação rádio-base adotou-se o cálculo do nível de recepção do sinal para um raio que atenda à restrição dada pela equação (5.12) cujo valor é sensibilidade do rádio. A maior distância entre uma rádio-base e um receptor é de 1810 metros. Os dados dos receptores e parâmetros do modelo de propagação estão representados na Tabela 5.8. Considerando-se os parâmetros utilizados, a Figura 5.13 ilustra uma comparação da potência máxima e mínima recebida, do limiar de sensibilidade do rádio e a atenuação por propagação adicionado de 3σ . Para uma probabilidade de 99.7% (3 vezes o desvio padrão) para dar cobertura com essa distância, o ganho da antena deverá ser de 5dBi para ambas as antenas.

Largura de banda	5 MHz
Altura da antena da rádio-base	15 m
Altura das antenas dos transmissores	10 m
Ganho antena _{rádio-base}	5 dBi omni
Ganho antena _{receptor}	5 dBi omni
Potência Tx _{rádio-base}	25,44 dBm
Potência Tx _{receptor}	25,44 dBm
Sensibilidade	-93 dBm @ 1 Mbps
σ	3,0 dB

Tabela 5.8: Tabela de parâmetros utilizados no modelo ponto multiponto

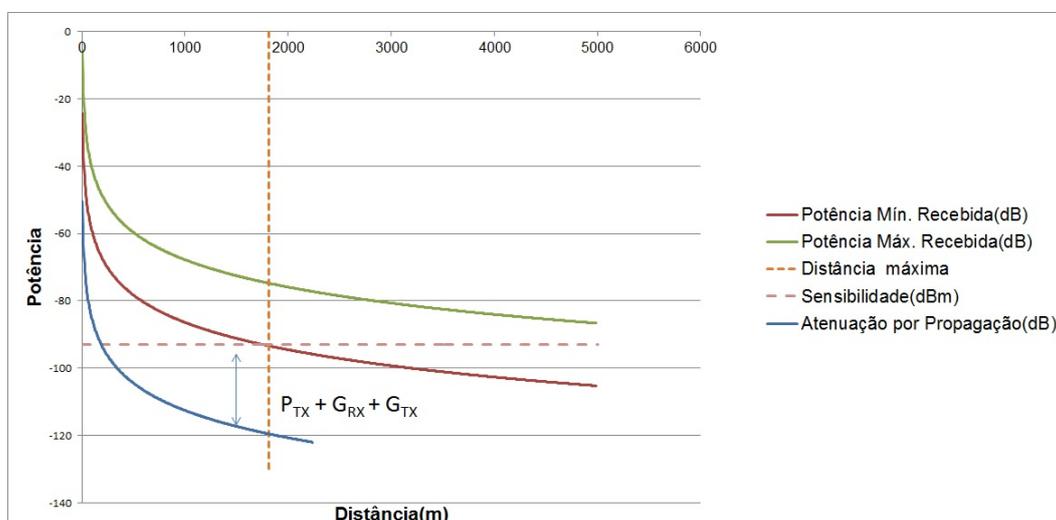


Figura 5.13: Potência do sinal recebido por um nó da rede PMP

A topologia utilizada permite a cobertura da área utilizando-se 4 canais diferentes de tal forma que a interferência cocanal seja reduzida.

5.8.3 Capacidade da rede ponto-multiponto

Para uma rede sem-fio com um único salto, como no caso da rede ponto-multiponto, o critério para cálculo da capacidade máxima de fluxo de um nó é também representada pela expressão (5.7). A rede possui ao todo 144 nós, sendo 103 nós associados aos componentes elétricos da rede e 41 coletores de dados. A rádio-base com a maior cobertura possui 40 nós associados. Nesse caso, o limite máximo da capacidade de um nó é de $\Theta(R/40)$ ou 25 Kbps em que R é a velocidade máxima do enlace cujo valor é de 1 Mbps.

5.8.4 Resultados da simulação

Considerando-se os parâmetros adotados da Tabela 5.8, os resultados foram insatisfatórios, pois em nenhum caso a simulação foi concluída com sucesso. Isso se deve ao fato do possível nível de interferência ocasionado pelos outros receptores dentro da área de cobertura de uma mesma rádio-base. Nas cossimulações identificou-se que, ao se reduzir a potência dos rádios de 25 dBm para 21dBm, obtém-se melhores resultados que estão ilustrados na Tabela 5.9. Na Figura 5.14 estão ilustrados os valores do tempo médio de reconfiguração e as taxas dos coletores de dados dos medidores inteligentes. No entanto, observa-se ainda um percentual de insucesso, podendo ser ocasionado por colisões de pacotes ou por falta de cobertura em algum período da execução da simulação. Nesse aspecto, sendo a sensibilidade de -93 dBm e $\sigma=3.0$ dB e calculando-se a função densidade de probabilidade para que o mínimo seja alcançado, tem-se uma cobertura de 91.7% do tempo. Nessa condição, quando o pacote não chega ao seu destino pode ocorrer retransmissão do sinal até que não o limite seja alcançado.

Taxa (Bytes/seg)	Tempo médio de reconfig. (seg)	Intervalo de confiaça (95%)	% Finalização
1854	72,67	$\pm 11,88$	100
1236	48,01	$\pm 4,87$	99
927	48,31	$\pm 7,44$	97
618	56,33	$\pm 9,21$	97
412	62,79	$\pm 5,89$	98
309	60,88	$\pm 10,12$	100
206	49,20	$\pm 7,98$	95
155	40,01	$\pm 0,37$	98
103	43,35	$\pm 5,58$	99
52	42,37	$\pm 4,67$	100

Tabela 5.9: Tabela dos resultados da simulações com tráfego dos coletores

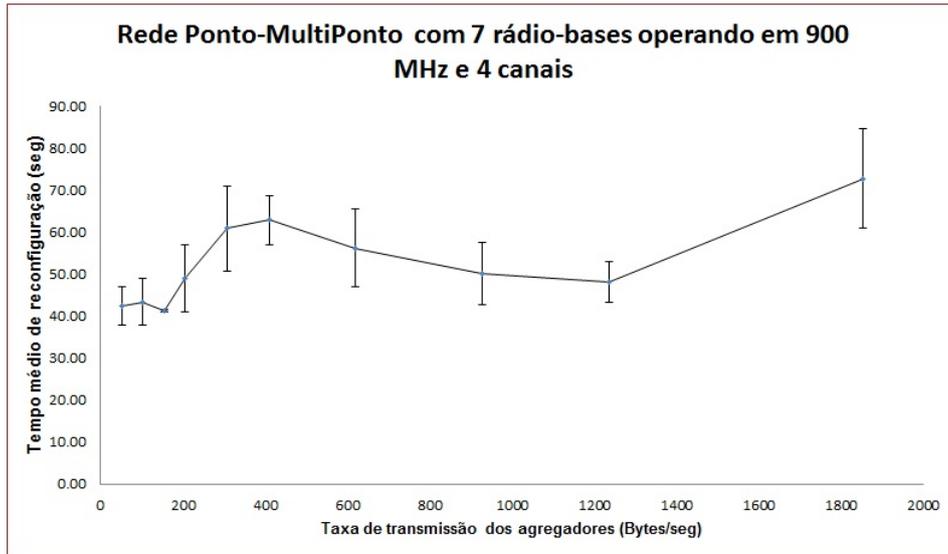


Figura 5.14: Resultado das simulações com o tráfego dos coletores

Reproduzindo a simulação sem a presença do tráfego dos coletores, obtém-se os resultados representados na Tabela 5.10 em que \bar{X} é o tempo médio, $T_{\text{máx}}$ e $T_{\text{mín}}$ são o maior tempo de simulação observando em 100 rodadas, respectivamente, σ é o desvio padrão, IC é o intervalo de confiança e P_f é o percentual de finalização das 100 rodadas de simulação.

\bar{T}	41 seg
$T_{\text{máx}}$	114 seg
$T_{\text{mín}}$	38 seg
σ	7,58 seg
IC	$\pm 1,49$ seg
P_f	99%

Tabela 5.10: Tabela de resultados das simulações da rede PMP

5.9 Cenário composto com rede LTE

O padrão LTE (*Long Term Evolution*) foi desenvolvido visando atender à crescente demanda de tráfego de vídeo, dados e voz nas redes móveis com capacidade para prover serviços com qualidade e eficiência. A tecnologia foi criada pelo 3GPP e é uma evolução da rede GSM, com a promessa de alcançar velocidades de 300 Mbps *downlink* e 75 Mbps no *uplink*, com baixa latência e escalabilidade.

Para oferecer essas taxas, o LTE utiliza a técnica de modulação *Orthogonal Frequency Division Multiplexing Access* (OFDMA) no enlace da estação rádio-base para o terminal e *Single Carrier Frequency Division Access* (SC-FDMA) no

sentido inverso. Ambas usam técnica de modulação digital. A técnica consiste na transmissão paralela de dados em um grande número de subportadoras ortogonais com baixas taxas de transmissão. Cada grupo de doze subportadoras tem largura de 180kHz e cada subportadora está separada por um intervalo de 15 kHz. Esse espaçamento garante a ortogonalidade do sinal pela localização em pontos de cruzamento zero do espectro das outras subportadoras. O OFDMA divide as subportadoras no domínio do tempo. As doze subportadoras formam um bloco, conhecido como bloco de recurso, e cada subportadora tem sete elementos de recurso, conforme ilustrado na Figura 5.15. O bloco representa a menor alocação possível e tem duração de 0.5 ms (1 *slot*). Dois *slots* são agrupados para formar um subquadro denominado *Transmit Time Interval* (TTI) com duração de 1 ms.

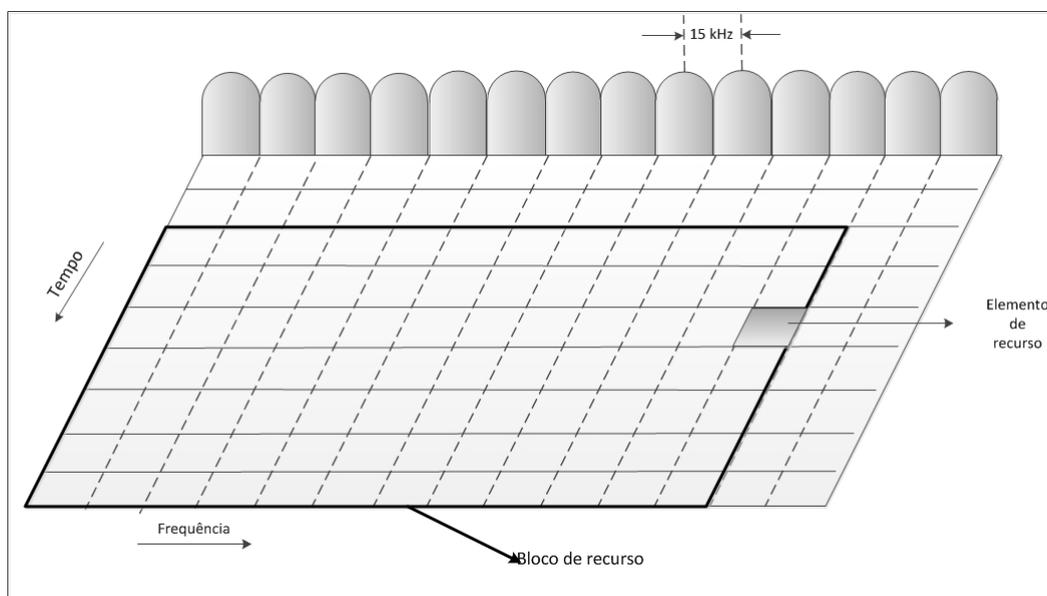


Figura 5.15: OFDMA

A tecnologia permite o uso de diferentes tipos de larguras de banda entre 1.4 MHz e 20 MHz. No caso do Brasil são utilizados valores de 10 MHz e 20 MHz nas portadoras de 2.5 e 2.6 GHz, conforme resolução da Anatel [93]. Pode-se operar com no modo *Frequency Division Duplex* (FDD)) ou *Time Division Duplex* (TDD)). No FDD, duas portadoras de frequências diferentes são utilizadas para o *uplink* e *Downlink*, enquanto que no TDD uma única portadora é utilizada em ambos sentidos, porém o envio de quadros é separado no tempo.

A arquitetura, ilustrada na Figura 5.16, é conhecida como *Evolved Packet System* (EPS), baseada totalmente em protocolo IP, o que permite a utilização de múltiplos serviços como voz sobre IP (VoIP), comunicação M2M, vídeo *streaming*, dentre outras que requerem priorização de tráfego.

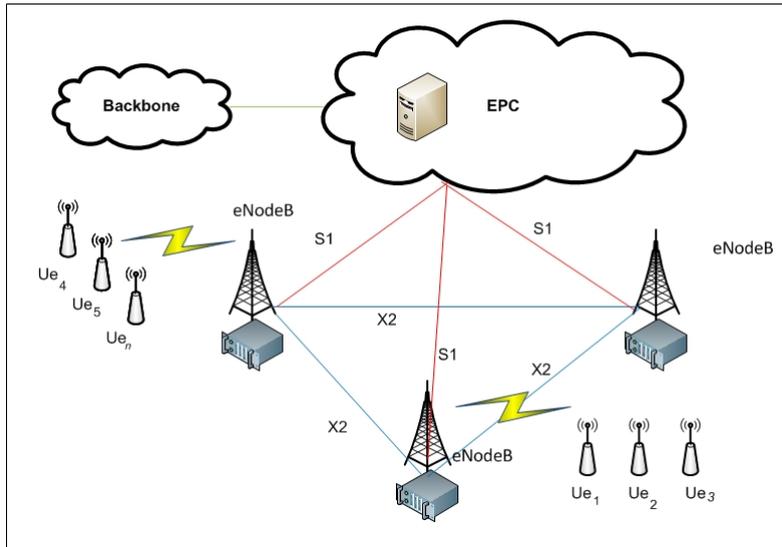


Figura 5.16: Arquitetura básica do LTE

Nessa arquitetura, a rádio-base LTE é conhecida como *Enhanced NodeB* (eNodeB), o *Evolved Packet Core* (EPC), que representa o núcleo da arquitetura LTE, é composto por diversos subsistemas, e o *User Equipment* (Ue) que representa o terminal do usuário. O X2 é um tipo de interface da rede de rádio de acesso que conecta os nós eNodeBs e gerencia o *handover* dos terminais móveis. O S1 é um tipo de interface que o núcleo da rede acessa as estações rádio-bases e consiste em um protocolo que roda sobre IP e gerencia o tráfego do usuário, dos subsistemas que compõem o EPC e as rádio-bases.

5.9.1 Modelo computacional

O modelo computacional foi desenvolvido pela universidade de Pisa [94] e contempla apenas o plano de dados, isto é, as mensagens de controle não estão presentes que permitem o sincronismo das diferentes estações rádio-bases, a alocação das subportadoras e de controle dos eNodeBs. O modelo de rádio do eNodeB pode operar apenas no modo FDD.

Do ponto de vista de camada OSI, o modelo desenvolvido envolve apenas as camadas físicas, MAC e a interface com IP. Todo resto da pilha de protocolos utiliza a biblioteca do Omnet++.

A topologia física implementada no Omnet++ é ilustrada na Figura 5.17, cuja distância entre sites para o cenário de área suburbana é de 1299 metros. Esse modelo de distribuição das rádio-bases está de acordo com a recomendação do ITU-R [95] para testes de sistemas de simulação. Com isso, para dar cobertura a todos os receptores no cenário avaliado, são utilizadas 31 rádio-bases com três setores cada.

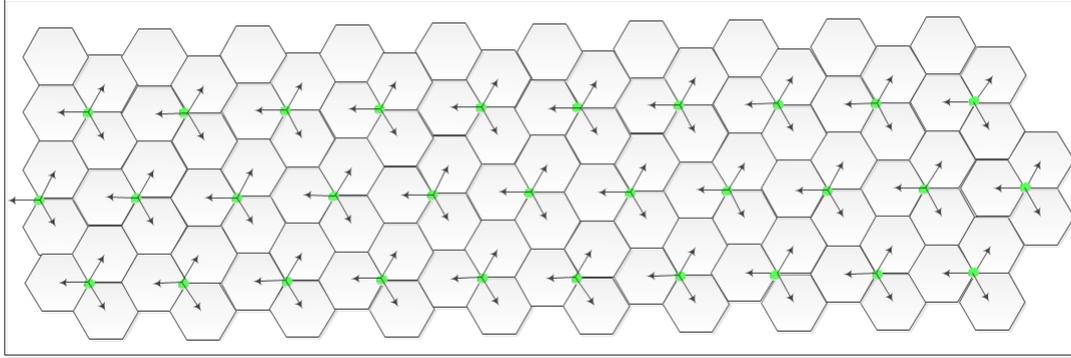


Figura 5.17: Topologia física das rádio-bases LTE implementadas no Omnet++

5.9.2 Modelo de propagação

Todo o desenvolvimento da camada física baseou-se no relatório técnico do 3GPP TR 36.814 V9.0.0 [96], do grupo de rede de rádio de acesso, o qual descreveu o funcionamento da rede de acesso via rádio. O objetivo desse relatório foi definir e descrever a camada física em estudo. O relatório também estabeleceu um conjunto de cenários para simulação e avaliação de desempenho para ambientes internos e externos, entre os quais destacam-se modelos para picocélulas, microcélulas, macrocélula urbana, suburbana e rural.

