

UNIVERSIDADE FEDERAL DO PARANÁ

RAFAEL VIDAL IASCHITZKI

**CO-SIMULAÇÃO DE SISTEMA DE POTÊNCIA E REDES DE
COMUNICAÇÃO**

CURITIBA

2019

RAFAEL VIDAL IASCHITZKI

**CO-SIMULAÇÃO DE SISTEMA DE POTÊNCIA E REDES DE
COMUNICAÇÃO**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Engenheiro Eletricista, Ênfase em Eletrotécnica, do Departamento de Engenharia Elétrica, da Universidade Federal do Paraná.

Orientador: Prof. Dr. Eduardo Parente Ribeiro

CURITIBA

2019

TERMO DE APROVAÇÃO

RAFAEL VIDAL IASCHITZKI

CO-SUMLAÇÃO DE SISTEMAS DE POTÊNCIA E REDES DE COMUNICAÇÃO

Trabalho de Conclusão de Curso aprovado como requisito parcial à obtenção do título de Bacharel em Engenharia Elétrica, Curso de Graduação em Engenharia Elétrica, Setor de Tecnologia da Universidade Federal do Paraná, pela seguinte banca examinadora.

Orientador Prof. Dr. Eduardo Parente Ribeiro
Departamento de Engenharia Elétrica, UFPR

Prof. Dr. Oscar da Costa Gouveia filho
Departamento de Engenharia Elétrica, UFPR

M. Sc. Felipe Kurpiel José
Departamento de Engenharia Elétrica, UFPR

Curitiba, 22 de novembro de 2019

AGRADECIMENTOS

Agradeço primeiramente a Deus por ter me dado saúde e força nas adversidades.

Ao meu orientador Prof. Dr. Eduardo Parente Ribeiro pelo apoio, sugestões precisas e exemplo profissional.

Aos meus pais Ezero e Rosana Iaschitzki pelo amor, carinho e incentivo incondicional.

Aos meus familiares e amigos que de alguma forma sempre estiveram comigo nos momentos difíceis.

RESUMO

A transição do sistema elétrico atual para um sistema inteligente (ou smart grids, no termo original em inglês) demanda o desenvolvimento de tecnologias capazes de realizar a simulação de uma rede elétrica mais complexa, integrando parâmetros de comunicação e potência. Softwares de código aberto de potência como GridLab-D, Open-DSS, CYMDIST e softwares de comunicação como OMNeT++, Network Simulator 2 e 3, Opnet, já se consolidaram no ambiente prático e acadêmico, e uma forma de realizar sua integração seria útil para utilizar as características de cada um em um sistema inteligente e robusto. A Co-simulação é uma forma de realizar essa integração, e será objeto de estudo desse trabalho. Os softwares utilizados foram o GridLab-D e o OMNeT++, unidos através de uma interface Matlab que realiza a comunicação e a troca de dados entre ambos.

Palavras-chave: Rede inteligente, Co-simulação, Sistemas de Potência, Redes de Comunicação.

ABSTRACT

The transition from the current electrical system to an intelligent system (or smart grids) demands the development of technologies capable of simulating a more complex electrical network, integrating communication and power parameters. Powerful open source software such as GridLab-D, Open-DSS, CYMDIST and communication software such as OMNeT ++, Network simulator 2 and 3, Opnet, have already consolidated into the practical and academic environment, and a way to integrate them would be useful for utilize the characteristics of each in a robust intelligent system. Co-simulation is a way to achieve this integration and will be the subject of this article. The software used was GridLab-D and OMNeT ++, united through an interface that communicates between them.

Keywords: Smart grid, Co-simulation, Power Systems, Communication Networks, Data Exchange.

LISTA DE FIGURAS

Figura 1: Interface GridLAB-D	18
Figura 2: Sincronismo entre simuladores	23
Figura 3: Sincronização Mestre/Escravo	24
Figura 4: Co-simulação de sistemas de Potência e Redes de Comunicação	26
Figura 5: Sistema mestre escravo aplicados aos simuladores GridLAB-D e OMNeT++	28
Figura 6: Diagrama de blocos co-simulação de nuvem transitória	28
Figura 7: Topologia sistema 70 Barras	30
Figura 8: Sistema 70 Barras com usina solar e unidades de armazenamento distribuídas	31
Figura 9: Demanda de potência ativa Sistema 70 Barras	32
Figura 10: Variação da tensão (fase A) Sistema 70 barras	32
Figura 11: Variação da temperatura em Curitiba, 12 de fevereiro de 2014	33
Figura 12: Demanda de potência Sistema 70 Barras com usina solar	34
Figura 13: Potência ativa no período da nuvem transitória	35
Figura 14: Inclusão da Potência de saída do fotovoltaico	36
Figura 15: Topologia Sistema de armazenamento	39
Figura 16: Distância das unidades de armazenamento	39
Figura 17: Associação IEEE 802.11	40
Figura 18: UDP Frame format. Fonte: INET framework Manual (2018)	41
Figura 19: Transmissão RF 10mW	42
Figura 20: Nuvem transitória SE caso 10mW de sinal transmitido	43
Figura 21: Transmissão RF 30mW	43
Figura 22: Nuvem transitória SE caso 30mW de sinal transmitido	44
Figura 23: Transmissão RF 150mW	44
Figura 24: Nuvem transitória caso 150mW de sinal transmitido	45

LISTA DE TABELAS

Tabela 1: Comparativo entre os Softwares de Potência	19
Tabela 2: Comparativo entre softwares de Redes de Comunicação.....	23
Tabela 3: Especificações usina solar	34
Tabela 4: Especificações unidade de armazenamento	37
Tabela 5: Especificações unidade de armazenamento	38

LISTA DE SIGLAS

ANEEL	Agência Nacional de Energia Elétrica
AP	<i>Access Point</i>
COM	<i>Component Object Model</i>
DLL	<i>Dynamic Link Library</i>
GLM	<i>GridLAB-D Model File</i>
GUI	<i>Graphical User Interface</i>
IDE	<i>Integrated Development Environment</i>
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
IP	<i>Internet Protocol</i>
NAM	<i>Network Analysis Module</i>
NED	<i>Network Description</i>
NS	<i>Network Simulator</i>
P&D	Pesquisa e Desenvolvimento
RMS	<i>Root Mean Square</i>
SE	Sistema Elétrico
SSID	<i>Service set Identifier</i>
TCP	<i>Transmission Control Protocol</i>
UDP	<i>User Datagram Protocol</i>
WI-FI	<i>Wireless Fidelity</i>

SUMÁRIO

1 INTRODUÇÃO	13
1.1 OBJETIVOS	14
1.1.1 Objetivo Geral	14
1.1.2 Objetivos Específicos	14
1.2 JUSTIFICATIVA	14
2 REVISÃO BIBLIOGRÁFICA	16
2.1 SIMULADORES DE POTÊNCIA	16
2.1.1 Open-DSS	16
2.1.2 GridLAB-D	17
2.1.3 CYMDIST	19
2.2 SIMULADORES DE REDES DE COMUNICAÇÃO	20
2.2.1 OMNeT++	20
2.2.2 NS2/NS3	20
2.2.3 NetSim	21
2.2.4 OPNet	22
2.3 PARADIGMAS DO SISTEMA CO-SIMULADO	23
2.4 REVISÃO DE ESTUDOS DE CO-SIMULAÇÃO	24
3 MÉTODOS	27
3.1 ARQUITETURA PROPOSTA	27
3.2 ESTUDO DE CASO	29
3.2.1 Nuvem Transitória	29
3.2.1.1 Sistema Elétrico	30
3.2.1.2 Sistema de armazenamento	37
3.2.1.3 Sistema de comunicação (IEEE 802.11)	38
4 ANÁLISE DOS RESULTADOS E DISCUSSÕES	42
4.1 POTÊNCIA DE TRANSMISSÃO 10 MW	42
4.2 POTÊNCIA DE TRANSMISSÃO 30 MW	43
4.3 POTÊNCIA DE TRANSMISSÃO 150 MW	44
5 CONCLUSÃO	46
REFERÊNCIAS	47

1 INTRODUÇÃO

O sistema elétrico atual está passando por uma transição com a introdução das redes elétricas inteligentes (ou *smart grids*, no termo original em inglês). O conceito “redes inteligentes” surge com a inserção de tecnologias no sistema elétrico que envolve comunicação, controle, armazenamento e monitoramento de dados a fim de melhorar o seu desempenho, permitindo assim, a utilização mais eficiente e segura dos recursos. A consequência das redes inteligentes é trazer serviços mais eficientes a um custo menor, com melhor e mais rápida detecção de eventuais faltas no sistema além de uma melhor gestão da energia, beneficiando concessionárias e consumidores. Por conta disso, a Agência Nacional de Energia Elétrica (Aneel) através de seu programa de pesquisa e desenvolvimento (P&D) vem intensificando o incentivo em pesquisas nessa área, o que pode impactar o uso de energia elétrica em breve.

Apesar dos inúmeros benefícios, ainda é incerto os tipos de serviços que realmente o sistema *smart grid* irá oferecer, e como o sistema irá integrá-los para um funcionamento confiável e integrado da rede. Questões como segurança também são temas de discussões, afinal, o fluxo de informações seria altíssimo. Nesse contexto, o desenvolvimento de tecnologias capazes de realizar a simulação de uma rede elétrica mais complexa, integrando parâmetros de comunicação e potência teria enorme importância antes de viabilizar um projeto ou um investimento.

Historicamente, a simulação é uma ferramenta importante para o *design* de sistemas de potência e redes de comunicação, usados para desenvolver e avaliar novas arquiteturas além de permitir o estudo de iterações mais complexas. A simulação é utilizada para reduzir os custos associados a atualizações dos sistemas de energia e redes de comunicação. Outro benefício é a redução do tempo necessário para desenvolver novas tecnologias, oferecendo maior flexibilidade comparada a estudos que dependem de implantações na realidade, portanto, o desenvolvimento de simuladores de redes inteligentes ajuda na melhora do desenvolvimento do processo e permite que os usuários avaliem diferentes soluções.

Atualmente, já existem inúmeras ferramentas de simulação e trazendo para o universo de programas de código aberto, softwares de potência como Open-DSS,

GridLAB-D, CYMDIST e de redes de comunicação como OMNeT++, NS2, NS3 e OPNet são exemplos que já obtiveram êxito no ambiente prático e acadêmico, porém, ainda não existe ferramentas de simulação *smart grid* consolidadas no mercado, e seu desenvolvimento ainda é foco de pesquisa no mundo todo.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

O objetivo geral desse trabalho, é desenvolver um método de co-simulação utilizando softwares de código aberto já consolidados no ambiente de potência e redes de comunicação e integrar suas funcionalidades para desenvolver projetos aplicados em redes inteligentes.

1.1.2 Objetivos Específicos

Entre os objetivos específicos destacam-se:

1. Utilizar eficientemente as funcionalidades pré-existentes nos softwares de potência e redes de comunicação.
2. Desenvolver um meio de integrar as funções de ambos os softwares para que possam ser usados em conjunto.
3. Comunicar os resultados adquiridos de forma simultânea a fim de obter uma gama de resultados aplicados em ambientes inteligentes.
4. Desenvolver um estudo de caso aplicado em redes inteligentes utilizando a co-simulação entre os softwares de potência e redes de comunicação.

1.2 JUSTIFICATIVA

Muitos estudos estão sendo desenvolvidos para simular sistemas elétricos inteligentes, porém demandam um elevado tempo de desenvolvimento e de recursos devido a sua alta complexidade. Além disso, ainda há dúvidas nos serviços que os

sistemas *smart grids* podem oferecer, devido à vasta gama de possibilidades que a tecnologia da rede de comunicação traz ao sistema elétrico, podendo limitar o usuário na hora de realizar simulações. O diferencial deste projeto é utilizar softwares já existentes e consolidados em cada um dos meios destacados e integrá-los na simulação de sistemas inteligentes. A co-simulação também fornece ao usuário a flexibilidade característica deste tipo de sistema, podendo realizar a simulação dos projetos mais diversificados possíveis.

2 REVISÃO BIBLIOGRÁFICA

2.1 SIMULADORES DE POTÊNCIA

A implementação de qualquer co-simulador requer primeiro entendimento e definição das diferentes interações entre os simuladores e o software auxiliar necessário para fornecer interoperabilidade. Esta seção apresenta os principais simuladores de sistemas de potência, bem como suas principais características e funcionalidades.

2.1.1 Open-DSS

O Open-DSS (*Open Distribution System Simulator*) é um software de simulação para sistemas elétricos de distribuição de energia elétrica. O OpenDSS é implementado em duas versões, um programa executável autônomo e um servidor COM (*Component Object Model*) implementado a partir de uma DLL. Na versão autônoma há uma interface para o usuário que permite a descrição de circuitos, a solução e a visualização de resultados. No segundo caso, o OpenDSS pode ser controlado por uma variedade de plataformas de softwares existentes.

O programa suporta a maioria das análises em regime permanente senoidal (Sigla em inglês RMS) comumente realizada para a análise e planejamento de sistemas de distribuição de energia elétrica. Além disso, o OpenDSS é capaz de realizar novos tipos de análises que são necessárias para atender as necessidades futuras relacionadas as redes elétricas inteligentes. Muitos dos recursos encontrados no programa foram originalmente destinados a dar suporte às necessidades das análises de eficiência energética do transporte de energia e análise harmônica. Por fim, o OpenDSS foi projetado para ser indefinidamente expansível de tal modo que possa ser facilmente modificado para atender as necessidades futuras ou para atender as necessidades específicas dos usuários.

O software OpenDSS é um programa que faz usos de linhas de comando para a realização dos cadastros de dados, geração de relatórios, escolha de métodos de resolução, entre outros. As linhas de comando podem ser escritas pelo usuário ou ter origem através de outros softwares. A interface COM, oferece acesso

a algumas funções bem como acesso a solução de banco de dados em estudos. Outra característica é que o software permite que desenvolvedores de software possam customizar o programa criando suas próprias DLL.

A execução e os modelos de circuito do OpenDSS são atualmente escritos em linguagem de programação Delphi que combina a programação orientada a objeto com Pascal, além do rápido desenvolvimento de aplicativo para a plataforma Windows. Uma das razões para escolher Delphi foi a facilidade que a linguagem oferece para a implementação da interface COM. Para a matriz impedância do sistema e matrizes associadas são usados solucionadores de espaçamento implementados em DLL

2.1.2 GridLAB-D

O GridLAB-D é um software de código aberto de simulação e análise de sistemas de distribuição de energia que fornece informações para usuários que projetam e operam sistemas de distribuição ou que desejam tirar proveito das mais recentes tecnologias de energia. Ele incorpora as técnicas de modelagem mais avançadas, com algoritmos de alto desempenho para oferecer o melhor em modelagem de uso final.

Se trata de um software de simulação discreta desenvolvido em linguagem C++, porém, a interface com o usuário é realizada através de arquivos GLM (*GridLAB-D Models*) que define a estrutura do modelo, como condições iniciais e de fronteira do sistema, além de direcionar as entradas e saídas dos dados simulados. Destinado a sistemas de distribuição de energia elétrica, com foco em sistemas inteligentes conhecidos como *Smart-Grid*, o projeto foi iniciado em 2004 pelo Departamento de Energia dos EUA, e atualmente está em sua versão 4.0 (disponibilizada em setembro de 2017), com crescimento constante graças a contribuição industrial e acadêmica.

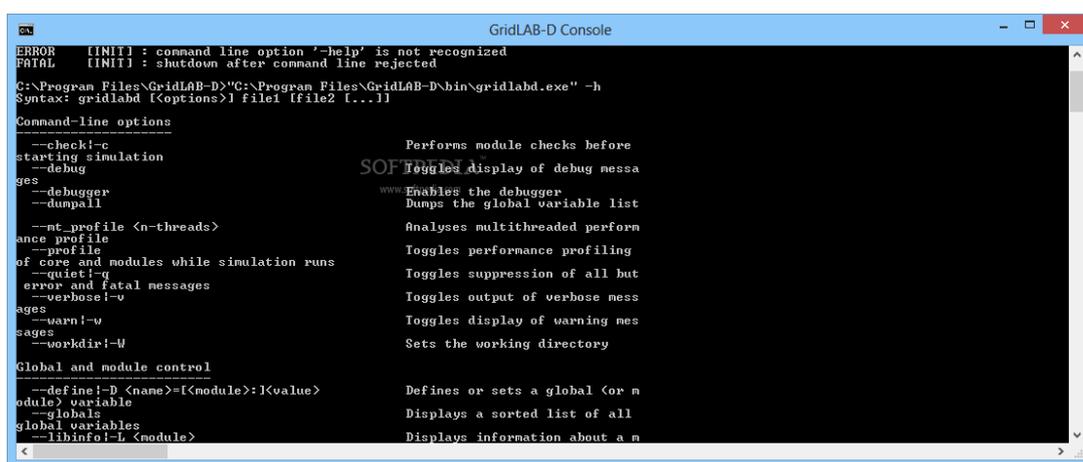
O diferencial do programa é a modelagem de objetos não tradicionais em sistemas elétricos de potência, como residências, prédios, aparelhos eletrodomésticos entre outros. Essas modelagens são carregadas através de módulos que operam como bibliotecas pré-definidas.

A lógica de programação é composta por 6 estruturas básicas:

1. Módulos: Definem as classes e devem ser carregados no programa antes das classes a eles associada.
2. Classes: Podem ser criadas pelo usuário dentro do módulo desejado, desde que seu comportamento esteja pré-definido. Os objetos são agrupados pelas classes de acordo com a similaridade das propriedades.
3. Objetos: São as instâncias das classes que apresentam uma relação de hereditariedade.
4. Players: Altera determinado objeto como uma função do tempo e podem ser implementados em loop.
5. Schedules: Operam similarmente aos players, porém atuam no núcleo de programação do programa, atualizando especificações recorrentes em um tempo específico de execução.

Para executar o programa, o software utiliza um *prompt* de comando que funciona como interface do GridLAB-D, onde a linha de comando deve indicar o caminho do arquivo GLM a ser executado ou o próprio comando diretamente. Um exemplo da interface do programa é visto na Figura 1.

Figura 1: Interface GridLAB-D



```

GridLAB-D Console
ERROR [INIT]: command line option '-help' is not recognized
FATAL [INIT]: shutdown after command line rejected

C:\Program Files\GridLAB-D>"C:\Program Files\GridLAB-D\bin\gridlabd.exe" -h
Syntax: gridlabd [<options>] file1 [file2 [...]]

Command-line options
-----
--check!-c          Performs module checks before
starting simulation
--debug            Toggles display of debug messa
ges
--debugger         Enables the debugger
--dumpall          Dumps the global variable list
--nt_profile <n-threads>
Analyses multithreaded perform
ance profile
--profile          Toggles performance profiling
of core and modules while simulation runs
--quiet!-q        Toggles suppression of all but
error and fatal messages
--verbose!-v      Toggles output of verbose mess
ages
--warn!-w         Toggles display of warning mes
sages
--workdir!-W      Sets the working directory

Global and module control
-----
--define!-D <name>=[<module>]:<value>
Defines or sets a global (or m
odule) variable
--globals          Displays a sorted list of all
global variables
--libinfo!-L <module>
Displays information about a m

```

FONTE: O Autor (2019)

Para a saída de dados, é utilizado a aplicação dos objetos *collector* e *recordes*, que geram uma série temporal dos valores do módulo.

2.1.3 CYMDIST

O software CYME é uma ferramenta focada em modelar todo o sistema de distribuição de energia e atender às necessidades de simulação para análises de capacidade, contingência, qualidade de energia e otimização.