5.9.2.1 Modelo de propagação do 3GPP

O relatório fornece um conjunto de equações que regem os modelos de propagação para os seguintes tipos de cenários: macrocélula urbana, suburbana, rural, microcélular *indoor* e microcélular urbana. De acordo com o cenário de macrocélula suburbana, o relatório determina as equações para perda de propagação do sinal para visada direta entre a estação rádio-base e o terminal do usuário móvel, expressos pelas equações (5.13) e (5.14), e sem visada direta, expresso pela equação (5.15). Acrescenta-se às equações uma função distribuição normal, com média 0 e desvio-padrão σ , que representa os efeitos de multipercursos do sinal e as reflexões.

$$PL_1 = 22 \log_{10}(40\pi df_c) + \min(0.03h^{1.72}, 10) \log_{10}(d) - \min(0.044h^{1.72}, 14.77) + 0.002 \log_{10}(h)d + Z \quad (5.13)$$

em que $10 \text{ m} < d < d_{BP}$, Z é uma variável aleatória cuja distribuição é do tipo $\mathcal{N}(0, \sigma)$ e $\sigma=4$. Além disso,

$$PL_2 = PL_1 + 40 \log_{10}(d/d_{BP}) \quad (5.14)$$

em que $d_{BP} < d < 5000$ m e $\sigma=6$. Finalmente,

$$\begin{aligned}
PL_3 = & 161.04 - 7.1 \log_{10}(Ws) + 7.5 \log_{10}(h) - (24.37 - 3.7(h/h_{BS})^2) \log_{10}(h_{BS}) \\
& + (43.42 - 3.1 \log_{10}(h_{BS})(\log_{10}(d) - 3) + 20 \log_{10}(f_c) \\
& - (3.2(\log_{10}(11.75h_{UT}))^2 - 4.97) + Z \quad (5.15)
\end{aligned}$$

em que h é a média das alturas dos prédios, em metros, Ws é a largura da rua em metros, h_{BS} é a altura da da antena da rádio-base, h_{UT} é a altura da antena do equipamento do usuário, f_c é a frequência da portadora em Hertz, d é a distância entre o terminal do usuário e a rádio-base, em metros, d_{BP} é uma distância estabelecida expressa pela equação (5.16) e Z é uma função do tipo $\mathcal{N}(0, \sigma)$.

$$d_{BP} = \frac{2\pi h_{BS} h_{UT} f_c}{3.10^8} \quad (5.16)$$

Note que a equação (5.15) é válida para distâncias entre $10 \text{ m} < d < 5000 \text{ m}$ e desvio padrão $\sigma=8$.

5.9.2.2 Modelo SUI

O modelo SUI (*Stanford University Interim*) foi desenvolvido pela Universidade de Stanford para faixa de frequência entre 2,5-2,7 GHz é muito utilizado para estudo de propagação de rádios com tecnologia Wimax(IEEE 802.16m). O modelo, que foi desenvolvido a partir de medições em campo, categoriza três tipos de terrenos e suas características: urbano (tipo A), que é identificado para áreas urbanas por ter a maior perda de propagação; suburbano (tipo B), que é identificado para áreas suburbanas, com média densidade e pouco montanhoso; rural (tipo C), que é identificado para áreas rurais, baixa densidade, poucas árvores e uma superfície plana. A perda de propagação, acrescida de fatores de correção, é dada pela expressão (5.17), válida para $d > d_0$.

$$PL = A + 10\gamma \log(d/d_0) + \chi_f + \chi_h + s, d > d_0 \quad (5.17)$$

$$A = 20 \log_{10}(4\pi d_0 / \lambda), d \leq d_0 \quad (5.18)$$

$$\chi_f = 6.0 \log_{10}(f_c/2000) \quad (5.19)$$

$$\chi_h = -10.8 \log_{10}(h_{UT}/2) \quad (5.20)$$

$$\chi_h = -20 \log_{10}(h_{UT}/2) \quad (5.21)$$

em que PL é dado em dBm, d é a distância entre a estação rádio-base e o receptor instalado em poste de energia elétrica, f_c é a frequência da portadora em MHz, d_0

é uma distância referência (100 m) e s é um fator que representa o desvanecimento lento modelado cujo valor varia entre 8,6 e 10,2 dB. As equações (5.20) e (5.21) são válidas para os tipos de terrenos A, B e C, respectivamente, e representam fatores de correção para frequências superiores a 2 GHz. A variável λ representa o comprimento de onda em metros enquanto γ é expoente da perda de propagação dado por

$$\gamma = a - bh_{BS} - c/h_{BS} , \quad (5.22)$$

na qual h_{BS} é a altura da antena da rádio-base em metros, h_{UT} é a altura da antena dos receptores localizados nos postes. Os valores adotados para as constantes j , k e l encontram-se na Tabela 5.11 e variam em função do tipo de terreno:

Parâmetro	Terreno A	Terreno B	Terreno C
j	4,6	4	3,6
k	0,0075	0,0065	0,005
l	12,6	17,1	20

Tabela 5.11: Tabela de parâmetros do modelo SUI

O cenário selecionado é o do tipo B, apropriado para macrocélula suburbana. Os parâmetros do modelo são apresentados na Tabela 5.12.

Frequência	2500 MHz
Largura de banda	10 MHz
Número de <i>Resource Blocks</i>	50
h_{UT}	10 m
h_{BS}	15 m
Ganho antena rádio-base (G_{TX})	18 dBi/70° setorizada
Potência Tx rádio-base (P_{TX})	46 dBm
Potência Tx terminal receptor (P_{RX})	24 dBm
Taxa de blocos errados	10%

Tabela 5.12: Tabela de parâmetros utilizados no modelo LTE

Para efeito de comparação é apresentado na figura 5.18, as perdas de propagação entre os modelos SUI e 3GPP em função da distância entre a rádio-base e o receptor, considerando-se as componentes aleatórias de desvanecimento lento (*shadowing*) aplicáveis ao modelo do 3GPP.

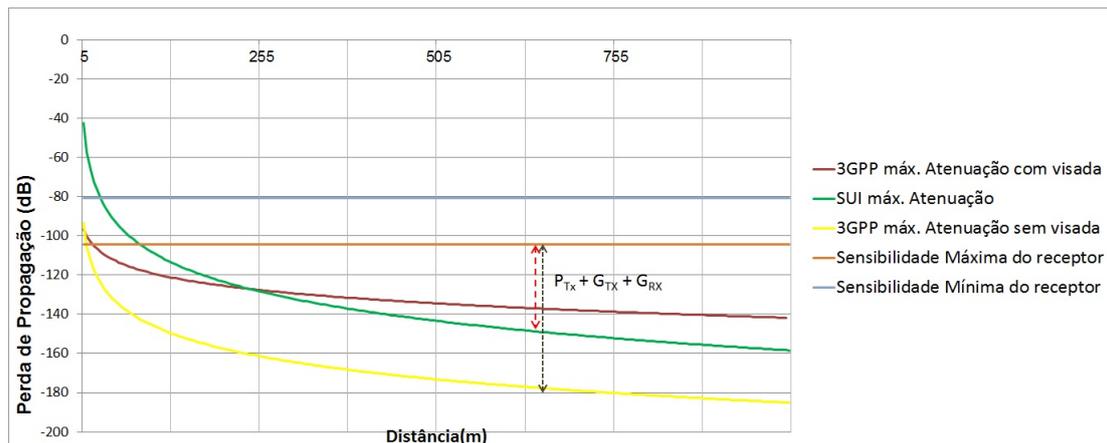


Figura 5.18: Comparação dos modelos de propagação SUI e 3GPP

Da Figura 5.18, a linha amarela representa a maior atenuação possível para o modelo 3GPP, considerando-se que esta é a pior condição quando o receptor não está apontado diretamente para o lobo principal da antena da estação rádio base. As setas indicam que na borda da célula, cujo raio é de 650 metros, o nível de potência recebido deve ser tal que seja maior que a soma dos ganhos das antenas e da potência transmitida pelo receptor. Considerando-se que o ganho máximo de potência de um receptor, pela legislação é de 24 dBm, chega-se ao valor de 31 dBm.

5.9.3 Cálculo da sensibilidade dos rádios

A sensibilidade de um receptor é a menor potência do sinal na sua entrada que seja capaz de ser decodificado corretamente. No presente estudo o cálculo permitirá determinar se o nível de potência transmitida pelo receptor e o ganho da antena são suficientes para alcançar a rádio-base na pior condição possível.

Determina-se a sensibilidade S pela equação (5.23), cuja primeira parcela representa o ruído térmico, a segunda representa a figura de ruído do equipamento e a terceira, representa a razão sinal/ruído do canal para que o mesmo possa ser decodificado.

$$S = 10 \log_{10}(K.T.B) + F_r + SNR \quad (5.23)$$

em que K é a constante de Boltzman ($1,38 \times 10^{-23}$), T é a temperatura ambiente na entrada do receptor (290 Kelvin) e B é a largura de banda do canal, que no caso é de 10 MHz, as figuras de ruído típicas para o receptor e a estação rádio-base são 7 e 5 dB, respectivamente e SNR é dado pela Tabela 5.13 [97]. Essa última parcela é dependente do tipo de esquema modulação selecionado pela rádio-base. Baseando-se nesses dados, o terminal calcula o *Channel Quality Indicator* (CQI) mais elevado que permita um bloco ser decodificado com uma probabilidade de erro inferior a 10%. Essa informação é então enviada para a estação rádio-base, indicando qual o

esquema que pode ser suportado.

Tipo de modulação	Taxa do código	SNR [dB]
QPSK	1/8	-5,1
QPSK	1/5	-2,9
QPSK	1/4	-1,7
QPSK	1/3	-1,0
QPSK	1/2	2,0
QPSK	2/3	4,3
QPSK	3/4	5,5
QPSK	4/5	6,2
16 QAM	1/2	7,9
16 QAM	2/3	11,3
16 QAM	3/4	12,2
16 QAM	4/5	12,8
64 QAM	2/3	15,3
64 QAM	3/4	17,5
64 QAM	4/5	18,6

Tabela 5.13: Tabela de relação modulação e sinal-ruído

Considerando-se a banda em questão e os parâmetros de modulação, a sensibilidade do rádio receptor e da estação rádio-base são calculadas os valores máximos e mínimos são expressos pelas desigualdades (5.24) e (5.25).

$$-102,6dBm \leq Sensibilidade_{receptor} \leq -78,6dBm \quad (5.24)$$

$$-104,6dBm \leq Sensibilidade_{rádio-base} \leq -80,6dBm \quad (5.25)$$

5.9.4 Determinação do ganho da antena do receptor

A limitação de cobertura de um sistema de rádio é determinada pelo enlace de subida porque a potência do receptor e a sua antena são menores do que as da estação rádio-base. Admitindo-se que a distância entre as estações rádio-bases é de 1299m, o pior caso para recepção da estação rádio-base ocorre quando o receptor está na borda da célula, isto é, quando estiver a uma distância de 650 metros. A partir da sensibilidade dos rádios, das curvas de perda por atenuação, da equação de balanço de potência, vide equação (5.11) e do raio da célula (650 m), pode-se calcular o ganho mínimo da antena do receptor a partir da equação (5.27), para garantir que o sinal seja demodulado pela rádio-base. Nesse caso, a desigualdade (5.26) deve ser satisfeita em que $P_{recebida}$ é a potência do sinal recebida pela estação

rádio-base e pelo receptor.

$$P_{recebida} \geq \text{Sensibilidade} \quad (5.26)$$

$$P_{Rx} \geq P_{Tx} + G_{Tx} + G_{Rx} - PL_{650m} \quad (5.27)$$

em que P_{Rx} é a potência do sinal recebido, P_{Tx} é a potência do sinal transmitido, G_{Tx} e G_{Rx} são os ganhos das antenas em dBi e PL_{650m} é a perda por propagação na borda da célula que está a uma distância de 650 metros.

A antena da estação rádio-base é diretiva e o lobo principal tem ângulo 1/2 potência de 7° , conforme o diagrama de irradiação no plano horizontal ilustrado na Figura 5.19. A pior condição de apontamento entre a estação rádio-base e o receptor ocorre quando o lobo principal da antena tem a máxima atenuação.

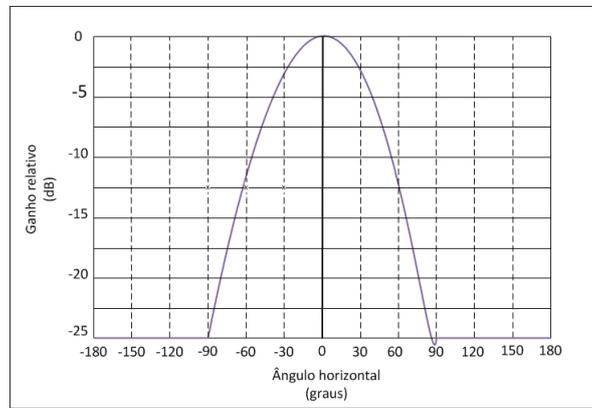


Figura 5.19: Padrão de irradiação no plano horizontal

Para se calcular a atenuação da antena diretiva, o padrão horizontal é regido pela equação (5.28), na qual $A(\theta)$ é o ganho relativo da antena na direção dos ângulos, A_m é o valor máximo de atenuação (25dB), θ_{3dB} é o ângulo de 1/2 potência, sendo que $-180^\circ \leq \theta \leq 180^\circ$ e $\min(\cdot)$ é o menor entre os valores.

$$A(\theta) = -\min \left(12 \left(\frac{\theta}{\theta_{3dB}} \right)^2, A_m \right) \quad (5.28)$$

Logo, para se determinar o ganho mínimo da antena do receptor reordena-se a equação 5.27 resulta na inequação (5.29), levando-se em consideração a atenuação angular da antena da rádio-base.

$$G_t \geq \text{Sensibilidade}_{mínima} - [P_{Tx} + G_r + \sigma - A_{t\theta} - PL_{650m}] \quad (5.29)$$

Para garantir com 99% de probabilidade que o sinal seja demodulado por qualquer rádio receptor, leva-se em consideração a maior distância entre nó e rádio-

base, o máximo desvio por *fading* ($3\sigma=24\text{dB}$) e a máxima atenuação da antena. Obtém-se então que para o modelo de propagação SUI, o ganho mínimo deveria ser de 3 dBi. Por segurança, arbitrou-se um ganho de 7dBi. Se fosse adotado o modelo do 3GPP, o ganho mínimo necessário seria de 31 dBi, que é um valor elevado demais para uma típica antena *omni*. É possível utilizar antena com ganho de diretividade com menor valor, porém o sinal pode não ter potência suficiente para ser demodulado. Estatisticamente um pacote pode alcançar a estação rádio-base, porém no instante seguinte, dada que a potência pode flutuar e estar abaixo do limiar de sensibilidade, poderá haver retransmissão do pacote até que a mensagem não possa ser mais enviada.

5.9.5 Resultados

Foram feitas simulações utilizando-se os dois modelos de propagação:

- Análise do tempo de reconfiguração da rede sem o tráfego dos coletores;
- Análise do tempo de reconfiguração do algoritmo com o tráfego dos coletores de dados de medidores inteligentes;

Para se obter maior fidelidade e confiabilidade aos resultados obtidos, cada cenário foi executado 100 vezes com diferentes sementes para as variáveis aleatórias do simulador. Dessa forma, determinou-se o intervalo de confiança representado nos gráficos por barras verticais.

5.9.5.1 Cenário com modelo de propagação SUI

Foram feitas duas simulações considerando-se o modelo de propagação e um ganho de antena do receptor de 7dBi. No cenário com a adição do tráfego dos coletores de dados, a Tabela 5.14 ilustra o tempo de reconfiguração com o intervalo de confiança de 95%, assim como o percentual de finalização do algoritmo.

Taxa (Bytes/seg)	Tempo médio de reconfig. (seg)	Intervalo de confiança (95%)	% Finalização
1854	66,57	$\pm 1,80$	100
1236	66,90	$\pm 0,18$	100
927	67,79	$\pm 0,07$	100
618	66,64	$\pm 0,11$	100
412	66,70	$\pm 0,10$	100
309	67,00	$\pm 0,14$	100
206	66,93	$\pm 2,06$	100
155	66,92	$\pm 2,01$	100
103	67,70	$\pm 1,48$	100
52	66,67	$\pm 0,11$	100

Tabela 5.14: Tabela de simulação LTE com tráfego dos coletores

O resultado das simulações, no cenário sem a presença dos coletores, está representado na Tabela 5.15, em que \bar{T} é o tempo médio de reconfiguração da rede de distribuição de energia elétrica, $T_{\text{máx}}$ e $T_{\text{mín}}$ são os tempos máximos e mínimos encontrados após 100 rodadas, respectivamente, σ é o desvio-padrão, IC é o intervalo de confiança a 95%, P_f é o percentual de finalização do algoritmo.

\bar{T}	48 seg
$T_{\text{máx}}$	141 seg
$T_{\text{mín}}$	66 seg
σ	7,48 seg
IC	1,47 seg
P_f	100%

Tabela 5.15: Tabela de simulação LTE sem coletores de tráfego

O resultado gráfico está ilustrado na Figura 5.20. As variações dos intervalos de confiança observadas em alguns pontos do gráfico podem ser explicadas pela inexistência de sincronismo dos escalonadores das subportadoras entre as estações rádio-bases do modelo computacional do LTE. Isto significa que, estatisticamente, há uma probabilidade de que a alocação de blocos de recursos e *slots* seja coincidente entre as estações rádio-bases e, conseqüentemente, gerar interferência e requisitar retransmissão de pacotes.

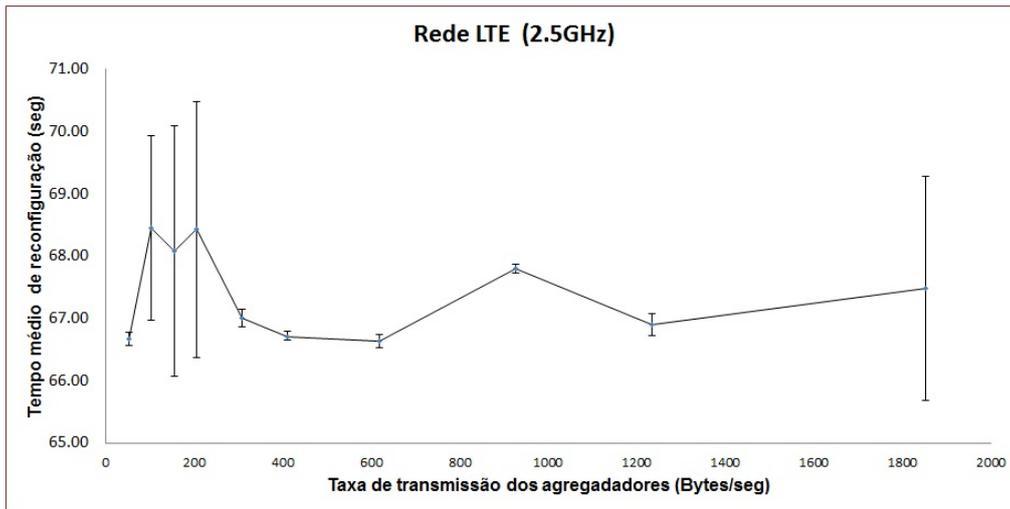


Figura 5.20: Tempos de reconfiguração da rede elétrica com tecnologia LTE

Da Figura 5.20 e da Tabela 5.14 depreende-se que que o compartilhamento do tráfego dos religadores não é influenciado pelo tráfego dos coletores, fato que já se esperava pois a tecnologia de rádio suporta tráfegos superiores a 100 Mbps.

5.10 Conclusão

Dentre as diversas aplicações existentes no âmbito das redes elétricas inteligentes, optou-se por estudar o cenário de reconfiguração de redes de distribuição de energia elétrica e a criação de um algoritmo distribuído baseado em sistemas multiagentes. Foi criada uma estrutura hierárquica híbrida de tal forma que a solicitação de uma informação passa por um agente intermediário. Existem diversas estratégias heurísticas, meta-heurísticas propostas em estudos para resolução de problemas de reconfiguração, porém implementou-se uma na qual é realizado o cálculo de grafos direcionados e não-direcionados e a descoberta de possíveis rotas entre a falha e o ramo desconexo. Essas rotas são trocadas entre os agentes e a cada passo é calculado o fluxo de potência da rede elétrica observada as restrições de tensão nas linhas de distribuição e a corrente máxima admissível. Se o fluxo atende, então uma solução é encontrada; caso contrário folhas da árvore do grafo são determinadas e a menor carga associada é cortada. O algoritmo executa até que encontre uma topologia que atenda a todos os requisitos ou então finaliza sem encontrar. Os agentes enviam as mensagens que passam pelo RTI e simulados como se fossem pacotes HTTP na rede de telecomunicações. Dessa forma, é possível analisar aspectos da rede de telecomunicações, tais como latência e perda de pacotes, que possam influenciar nos indicadores de desempenho da cossimulação.

Analisando-se os três cenários é possível verificar que uma das dificuldades para se chegar a um resultado é a correta definição do modelo de propagação a ser adotado.

Nesse caso, embora os modelos produzam atenuações diferentes, a seleção depende de uma verificação em campo para identificar aquele que mais se aproxima da realidade.

Para as cossimulações sem a presença do tráfego dos coletores de dados, a tecnologia ponto-multiponto é a que apresentou o menor tempo médio de reconfiguração, conforme ilustrado na Tabela 5.16.

Tecnologia	Tempo médio de reconfig. (seg)	% Finalização
LTE	67,48	100
Malha	104,48	100
Ponto-multiponto	40,62	99

Tabela 5.16: Tabela comparativa sem coletores de dados

Comparando-se os resultados das cossimulações dos três cenários, conforme ilustrado na Figura 5.21 observa-se que a tecnologia de rádio ponto-multiponto oferece o menor tempo médio de reconfiguração. No entanto, a tecnologia LTE é a que oferece a menor variação do tempo de reconfiguração, já que o escalonamento para envio de pacote é feito por meio de oferta de *slots* de tempo de ocupação das subportadoras OFDMA. Por outro lado, a rede em malha, pelo fato que o pacote necessita passar por múltiplos saltos até chegar ao seu destino e pelo padrão IEEE 802.11 possuir o mecanismo que previne a colisão de pacotes através de sensoreamento da interface aérea, resulta em uma latência maior para enviar pacotes.

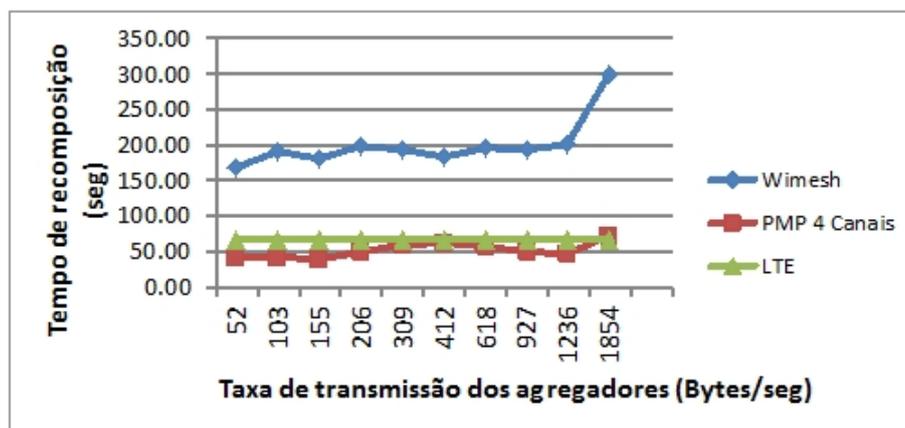


Figura 5.21: Comparação dos tempos de reconfiguração

Na rede ponto-multiponto uma pequena quantidade de cossimulações não foi finalizada, conforme indicado na Tabela 5.17, ainda que o protocolo TCP retransmita os pacotes para o nó de destino. Uma potencial causa a ser investigada é a ocorrência de colisão de pacotes que resultaram em sucessivas retransmissões sem sucesso até que o tempo expirasse.

Taxas dos coletores de dados (Bytes/seg)	LTE	Malha	Ponto multiponto
52	100%	100%	98%
103	100%	100%	99%
155	100%	99%	98%
206	100%	100%	95%
309	100%	100%	100%
412	100%	100%	98%
618	100%	100%	97%
927	100%	100%	100%
1236	100%	100%	99%
1854	100%	100%	100%

Tabela 5.17: Percentual de finalização das simulações considerando-se tráfego dos coletores de dados

É visível a influência do compartilhamento do tráfego de dados dos coletores à medida que o aumento da taxa implica em uma limitação na capacidade de fluir o tráfego de automação dos religadores, devido a maior competição pelo meio de acesso. Essa competição é maior na rede em malha devido à quantidade de saltos entre os nós para que a mensagem alcance o seu destino, o fato de se estar operando com um único canal e pelo protocolo de roteamento que contribui com mais tráfego. Para se implantar uma rede de telecomunicações em malha para cobrir uma grande área geográfica é de difícil implementação já que existem obstáculos físicos em um cenário real que podem interferir nas conexões entre os nós. Dos resultados das cossimulações, observa-se que a rede em malha não é escalável e à medida que o número de nós aumenta, maior é a latência entre pacotes e menor a taxa de transmissão. Para se criar uma rede tipo apenas para atender a uma ou duas aplicações de redes elétricas inteligentes pode não ser economicamente viável.

Uma rede ponto-multiponto, embora apresente melhores resultados com relação ao aspecto da latência quando comparado com a rede em malha, apresenta desvantagens como a questão da limitação de escala, isto é, à medida que mais nós são atendidos pela mesma cobertura de uma rádio-base, maior é a quantidade de colisões o que pode inviabilizar o tráfego de outras aplicações. Para dar melhor cobertura e permitir um aumento de tráfego, a alternativa é reduzir o tamanho da cobertura da célula e aumentar a quantidade delas. Mais uma vez, é preciso uma avaliação de viabilidade econômica de um possível projeto desse tipo de rede para apenas passar uma pequena quantidade de aplicações. Os dois tipos de rede possuem ainda outro fator mais crítico que é o nível de interferência nas faixas de frequências em 900 MHz e 2.4 GHz. A interferência externa não foi levada em

consideração nos modelos computacionais e podem levar a resultados totalmente diferentes. Finalmente, entre as três a tecnologia LTE parece ser a mais promissora em termos de capacidade e disponibilidade. No entanto, há que se entender que esse tipo de tecnologia somente está disponível em empresas operadoras de serviço de telecomunicações que detém a licença do espectro. Dessa forma, a única forma de se utilizá-la é através de contratação de serviço de transmissão de dados que pode ser uma boa alternativa econômica. Nesse caso é preciso avaliar também a disponibilidade do serviço e se é adequado para a aplicação que se deseja utilizar. Convém avaliar outras alternativas como outro tipo de arquitetura, em malha, porém com múltiplos canais, avaliar o uso da tecnologia Wimax, adotar outro protocolo de roteamento como o RPL [98] ou construir uma arquitetura híbrida com tecnologias, tais como LTE, malha e ponto-multiponto.

Capítulo 6

Conclusões e Próximos Passos

A tese optou pela arquitetura HLA pelo fato de permitir um fraco acoplamento entre simuladores, por ser um padrão IEEE e por dispor de ferramentas de desenvolvimento de domínio público. Nesse contexto, foram desenvolvidas interfaces para os simuladores que se integram com o RTI. Embora não seja uma proposta nova, a arquitetura é bastante complexa e o desenvolvimento requer conhecimento dos passos necessários para a ativação e envio dos dados que foram descritos no capítulo 4.

Para o desenvolvimento da integração, foi criado primeiramente uma estrutura de modelo de dados comuns a todos os simuladores. Essa estrutura é baseada em uma ontologia que representa um agrupamento de objetos de um determinado domínio de conhecimento. Para isso, utilizou-se a maior parte do modelo de objetos que compõem uma rede elétrica elaborada pela norma IEC 61970. Posteriormente, a mesma foi estendida para acomodar novos objetos que poderão ser utilizados em simulações de redes elétricas inteligentes. Essa estrutura do modelo permite que todas os atributos dos objetos e interações sejam trocados por intermédio do paradigma “publicar/assinar”. Assim, os simuladores que precisam receber informações sobre certos objetos solicitam assinatura e, a cada publicação, são notificados pelo RTI para leitura do objeto e seus atributos.

A tese avaliou três diferentes redes de telecomunicações: duas baseadas no padrão IEEE 802.11 e outra no LTE. O primeiro cenário com rede sem-fio em malha, apresentou desempenho inferior comparado aos demais, em termos de tempo médio de reconfiguração da rede de distribuição de energia elétrica, devido ao número de mensagens de rotas divulgadas pelo protocolo de roteamento e pela quantidade de saltos.

O segundo cenário, composto com a rede ponto-multiponto, apresentou um desempenho melhor, porém demonstrou que o tráfego de dados compartilhado, mesmo com um número maior de canais influencia nos resultados à medida que a taxa dos coletores de dados dos medidores inteligentes aumenta. Em todas as

ocasiões observou-se um baixo percentual de não finalização devido à possíveis retransmissões. Para se aumentar a capacidade da rede ponto-multiponto existem duas alternativas: aumentar o número de rádio-bases e, conseqüentemente, reduzir a área de cobertura e utilizar outro modelo de rádio como o IEEE 802.11g. Nesse segundo caso, somente um único canal pode ser acomodado em torno de 900MHz (banda ISM) . Dessa forma, poderá ocorrer maior interferência cocanal e, portanto, redução do desempenho com aumento do número de retransmissões, colisões e um possível aumento percentual de não finalização do algoritmo.

O terceiro cenário, composto com a rede LTE, apresenta bom desempenho sob a ótica da métrica de percentual de finalização, mesmo com tráfego competindo com o de reconfiguração da rede de distribuição de energia elétrica. Era de se esperar que não houvesse grandes variações, já que a capacidade de tráfego do rádio é superior ao tráfego gerado. Porém, há impacto nos resultados quando se utiliza o modelo de propagação 3GPP. O cálculo do ganho das antenas dos receptores para garantir cobertura em 100% do tempo, leva a um valor que na prática não é possível de se ter para antenas omnidirecionais. Por outro lado, quando se utilizou o modelo de propagação SUI, obteve-se um valor bastante satisfatório e um resultado da cossimulação adequado. Para mitigar o problema da seleção mais adequada do modelo de propagação recomenda-se um levantamento em campo para verificar qual é aquele que mais se aproxima dos modelos existentes e fazer as necessárias adequações. O modelo computacional desenvolvido para a tecnologia LTE carece de melhorias, pois não leva em consideração o as mensagens de controle e sincronismo entre as estações rádio-bases e não implementa mecanismo os diferentes tipos de filas de QoS. Isso explica os eventuais intervalos de confiança com valores maiores do que os outros observados. Verificou-se que entre os três tipos, a LTE é a que permite uma convergência mais rápida do algoritmo.

Das três topologias apresentadas a menos recomendada é a rede em malha com um gateway para a Internet e com um canal. Isso se deve à dificuldade de implementação real, já que há necessidade da presença de nós de repetição devido à baixa cobertura de cada estação. Observa-se o aumento da latência com a presença de outro tráfego de dados compartilhado.

A existência de um agente que centraliza todos os dados e outro para cálculo de fluxo de carga não parece ser adequado para uma topologia de rede com múltiplos saltos, já que nós muito afastados demandam um longo tempo de resposta. É preciso então propor melhorias no algoritmo e distribuir mais ações para tentar mitigar este problema.

As duas outras tecnologias se mostraram bastante eficientes em termos de tempo de reconfiguração, porém a rede ponto-multiponto ainda apresentou algumas condições nas quais os resultados não foram satisfatórios, principalmente pelo fato

de que algumas rádio-bases não dispõem de cobertura de sinal em 100% do tempo para todos os receptores e pelo fato de que potência excessiva poderia causar maior interferência sobre as rádio-bases. Essa é uma solução que pode ser implantada pelas distribuidoras de energia elétrica, adequando-se a área de cobertura e aumentando o número de rádio-bases.

O modelo LTE se mostrou estável e insensível com a introdução de tráfego compartilhado. No entanto, é uma solução tipicamente de operadora celular, ou seja, a concessionária terá que contratar serviço de transmissão de dados para cada nó da rede e para cada agregador de dados existente. Na verdade, as duas tecnologias se complementam já que o LTE pode ser adequado em algumas áreas urbanas, onde a tecnologia já está mais estabelecida, enquanto que o ponto-multiponto pode ser mais adequado em áreas suburbanas ou mesmo rurais pelo fato de permitir maior cobertura.

A escolha do tipo de arquitetura e tecnologia a ser adotada depende, entre outros fatores, da quantidade de serviços que serão ofertados, da disponibilidade, da qualidade e cobertura da última milha. A viabilidade, no entanto, depende também de uma análise custo-benefício e planejamento para se definir a rentabilidade e a estratégia a ser adotada entre a criação de uma infraestrutura própria ou contratação de serviços de terceiros.

Para os próximos passos é preciso melhorar os modelos computacionais disponíveis, possivelmente adotando outra ferramenta de simulação como o NS-3 ou Opnet Modeler, que possui modelos mais refinados, e comparar com os resultados obtidos até aqui. O modelo de um único canal para a rede em malha deve ser alterado para o modelo 802.11a operando em 5GHz, com 8 canais não sobrepostos, operando como *backhaul* dos coletores de dados que podem utilizar a frequência de 2.4GHz. No entanto, outro desafio se delineia com o aumento da frequência e o consequente aumento da atenuação do sinal, ensejando um possível aumento do número de nós da rede.

Os atuais modelos computacionais não tiram proveito do paralelismo da máquina na qual são executados. À medida que o número de nós da rede de telecomunicações aumenta, mais recursos de processamento e memória são necessários. É preciso desenvolver modelos que permitam a distribuição de carga nos múltiplos processadores. Todos os simuladores utilizados dispõem dessa funcionalidade e o seu uso é capaz de acelerar as simulações.

Outra proposta é desenvolver outro modelo computacional de rádio baseado em salto de frequência na faixa de 900 MHz, que tem sido adotado pelas concessionárias no caso da criação de uma infraestrutura própria para leitura de medidores e para reconfiguração de rede de distribuição de energia elétrica.

Por fim, uma proposta importante para acelerar o desenvolvimento da integração,

é a criação de uma aplicação que, a partir de um modelo em alto nível chamado *Model Driven Architecture* (MDA), gere todos os códigos de interface para diferentes simuladores em linguagens de programação como Matlab, Java ou C++. Isso proporciona um ganho de produtividade além de permitir que se foque no desenvolvimento dos modelos de cada domínio e nas aplicações dentro do contexto de redes elétricas inteligentes.

Referências Bibliográficas

- [1] AMIN, S. “For the Good of the Grid”, *Power and Energy Magazine, IEEE*, v. 6, n. 6, pp. 48–59, November 2008.
- [2] COMMISSION, E. Disponível em: <http://ec.europa.eu/energy/renewables/action_plan_en.htm>. Acessado em: 28 jun. 2014.
- [3] MME, 2015. Disponível em: <<http://www.mme.gov.br/documents/1138787/1732840/Resenha+Ener%C3%A9tica+-+Brasil+2015.pdf/4e6b9a34-6b2e-48fa-9ef8-dc7008470bf2>>. Acessado em: 04 jan. 2016.
- [4] SURYANARAYANAN, S., RIBEIRO, P. F., SIMÕES, M. G. “Grid modernization efforts in the USA and Brazil—some common lessons based on the smart grid initiative”. In: *IEEE Power and Energy Society General Meeting*, pp. 1–5, 2010.
- [5] NIST. “NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 2.0”. 2012. Disponível em: <http://www.nist.gov/smartgrid/upload/NIST_Framework_Release_2-0_corr.pdf>. Acessado em: 05 ago. 2015.
- [6] HERNANDEZ, L., BALADRON, C., AGUIAR, J., et al. “A multi-agent system architecture for smart grid management and forecasting of energy demand in virtual power plants”, *Communications Magazine, IEEE*, v. 51, n. 1, pp. 106–113, January 2013.
- [7] GOMES, F. V., CARNEIRO JR, S., PEREIRA, J. L. R., et al. “A new heuristic reconfiguration algorithm for large distribution systems”, *Power Systems, IEEE Transactions on*, v. 20, n. 3, pp. 1373–1378, 2005.
- [8] DCDERMOTT, T., DREZGA, I., BROADWATER, R. “A heuristic nonlinear constructive method for distribution system reconfiguration”, *Power Systems, IEEE Transactions on*, v. 14, n. 2, pp. 478–483, 1999.

- [9] NAGATA, T., SASAKI, H., YOKOYAMA, R. “Power system restoration by joint usage of expert system and mathematical programming approach”, *Power Systems, IEEE Transactions on*, v. 10, n. 3, pp. 1473–1479, 1995.
- [10] NUNNA, K., DOOLLA, S. “Responsive End-User-Based Demand Side Management in Multimicrogrid Environment”, *Industrial Informatics, IEEE Transactions on*, v. 10, n. 2, pp. 1262–1272, May 2014.
- [11] HERNANDEZ, L., BALADRON, C., AGUIAR, J., et al. “A multi-agent system architecture for smart grid management and forecasting of energy demand in virtual power plants”, *Communications Magazine, IEEE*, v. 51, n. 1, pp. 106–113, January 2013.
- [12] REN, F., ZHANG, M., SUTANTO, D. “A multi-agent solution to distribution system management by considering distributed generators”, *Power Systems, IEEE Transactions on*, v. 28, n. 2, pp. 1442–1451, 2013.
- [13] KARFOPOULOS, E. L., HATZIARGYRIOU, N. D. “A multi-agent system for controlled charging of a large population of electric vehicles”, *Power Systems, IEEE Transactions on*, v. 28, n. 2, pp. 1196–1204, 2013.
- [14] LEE, E. A. “Cyber physical systems: Design challenges”. In: *Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on*, pp. 363–369, 2008.
- [15] WIETFELD, C., GEORG, H., GRONING, S., et al. “Wireless M2M communication networks for smart grid applications”. In: *Wireless Conference 2011-Sustainable Wireless Technologies (European Wireless), 11th European*, pp. 1–7. VDE, 2011.
- [16] CONSUMER, S. G. “ITU-T STUDY GROUP 20 ON IOT AND ITS APPLICATIONS, INCLUDING SMART CITIES AND COMMUNITIES”, *IEEE Communications Magazine*, v. 5, 2015.
- [17] ELMANGOUSH, A., COSKUN, H., WAHLE, S., et al. “Design aspects for a reference M2M communication platform for Smart Cities”. In: *Innovations in Information Technology (IIT), 2013 9th International Conference on*, pp. 204–209, March 2013.
- [18] ALAM, M., NIELSEN, R., PRASAD, N. “The evolution of M2M into IoT”. In: *Communications and Networking (BlackSeaCom), 2013 First International Black Sea Conference on*, pp. 112–115, July 2013.