CYMDIST é o pacote base de análise do sistema de distribuição do software CYME. Ele agrupa todas as ferramentas de modelagem e análise necessárias para executar os vários tipos de simulações envolvidas no planejamento do sistema de distribuição elétrica. Os mecanismos de cálculo suportam modelos de distribuição balanceados ou não balanceados que são construídos com qualquer combinação de fases e operados em configurações radiais, em *loop* ou em malha.

Os recursos de modelagem incluem a representação detalhada de todas as partes de uma rede de distribuição, radial ou em malha, bem como o sistema de subtransmissão. Sua interface gráfica personalizável fornece representação, resultados e relatórios no nível de detalhe necessário.

Na Tabela 1 é visualizada uma tabela que mostra um comparativo das funcionalidades dos softwares de potência estudados.

FONTE: Mets, Kevin (2015)

Funcionalidades	CYMDIST	OPEN-DSS	GridLab-D
Análise de fluxo de energia para redes radiais / em loop	Sim	Sim	Sim
Análise de fluxo de potência para sistemas com desbalanceamento cargas	Sim	Sim	Sim
Desequilíbrio do barramento de referência	Não	Não	Sim
Análise de fluxo de potência trifásica / monofásica	Sim	Sim	Sim
Possibilidade de escolha do tipo de técnica de fluxo de potência	-	-	Sim
Suporte à regulação de tensão fotovoltaica no estudo do fluxo de potência	Não	Sim	Sim
Análise de Curto-Circuito	Sim	Sim	Não
Análise de falha de circuito aberto	Não	Sim	Não
Controle de Bancos de Capacitores	Sim	Sim	Sim
Controle do banco de capacitores baseado em nó remoto de detecção	Não	Não	Sim
GIS	Não	Não	Não
Pi equivalente à linha ou cabos de transmissão	Sim	Sim	Sim
Geradores (síncronos)	Sim	Sim	Sim
Regulação de tensão Transformador	Não	Não	Sim
Modelo de geração PV	Não	Sim	Sim

Tabela 1: Comparativo entre os Softwares de Potência

2.2 SIMULADORES DE REDES DE COMUNICAÇÃO

2.2.1 OMNeT++

O OMNeT++ é uma ferramenta de simulação de eventos discretos, de código aberto, extensível, modular e baseada em componentes, como o NS2 e o NS3 para simular redes com e sem fio. O software desenvolvido completamente em C++ é usado principalmente para fins educacionais, pesquisa e na comunidade científica global. Ele oferece um IDE baseado em Eclipse, um ambiente de tempo de execução gráfico e uma série de outras ferramentas. É um simulador de uso geral capaz de simular qualquer sistema composto por dispositivos que interagem entre si. O OMNeT++ fornece uma arquitetura hierárquica, modular e extensível baseada em componentes. Os componentes e módulos são programados em C++ e, em seguida, montados em componentes e modelos maiores usando uma linguagem de alto nível (NED). Ele fornece reutilização do modelo gratuitamente. O OMNeT++ possui amplo suporte a GUI e, devido à sua arquitetura modular, o núcleo (e os modelos) de simulação podem ser facilmente incorporados aos aplicativos. Além da biblioteca do kernel de simulação, o ambiente de simulação contém interfaces de editor de rede gráfica, um compilador NED, interfaces gráficas (Tkenv) e de linha de comando (Cmdenv) para simulação.

A base de protocolos é provida por um *framework* chamado INET com suporte para IPv4, IPv6, TCP, UDP, Ethernet entre outros protocolos.

2.2.2 NS2/NS3

O NS2 (Network Simulator Versão-2) é um simulador de eventos discretos de código aberto desenvolvido especialmente para pesquisas em rede. Entre 1996 e 1997, o NS2 foi iniciado e licenciado para uso sob a *General Public License* (GNU). Ele fornece suporte para simulação com e sem fio de funções e protocolos como TCP, UDP etc. O NS2 foi desenvolvido a partir da ideologia do simulador REAL. O NS2 é um dos simuladores populares devido à sua flexibilidade e comportamento modular. Está escrito em duas linguagens principais: C++ e Linguagem de comando de ferramenta orientada a objeto (OTcl). C++ define o mecanismo interno de objetos

de simulação ao passo que OTcl é usada para os usuários controlarem o cenário de simulação e agendarem os eventos. O C++ e o OTcl são vinculados usando o Tcl. O NS2 fornece uma vasta gama de objetos C++ internos e é recomendável usar esses objetos C++ para configurar uma simulação usando um script de simulação Tcl. No entanto, usuários avançados podem achar esses objetos insuficientes, portanto, eles precisam desenvolver seus próprios objetos C++ e usar uma interface OTcl para montar esses objetos. O NS2 executa a simulação para explorar diferentes problemas, como interação de protocolo, controle de congestionamento, efeito da dinâmica de rede, escalabilidade etc. Ele é executado no tipo certo de cenário que inclui, mas não se limita ao tamanho da topologia, distribuição de densidade, geração de tráfego, distribuição de associação, variação em tempo real da associação, dinâmica de rede etc. O NS2 gera saídas baseadas em texto ou de animação. Para interpretar esses resultados gráfica e interativamente, ferramentas como NAM (Network AniMator) XGraph e Gnuplot são usadas. Para analisar um comportamento específico da rede, os usuários podem extrair um subconjunto relevante de dados baseados em texto e transformá-lo em uma apresentação mais flexível.

2.2.3 NetSim

O NetSim é uma ferramenta de simulação de rede estocástica de eventos discretos usada para experimentação e pesquisa em laboratório de rede. É um software líder de simulação de rede para modelagem e simulação de protocolos, permitindo analisar redes de computadores com profundidade, potência e flexibilidade. O NetSim vem com um ambiente de desenvolvimento integrado, que serve como interface entre o código do usuário, as bibliotecas de protocolos e o núcleo de simulação do NetSim. Além disso, fornece métricas de desempenho de rede em vários níveis de abstração, como rede, sub-rede, nó e um rastreamento de pacote detalhado. Ele possui recursos e funcionalidades exclusivas. O NetSim está disponível nas versões *Standard* ou *Academic* e é construído sobre uma estrutura de design comum de arquitetura e código de alto nível.

2.2.4 OPNet

O OPNET pode ser usado com flexibilidade para estudar redes de comunicação, dispositivos, protocolos e aplicativos. Oferece suporte gráfico relativamente poderoso para os usuários e sua interface do editor gráfico pode ser usada para criar topologia de rede e entidades da camada de aplicativo para a camada física. A técnica de Programação Orientada a Objetos é usada para criar mapeamento desde o design gráfico até a implementação dos sistemas reais. Todos os resultados de configuração e simulação de topologias podem ser apresentados de maneira muito intuitiva e visual. Os parâmetros também podem ser ajustados e os experimentos podem ser repetidos facilmente através do GUI. O OPNet, é baseado em um mecanismo de sistema de eventos discretos, com uma estrutura hierárquica usada para organizar as redes. Este simulador é muito útil quando se trabalha com redes complexas com muitos dispositivos e fluxos de tráfego ou em redes onde uma pequena mudança pode ser crítica. Antes de implementar qualquer alteração, é possível prever o comportamento e verificar as configurações dos dispositivos. O OPNET possui diferentes ferramentas que permitem aos administradores analisar suas redes e as implementações futuras que desejam fazer.

Na

Tabela 2 é visualizado um comparativo das funcionalidades dos softwares de redes de comunicação estudados.

FONTE: Mohammed, Kabir (2014)

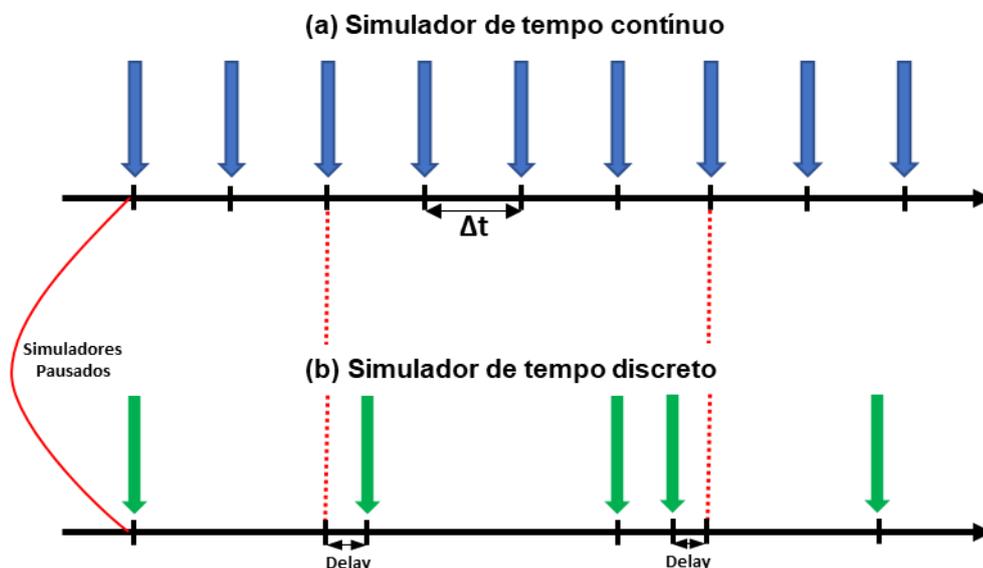
Funcionalidades	NS2	NS3	OMNeT++	OPNET
Licença	Código aberto	Código aberto	Código aberto	Código aberto
Linguagem de Programação	C++ e OTCL	C++ e Python	C++	C e C++
Sistema Operacional	GNU/Linux, FreeBSD, Mac OS X, Windows	GNU/Linux, FreeBSD, Mac OS X, Windows	GNU/Linux, FreeBSD, Mac OS X, Windows	Windows
Suporte GUI	Limitada	Sim	Sim	Sim
Documentação Disponível	Excelente	Excelente	Bom	Bom
Facilidade de uso	Difícil	Difícil	Fácil	Fácil

Tabela 2: Comparativo entre softwares de Redes de Comunicação

2.3 PARADIGMAS DO SISTEMA CO-SIMULADO

Os simuladores de potência e de redes de comunicação tendem a adotar diferentes abordagens de modelagem. A simulação dinâmica de sistemas de potência geralmente usa modelagem de tempo, em que variáveis de estado são descritas como funções contínuas do tempo. Assim, o elemento do sistema de energia dinâmica é expresso por equações diferenciais que definem a relações entre variáveis de estado contínuas. No entanto, algumas dinâmicas discretas são introduzidas por disjuntores, relés etc. Portanto, é utilizada uma abordagem em etapas, uma vez que resolver as equações analiticamente só é possível em casos triviais, são utilizados algoritmos numéricos usando intervalos de tempo discretos. Isso leva ao modelo de tempo ilustrado na Figura 2(a).

Figura 2: Sincronismo entre simuladores



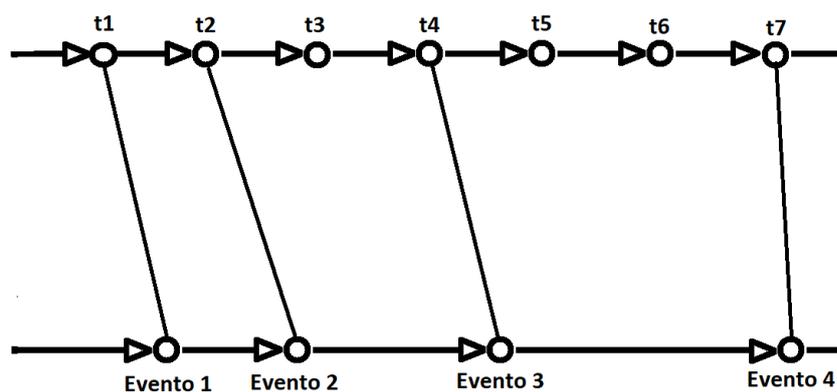
FONTE: O Autor (2019)

As redes de comunicação são tipicamente comutação de pacotes redes (tecnologias baseadas em IP), que são adequadamente modelados como sistemas de eventos discretos caracterizados por eventos como envio e recebimento de pacotes, expiração de temporizadores etc. Tais eventos ocorrem desigualmente

distribuídos no tempo. Isto é claramente diferente da abordagem temporizada comumente usados para simulação dinâmica de sistema de potência, onde um intervalo entre eventos está selecionado. Um planejador de eventos é responsável por manter uma lista ordenada por tempo de todos os eventos e o tempo da simulação progride de um evento para outro esboçado na Figura 2(b).

Uma abordagem para combinar as duas situações é o uso de pontos de sincronização predefinidos, onde cada simulador faz uma pausa quando sua simulação atinge um ponto de sincronização como ilustra a Figura 3. Essa abordagem é chamada de Mestre-Escravo, onde após cada simulador ser pausado, as informações são trocadas. No entanto, isso pode levar a imprecisões da simulação: mensagens que precisam ser trocadas entre os dois simuladores são atrasadas se ocorrerem entre pontos de sincronização. Uma solução para esse problema é reduzir o intervalo de tempo entre os pontos de sincronização (e possivelmente refinando a escala de tempo usada para o tempo contínuo simulador), mas isso claramente prejudica o desempenho. Assim, a co-simulação precisa encontrar o equilíbrio certo entre precisão e velocidade de simulação. Além disso, nem todos os instantes de tempo em que comunicação entre os diferentes simuladores deve ocorrer são conhecidos a priori.

Figura 3: Sincronização Mestre/Escravo



FONTE: O Autor (2019)

2.4 REVISÃO DE ESTUDOS DE CO-SIMULAÇÃO

Vários simuladores podem ser integrados para criar um co-simulador de sistemas de potência e comunicação. Para simular redes de energia, por exemplo, é

possível usar o OpenDSS, GridLAB-D, Matpower, Matlab/Simulink, SimPowerSystems, Modelica, OPAL-RT e RTDS. Para simular redes de comunicação, OMNeT++, NS2, NS3 e NetSim estão entre as ferramentas mais populares.

O simulador de sistema de potência deve ser definido primeiro de acordo com a análise a ser realizada. Por exemplo, o OpenDSS e o GridLAB-D são mais adequados para redes de distribuição de energia, enquanto o Matpower é voltado principalmente para redes de transmissão de energia. No entanto, co-simuladores podem ser construídos para atender a análises completas do sistema de energia (por exemplo, o *Framework Network Simulation* integra Matpower, GridLAB-D e NS-3) e até análises avançadas em tempo real (por exemplo, OPAL-RT com OPNET).

Também podem ser usadas plataformas de co-simulação projetadas para controlar a troca de informações entre as ferramentas de software de potência e comunicação. Por exemplo, o Mosaik (*framework* de código aberto), desenvolvido em Python, foi usado para integrar o OMNeT++ (também de código aberto) com simuladores de rede de potência.

Embora os simuladores de potência e redes de comunicação estejam sendo usados extensivamente nos dois domínios, é a simulação combinada do sistema de potência e rede de comunicação que atraiu mais atenção devido ao crescente interesse de governos, indústria e academia.

A construção de um novo ambiente de simulação combinada é potencialmente demorada e cara. Portanto, o método co-simulado é uma abordagem que combina simuladores especializados já existentes. Nesse contexto, um co-simulador consistiria em um simulador de rede de comunicação especializado (por exemplo, OMNeT++) e um simulador de sistema de potência especializado (por exemplo, GridLab-D). A Figura 4 ilustra a arquitetura de um ambiente com essas características. O principal desafio é conectar, manipular e sincronizar dados e interações entre ambos os simuladores usando as respectivas interfaces dos simuladores.

Especialmente o gerenciamento de tempo entre os dois simuladores é um desafio, porque cada simulador gerencia seu tempo de simulação individualmente. No entanto, a principal vantagem é que os existentes modelos de simulação, algoritmos etc. que já foram implementados e validados podem ser reutilizados. De fato, a maioria do esforço de desenvolvimento é colocada na modelagem de

componentes específicos de redes inteligentes: sistemas como energia fotovoltaica, turbinas eólicas, além de redes de baixa e média tensão. Portanto, em uma co-simulação, essa abordagem reduz o tempo de desenvolvimento e o risco de erros.

Figura 4: Co-simulação de sistemas de Potência e Redes de Comunicação



FONTE: O Autor (2019)

3 MÉTODOS

3.1 ARQUITETURA PROPOSTA

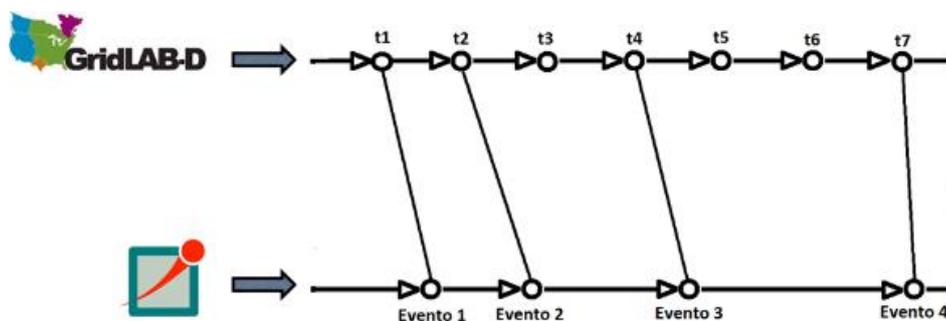
Para a arquitetura proposta, foram escolhidos os softwares GridLAB-D e OMNeT++.

O GridLAB-D, por se tratar de um software destinado ao sistema de distribuição que é o ambiente mais propício para as simulações que necessitem de um ambiente co-simulado e também pela capacidade de modelar elementos não tradicionais do sistema elétrico, como eletrodomésticos, fontes de energia alternativas e unidades de armazenamento, tudo isso aliado a confiabilidade que o software dispõe em conjunto com uma infraestrutura de manuais de utilização e fóruns que trazem segurança em eventuais dúvidas que possam surgir nas simulações, mostrou-se uma opção muito interessante para ser aplicada em co-simulações destinadas aos sistemas inteligentes.

O software de comunicação OMNeT++ é um simulador de redes de código aberto já consolidado no mercado, e conta com um desempenho excelente frente aos softwares que também realizam esse tipo de simulação. Além disso, o OMNeT++ mostrou-se um software amigável para realizar um ambiente co-simulado. As simulações dos protocolos de comunicação serão realizadas através do framework INET que vem instalado junto com o software de OMNeT++.

Como abordado no capítulo anterior, para a sincronização de ambos os simuladores será utilizado o sistema mestre-escravo onde a cada ocorrência da necessidade troca de dados, o simulador mestre é pausado para a troca de informações com o software escravo, e logo após o software mestre continua sua simulação normalmente. Na arquitetura proposta, o simulador mestre será o GridLAB-D e o simulador escravo o OMNeT++ como é visualizado na Figura 5.

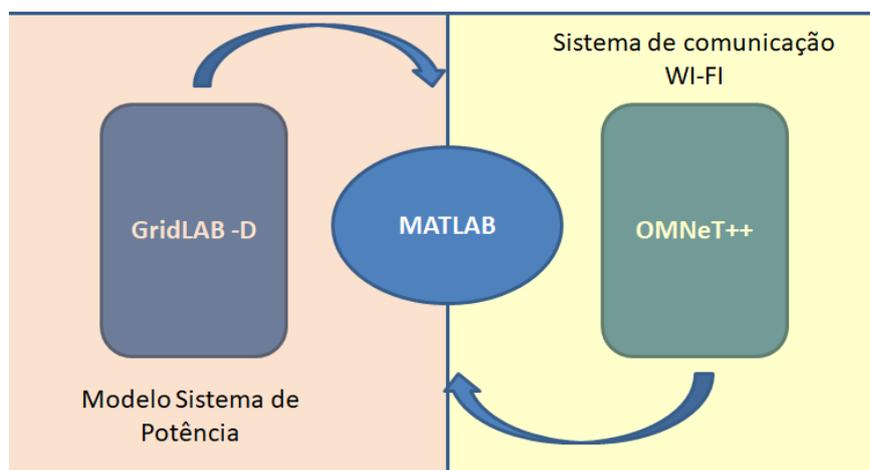
Figura 5: Sistema mestre escravo aplicados aos simuladores GridLAB-D e OMNeT++



FONTE: O Autor (2019)

Para a realização da troca de dados entre os simuladores é necessário um programa que realize a interface entre o GridLAB-D e o OMNeT++. Para isso, será usado o software de programação Matlab. A arquitetura geral da metodologia proposta é vista na Figura 6 que ilustra de maneira simplificada o funcionamento da co-simulação.