- [19] IEEE. “IEEE Standard for Modeling and Simulation (M & S) High Level Architecture (HLA)– Federate Interface Specification”, *IEEE Std 1516.1-2010 (Revision of IEEE Std 1516.1-2000)*, pp. 1–378, Aug 2010.
- [20] MATLAB, . Disponível em: <<http://www.mathworks.com/products/matlab/>>. Acessado em: 18 jun. 2016.
- [21] MOBILE, T. I. “Java Agent Development Framework”. Disponível em: <<http://jade.tilab.com/>>. Acessado em: 30 mar. 2014.
- [22] IEEE. “The Foundation for Intelligent Physical Agents”. . Disponível em: <<http://www.fipa.org/>>. Acessado em: 30 mar. 2014.
- [23] ALBAGLI, A. N., FALCÃO, D. M., DE REZENDE, J. F. “Smart grid framework co-simulation using HLA architecture”, *Electric Power Systems Research*, v. 130, pp. 22–33, 2016.
- [24] LI, W., ZHANG, X. “Simulation of the smart grid communications: Challenges, techniques, and future trends”, *Computers & Electrical Engineering*, v. 40, n. 1, pp. 270–288, 2014.
- [25] HOPKINSON, K., WANG, X., GIOVANINI, R., et al. “EPOCHS: a platform for agent-based electric power and communication simulation built from commercial off-the-shelf components”, *Power Systems, IEEE Transactions on*, v. 21, n. 2, pp. 548–558, 2006.
- [26] SHUM, C., LAU, W., LAM, K. L., et al. “The development of a smart grid co-simulation platform and case study on Vehicle-to-Grid voltage support application”. In: *Smart Grid Communications (SmartGridComm), 2013 IEEE International Conference on*, pp. 594–599, 2013.
- [27] IEEE, . Disponível em: <<http://www.feis.unesp.br/#!/departamentos/engenharia-eletrica/pesquisas-e-projetos/lapsee/downloads/materiais-de-cursos1193/>>. Acessado em: 18 jun. 2016.
- [28] PERKONIGG, F., BRUJIC, D., RISTIC, M. “MAC-Sim: A multi-agent and communication network simulation platform for smart grid applications based on established technologies”. In: *Smart Grid Communications (SmartGridComm), 2013 IEEE International Conference on*, pp. 570–575, 2013.
- [29] IEEE, . Disponível em: <<http://sys.elec.kitami-it.ac.jp/ueda/demo/WebPF/39-New-England.pdf>>. Acessado em: 28 jan. 2016.

- [30] LI, W., LUO, M., ZHU, L., et al. “A co-simulation method as an enabler for joint analysis and design of MAS-based electrical power protection and communication”, *Simulation*, v. 89, n. 7, pp. 790–809, 2013.
- [31] VTB. Disponível em: <<http://vtb.engr.sc.edu/vtbwebsite/#/Overview>>. Acessado em: 30 mar. 2014.
- [32] LIBERATORE, V., AL-HAMMOURI, A. “Smart grid communication and co-simulation”. In: *Energytech, 2011 IEEE*, pp. 1–5, 2011.
- [33] MODELICA. Disponível em: <<http://www.openmodelica.org>>. Acessado em: 29 jan. 2016.
- [34] NASIRIANI, N., RAMACHANDRAN, R., FALLAH, K. R. Y. P., et al. “An embedded communication network simulator for power systems simulations in PSCAD”. In: *Power and Energy Society General Meeting (PES), 2013 IEEE*, pp. 1–5. IEEE, 2013.
- [35] PSCAD. Disponível em: <<https://hvdc.ca/pscad/>>. Acessado em: 28 jan. 2016.
- [36] 57, I. T. C. Disponível em: <<http://www.dnp.org/pages/aboutdefault.aspx>>. Acessado em: 18 jun. 2016.
- [37] MCDONALD, J. D. “Developing and defining basic SCADA system concepts”. In: *Rural Electric Power Conference, 1993. Papers Presented at the 37th Annual Conference*, pp. B3–1. IEEE, 1993.
- [38] IEEE, 2016. Disponível em: <<http://ewh.ieee.org/soc/pes/dsacom/testfeeders/>>. Acessado em: 15 fev. 2016.
- [39] LIN, H., VEDA, S., SHUKLA, S., et al. “GECO: Global Event-Driven Co-Simulation Framework for Interconnected Power System and Communication Network”, *Smart Grid, IEEE Transactions on*, v. 3, n. 3, pp. 1444–1456, Sept 2012.
- [40] GEORG, H., MULLER, S., DORSCH, N., et al. “INSPIRE: Integrated co-simulation of power and ICT systems for real-time evaluation”. In: *Smart Grid Communications (SmartGridComm), 2013 IEEE International Conference on*, pp. 576–581, Oct 2013.
- [41] IEC, 2016. Disponível em: <<http://www.iec.ch/smartgrid/standards/>>. Acessado em: 29 jan. 2016.

- [42] MULLER, S. C., GEORG, H., REHTANZ, C., et al. “Hybrid simulation of power systems and ICT for real-time applications”. In: *Innovative Smart Grid Technologies (ISGT Europe), 2012 3rd IEEE PES International Conference and Exhibition on*, pp. 1–7, 2012.
- [43] BABAZADEH, D., CHENINE, M., ZHU, K., et al. “A platform for wide area monitoring and control system ICT analysis and development”. In: *PowerTech (POWERTECH), 2013 IEEE Grenoble*, pp. 1–7. IEEE, 2013.
- [44] TECHNOLOGIES, O.-R. Disponível em: <<http://www.opal-rt.com/>>. Acessado em: 15 fev. 2016.
- [45] OPENPDC. Disponível em: <<http://openpdc.codeplex.com/>>. Acessado em: 15 fev. 2016.
- [46] FUJIMOTO, R. M. *Parallel and distributed simulation systems*, v. 300. 1 ed. New York, Wiley New York, 2000.
- [47] FISCHER, M. C. “Aggregate level simulation protocol (ALSP) managing confederation development”. In: *Simulation Conference Proceedings, 1994. Winter*, pp. 775–780, Dec 1994. doi: 10.1109/WSC.1994.717432.
- [48] POWELL, E. T., NOSEWORTHY, J. R. *Engineering Principles of Combat Modeling and Distributed Simulation*. New Jersey, John Wiley & Sons, Inc., 2012. ISBN: 9781118180310. doi: 10.1002/9781118180310.ch20. Disponível em: <<http://dx.doi.org/10.1002/9781118180310.ch20>>.
- [49] IEEE. “IEEE Standard for Modeling and Simulation (M & S) High Level Architecture (HLA)– Object Model Template (OMT) Specification”, *IEEE Std 1516.2-2010 (Revision of IEEE Std 1516.2-2000)*, pp. 1–110, Aug 2010.
- [50] DAHMANN, J. S., FUJIMOTO, R. M., WEATHERLY, R. M. “The department of defense high level architecture”. In: *Proceedings of the 29th conference on Winter simulation*, pp. 142–149. IEEE Computer Society, 1997.
- [51] FUJIMOTO, R. M. “Time management in the high level architecture”, *Simulation*, v. 71, n. 6, pp. 388–400, 1998.
- [52] TECHNOLOGIES, P., 2016. Disponível em: <<http://www.pitchtechnologies.com/>>. Acessado em: 18 jun. 2016.
- [53] M’AK, V., 2016. Disponível em: <<http://www.mak.com/>>. Acessado em: 18 jun. 2016.

- [54] SIMWARE, 2016. Disponível em: <<http://www.simware.es/>>. Acessado em: 18 jun. 2016.
- [55] RAYTHEON, 2016. Disponível em: <http://www.raytheon.com/news/technology_today/2015_i1/eye_on_tech.html>. Acessado em: 18 jun. 2016.
- [56] OPENRTI. Disponível em: <<http://sourceforge.net/projects/openrti>>. Acessado em: 30 mar. 2014.
- [57] GERTICO. Disponível em: <<http://www.iosb.fraunhofer.de/servlet/is/18067/>>. Acessado em: 30 mar. 2014.
- [58] OPENHLA. Disponível em: <<http://sourceforge.net/projects/ohla/>>. Acessado em: 30 mar. 2014.
- [59] “poRTIco”. Disponível em: <<http://www.portico.org>>. Acessado em: 17 abr. 2014.
- [60] CERTI. Disponível em: <<http://savannah.nongnu.org/projects/certi/>>. Acessado em: 30 mar. 2014.
- [61] OMNET. Disponível em: <<http://inet.omnetpp.org/>>. Acessado em: 02 fev. 2016.
- [62] IEEE. “IEEE Recommended Practice for Distributed Simulation Engineering and Execution Process”, *IEEE Std 1730-2010 (Revision of IEEE Std 1516.3-2003)*, pp. 1–79, Jan 2011.
- [63] GÓMEZ-PÉREZ, A. “Ontological engineering: A state of the art”, *Expert Update: Knowledge Based Systems and Applied Artificial Intelligence*, v. 2, n. 3, pp. 33–43, 1999.
- [64] GRUBER, T. R. “A translation approach to portable ontology specifications”, *Knowledge acquisition*, v. 5, n. 2, pp. 199–220, 1993.
- [65] GUARINO, N. *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy*, v. 46. Italy, IOS press, 1998.
- [66] NOY, N. F., MCGUINNESS, D. L., OTHERS. “Ontology development 101: A guide to creating your first ontology”. 2001. Disponível em: <<http://www.bibsonomy.org/bibtex/2f1254205d255e390b6bb45dc0e234238/sustymenko>>.

- [67] UNIVERSITY, S. “Protégé”. Disponível em: <<http://protege.stanford.edu/>>. acessado em 15 de abr 2014.
- [68] IEC. Disponível em: <<http://www.iec.ch/smartgrid/standards/>>. Acessado em: 02 fev. 2016.
- [69] BELLIFEMINE, F. L., CAIRE, G., GREENWOOD, D. *Developing Multi-Agent Systems with JADE(Wiley Series in Agent Technology)*. Sussex, John Wiley & Sons, 2007.
- [70] “SimGe”. Disponível em: <<http://www.ceng.metu.edu.tr/~otopcu/simge/>>. acessado em 12 abr. 2014.
- [71] MATLAB, . Disponível em: <http://www.mathworks.com/help/pdf_doc/simulink/sfunctions.pdf>. Acessado em: 02 fev. 2016.
- [72] MIU, K. N., CHIANG, H.-D., YUAN, B., et al. “Fast service restoration for large-scale distribution systems with priority customers and constraints”. In: *Power Industry Computer Applications., 1997. 20th International Conference on*, pp. 3–9, May 1997. doi: 10.1109/PICA.1997.599366.
- [73] NAGATA, T., SASAKI, H. “A multi-agent approach to power system restoration”, *Power Systems, IEEE Transactions on*, v. 17, n. 2, pp. 457–462, 2002.
- [74] SOLANKI, J. M., KHUSHALANI, S., SCHULZ, N. N. “A multi-agent solution to distribution systems restoration”, *Power Systems, IEEE Transactions on*, v. 22, n. 3, pp. 1026–1034, 2007.
- [75] KOVAC, M., BRACINIK, P., HOGER, M., et al. “Autonomous power restoration of medium voltage distribution network”. In: *ELEKTRO, 2012*, pp. 212–215, May 2012.
- [76] BARAN, M. E., WU, F. F. “Network reconfiguration in distribution systems for loss reduction and load balancing”, *Power Delivery, IEEE Transactions on*, v. 4, n. 2, pp. 1401–1407, 1989.
- [77] EPE, 2015. Disponível em: <<http://www.epe.gov.br/mercado/Documents/S%C3%A9rie%20Estudos%20de%20Energia/DEA%2001%20-%20Gera%C3%A7%C3%A3o%20Distribu%C3%ADa%20no%20Hor%C3%A1rio%20de%20Ponta.pdf>>. Acessado em: 23 mar. 2016.
- [78] CORMEN, T. H. *Introduction to algorithms*. Cambridge, MIT press, 2009.

- [79] GEORIO, 2016. Disponível em: <[http://portalgeo.rio.rj.gov.br/indice/flanali.asp?codpal=697&pal=FREGUESIA\(ILHADOGOVERNADOR\)-bairro](http://portalgeo.rio.rj.gov.br/indice/flanali.asp?codpal=697&pal=FREGUESIA(ILHADOGOVERNADOR)-bairro)>. Acessado em: 20 fev. 2016.
- [80] KUZLU, M., PIPATTANASOMPORN, M., RAHMAN, S. “Communication network requirements for major smart grid applications in HAN, {NAN} and {WAN}”, *Computer Networks*, v. 67, pp. 74 – 88, 2014. ISSN: 1389-1286.
- [81] GROUP, O. S. G. U. Disponível em: <http://osgug.ucaiuug.org/UtiliComm/Shared%20Documents/Latest_Release_Deliverables/SG%20Network%20SRS%20Version%20V5%20Final.pdf>. Acessado em: 21 fev. 2016.
- [82] LUAN, W., SHARP, D., LANCASHIRE, S. “Smart grid communication network capacity planning for power utilities”. In: *Transmission and Distribution Conference and Exposition, 2010 IEEE PES*, pp. 1–4, April 2010.
- [83] PLAN, N. P. A. “2-Guidelines for Assessing Wireless Standards for Smart Grid Applications”, *National Institute of Standards and Technology Std*, 2011.
- [84] GUPTA, P., KUMAR, P. R. “The capacity of wireless networks”, *IEEE Transactions on Information Theory*, v. 46, n. 2, pp. 388–404, Mar 2000.
- [85] GASTPAR, M., VETTERLI, M. “On the capacity of wireless networks: the relay case”. In: *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, v. 3, pp. 1577–1586 vol.3, 2002.
- [86] JUN, J., SICHITIU, M. L. “The nominal capacity of wireless mesh networks”, *Wireless Communications, IEEE*, v. 10, n. 5, pp. 8–14, 2003.
- [87] MISHRA, A. R. *Fundamentals of cellular network planning and optimisation: 2G/2.5 G/3G ... evolution to 4G*. Sussex, John Wiley & Sons, 2004.
- [88] TRANTER, W., SHANMUGAN, K., RAPPAPORT, T., et al. *Principles of Communication Systems Simulation with Wireless Applications*. First ed. Upper Saddle River, NJ, USA, Prentice Hall Press, 2003. ISBN: 0134947908.
- [89] GUPTA, A. K., SADAWARTI, H., VERMA, A. K. “Implementation of DYMO routing protocol”, *arXiv preprint arXiv:1306.1338*, 2013.

- [90] OUNI, A., RIVANO, H., VALOIS, F. “Capacity of wireless mesh networks: determining elements and insensible characters.” In: *PlanNet 2010, International Workshop on Planning and Optimization of Wireless Communication Networks (IEEE WCNC2010 Workshop)*, 2010.
- [91] Disponível em: http://www.orbitel.com.br/produtos/wireless/serie-954.html?gclid=CjwKEAjwtL07BRDax4-I4_6G71USJAA6FjN10jhFKzLQ5WCxd4w7Xol-UDv8b8cW2aMnAkVgKLhiCBoCpcf_wcB>. Acessado em: 18 jun. 2016.
- [92] Disponível em: http://www.alcon.tw/apro_900.htm>. Acessado em: 18 jun. 2016.
- [93] DE TELECOMUNICAÇÕES, A. N. “Resolução N° 544”. 2010. Disponível em: http://www.anatel.gov.br/Portal/documentos/sala_imprensa/16-8-2010--14h39min37s-Res544.pdf>. Acessado em: 08 abr. 2016.
- [94] DE PISA, U., 2016. Disponível em: <http://simulte.com>>. Acessado em: 20 fev. 2016.
- [95] ITU-R. “REPORT ITU-R M.2134, Requirements related to technical performance for IMT-Advanced radio interface(s)”, *ITU-R, Tech. Rep.*, 2008.
- [96] 3GPP, 2015. Disponível em: <http://www.3gpp.org/dynareport/36814.htm>>. Acessado em: 21 fev. 2016.
- [97] ROHDE-SCHWARZ, 2016. Disponível em: http://cdn.rohde-schwarz.com/pws/dl_downloads/dl_application/application_notes/1ma221/1MA221_1e_LTE_system_specifications.pdf>. Acessado em: 17 mar. 2016.
- [98] HO, Q.-D., GAO, Y., RAJALINGHAM, G., et al. “Robustness of the routing protocol for low-power and lossy networks (RPL) in smart grid’s neighbor-area networks”. In: *2015 IEEE International Conference on Communications (ICC)*, pp. 826–831. IEEE, 2015.
- [99] FOR STANDARD TECHNOLOGY, N. I. “NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 2.0”. Disponível em: https://www.smartgrid.gov/sites/default/files/doc/files/NIST/_Special_Publication_1108R2_NIST_Framework_d_Roadmap_for.pdf>. Acessado em: 30 mar. 2014.