Figura 6: Diagrama de blocos co-simulação de nuvem transitória



FONTE: O Autor (2019)

A topologia e configuração do sistema de potência simulado, é modelado pela linguagem GLM no GridLAB-D e o Matlab será o responsável por realizar a escrita desse código e execução do programa GridLAB-D. A partir disso o sistema

de potência estará sendo executado, porém, como a simulação em questão trata-se de um sistema característico de *smart-grid* em determinado momento far-se-á necessário algum dado do sistema de comunicação, nesse momento, o software específico de potência não será capaz de fornecer estes dados, pois é um software exclusivo de sistemas de potência. A interface Matlab, então pausará a simulação de potência, para realizar a simulação do sistema de comunicação utilizado no software OMNeT++, este sim, capaz de realizar simulações de sistemas de comunicação de dados. Após executar a simulação característica do sistema de comunicação, o Matlab então reescreverá o código GLM com a mesma topologia do momento em que este foi pausado, porém, com as configurações dos dados atualizados pelo software de comunicação, e reiniciará a simulação do GridLAB-D no exato instante de tempo em que este foi pausado, finalizando a troca de dados entre ambos os simuladores.

3.2 ESTUDO DE CASO

3.2.1 Nuvem Transitória

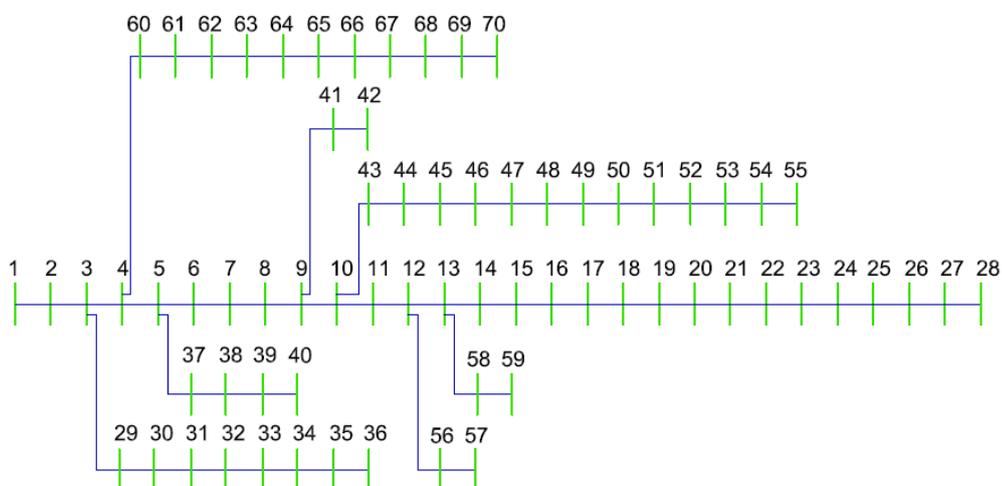
O fenômeno da nuvem transitória (*Cloud Transient*) em painéis fotovoltaicos de pequena escala, como em residências ou sistemas *off-grids*, não se mostra algo preocupante, porém, quanto tem-se o contexto aplicado em usinas fotovoltaicas, a passagem de nuvens durante o dia, afeta consideravelmente o desempenho dos painéis, pela grande área afetada, causando uma queda brusca da potência de saída e uma rápida variação de tensão afetando a qualidade de energia fornecida, e reduzindo substancialmente o tempo de vida útil dos reguladores. Este estudo de caso propõe um sistema elétrico de distribuição de 70 Barras em conjunto com um gerador e uma usina fotovoltaica de 200 kW para suprir a demanda de energia elétrica durante o dia, assim como um sistema de armazenamento de energia que deve despachar a energia armazenada de modo a suprir a queda de potência na ocorrência das nuvens transitórias. Em um sistema característico de distribuição, a unidade de armazenamento não se concentra em um lugar específico, e sim distribuído em pequenas unidades, neste caso foram utilizadas 11 unidades de armazenamento. Nesta situação, deve-se levar em conta a distância entre elas e o

modo de comunicação usado para o controle de seu despacho, a fim de que seja realizado de forma eficiente. A partir disso, faz-se necessário a utilização de um software, para a análise da comunicação entre o controlador e as unidades de armazenamento. Neste relatório, será abordado a co-simulação entre os softwares GridLAB-D e OMNeT++, aplicados no estudo deste fenômeno característico de sistemas *smart grids*, utilizando o modelo de comunicação baseado no IEEE 802.11 também conhecido como rede WI-FI.

3.2.1.1 Sistema Elétrico

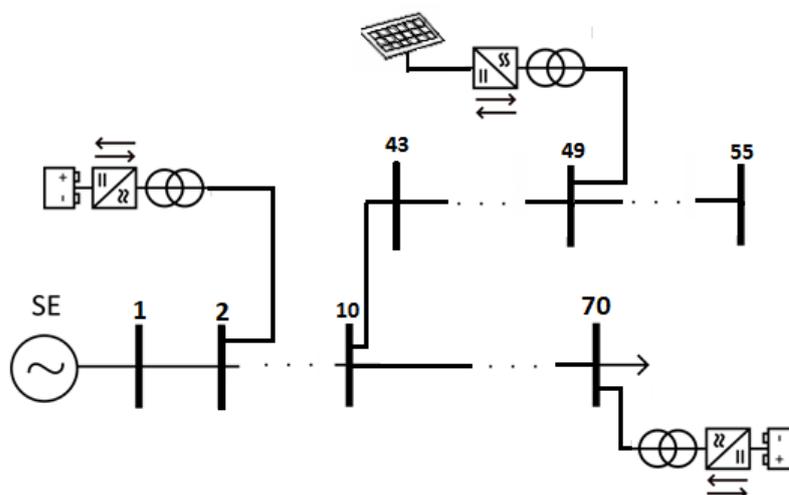
O cenário genérico do sistema de distribuição utilizado é visto na Figura 7 e Figura 8.

Figura 7: Topologia sistema 70 Barras.



FONTE: O Autor (2019)

Figura 8: Sistema 70 Barras com usina solar e unidades de armazenamento distribuídas

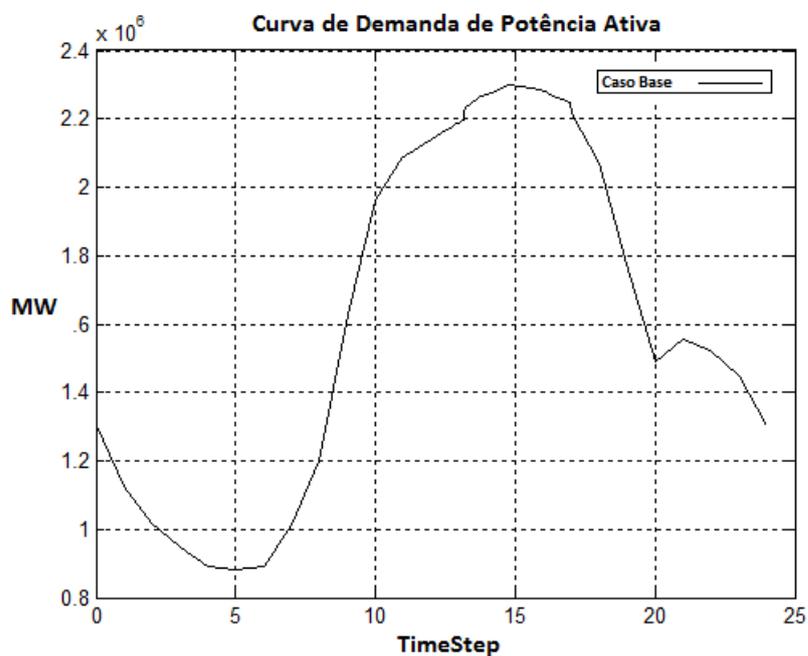


FONTE: O Autor (2019)

Na Figura 7, tem-se a topologia do sistema elétrico de 70 barras [8]. Na Figura 8, observa-se uma simplificação do sistema gerador e usina fotovoltaica, em conjunto com unidades de armazenamento que estão distribuídas ao longo de toda topologia. A co-simulação é realizada entre o software GridLAB-D que realizará a simulação do sistema de potência e o software OMNeT++ que fará a simulação do sistema de comunicação. A interface é realizada através do software Matlab que irá concentrar as informações advindas de ambos os softwares, realizará a troca de dados e efetivará os resultados.

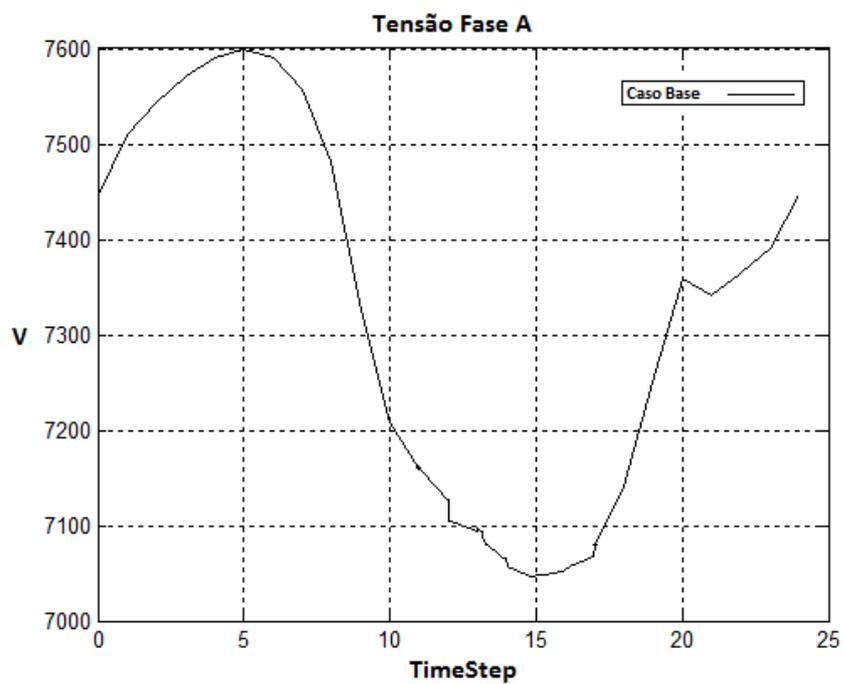
A curva de demanda de potência do sistema 70 barras obtida durante o dia é vista na Figura 4. Nela nota-se uma curva de carga característica do sistema elétrico brasileiro. Nas primeiras horas do dia, a demanda fica abaixo da média diária, ao passo que no período da tarde, a demanda de potência alcança seu pico, isso se deve ao grande consumo de energia elétrica nesses horários, devido, por exemplo, ao uso de ar-condicionado quando as temperaturas se elevam.

Figura 9: Demanda de potência ativa Sistema 70 Barras



FONTE: O Autor (2019)

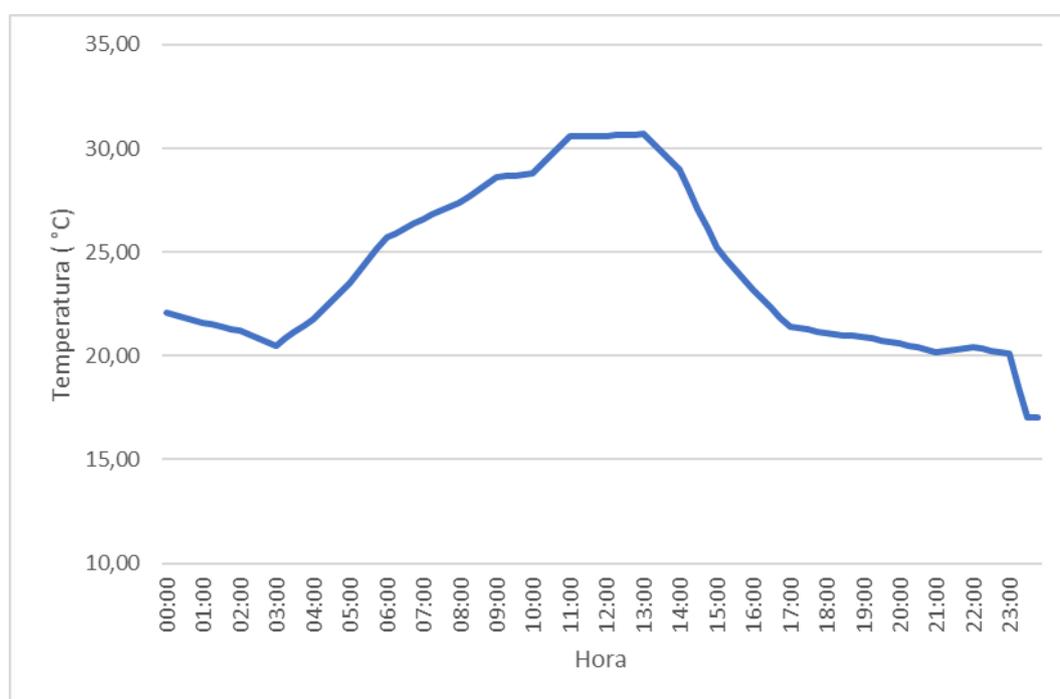
Figura 10: Variação da tensão (fase A) Sistema 70 barras



FONTE: O Autor (2019)

A curva de tensão ao longo do dia é visualizada na Figura 10. Nota-se que no período de maior demanda de potência, a tensão cai de maneira substancial, e o caso oposto observa-se no período matutino. A operação do sistema elétrico é fortemente influenciada pela temperatura devido a dois fatores principais, são eles as perdas associadas as temperaturas dos equipamentos elétricos e o maior consumo de energia devido ao uso do ar-condicionado, como mostra a Figura 11.

Figura 11: Variação da temperatura em Curitiba, 12 de fevereiro de 2014



Fonte: LACHOVICZ, FELIPE (2017)

Para auxiliar no período de maior demanda, foi instalado uma usina solar que terá maior taxa de fornecimento de energia, justamente nos períodos de maior temperatura, deste modo, a energia fotovoltaica auxiliará o gerador nos períodos de maior demanda. A inserção foi feita na barra 49 do sistema, e os dados de temperatura são característicos da região de Curitiba/PR. As configurações da usina fotovoltaica que possui dimensão de aproximadamente 1400 m² são vistas na Tabela 3.

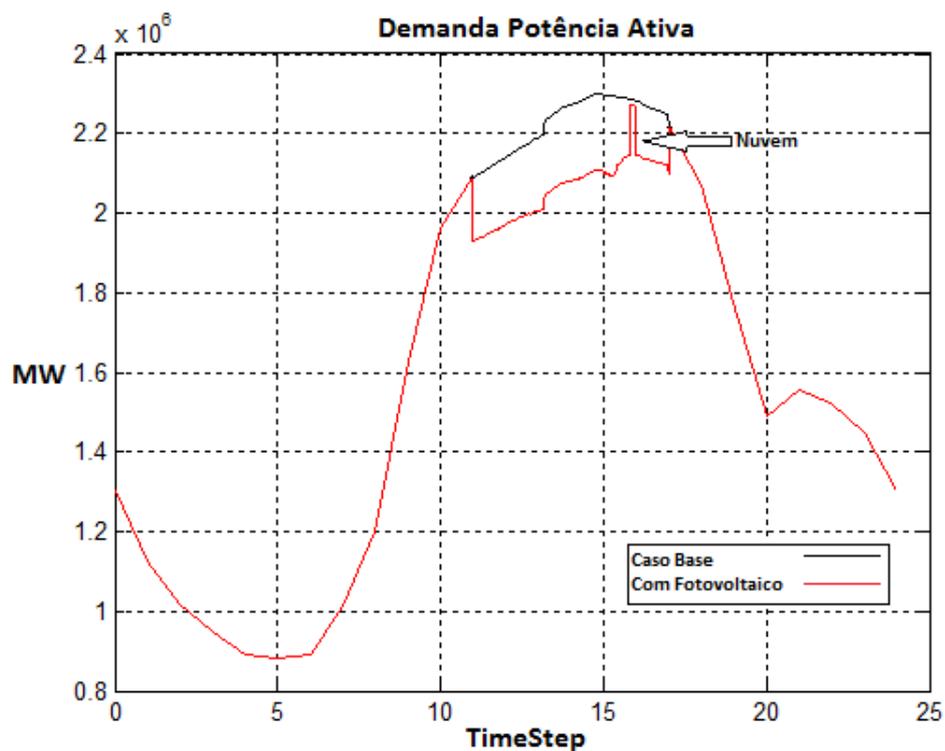
FONTE: O Autor (2019)

Inversor Solar		Painel solar	
Fator de potência	1.0	Tipo do painel	SINGLE CRYSTAL SILICON
Eficiência do Inversor	0.95	Eficiência	0.2
Potência Nominal	200 kW	Área	15000 sf

Tabela 3: Especificações usina solar

No período da tarde foi simulada a passagem de uma nuvem transitória que reduziu os níveis de potência de saída do fotovoltaico significativamente. A Figura 12 mostra a curva de demanda de potência sem e com a usina fotovoltaica, já com o efeito da nuvem transitória.

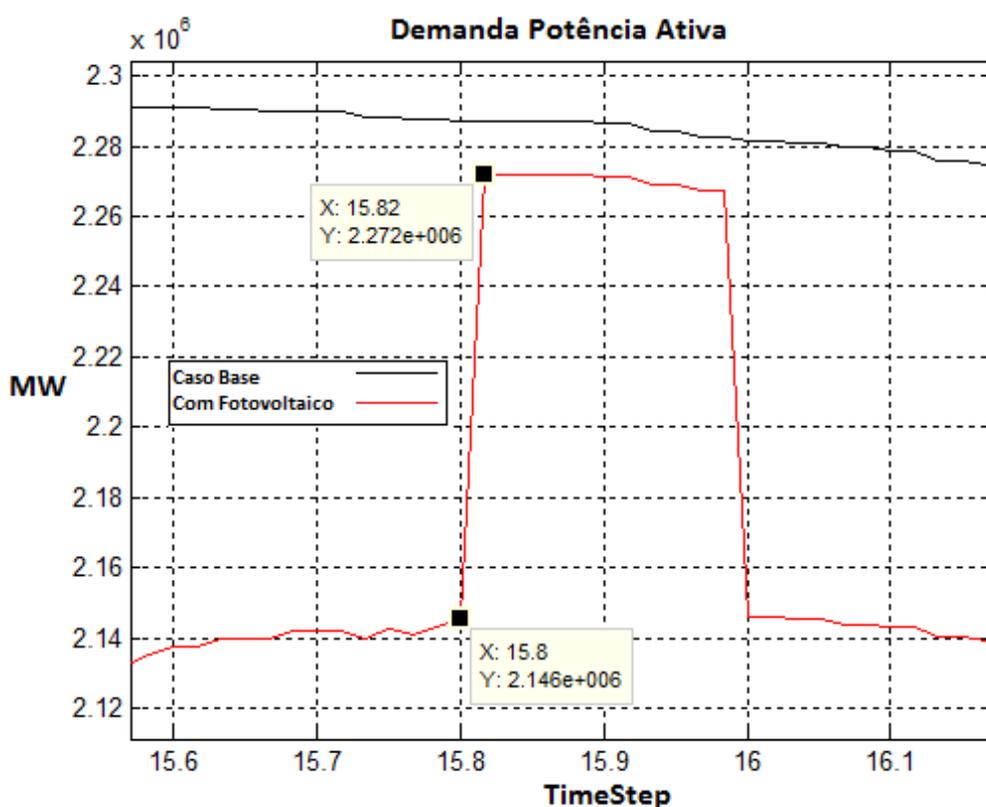
Figura 12: Demanda de potência Sistema 70 Barras com usina solar



FONTE: O Autor (2019)

Devido a limitações do software GridLAB-D a usina foi conectada ao sistema das 10h às 17h pois os dados sobrecarregaram o *buffer* do software caso usadas as 24h de simulação. Futuramente pretende-se prolongar este período. Na Figura 13 tem-se uma visão mais detalhada do comportamento no período da nuvem.