- [100] EPRI. “IntelliGrid - Program 161”. Disponível em: <http://mydocs.epri.com/docs/Portfolio/PDF/2010_P161.pdf>. Acessado em: 30 mar. 2014.
- [101] IEEE. “IEEE Draft Guide for Smart Grid Interoperability of Energy Technology and Information Technology Operation with the Electric Power System (EPS), and End-Use Applications and Loads”, *IEEE P2030/D7.0, July 2011*, pp. 1–121, Aug 2011.
- [102] KARNOUSKOS, S. “The cooperative internet of things enabled smart grid”. In: *Proceedings of the 14th IEEE international symposium on consumer electronics, Braunschweig, Germany*, pp. 07–10, 2010.
- [103] CONSORTIUM, E. “The Internet of Things Initiative”. 2014. Disponível em: <<http://www.iot-i.eu/public>>. Acessado em: 30 mar. 2014.
- [104] VON DOLLEN, D. “Report to NIST on the smart grid interoperability standards roadmap”, *Electric Power Research Institute (EPRI) and National Institute of Standards and Technology*, 2009.
- [105] ZIGBEEALLIANCE. “Zigbee SmartEnergy”. 2014. Disponível em: <<http://zigbee.org/Standards/ZigBeeSmartEnergy/Overview.aspx>>. Acessado em: 30 mar. 2014.
- [106] GALLI, S., SCAGLIONE, A., WANG, Z. “For the grid and through the grid: The role of power line communications in the smart grid”, *Proceedings of the IEEE*, v. 99, n. 6, pp. 998–1027, 2011.
- [107] CHAO, C.-W., HO, Q.-D., LE-NGOC, T. “Challenges of Power Line Communications for Advanced Distribution Automation in Smart Grid”. In: *Power and Energy Society General Meeting (PES), 2013 IEEE*, pp. 1–5. IEEE, 2013.
- [108] HORN, P. *Autonomic Computing: IBM’s Perspective on the State of Information Technology*. Relatório técnico, IBM, 2001.
- [109] KEPHART, J., CHESS, D. “The vision of autonomic computing”, *Computer*, v. 36, n. 1, pp. 41–50, Jan 2003.
- [110] FRANKLIN, S., GRAESSER, A. “Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents”. In: *Intelligent agents III agent theories, architectures, and languages*, Springer, pp. 21–35, Budapest, Hungary, 1997.

- [111] WOOLDRIDGE, M., JENNINGS, N. R. “Intelligent agents: Theory and practice”, *The knowledge engineering review*, v. 10, n. 02, pp. 115–152, 1995.
- [112] JENNINGS, N. R., SYCARA, K., WOOLDRIDGE, M. “A roadmap of agent research and development”, *Autonomous agents and multi-agent systems*, v. 1, n. 1, pp. 7–38, 1998.
- [113] WOOLDRIDGE, M. *An introduction to multiagent systems*. Glasgow, John Wiley & Sons, 2009.
- [114] WEISS, G. *Multiagent systems: a modern approach to distributed artificial intelligence*. London, MIT press, 1999.
- [115] FERBER, J. *Multi-agent systems: an introduction to distributed artificial intelligence*, v. 1. Manchester, Addison-Wesley Reading, 1999.
- [116] HUHNS, M. N., SINGH, M. P. “Ontologies for agents”, *Internet Computing, IEEE*, v. 1, n. 6, pp. 81–83, 1997.
- [117] PIPATTANASOMPORN, M., FEROUZE, H., RAHMAN, S. “Multi-agent systems in a distributed smart grid: Design and implementation”. In: *Power Systems Conference and Exposition, 2009. PSCE'09. IEEE/PES*, pp. 1–8. IEEE, 2009.
- [118] JIANG, Z. “Computational intelligence techniques for a smart electric grid of the future”. In: *Advances in Neural Networks–ISNN 2009*, Springer, pp. 1191–1201, China, 2009.
- [119] FRIEDMAN-HILL, E. “The Java Engine Rules”. Disponível em: <<http://www.jessrules.com/jess/index.shtml>>. Acessado em: 30 mar. 2014.
- [120] LUKE, S., BALAN, G. C., SULLIVAN, K., et al. “Multi-Agent Simulator of Neighborhood”. Disponível em: <<http://cs.gmu.edu/~eclab/projects/mason/>>. Acessado em: 30 mar. 2014.

Apêndice A

Conceitos Gerais de Redes Elétricas Inteligentes

A.1 Redes elétricas inteligentes

As redes elétricas inteligentes representam um conjunto de áreas multidisciplinares, tais como tecnologia da informação, comunicações, segurança, energia elétrica, novos materiais, sistemas especialistas e redes de sensores aplicados aos domínios da geração de energia, da transmissão, da distribuição e do consumidor.

A inclusão de novas tecnologias de comunicação e informação no processo de automação deixa o sistema elétrico mais resiliente, mais tolerante à falhas e menos propenso às perturbações. Isso traz reflexos positivos para o consumidor, pois proporciona um aumento da qualidade e disponibilidade da rede elétrica e representa uma mudança de paradigma, ampliando os possíveis cenários de aplicações dentro do contexto de redes elétricas inteligentes.

Até alguns anos atrás apenas as grandes fontes geravam energia para o consumidor, porém esse cenário será transformado com a introdução de fontes alternativas de energia ligadas à rede de distribuição de energia elétrica, tornando o fluxo de potência bidirecional. Para que essa transformação seja viável, dois aspectos são relevantes:

1. O nível de controle da rede precisa ser muito mais sofisticado, mais robusto e ao mesmo tempo mais inteligente. Esse nível de inteligência permeará toda a rede desde a operação da rede até a casa do consumidor. Esse grau de complexidade vai gerar uma grande massa de dados para serem processados de forma descentralizada. Isso somente será possível se a infraestrutura de telecomunicações e a tecnologia da informação dispuserem de uma boa cobertura geográfica para que possa ocorrer com qualidade e em tempos de resposta adequados.

2. A mudança de paradigma depende muito do comportamento e da conscientização do consumidor que participará também da geração e do uso racional de energia.

A.1.1 Benefícios

O desenvolvimento de uma rede inteligente trará grandes benefícios aos consumidores e fornecedores de energia elétrica entre os quais a possibilidade de se ter uma gestão mais eficiente do seu consumo permitindo redução do pico de demanda.

Todo o processo de automação da rede também facilitará o atendimento ao usuário em caso de falha com a rápida localização do problema.

Uma automação inteligente permite que manobras de energia possam ser feitas reduzindo o tempo de recuperação. Nesse cenário, essa inteligência é capaz de reconfigurar a rede elétrica através de abertura e fechamento de chaves em um curto intervalo de tempo e isolar a falha garantindo o retorno da energia para maior parte dos consumidores.

Além de recuperar a rede, sistemas distribuídos permitirão um controle mais eficiente da demanda, otimizando recursos disponíveis através de controles de potência de aparelhos e acionando-os quando for mais conveniente para a rede o quando mesmo o horário tarifário for o mais benéfico para o consumidor.

Um outro aspecto resultante desse novo ramo do conhecimento é a possibilidade de desenvolvimento de novas tecnologias, entre as quais relacionadas com eficiência energética, iluminação baseadas em LEDs, motores mais eficientes, aquecimento inteligente e integrado à rede. Outras tecnologias que associam inteligência distribuída, inteligência artificial na otimização dos recursos, também representam inovações nessa área de conhecimento.

A.1.2 Histórico e padrões

Como o tema envolve diversas áreas do conhecimento, isso atraiu o interesse de inúmeras empresas e universidades. Diversas iniciativas de padronização começaram a surgir com o objetivo de garantir uma futura interoperabilidade entre sistemas e tecnologias.

Em 2007, o governo americano determinou através de uma norma denominada *Energy Independent Security Act* (EISA), que o órgão regulador *National Institute for Standard Technology* (NIST) fosse responsável pela coordenação da criação de uma arquitetura de rede inteligente [99] para a rede de energia elétrica. Essa arquitetura propõe que os protocolos de comunicação e modelos de dados sejam padronizados a fim de que o gerenciamento da informação permita a

interoperabilidade entre sensores e subsistemas. Segundo a norma, esse conjunto de padrões deve ser “flexível, uniforme e tecnologicamente neutro”. Dessa forma, o governo deseja garantir a criação de um padrão aberto para que empresas e fabricantes possam participar em um mercado altamente competitivo e garantir que a tecnologia estivesse sempre no estado da arte.

Desde 2008, o *Electric Power Research Institute* (EPRI), uma instituição sem fins lucrativos, trabalha em um programa conhecido como *Intelligrid* [100], que avalia tecnologias da informação e comunicação e segurança. Além desse papel, o EPRI envolve diversos fornecedores e empresas prestadoras de serviço na busca das melhores soluções através de uma série de testes e projetos pilotos envolvendo tecnologias. Através dos resultados dos testes, ou das lições aprendidas desses projetos, o EPRI desenvolveu um conjunto de requisitos de tecnologias e infraestrutura de controle que permite suportar a integração de diversas fontes de energia distribuídas. O EPRI também participou com o NIST na criação da arquitetura e contou com diversas contribuições de fabricantes de equipamentos e distribuidores de energia.

Em 2010, o IEC, órgão regulador internacional, também apresentou um *roadmap* de padrões a serem aplicados às redes elétricas inteligentes, similares ao conjunto definido pelo NIST. O IEC tem interesses similares com padronização de geração, transmissão e distribuição de energia elétrica e, avaliando a evolução do assunto, tratou de criar um grupo estratégico denominado *Study Group 3*(SG3).

O IEEE, com o intuito de suportar outros órgãos normativos e, baseando-se nos padrões já incorporados pelo NIST e por outras instituições, resolveu também criar um padrão conhecido como IEEE Std 2030 [101]. Esse padrão tem como foco principal a criação de um guia de interoperabilidade de componentes de comunicação, sistemas de potência e plataformas de tecnologia da informação. O modelo de referência adotado pelo IEEE é apresentado na Figura A.1.

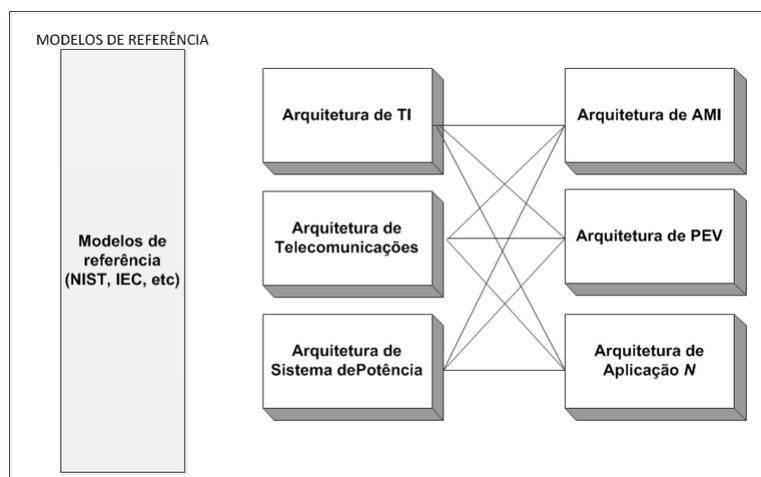


Figura A.1: Modelo padrão de referência do IEEE

Independentemente dos padrões aqui citados, todos eles têm uma preocupação em comum: a capacidade de se ter uma arquitetura totalmente aberta, interoperável e que seja um guia aplicável de uma forma genérica. Isso permitirá que o consumidor tenha capacidade de gerenciar o uso e a geração de energia elétrica.

A rede elétrica inteligente permitirá a introdução de novos produtos e serviços associados. Para que todos os aparelhos residenciais, medidores eletrônicos, fontes de energia renováveis, veículos elétricos e sistemas de armazenamento de energia sejam integráveis, é fundamental que ocorra de forma aberta e padronizada. Isso significa que no futuro todo e qualquer fabricante que estiver aderente aos padrões fixados será capaz de conectar o seu aparelho à rede elétrica e ser gerenciado pelo consumidor ou até mesmo pela concessionária de energia.

A criação de um padrão tem recebido inúmeras contribuições de fabricantes, empresas de energia e de governo. É necessário uma completa harmonização pois centenas de padrões fazem parte da definição global da rede elétrica inteligente. Dada a extensão do assunto, o NIST elegeu oito áreas prioritárias na definição do padrão:

1. **Conhecimento da área geográfica** - Refere-se ao monitoramento e a visualização da situação de grandes áreas geográficas de energia a fim de se ter um status das ocorrências e do desempenho da rede. Com essa gestão é possível tomar ações imediatas que não permitam falhas na rede.
2. **Eficiência energética e resposta a demanda** - Refere-se ao mecanismo de otimização do recurso energético através da oferta diferenciada de preços para os diversos tipos de usuários. O objetivo é espalhar o pico da demanda ao longo de um período trazendo benefícios para as geradoras e os consumidores.
3. **Armazenamento de energia** - Refere-se às novas tecnologias de armazenamento distribuído e que, futuramente, poderão auxiliar na redução da demanda durante o horário de pico.
4. **Veículos elétricos** - Refere-se ao uso de carros ou veículos elétricos ou híbridos e que contribuirão para redução da demanda por petróleo significativamente. Esse assunto tem se destacado, pois veículos também são vistos como meio de armazenamento de energia. Uma vez conectados à rede de distribuição, poderão devolver a energia armazenada. Portanto, é necessário uma padronização na forma de comunicação do veículo com as empresas de distribuição de energia, o momento em que o usuário poderá ou desejará fornecer e quanto o mesmo será remunerado por isso. Por detrás desse conceito, existe um sistema inteligente que monitorará o veículo, a sua autonomia e

estabelecerá uma comunicação com a empresa, informando quando o mesmo poderá estar disponível para devolver energia para a rede.

5. **Segurança da Informação** - Refere-se à manutenção da segurança dos dados trafegados sob a ótica dos três pilares de uma comunicação segura: confidencialidade, integridade e disponibilidade. Isso se aplica à infraestrutura de telecomunicações e sistemas de informação. Esse é um dos aspectos mais críticos da rede pois a mesma será vista como a “rede das redes” pois uma vez conectada à Internet, por exemplo, a mesma ficará exposta aos mais diversos tipos de ataques. A preocupação com *hackers* e invasões é motivo para se criar mecanismos inteligentes para se evitar possíveis problemas nos consumidores e fornecedores de energia.
6. **Telecomunicações** - Refere-se ao uso de diversas tecnologias de comunicação e à segurança envolvida. É muito importante, pois a rede de telecomunicações será sobreposta à rede elétrica para o transporte da informação. Uma malha de distribuição de energia elétrica deverá ser suportada por uma rede de telecomunicação robusta, permitindo que em caso de falha de um ou mais nós da rede de comunicação não afete as aplicações de controle e automação.
7. **Infra-estrutura de medição de energia (AMI)** - Refere-se à arquitetura de *software*, *hardware* e telecomunicações que permitem que o usuário tenha um controle efetivo da energia consumida nas residências. O gerenciamento associado à possibilidade do usuário receber tarifas diferenciadas de energia ao longo do dia, permitirá a redução de custo e o uso mais eficiente da energia elétrica.

No futuro, cada aparelho residencial será capaz de se comunicar inclusive com a empresa fornecedora e, remotamente, com o próprio usuário através da Internet. O assunto é visto como a “Internet das coisas” onde todo e qualquer aparelho terá um endereço de rede Internet. Isto permite uma granularidade no controle de energia mais eficiente e inteligente.

As empresas de serviço, por exemplo, podem utilizar essa informação em uma grande base de informações, para realizar mineração de dados sobre demandas de energia. Esse é um tema que tem um potencial enorme para alavancar outras tecnologias [102] relacionadas a *Web Services*, redes cooperativas, arquitetura orientada a serviços. No futuro, quando cada aparelho tiver o seu endereço e conectado à Internet, seja ele de qualquer tipo, ligado à rede elétrica, poderá ser acessado. Já existem diversas iniciativas [103] que contam com a colaboração de instituições de pesquisa e grandes empresas para criar e consolidar uma arquitetura padronizada.

8. **Gerenciamento da rede de distribuição** - Refere-se ao controle eficiente da rede de distribuição elétrica integrada à rede de transmissão, monitorando transformadores, alimentadores e outros componentes da rede. Isto aumenta a disponibilidade e a confiança na rede através de recursos automáticos que permitam remanejamento de energia. Em caso de falha, a informação é enviada para os centros de operação a fim de que equipes de manutenção possam ser despachadas automaticamente.

A.1.3 Arquitetura conceitual

Para o NIST [104], uma arquitetura conceitual divide a rede em partes conhecidas como domínios, interligados por uma infraestrutura de comunicação segura, conforme representado no diagrama da Figura A.2. Cada domínio (geração, distribuição, transporte, consumo, mercado, provedores de serviços e operação) é constituído por um conjunto de aplicações próprias destinadas à automação e comunicação, além de um conjunto de protocolos, arquitetura de dados e de informação. Nesses domínios o fluxo da informação também obedece a um determinado conjunto de recomendações já elaborado pelo NIST e pelo IEEE.

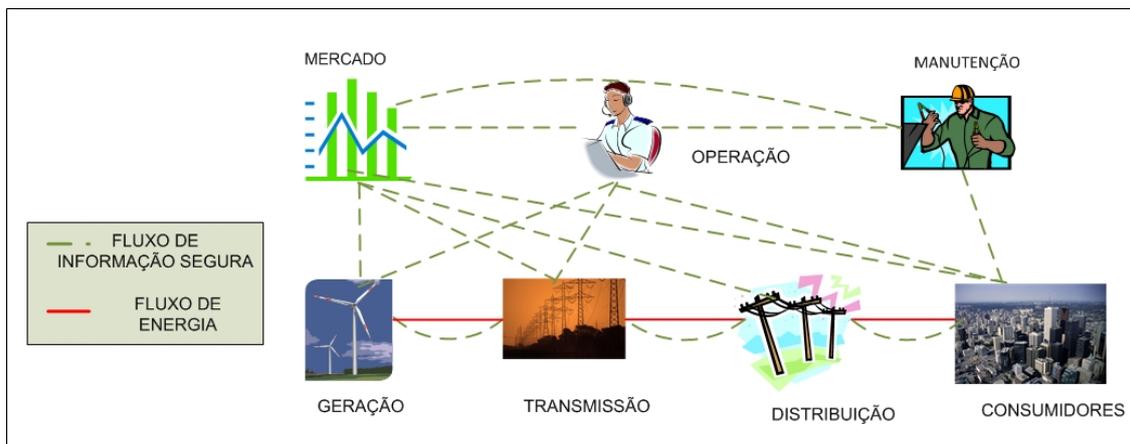


Figura A.2: Arquitetura conceitual elaborada pelo NIST

A Figura A.3 representa o modelo conceitual e genérico da infraestrutura de telecomunicações aplicada a cada um dos domínios envolvidos. Diversas tecnologias de comunicação estão envolvidas em cada um dos domínios adequadas a cada ambiente. A arquitetura genérica permite que novas tecnologias possam ser incorporadas. Cada um desses domínios possui interfaces de comunicação específicas com outros domínios.