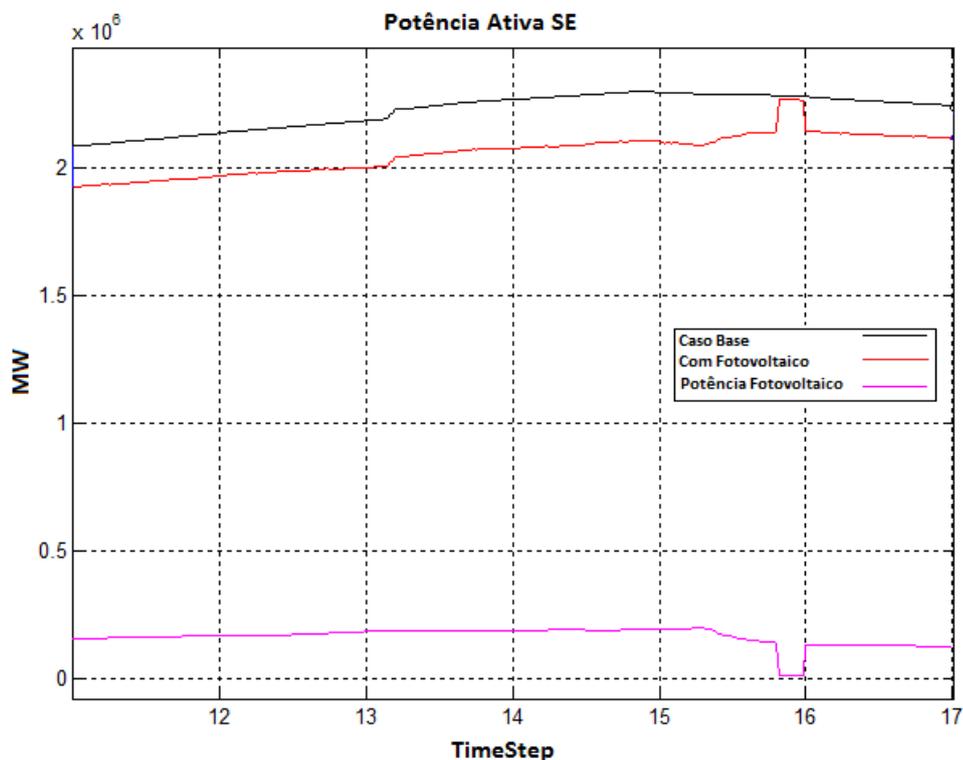
Figura 13: Potência ativa no período da nuvem transitória



FONTE: O Autor (2019)

Através da figura, observa-se uma queda de 126 kW de potência ocasionado pela passagem da nuvem. O objetivo será suprir essa queda pelo sistema distribuído de baterias.

Figura 14: Inclusão da Potência de saída do fotovoltaico



FONTE: O Autor (2019)

A Figura 14 insere o gráfico da potência na saída do painel solar durante o período especificado. Nota-se que está de acordo com o esperado devido ao comportamento da demanda de potência ativa no sistema com o sistema solar incluído.

O sistema de armazenamento que será utilizado foi do tipo *Lithium-Ion*. Este tipo de bateria possui como principal característica, não possuir o efeito memória, ao contrário de baterias de chumbo ácido que possuem o tempo de vida útil prejudicado operando a menos de meia carga.

FONTE: O Autor (2019)

Inversor		Bateria	
Rated power (kW)	150	Tipo	Lithium-Ion
Eficiência	0.95	Potência Nominal (kW)	50
		Tensão Nominal (kV)	0.48

Tabela 4: Especificações unidade de armazenamento

A capacidade total do sistema de armazenamento é de 25 kWh. Após as mensagens de controle ser enviadas até as unidades de estoque, é realizado o despacho de energia a uma taxa aproximadamente 8% por minuto. O controle de carga e descarga é realizado pelo próprio simulador GridLAB-D, onde são setados os gatilhos de disparo do despacho de energia para quando houver a ocorrência de uma queda brusca de potência do painel solar ocasionado por uma nuvem transitória.

3.2.1.2 Sistema de armazenamento

O sistema de armazenamento utilizado foi do tipo *Lithium-Ion*. Este tipo de bateria possui como principal característica, não possuir o efeito memória, ao contrário de baterias de chumbo ácido que possuem o tempo de vida útil prejudicado operando a menos de meia-carga. O estado inicial de simulação das baterias é completamente carregado devido à limitação do software anteriormente citada. Pretende-se futuramente, efetuar o carregamento dela no período de menor demanda do sistema.

FONTE: O Autor (2019)

Inversor		Bateria	
Rated power (kW)	150	Tipo	Lithium-Ion
Eficiência	0.95	Potência Nominal (kW)	50
		Tensão Nominal (kV)	0.48

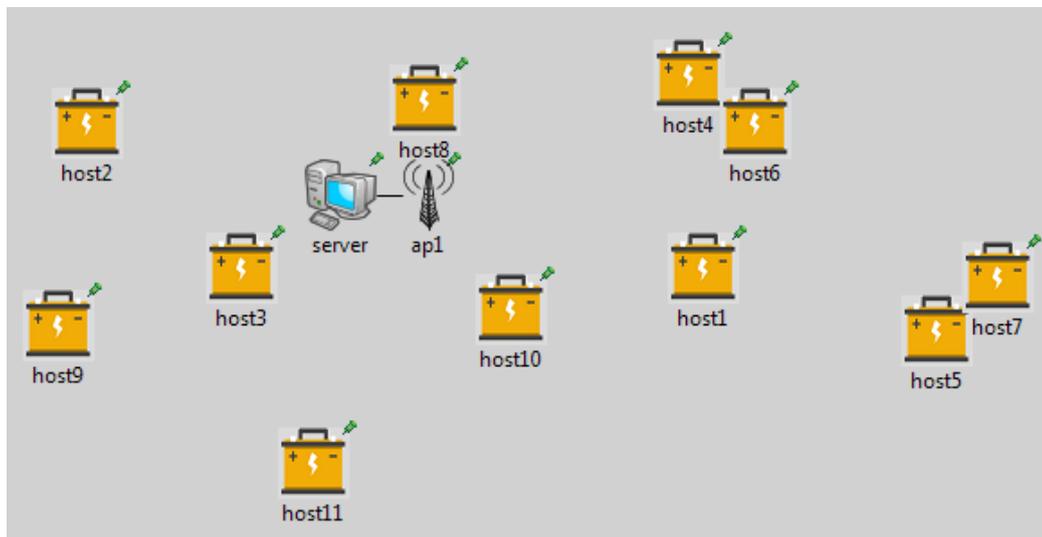
Tabela 5: Especificações unidade de armazenamento

A capacidade total do sistema de armazenamento é de 25 kWh. Após as mensagens de controle ser enviadas até as unidades de estoque, é realizado o despacho de energia a uma taxa aproximadamente 8% por minuto. O controle de carga e descarga é realizado pelo próprio simulador GridLAB-D, onde são setados os gatilhos de disparo do despacho de energia para quando houver a ocorrência de uma queda brusca de potência do painel solar ocasionado por uma nuvem transitória.

3.2.1.3 Sistema de comunicação (IEEE 802.11)

Para a simulação do sistema de comunicação, foi utilizado o software de eventos discreto OMNeT++, e os protocolos de comunicação são providos pelo framework INET. A topologia do sistema de armazenamento distribuído é observada na Figura 15.

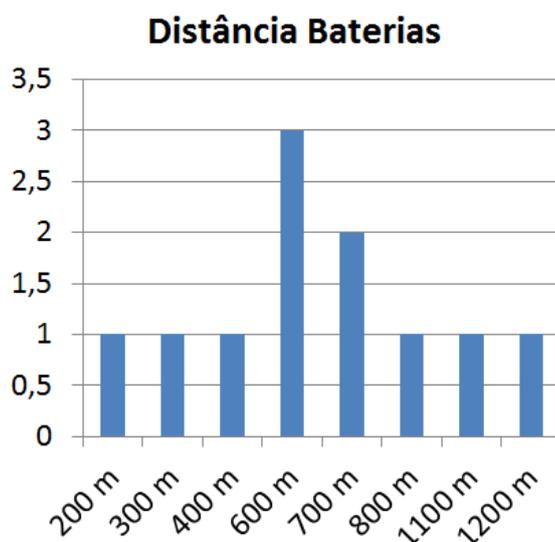
Figura 15: Topologia Sistema de armazenamento



FONTE: O Autor (2019)

As distâncias das unidades de armazenamento do Sistema elétrico são especificadas através do gráfico da Figura 16.