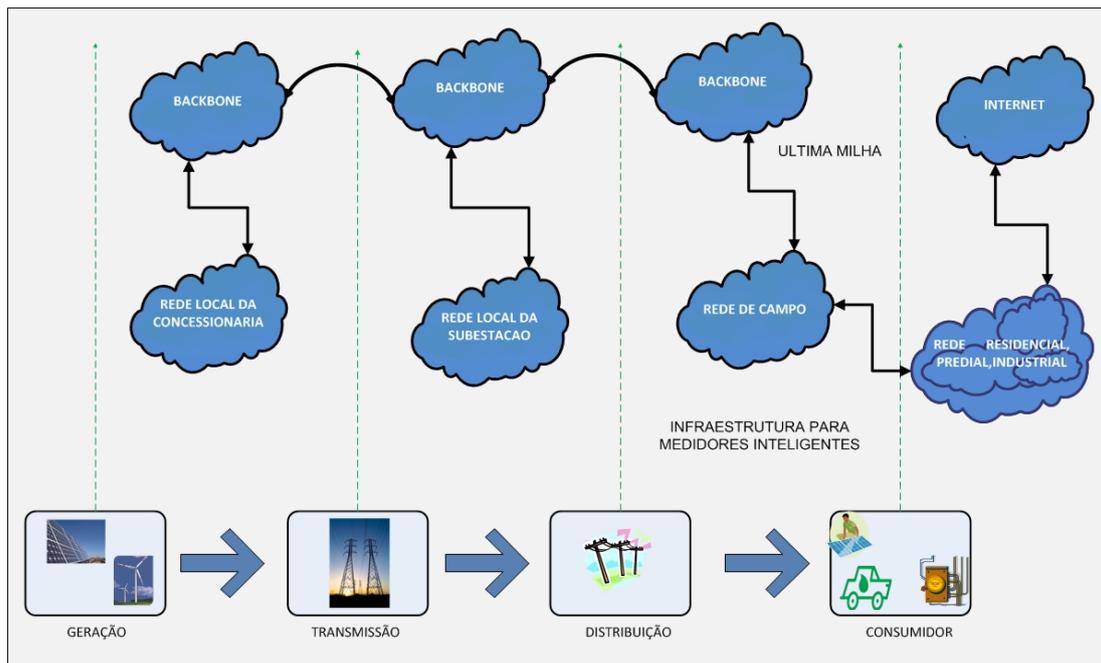


Figura A.3: Modelo conceitual de arquitetura de telecomunicações

Para entender essa representação, é preciso apresentar alguns conceitos básicos descritos pelo NIST.

- **Atores** - Representam aparelhos eletrônicos, computadores, programas de computador ou organizações que os controlam.
- **Aplicações** - Representam tarefas que são desempenhadas por um ou mais atores dentro do domínio operando de forma cooperativa como exemplo, a aplicação de reconfiguração de redes elétricas de média tensão e controle de resposta à demanda.
- **Associações** - São conexões lógicas estabelecidas entre atores através de interfaces.
- **Interfaces** - Representam pontos lógicos de conexão entre domínios que podem ser elétricos ou de comunicação.
- **Domínio** - Representam agrupamentos de atores com um objetivo em comum.

É possível observar, na norma IEEE P2030, as diversas interconexões existentes entre as entidades lógicas pertencentes a domínios diferentes. O modelo descreve interfaces entre esses domínios e atores e diversas aplicações para cada um deles. Esse tipo de abstração permite a implementação de novas tecnologias e novas aplicações, além de estabelecer um grau de interoperabilidade entre as entidades.

Diversas tecnologias existentes no mercado são apontadas para atuar nos domínios representados na Figura A.3 e nas interfaces entre os domínios, como por

exemplo, GPRS, fibra ótica, Wimax, LTE 4G, 3G, rádio microondas. No entanto, os cenários de uso dependem muito dos aspectos de demanda de tráfego gerada, da aplicação envolvida e das características de uma rede de comunicação como latência e perda de pacotes.

A.1.3.1 Infraestrutura Tecnológica

Cada domínio possui arquitetura própria adequada ao seu ambiente. Do ponto de vista de comunicação, vislumbram-se as seguintes tecnologias potencializadoras para redes inteligentes:

- **Domínio cliente** Esse domínio divide-se em quatro outros:

Rede residencial (RR) Restrito ao ambiente doméstico e seus aparelhos internos que podem ser desde o refrigerador, ar condicionado até o aquecedor. Todo o arcabouço tecnológico que permite a transmissão de dados e o gerenciamento desses equipamentos para a rede da concessionária é conhecido como medidor inteligente.

O medidor de energia inteligente será capaz não só de fazer a interface com os equipamentos mas também de receber os planos tarifários das empresas para que o consumidor possa adequar a sua demanda ao preço cobrado em um determinado momento. O medidor terá, no futuro, outros papéis como a capacidade de se comunicar com cada um dos aparelhos dentro de uma residência, fábrica ou prédio e obter e transmitir informações. Dessa forma, a concessionária da energia poderá saber inclusive o consumo individual de cada tipo de aparelho, criando uma base de dados do qual poderão ser extraídas informações de comportamento do usuário.

Existem diversas iniciativas para estabelecer um padrão de comunicação entre o medidor inteligente e os aparelhos, porém há uma iniciativa, coordenada por diversas empresas, conhecida como *Zigbee Alliance* [105]. Essa iniciativa trata da utilização do padrão IEEE 802.15.4, que especifica a camada física e de acesso de uma rede sem fio, no espectro de faixa não licenciada, com o objetivo de utilizá-la em aparelhos interligados ao medidor eletrônico e à operadora de energia. A versão 2.0 (IEEE 2030.5), finalizada em 2013, representa um avanço significativo no sentido de incorporar novas funcionalidades para a criação de uma rede local, destinada ao gerenciamento, controle de aplicações de informação de consumo de aparelhos residenciais com o objetivo de garantir a interoperabilidade entre os diversos fabricantes.

Entre o medidor eletrônico e as operadoras de energia, as tecnologias mais adequadas são as que permitem comunicação sem fio com o usuário como

Wimax (IEEE 802.16m), LTE 3G, 4G ou ainda *Power Line Communications* (PLC). Essa comunicação é estabelecida entre o medidor eletrônico e a concessionária de energia. Dessa forma dados como consumo, tensão, corrente podem ser monitorados periodicamente.

Rede predial (RP) - Representa a arquitetura de rede que faz uso de tecnologia de monitoração em ambientes prediais. Sistemas de automação predial podem se beneficiar da tecnologia zigbee.

Rede industrial (RI) - Representa a arquitetura de rede que utiliza tecnologia de monitoração de ambientes industriais. Como a área industrial é mais complexa, normalmente esse tipo de rede está associado a outras tecnologias como redes sem fio padrão IEEE 802.11 a/b/g/n, sistemas ópticos e roteadores. As fibras óticas são largamente utilizadas nesses ambientes devido à sua imunidade a ruído e longo alcance.

Rede de campo (RC) - Representa a infraestrutura de rede de comunicação e segurança, composta por diversas tecnologias, para prover serviços de última milha para automação de consumidores e ambientes industriais e também para operação das concessionárias. Os exemplos de tecnologia que suportam esse conceito são: Wimax (IEEE 802.16), IEEE 802.11 a/b/g/n, Zigbee (IEEE 802.15.4).

- **Domínio distribuição** Abrange as redes de energia elétrica de média e baixa tensão. Sobre essa rede está sobreposta uma infraestrutura de sensores com capacidade de comunicação entre os nós. Esse é um tema ainda pouco explorado pelos artigos acadêmicos que abordam mais a questão do controle distribuído do que o meio de comunicação propriamente dito.

Alguns artigos apontam uma solução de comunicação baseada em PLC. Em [106], os autores descrevem detalhadamente as diversas tecnologias já existentes no mercado e outras que foram padronizadas como o *Broad Band PLC* (IEEE 1901), com promessas de taxas de até 200 Mbps. No entanto, entre todas as tecnologias apresentadas, poucas são adequadas a transmissão em linhas de média tensão devido a problemas de ruído impulsivo e a não linearidade existente em de linhas de distribuição. Nesse caso, o artigo apresenta modelos de PLC mais adequados nesse nível de tensão e com alcance da ordem de dezenas de quilômetros. No entanto, há limitação de taxa de transmissão entre 100 Kbps e 500 Kbps e isso pode ser um forte fator limitante para as aplicações de automação que demandem capacidade. Por outro lado, em [107] são apresentados os desafios que ainda persistem nessa tecnologia quando se trata de uso sobre linhas de distribuição não só tecnicamente pois requer mais repetidores, como economicamente também.

Tipicamente, a arquitetura sobre a rede de distribuição é composta de sensores interligados em malha e suportados por protocolos de comunicação robustos e tolerantes a falhas.

- **Domínio transmissão** - Abrange as redes de transmissão em alta tensão e com sensores PMUs instalados em subestações de energia ou em fontes geradoras. Os PMUs são responsáveis pela medição precisa de tensão e corrente a fim de garantir a qualidade da energia transmitida. Tratando-se de subestação, os mesmos são bem atendidos por sistemas rádio enlace e fibras óticas, que são muito utilizadas em linhas de transmissão e são conhecidas como cabos tipo *Optical Ground Wire* (OPGW).
- **Domínio geração** - São as unidades geradoras de energia, como hidroelétrica, eólica ou solar. Nesses elementos, sensores de energia coletam informações da qualidade, potência, tensão e corrente.

A arquitetura de redes elétricas inteligentes é então composta por agrupamentos de subredes de sensores, alocados em cada um desses domínios, atuando com autonomia, desde a casa do consumidor até a usina geradora de energia.

Apêndice B

Sistemas Multiagentes

B.1 Conceito de Controle Autônômico

Em 2001, o vice-presidente de pesquisas da IBM, Paul Horn [108], lançou um manifesto no qual declarava que os grandes obstáculos que a área de tecnologia da informação das grandes empresas passaria a experimentar seria a crescente complexidade das aplicações desenvolvidas em função dos milhões de linhas de código para implementá-las. Isso significava um forte investimento em pessoas e um sem-número de configurações, além da integração com diversos subsistemas. Para quebrar esse paradigma, a IBM propôs uma nova abordagem sobre o tema. Essa proposição baseava-se na criação de *hardwares* e *softwares* com características autônômicas e autoconfiguráveis. O conceito era baseado na modelagem de sistemas biológicos como o sistema nervoso, responsável pelo controle de todo o nosso organismo, capaz de gerenciar tudo de forma equilibrada sem que se pense no próprio funcionamento. A partir dessa ideia, criou-se um conceito de computação análogo e que era constituído de pequenos elementos, análogo à células de um organismo, capazes de se agrupar para trabalhar de forma coordenada. Esse modelo ficou conhecido como computação autônômica. Dessa forma, sistemas autônômicos permitiriam que os usuários focassem nas suas reais necessidades, deixando a parte configurável nas mãos desses sistemas.

Na visão da IBM [109], sistemas autônômicos devem possuir cinco fundamentos para que sejam considerados sistemas independentes:

1. **Autogerenciamento** - É a capacidade de se automonitorar verificando o status interno e capaz de tomar decisões independentes de um operador. Estas monitorações podem ser observadas em sistemas que analisam falhas que monitoram o funcionamento e o comportamento de programas e podem gerar uma configuração alternativa de forma proativa.
2. **Autoconfiguração** - É a capacidade que um sistema possui para se

configurar, de forma otimizada, baseada em políticas de mais alto nível, como por exemplo, de negócio. Desta forma, o usuário não precisaria se preocupar com os detalhes da implementação deixando esta parte para o sistema e foca apenas naquilo que é essencial. Uma vez que as políticas são modificadas, o sistema percebe as alterações e muda a configuração dos seus parâmetros.

3. **Auto-otimização** - É a capacidade, derivada do autogerenciamento, que um sistema possui para otimizar os parâmetros em tempo-real a fim de assegurar o melhor desempenho.
4. **Auto-recuperação** - É a capacidade que um sistema tem para coletar informações de problemas, diagnosticar e corrigí-las, através de uma base de conhecimento e de análise de dados de configuração, de seus parâmetros e histórico de problemas. Em sistemas mais complexos a análise é conhecida como análise causa-raiz. O próprio sistema é capaz de criar novas regras acrescentando-as à base de conhecimentos existente.
5. **Auto-proteção** - é a capacidade que um sistema possui de proteger os dados e um sistema inteiro através da detecção de falhas, maliciosas ou não, evitando que o problema se propague. Outra característica importante é a proatividade de monitoração através de sensores e tomada de determinadas ações antes que o fato ocorra.

Além destas características citadas, sistemas autônômicos possuem a capacidade de trocar ou solicitar informações com outros sistemas em uma linguagem de alto nível e de execução de ações independentes de tal forma que todos os sistemas interligados alcancem uma solução otimizada global.

Em um sistema autônômico, os sensores são monitorados e os dados são analisados realizando-se consultas em uma base de conhecimento. Disso resulta em um planejamento e execução de tomada de decisão que, por sua vez, reflete-se nos atuadores como uma resposta ao que foi monitorado.

B.2 Conceito de Sistemas Multiagentes

O sistemas multiagentes têm recebido diversos conceitos por parte dos pesquisadores e representam um reflexo e uma evolução do conceito de sistemas autônômicos. A existência de diversas definições torna o seu entendimento um pouco confuso já que alguns autores tratam como sistemas inteligentes ou que tenham um certo grau de inteligência [110]. De um modo geral, os conceitos convergem para a ideia de que o sistema multiagente é uma entidade, *hardware* ou *software*, que executa ações de forma independente ou autônoma. Para isso, o sistema precisa interagir com

o meio ambiente, realizando uma série de tarefas para atingir um determinado objetivo para o qual foi designado. Outros autores vão mais longe e incluem, entre outros aspectos, a capacidade de fazer inferências, realizar interpretações e resolver problemas, coincidindo com o conceito de sistemas autônômicos explicado anteriormente. No entanto, isso depende da aplicação envolvida, pois nem todas as características de um agente são desejadas. Em [111], o autor descreve um conceito mais amplo de agente e agentes inteligente. Agentes são simplesmente programas que executam uma determinada tarefa como, por exemplo, um controle digital de temperatura. O agente inteligente possui três características fundamentais que conceituam um agente que possui autonomia:

1. **Reatividade** - É a capacidade de sentir o ambiente no seu entorno e de reagir a mudanças a fim de atender a um objetivo definido.
2. **Proatividade** - É a capacidade de tomar a iniciativa de mudar o seu comportamento para alcançar um objetivo, regido por comportamentos definidos.
3. **Habilidade social** - É a capacidade de interação com outros agentes a fim de satisfazer um objetivo. Nesse caso, agentes devem ter uma característica importante que é a cooperação entre si. Essa interação atualmente é representada por uma linguagem de comunicação denominada *Agent Communication Language* (ACL) através da qual agentes inteligentes podem trocar informações. A linguagem promove um grande nível de abstração do código desenvolvido para o agente por se tratar de um padrão de envio de mensagens.

Em outro estudo [112], os autores descrevem sistemas multiagentes como sendo representantes de uma rede de solucionadores de problemas fracamente interligados e operando de forma conjunta e cooperativa. Além das características observadas, outras são atribuídas a um sistema multiagente:

1. As informações estão dispersas entre os agentes;
2. Cada agente possui apenas uma parte da informação sobre um problema, o que torna incapaz de resolvê-lo globalmente;
3. Cada agente processa sua informação sem sincronizar com outros.

Com esse conjunto básico de características é possível criar um sistema autônômico que possua propriedades de resiliência, tolerância a falhas e robustez. A resiliência de um sistema com multiagentes é uma característica que pode ser explorada em reconfiguração de redes elétricas. Em cenários de falha da rede elétrica

de distribuição ou de otimização das cargas são potenciais aplicações que um sistema distribuído e inteligente pode alcançar uma solução satisfatória em tempo hábil.

Em [113], quatro tipos de agentes são descritos de forma detalhada:

1. **Agentes baseados em lógica** - A decisão é baseada através de uma lógica dedutiva.
2. **Agentes reativos** - A decisão é mapeada em uma forma de mapeamento direto tipo situação \rightarrow ação.
3. **Agentes *Believe-Desire-Intention* (BDI)** - A decisão depende da manipulação da estrutura de dados que representa uma realidade, desejos e intenções do agente. Trata-se de uma atividade ou um comportamento que é realizado passo-a-passo a fim de alcançar um objetivo. Esse mecanismo lógico é conhecido como raciocínio prático e envolve a tomada de decisão, os objetivos que se deseja alcançar e as ações devem ser tomadas.
4. **Arquitetura em camadas** - A tomada de decisão é realizada através de várias camadas de software em que cada uma delas tem um nível de inteligência com diferentes níveis de abstração.

B.2.1 Benefícios

Existem diversos aspectos que os tornam apropriados para problemas de engenharia. Destacam-se [113]:

1. **Encapsulamento** - Análogo à linguagem orientada a objetos, um agente é capaz de “esconder” as suas tarefas do mundo exterior, encapsulando a rotina e deixando apenas as suas entradas e saídas disponíveis. A diferença entre os dois está no fato que os métodos de uma linguagem orientada são chamados enquanto os agentes recebem mensagens de solicitação de execução como interface.

Outro aspecto é que em linguagem orientada a objeto existe somente uma *thread* que controla todo o sistema enquanto o agente possui a sua própria *thread* de controle. Isso permite que um agente tenha autonomia ou independência de outros agentes, definindo a prioridade de suas ações, aceitando ou não as mensagens de outros agentes e se outras ações serão escalonadas. Essas características é que tornam o seu uso atrativo. Em vista disso, agentes podem ser substituídos por outros em tempo-real, mais atuais, ou ainda, novos agentes podem ser incluídos no sistema, o que permite que o ambiente seja bastante flexível.

2. **Interoperabilidade** - Essa característica permite que entidades distintas possam conversar entre si atendendo a um padrão de comunicação. Esse padrão aberto de comunicação foi normatizado pelo IEEE *Foundation for Intelligent Physical Agents* (FIPA), que é uma organização sem fins lucrativos responsável por estudar e deliberar sobre tecnologia baseadas em agentes e a interoperabilidade com outras tecnologias. A padronização está dividida nas seguintes categorias: agente de comunicação, agente de transporte, agente de gerenciamento, arquitetura abstrata e aplicações. O coração da padronização é o agente de comunicação, responsável pela troca de mensagens entre agentes.
3. **Distribuição** - Agentes inteligentes estão espalhados pelos elementos da rede e possuem total autonomia de ação. Múltiplos agentes podem ser instalados em um único *hardware*.
4. **Redundância** - Com diversos agentes inteligentes espalhados pela rede, é possível aumentar a disponibilidade de um serviço ou sistema, através da interação com seus vizinhos.
5. **Autoreconfiguração** - Sistemas auto-reconfiguráveis são aqueles capazes de mudar o seu comportamento diante de alterações no ambiente que cada agente sente e percebe. Entre estas condições, destaca-se a de falha de um agente. Quando esse evento ocorre, agentes trocam mensagens entre si e, de forma cooperativa, buscam uma alternativa de configuração do ambiente que atenda a um objetivo global, porém a informação sobre o ambiente que cada um possui é parcial. Para resolver esse problema, mecanismos de decisão local e hierárquicos devem ser aplicados, isto é, a inteligência é distribuída em camadas de *software*.

B.2.2 Aplicabilidade à redes elétricas inteligentes

Uma rede inteligente, no futuro, será composta por milhões de objetos, desde aparelhos de usuários, passando por milhares de sensores de rede e transformadores. Para resolver problemas relacionados a esse tipo de rede, uma arquitetura centralizada traria uma série de dificuldades, a começar pela demanda de tráfego que seria enorme e, em segundo lugar, aplicações que demandassem soluções rápidas e pontuais levariam tempo para processamento. Então, a abordagem por sistemas distribuídos parece ser uma boa alternativa, pois possibilita uma redução do tráfego de informação além de permitir tomadas de decisão mais rápidas e localizadas. Esse é o quadro ideal para se aplicar o conceito de agentes inteligentes sobre uma rede elétrica. Esta característica de flexibilidade permite que certos problemas complexos possam ser divididos em problemas menores, adequados a uma solução mais rápida.

Cada agente inteligente pode encapsular algoritmos inteligentes utilizando diversas técnicas já conhecidas, tais como inteligência artificial, redes neurais, lógica nebulosa ou estimadores bayesianos, sendo cada um focado em um problema específico. Da interação entre os agentes resulta em soluções que podem estar próximas do ótimo global.

B.3 Modelos de aplicações distribuídas

Existem dois modelos de aplicações distribuídas, sendo o primeiro conhecido como cliente/servidor. Nesta arquitetura, ambos papéis são bem conhecidos e mapeados. O cliente possui o papel de requisitante da informação enquanto o servidor é a fonte da informação. São papéis estáticos e reativos, isto é, quando o cliente necessita de alguma informação ele a envia para o servidor e o mesmo responde. O servidor não possui nenhuma capacidade para tomada de iniciativas independentes. O cliente, por outro lado, possui toda capacidade para tomá-las, porém somente mediante requisição do usuário, ou seja, não possui nenhuma forma de autonomia. Clientes não se comunicam diretamente com clientes e servidores só falam com clientes quando os mesmos estão disponíveis. Um exemplo desta arquitetura é a própria Internet em que o usuário acessa um determinado serviço.

O segundo modelo, conhecido como *peer-to-peer*, possui arquitetura com papéis mais flexíveis. O nó cliente é capaz de tomar iniciativas, começar uma comunicação e prover serviços. A aplicação não está mais concentrada no servidor, mas distribuída pelos clientes que podem entrar na rede ou sair a qualquer momento sem notificar a nenhum outro servidor. Essa é a diferença fundamental entre os dois modelos. Enquanto no cliente/servidor, o cliente necessariamente precisa saber com quem falar, no modelo *peer-to-peer* o conhecimento dos outros elementos é totalmente aleatório. Então é preciso possuir um mecanismo com o qual os nós possam saber que outros entraram ou saíram da rede e até mesmo buscar por um determinado tipo de serviço. Esses mecanismos são conhecidos como páginas brancas e páginas amarelas, análogo ao serviço de pesquisa em um catálogo telefônico.

Na base desses dois mecanismos estão dois modelos de rede: descentralizado e híbrido. O primeiro apresenta a ausência de uma entidade centralizadora e o mecanismo de entrada e saída da rede. O número de conexões cresce exponencialmente com o tamanho da rede assim como tráfego de comunicação.

Por outro lado, o modelo híbrido dispõe uma entidade centralizadora responsável pelo controle de entrada e saída dos nós e do provimento de serviços. Esse modelo gera menos tráfego na rede e é mais seguro, pois requer que o nó se autentique e se registre na rede. O aspecto contrário a esse modelo é a total dependência do nó central que pode ser um ponto crítico em caso de falha. Ele pode apresentar duas

abordagens distintas. Em primeiro lugar, o nó central pode ser utilizado apenas como *proxy* ou roteador de informações cujo destino seja outro nó da rede. A inteligência localiza-se nas extremidades da rede. Em segundo lugar, a inteligência pode encontrar-se no nó central e as extremidades fazem "consultas" ao nó central. Nessas duas estratégias a centralização traz impactos sobre as redes de comunicação e quanto ao tamanho da infraestrutura necessária para suportar esta concentração.

É fundamental para coordenação entre os agentes e para que todos se entendam através de uma linguagem em comum. Em [114], são citados quatro tipos de arquiteturas de comunicação entre agentes.

A primeira, a comunicação direta ou *peer-to-peer*, cujo modelo ilustrado na Figura B.1 ocorre quando todos os agentes trocam informações entre si através de um protocolo. Isso tem certas implicações quando se trata em quantidade de mensagens que pode ser elevada dependendo da implementação dos algoritmos. Quanto maior a topologia, mais complexa é a comunicação.

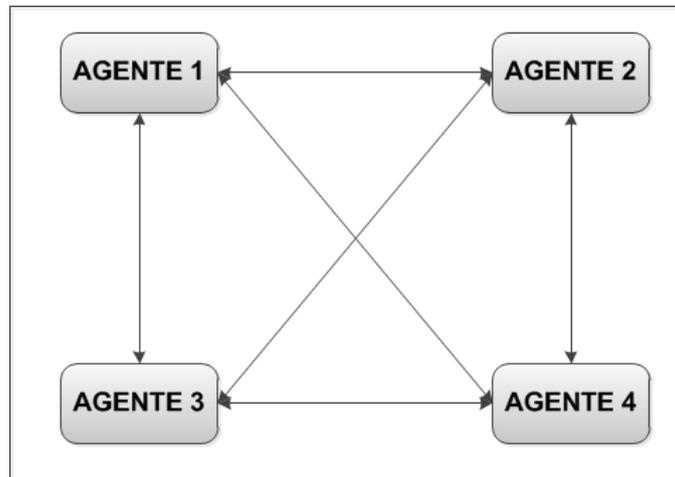


Figura B.1: Modelo de comunicação direta

A segunda, conhecida como federada ou coordenada, quando um agente é responsável pela comunicação entre diversos agentes, conforme ilustrado na Figura B.2. A vantagem é a existência de uma coordenação da comunicação exigindo menos tráfego e reduzindo a complexidade. É vantajoso quando a rede é muito grande porém exige uma robustez maior do centralizador, pois é o elo de comunicação entre todos os agentes.

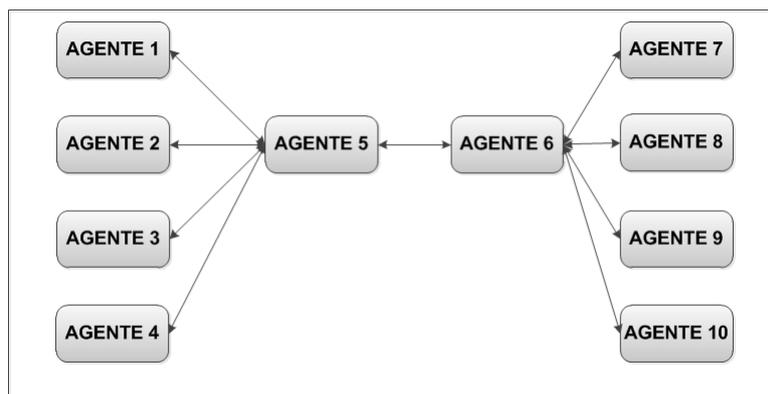


Figura B.2: Modelo de arquitetura híbrida

A terceira, conhecida como *broadcast*, na qual uma mesma mensagem que é enviada simultaneamente para todos os agentes, mas o agente somente toma uma ação quando identifica que a mensagem foi destinada a ele.

Por fim, existe um outro tipo de comunicação conhecido como *blackboard* [114], cuja arquitetura é composta de várias fontes de inteligência que contém algoritmos, regras, dados, um elemento de controle que informa qual a ordem de execução das bases de conhecimento e a área comum de escrita (*blackboard*) na qual todos os agentes escrevem suas contribuições quando um resultado é alcançado.

B.4 FIPA

A associação denominada FIPA é uma entidade sem fins lucrativos de empresas e organizações que cooperam entre si para desenvolver especificações para tecnologias de multiagentes genéricos, independentes da sua tecnologia de tal forma que seja possível interligá-los independente da empresa que o desenvolveu. A FIPA não desenvolve simplesmente uma tecnologia para uma aplicação específica mas um conjunto de especificações de tecnologias que permitam desenvolvedores integrar diversos sistemas complexos com alto grau de interoperabilidade e confiabilidade.

O padrão FIPA estabelece quatro tipos de linguagens que implementam a lógica dos agentes: *FIPA-Semantic Language*(FIPA-SL) , *Knowledge Interchange Format*(KIF) , *Constrain Choice Language* (CCL) e *Resource Description Framework**Resource Description Framework*(RDF) .

Conforme a norma, a mensagem possui uma série de parâmetros para que os agentes se comuniquem corretamente. A mensagem deve obrigatoriamente conter a origem(*sender*), destino(*receiver*) e o ato performativo(*performative act*). Outros campos são opcionais, porém são importantes para que a mensagem seja compreendida dentro do contexto de utilização. Os parâmetros utilizados estão apresentados na tabela B.1.

Campo da mensagem	Parâmetro	Significado
Tipo de ato comunicativo	<i>Performative</i>	Tipo de ato da mensagem
Participantes	<i>Sender</i>	Quem enviou a mensagem
	<i>Receiver</i>	Destino da mensagem
Conteúdo da mensagem	<i>Reply-to</i>	Novas mensagens deverão ser enviadas para destino Reply-to
	<i>Content</i>	É o conteúdo da mensagem
Descrição do conteúdo	<i>Language</i>	Denota linguagem na qual o conteúdo da mensagem é expresso
	<i>Encoding</i>	Especifica a codificação do conteúdo da mensagem
	<i>Ontology</i>	Define a ontologia que proporciona entendimento dos símbolos existentes no conteúdo da mensagem
Controle de conversação	<i>Protocol</i>	Define o protocolo de interação desta mensagem
	<i>Conversation Identifier</i>	Introduz uma identificação da conversação corrente
	<i>Reply-with</i>	Introduz uma expressão que será usada pelo agente receptor para identificar a mensagem
	<i>In-Reply-to</i>	É a expressão que se referencia a uma mensagem anterior para qual esta mensagem é uma resposta
	<i>Reply-by</i>	Expressa quanto tempo o emissor ficará esperando por uma resposta

Tabela B.1: Tabela de formato da mensagem

O primeiro documento, editado em 1997 e conhecido como especificação FIPA97, descreve um conjunto de regras para que um grupo de agentes possa coexistir, operar e ser mantidos. Nessa especificação é descrito um modelo de referência de uma plataforma, identificando alguns tipos de agentes obrigatórios em uma arquitetura. Esse possui os seguintes componentes:

- **Agent Management System (AMS)** - É o agente que executa a supervisão e controle dos acessos à plataforma e uma “biblioteca” ou “diretório” de agentes cujos ciclos de vida são gerenciados por ele.
- **Agent Communication Channel (ACC)** - É o agente responsável pelo contato entre os agentes dentro da arquitetura e os agentes externos. O ACC é o agente que representa a comunicação padrão e que oferece rotinas de mensagens confiáveis. Esse padrão suporta também *Internet Inter-ORB Protocol (IIOP)*. O IIOP é um protocolo que possibilita que programas distribuídos, desenvolvidos em diferentes linguagens, possam se comunicar de forma independente sobre a internet. O protocolo faz parte de uma estratégia da indústria no padrão *Common Object Request Broker Architecture (CORBA)*. Usando o padrão CORBA IIOP e os protocolos relacionados, uma empresa pode desenvolver programas e comunicar com outros sem saber o que o outro executa.

A FIPA tem por missão promover tecnologias e a especificação de interoperabilidade entre diferentes agentes em unidades industriais e comerciais modernas. O trabalho de especificação compreende os seguintes aspectos de um agente: gerenciamento de ciclo de vida, estrutura da mensagem trocada entre agentes, transporte da mensagem, protocolos de comunicação entre agentes, ontologia e segurança. A arquitetura proposta pela FIPA é dividida da seguinte forma:

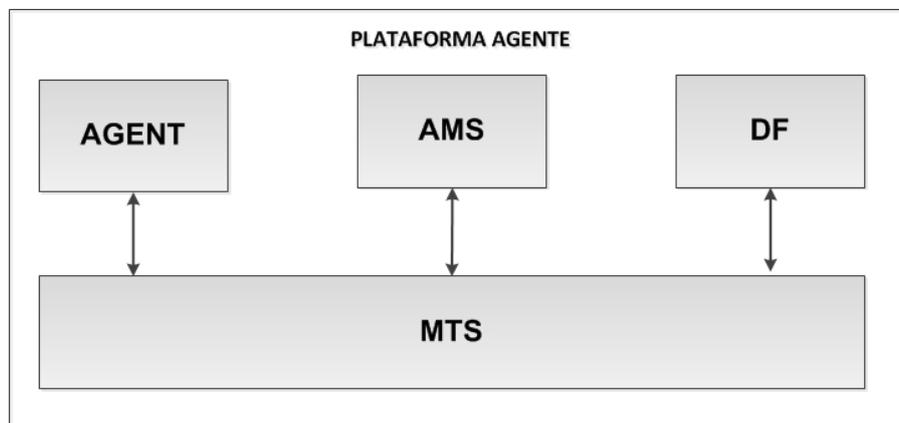


Figura B.3: Arquitetura FIPA

- **Agent** - Representa o programa propriamente dito. Possui todas as características já anteriormente citadas como proatividade, autonomia, reatividade, entre outras. Para essas características sejam realmente implementadas é necessário um “motor” inteligente em seu código. O agente por si só permite apenas um encapsulamento ou uma “auto-contenção”.
- **AMS** - Representa um agente de gerenciamento da plataforma, controlando os acessos e uso dos agentes além de garantir a sua identificação dentro do sistema.
- **Directory Facilitator (DF)** - Representa um serviço de páginas amarelas disponível para os agentes que podem fazer buscas por serviços localizados em outras plataformas. Cada agente pode registrar o seu serviço para que outros também possam buscá-lo, ou seja, representa um serviço de centralização de busca de informações.
- **Message Transport System (MTS)** - Representa o componente da arquitetura que controla toda troca de mensagem entre os agentes dentro da mesma plataforma ou entre plataformas. Esse bloco permite que agentes possam se comunicar com outros em qualquer lugar de uma rede, independente da forma como esses agentes tenham sido codificados porém obedecendo ao padrão FIPA ACL de mensagens. O padrão especifica como transportar estas mensagens sob três tipos de protocolo: HTTP, HTTPS e IIOP.

B.4.1 Mensagens FIPA

As mensagens trocadas entre agentes têm como base um conceito denominado de “atos de fala” [115] que basicamente representa um termo técnico estudado em linguística e filosofia de linguagens. É uma expressão que serve a uma função na comunicação como por exemplo comandar, desejar, descrever.

Esse conceito encontra potenciais aplicações como sistemas inteligentes e comunicação homem-máquina e a ciência da cognição.

Do ponto de vista de agentes, a mensagem deve ser entendida como um determinado nível de interação que o mesmo descreve. Essa interação está relacionada a desejos, crenças e intenções.

A FIPA estabelece um conjunto seletivo de tipos de mensagens, denominadas mensagens performativas, que atende à maioria das aplicações para comunicação entre agentes, descrito na tabela B.2.

Performativa	Significado
ACCEPT_PROPOSAL	Aceite de proposta numa negociação
AGREE	Aceite para desempenhar uma dada ação
CANCEL	Cancela a execução de uma dada ação
CFP	Utilizada para iniciar uma dada negociação (Call For Proposals)
CONFIRM	Confirma a veracidade de uma mensagem
DISCONFIRM	Inverso da mensagem anterior
FAILURE	Tentativa de execução de uma ação (solicitada por outro agente) que não pode ser executada
INFORM	Comunica uma informação para os agentes
INFORM-IF	Informa sobre a veracidade de uma informação
INFORM-REF	Informa sobre um determinado valor
QUERY-IF	Requisita a veracidade sobre uma informação
REFUSE	Recusa a execução de uma ação
REJECT_PROPOSAL	Recusa uma proposta após uma solicitação
REQUEST	Envia uma solicitação a outro agente para que o outro execute uma ação
REQUEST_WHEN	Pede para outro agente executar uma ação quando uma condição for verdadeira
REQUEST	Envia uma solicitação a outro agente para que o outro execute uma ação
REQUEST_WHEN	Pede para outro agente executar uma ação quando uma condição for verdadeira
REQUEST_WHENEVER	Pede para outro agente executar uma ação sempre que uma condição for verdadeira

SUBSCRIBE	Solicitação encaminhada a outros agentes para que o próprio seja notificado sobre determinados fatos ou ações
-----------	---

Tabela B.2: Tabela de mensagens FIPA

Cada uma destas mensagens possui um significado. Com o intuito de ilustrar a aplicação, a mensagem PROPOSE significa submeter uma proposta a um agente para realizar uma determinada operação. Como resposta, o agente pode enviar um ACCEPT_PROPOSAL ou um REJECT_PROPOSAL. Além da mensagem performativa, a mensagem completa contém outros atributos como o protocolo ao qual a mensagem pertence, a ontologia do contexto, a linguagem utilizada para expressar o conteúdo tais como, FIPA-SL, FIPA-KIF , FIPA-RDF), além de outros não citados.

B.4.1.1 Ontologia

Para que os agentes se comuniquem é necessário uma padronização da linguagem com um vocabulário de comum entendimento. É possível que agentes possuam, para a mesma linguagem, entendimentos diferentes. Para que isso não ocorra é necessário que os agentes compartilhem uma mesma ontologia. Existem diversas definições de ontologia. Em [116], ontologia é definida como sendo um modelo computacional de alguma entidade. É normalmente utilizado como uma rede semântica, um grafo cujos nós são os conceitos ou objetos individuais e os arcos representam as relações ou associações sobre esses conceitos. Esta rede é ampliada pelas propriedades, atributos, restrições, funções e regras que governam o comportamento dos conceitos. De uma maneira mais formal, ontologia é a representação de um conhecimento de um domínio específico que de comum entendimento entre os agentes mantendo uma coerência na forma de se comunicar. A cada ontologia listada existe uma ou mais mensagens associadas de tal forma que, unidas às mensagens FIPA, criam uma rede semântica de conhecimento comum dos agentes. Essa rede então permitirá que informações consistentes sejam trocadas entre os agentes para que, de forma coletiva, possam atingir a um objetivo global.

B.5 Softwares para desenvolvimento

Desde a concepção da idéia de sistemas multiagentes (MAS) até o presente momento foram desenvolvidas diversas *softwares* com diferentes graus de complexidade. Esse é um aspecto que representa um grande desafio para a aplicação em redes elétricas

inteligentes, já que não existe um padrão de fato para desenvolvimento.

Atualmente existe um conjunto de *softwares* capazes de criar agentes inteligentes, entre os quais destacam-se: Fipa-OS, Zeus e JADE. Todos eles aderentes ao padrão IEEE FIPA de comunicação. Diversos pesquisadores têm utilizado estes *softwares* para estudos de controle de geração distribuída [117] [118] no âmbito de simulação. Essa preferência por *softwares* aderentes ao padrão do IEEE tem uma motivação fundamental que é a garantia da interoperabilidade entre sistemas. Seguindo o conceito de padrão aberto, o IEEE FIPA define uma linguagem de comunicação de forma abstrata e que atende a maioria das aplicações.

Como foi explicado anteriormente, o fato dos *softwares* serem utilizados refere-se também a capacidade de “encapsulamento” do comportamento de agentes tornando-o de forma independente e isolada. Esta característica encontra fundamentos nas linguagens orientadas a objeto como Java que é a ferramenta básica utilizada para a criação dos agentes. Esse encapsulamento por si só não agrega capacidade inteligente ao agente. Falta complementar dentro do encapsulamento, o código inteligente. Nesse caso, outras ferramentas estão disponíveis para desenvolvimento como Jess [119], uma linguagem desenvolvida em Java e que implementa um conjunto de regras, cuja estrutura é muito similar a linguagem Lisp e Prolog, que é uma outra linguagem destinada ao desenvolvimento de aplicações em inteligência artificial. Esta linguagem permite programação de sistemas complexos com múltiplos paradigmas.

Além destas ferramentas de desenvolvimento destaca-se outra: Mason [120] sendo esta última, de maior complexidade quando comparado com JADE e pode gerar aplicações com maior escalabilidade. Esta também foi desenvolvida em Java. Algumas destas linguagens possuem como característica principal uma série de APIs que facilitam o desenvolvimento de aplicações. O fato de todas serem desenvolvidas em Java é a sua portabilidade, abstraindo-se da arquitetura de hardware além de permitir uma programação orientada a objetos em ambiente distribuído.

B.5.1 JADE

JADE é uma arquitetura para desenvolvimento de sistemas baseado nas especificações da FIPA para interoperabilidade entre sistemas multiagentes e totalmente desenvolvido em Java. Os serviços disponíveis em sua biblioteca consistem de comunicação, de acesso a dados, codificação de mensagens e definição de comportamento dos agentes. A biblioteca possui interface gráfica que permite acompanhar uma série de facilidades entre as quais a inicialização dos agentes assim como acompanhar a troca de mensagens FIPA entre os agentes, permitindo ao desenvolvedor depurar o programa com mais facilidade.

Apêndice C

Tabelas do Arquivo Padrão de Modelos

C.1 Tabela de objetos

A Tabela C.1 ilustra a lista de classes de objetos utilizados no OMT, cujos atributos são publicados e lidos pelos federados Matlab/Simulink e JADE através do RTI. Essas classes são definidas pelos seguintes campos:

- **Nome da classe:** Lista de objetos utilizados pelos federados ;
- **Nome da classe superior:** Classe de objeto hierarquicamente superior e a qual está vinculada;
- **P/A:** *Flag* que indica se o objeto será publicado, assinado, ambos ou nenhum. Nesse caso, todos os objetos são para ser registrados e publicados exceto o *HLAinteractionRoot*;

Classe	Classe superior
<i>HLAobjectRoot</i>	Raiz
RecursodeSistemadePotência	<i>HLAobjectRoot</i>
AgrupamentodeEquipamentos	RecursodeSistemadePotência
Subestação	AgrupamentodeEquipamentos
Equipamento	RecursodeSistemadePotência
EquipamentoCondutordeEnergia	Equipamento
Chave	EquipamentoCondutordeEnergia
ChaveProtetora	Chave
Seccionador	ChaveProtetora
Religador	Chave
BarramentoAC	EquipamentoCondutordeEnergia

Linha com Parâmetros Distribuídos	Segmento de Linha AC
Seção em PI	Segmento de Linha AC
Linha Trifásica	Segmento de Linha AC
Geração Distribuída	Equipamento
Célula Solar	Geração Distribuída
Turbina Eólica	Geração Distribuída
Unidade Geradora	Equipamento
Fonte Trifásica	Unidade Geradora
Máquina Síncrona	Unidade Geradora
Transformador	Equipamento
Trafo com Dois Enrolamentos	Transformador
Sincrofasores	Equipamento
Acumulador de Energia	Equipamento
Célula Combustível	Acumulador de Energia
Bateria	Acumulador de Energia
Bateria Industrial	Bateria
Bateria Automotiva	Bateria
Banco de Capacitores	Equipamento
Consumidor de Energia	Recurso de Sistema de Potência
Carga Trifásica Variável	Consumidor de Energia
Carga Trifásica RLCSérie	Consumidor de Energia
Carga RLCParalela	Consumidor de Energia
Medidor Inteligente	Consumidor de Energia
Aparelho Inteligente	Consumidor de Energia
Ar Condicionado	Aparelho Inteligente
Refrigerador	Aparelho Inteligente
Aquecedor	Aparelho Inteligente
Máquina de Lavar	Aparelho Inteligente
Microondas	Aparelho Inteligente
Nó	<i>HLAobjectRoot</i>
Agente	Nó
Fonte de Tráfego	Nó

Tabela C.1: Tabela de objetos do modelo padrão

C.2 Tabela de interações

A Tabela C.2 representa as classes de interações que ocorre entre os federados. Por definição, interações são tomadas de decisão de um federado que tem efeito ou impacto em outro federado dentro da federação. A estrutura de interações é composta pelos seguintes campos:

- **Classe:** Nome da classe de interação;
- **Classe superior:** Nome da classe hierarquicamente superior e a qual está ligada;
- **P/A:** Se a classe é publicada, assinada, ambas ou nenhuma delas. Nesse caso, todos os objetos são publicados e assinados exceto o *HLAInteractionRoot*;
- **Transporte:** Representa o protocolo de transporte utilizado pelas variáveis. Se for TCP (*HLA Reliable*) ou UDP (*HLA Besteffort*);
- **Ordenação:** Indica o tipo de ordenação que o RTI enviará a classe, isto é, se em ordem cronológica ou sem ordenamento no tempo.

Classe	Classe Superior	Transporte	Ordenação
<i>HLAinteractionRoot</i>	Raiz	<i>HLA Reliable</i>	cronológica
ProcessamentodoAgente	<i>HLAinteractionRoot</i>	<i>HLA Reliable</i>	Ordem de chegada
EnviaMsgFIPA	ProcessamentodoAgente	<i>HLA Reliable</i>	Ordem de chegada
AtivaStatus	ProcessamentodoAgente	<i>HLA Reliable</i>	Ordem de chegada
<i>LeStatus</i>	ProcessamentodoAgente	<i>HLA Reliable</i>	Ordem de chegada
ControleDaSimulação	<i>HLAinteractionRoot</i>	<i>HLA Reliable</i>	Ordem de chegada
InicioSimulação	ControleDaSimulação	<i>HLA Reliable</i>	Ordem de chegada
ParadaSimulação	ControleDaSimulação	<i>HLA Reliable</i>	Ordem de chegada
SuspendeSimulação	ControleDaSimulação	<i>HLA Reliable</i>	Ordem de chegada

Tabela C.2: Tabela de interações do modelo padrão

C.3 Tabela de atributos

A Tabela C.3 ilustra a lista de atributos das classes de objetos, cujos valores podem variar ao longo do tempo, associados às classes de objetos da Tabela C.1. Os valores são atualizados através do RTI e disponibilizados para todos os federados. Os objetos devem estar devidamente cadastrados no arquivo para que os federados troquem informações entre si. No topo da estrutura está o objeto *HLAobjectroot*

que representa uma “super-classe” para todos os objetos registrados. A estrutura é composta pelos seguintes campos:

- **Atributo** - Nome do atributo a qual a classe pertence. Por exemplo, em um sistema trifásico os atributos serão *tensão1*, *tensão2* , *tensão3*;
- **Classe**: Nome da classe de objeto que se associa ao atributo;
- **TipoDeDado**: Representa o formato do dado obtido da tabela de tipo de dados;
- **Atualização**: Representa a política para atualização do atributo. Os valores admissíveis são:
 - **Estático**: O valor do atributo é estático. O federado atualiza o atributo quando é registrado ou quando solicitado para tal;
 - **Periódico**: O federado atualiza o atributo em tempos regulares;
 - **Condicional**: O federado atualiza o atributo quando atende à certas regras;
 - **N/A** - O federado não disponibiliza um valor para este atributo.
- **Condição de atualização**: Se a atualização é periódica, um número de atualizações por unidade de tempo é indicado. Atributos com atualizações condicionais devem ser especificadas neste campo.
- **T/A**: Significa que a propriedade do atributo pode ser transferida (T), utilizando-se o serviço de gerenciamento de propriedade, ou adquirida (A), isto é, ao publicar o atributo a sua propriedade pode ser utilizada pelo próprio federado.
- **P/A**: Esta coluna indica se o federado é capaz de publicar o atributo, assinar, ambos ou nenhum.
- **Tipo de Transporte**: Especifica o tipo de camada de transporte do modelo OSI , que pode ser TCP (*HLAreliable*) ou UDP (*HLAbesteffort*).
- **Ordenação**: Especifica a ordem de entrega dos atributos para os federados. Pode ser entregue fora de ordem ou em ordem cronológica, determinado pela marcação de tempo atribuída quando o atributo foi enviado.

Atributo	Objeto	Tipo	Atualização	T/A	P/A	Transporte	Ordem
Tensão1	RecursodeSistemadePotência	Voltagem	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
Tensão2	RecursodeSistemadePotência	Voltagem	Periódico	No Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
Tensão3	RecursodeSistemadePotência	Voltagem	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
Corrente1	RecursodeSistemadePotência	Corrente	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
Corrente2	RecursodeSistemadePotência	Corrente	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
Corrente3	RecursodeSistemadePotência	Corrente	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
SubsPotAtiva	Substação	Kw	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
SubsPotAparente	Substação	Kva	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
Temperatura	Equipamento	GrauCelsius	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
EqPotAtiva	Equipamento	Mwatts	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
EqPotAparente	Equipamento	Kva	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
StatusChave	Chave	Status	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
Fase	Sincrofasor	grau	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
PosicaoSelecionada	CapacitorEmPoste	SeletordeBanco	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
PotDemandada	ConsumidorDeEnergia	Kw	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
FatordePotencia	ConsumidorDeEnergia	ParametrodePotencia	Periódico	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
NomedoNo	Agente	Nome	Estático	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
NumeroDoNo	Agente	Numero	Estático	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica
MsgFIPA	FonteDeTrafico	Mensagem	Estático	Não Transfere	P/A	<i>HLAreliable</i>	ordem cronológica

Tabela C.3: Tabela de atributos do modelo padrão

C.4 Tabela de parâmetros

A Tabela C.4 apresenta as funcionalidades dos parâmetros de interação entre federados. São utilizadas para associar informações importantes com as classes de interação. O parâmetro pode ser atribuído em qualquer nível hierárquico da classe de atributo e, se for colocado no nível mais elevado da estrutura, será herdado pelas classes inferiores através do mecanismo de herança. A tabela é composta pelos seguintes campos:

- **Parâmetro:** Representa o nome do parâmetro;
- **Interação:** Representa a classe a qual o parâmetro pertence;
- **Tipo de Dado:** Representa o tipo de dado, podendo ser um campo de um dos seis tipos de tabela existentes.

Parâmetro	Interação	Tipo de Dado
NomeDoNó	ProcessamentoDoAgente	Nome
NumerodoNo	ProcessamentoDoAgente	Número do nó
MsgFIPA	EnviaMsgFIPA	Mensagem
A_Status	AtivaStatus	Mensagem
L_Status	LeStatus	Mensagem
IniciaSimulação	IniciaSim	Status
ParaSimulação	ParaSim	Status
SuspendeSimulação	SuspendeSim	Status

Tabela C.4: Tabela de parâmetros

C.5 Tabela de sincronismo

O ponto de sincronismo é um método que suspende a execução do federado até que as execuções de todos os outros alcancem certos trechos de código, a partir dos quais todos estarão sincronizados. A Tabela C.5 apresenta todos os pontos de sincronismo criados para que os federados executem atividades em um mesmo ponto do código. É composta pelos seguintes campos:

- **Rótulo:** Contém um campo do tipo *string* que define o nome do ponto de sincronismo;
- **Tipo de Dado:** Representa o tipo de dado, listado na tabela “tipos de dados”, podendo ser um campo dos seis tipos de tabelas de dados existentes;

- **Capacidade:** Indica o nível de interação que um federado é capaz de cumprir. Os valores que são válidos para esta coluna são:
 - **Register:** Indica que o federado é capaz de iniciar serviços para registrar o ponto de sincronismo;
 - **Achieve:** Indica que o federado está apto para alcançar o ponto de sincronismo;
 - **RegisterAchieve:** Indica que o federado é capaz de registrar e alcançar o ponto de sincronismo;
 - **Nosync:** Indica que o federado não está apto para registrar nem alcançar o ponto de sincronismo;
- **Semântica** - É um campo de *string* e serve como comentário.

Rótulo	Tipo	Capabilidade	Semântica
PontoSinc	<i>HLAASCIIstring</i>	<i>RegisterAchieve</i>	Ponto de sincronismo para todos os federados

Tabela C.5: Tabela de pontos de sincronismo

C.6 Tabela de dados

Das seis tabelas de dados existentes, apenas duas serão utilizadas. No entanto, é importante ressaltar a existência de uma tabela de dados básicos que contempla uma série de representações de tipos de dados predefinidos (*HLAinteger16BE*, *HLAinteger32BE*, *HLAinteger64BE*, *HLAfloat32BE*, *HLAfloat64BE*, *HLAoctetPairBE*, *HLAinteger16LE*, *HLAinteger32LE*, *HLAinteger64LE*, *HLAfloat32LE*, *HLAfloat64LE*, *HLAoctetPairLE*, *HLAoctet*). A representação do tipo *BE* (*Big Endian*) o byte mais significativo é armazenado na posição de menor endereçamento de memória enquanto *LE* (*Little Endian*) o byte menos significativo é armazenado na posição de menor endereçamento de memória. Os tipos contidos nessa tabela servem para definir os atributos e parâmetros. A tabela de dados simples C.6 é utilizada para representar tipos simples de dados e é composta por cinco tipos de dados predefinidos: *HLAASCIIchar*, *HLAunicodeChar*, *HLAbyte*, *HLAinteger64time*, *HLAfloat64time* e é composta pelos seguintes campos:

- **Nome:** Representa o nome do tipo de dados simples;
- **Representação:** Identifica a representação do tipo de dado. Pode ser o nome de uma das representações da tabela de dados básicos;

- **Unidade:** Representa a unidade de medição como, por exemplo, volts, ampères, MW ou outro qualquer;
- **Resolução:** Representa a precisão da medida e especifica a menor diferença possível entre dois valores que possam ser discriminados. Em caso de números em ponto flutuante, a melhor opção é colocar “N/A” na qual a resolução neste caso não se aplica;
- **Acurácia:** Representa o quanto se está próximo do valor desejado. É expresso por um valor dimensional porém pode ser declarado como *perfect*, mesmo para valores discretos ou “NA”, quando não se aplica;
- **Semântica:** Representa um campo de *string* que serve como comentário.

Nome	Representação	Unidade	Resolução	Acurácia	Semântica
Volts	<i>HLAfloat32BE</i>	Volts	N/A	<i>Perfect</i>	tensão
Amperes	<i>HLAfloat32BE</i>	Amperes	N/A	<i>Perfect</i>	Corrente
W	<i>HLAfloat32BE</i>	Watts	N/A	<i>Perfect</i>	Pot. em Watts
Kw	<i>HLAfloat32BE</i>	Kwatts	N/A	<i>Perfect</i>	Pot. em Kw
NumerodoNo	<i>HLAinteger16BE</i>	N/A	N/A	N/A	Num nó
MW	<i>HLAfloat32BE</i>	Mwatts	0.01	N/A	Mwatts
KVA	<i>HLAfloat32BE</i>	KVA	0.01	N/A	KVA
KVAr	<i>HLAfloat32BE</i>	KVAr	0.01	N/A	KVAr
Celsius	<i>HLAfloat32BE</i>	Celsius	N/A	N/A	Temp.

Tabela C.6: Tabela de dados simples

C.7 Tabela de dados discretos

A Tabela C.7 define dados que assumem valores inteiros. A tabela já inclui um único tipo de dado predefinido (*HLAboolean*) que deve estar presente em todos os modelos de dados. Os campos que a compõe são:

- **Nome:** Campo que identifica o nome do tipo de dado;
- **Representação:** Campo que identifica o tipo básico de representação;
- **Especificação:** Texto atribuído ao campo de uma variável discreta como por exemplo, *HLAtrue*;
- **Valores:** É o valor atribuído à variável discreta como por exemplo, “1” para *HLAtrue* ou “0”, para *HLAfalse*;

- **Semântica:** Representa um campo de texto para ser preenchido com comentários.

Nome	Representação	Especificação	Valores	Semântica
<i>HLAboolean</i>	<i>HLAinteger32BE</i>			tipo <i>boolean</i> padrão
ChaveAberta	<i>HLAinteger32BE</i>	<i>HLAfalse</i>	0	chave aberta
ChaveFechada	<i>HLAinteger32BE</i>	<i>HLAfalse</i>	1	chave fechada
SeletorBCap	<i>HLAoctet</i>	PosicaoDoSeletor	0	seletor

Tabela C.7: Tabela de dados discretos