Figura 16: Distância das unidades de armazenamento



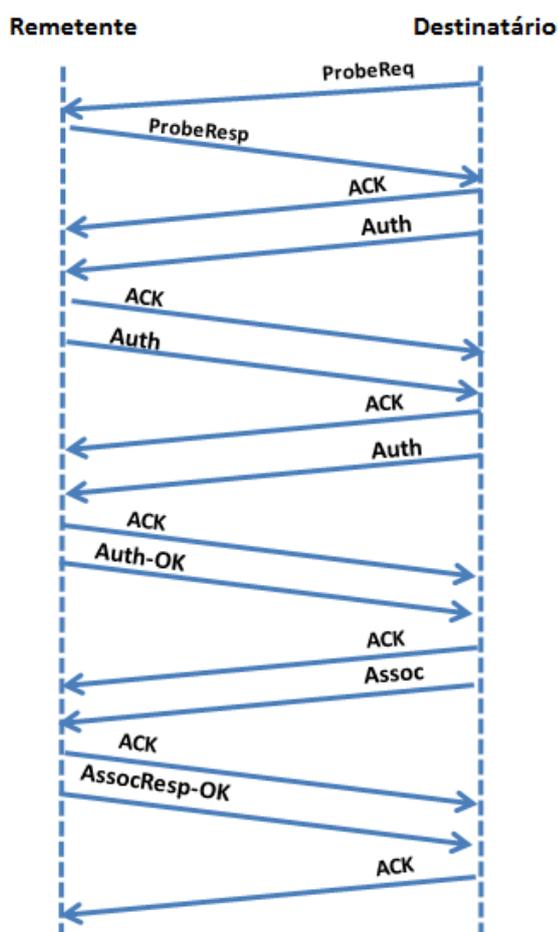
FONTE: O Autor (2019)

O sistema de comunicação utilizado é baseado no modelo IEEE 802.11 ou WI-FI, que define os protocolos da camada física e enlace de dados. Neste modelo,

existem dois modos de associação: ativo e passivo. No modo passivo, os hosts localizam seus AP (*Access Point*) recebendo seus quadros ou *frames bacon* que são sinalizadores de transmissão. Nesta simulação, será usado o modo ativo, onde os hosts transmitem quadros de solicitação de sondagem e os AP's enviam *frames* de resposta. O processo de conexão segue primariamente os passos de autenticação que normalmente é feita por um *password*, e associação que é um pedido do AP com uma série de elementos técnicos e o SSID (*Service set Identifier*) do AP. Os processos do protocolo são mais bem ilustrados na através da .

Figura 17.

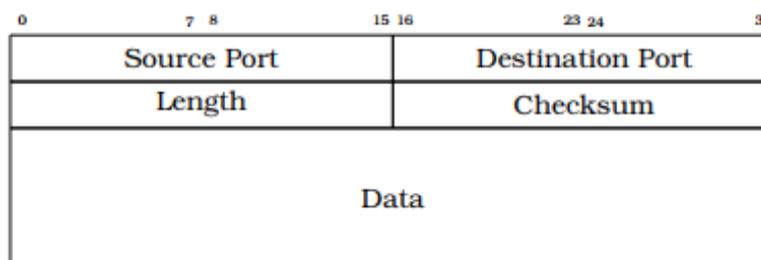
Figura 17: Associação IEEE 802.11



FONTE: O Autor (2019)

A camada de transporte e rede será definida pelos protocolos UDP (*User Datagram Protocol*) e IP (*Internet Protocol*) respectivamente. O protocolo UDP permite que seja enviado um datagrama encapsulado em um pacote IPV4 ou IPV6 a um destino sem a necessidade de conexão, o que o torna mais rápido. Outros protocolos de transporte mais confiáveis podem ser usados como TCP, por exemplo, porém com um aumento do tempo e do tamanho do pacote. Neste estudo de caso, o protocolo UDP mostrou-se eficiente para o caso em pequena escala. Optou-se por encapsular o datagrama em um pacote IPV4. Um exemplo do cabeçalho de um datagrama UDP pode ser visto na Figura 18.

Figura 18: UDP Frame format. Fonte: INET framework Manual (2018)



FONTE: O Autor (2019)

As portas representam os extremos da comunicação que estão alocadas as aplicações que irão enviar e receber datagramas. O campo *Data* é onde estão encapsulados os dados e *Length* é o campo que especifica o comprimento em bytes do cabeçalho UDP mais os dados carregados. O tamanho mínimo é de 8 bytes devido ao comprimento do cabeçalho. O tamanho máximo possível para um datagrama UDP é de 65.507 bytes (65.535 – 8 bytes do cabeçalho UDP – 20 bytes do cabeçalho IP), devido ao protocolo IPV4. Em IPV6 é possível termos pacotes de tamanho maior do que 65.535. *Checksum* pode ser usado para verificação de erros do cabeçalho e dos dados transmitidos. É opcional em IPV4 e mandatório em IPV6. Neste estudo de caso, será usado um comprimento de mensagem de 100 bytes, que é suficiente para transmitir os comandos desejados.

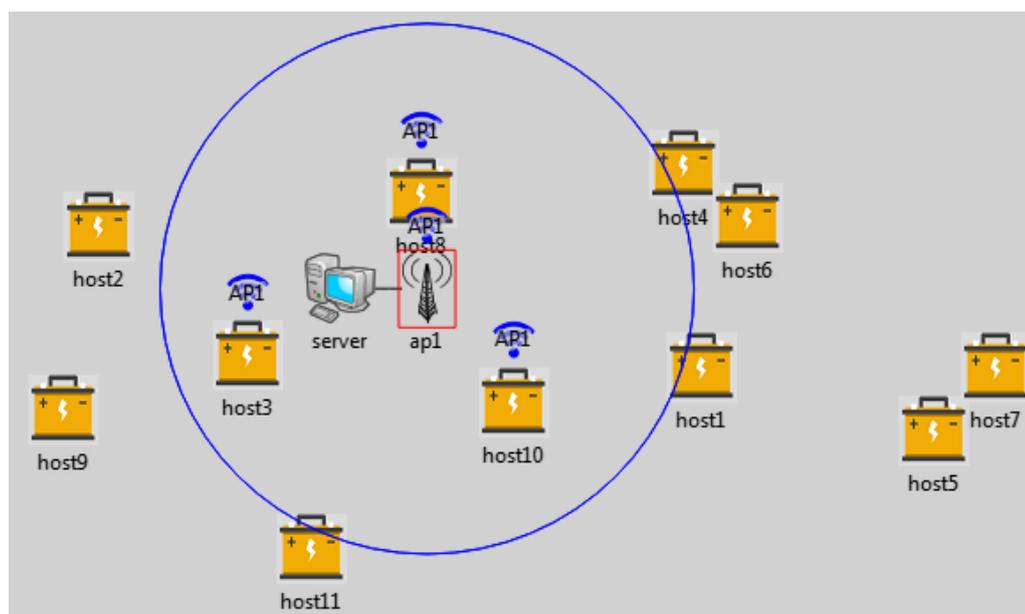
4 ANÁLISE DOS RESULTADOS E DISCUSSÕES

Para a análise do resultado foram simulados três casos de potência transmitida de sinal de rádio de frequência.

4.1 POTÊNCIA DE TRANSMISSÃO 10 mW

Para uma potência de transmissão de 10 mW, apenas 3 unidades de estoque conseguiram estabelecer conexão com o AP, como é ilustrado na Figura 19.

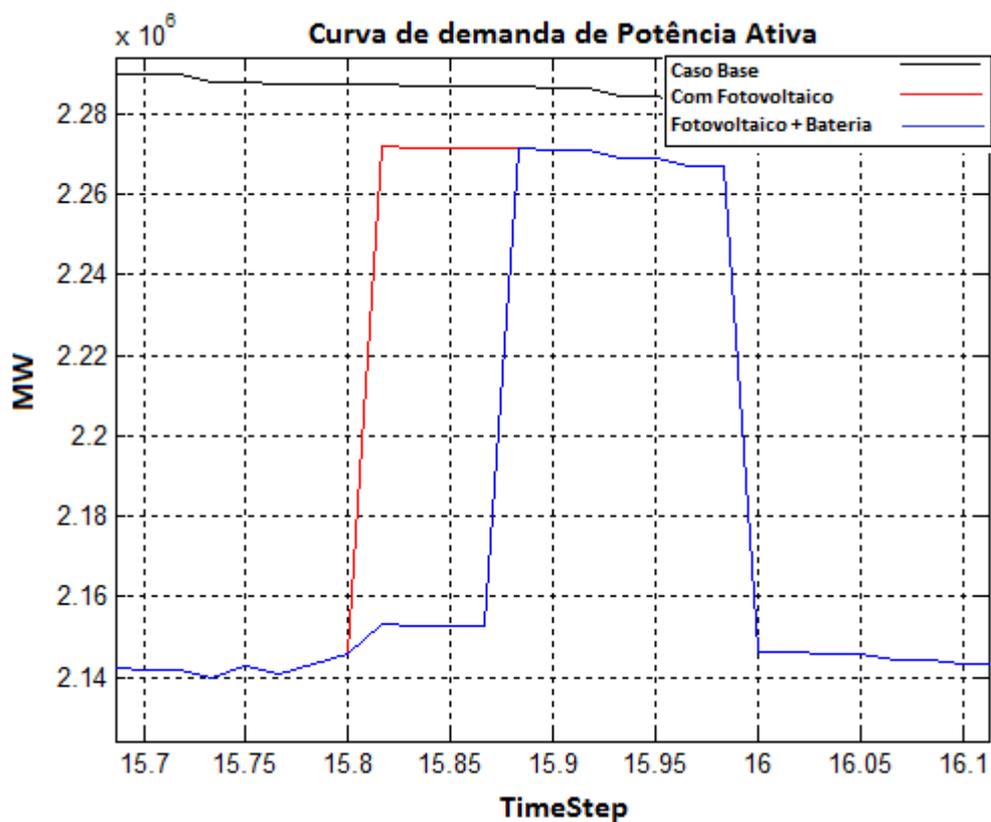
Figura 19: Transmissão RF 10mW



FONTE: O Autor (2019)

Na passagem da nuvem transitória, ocorre um aumento na demanda de potência do sistema devido à diminuição da potência de saída do fotovoltaico, assim, o sistema de armazenamento é acionado através do sistema de comunicação. No caso de uma potência de transmissão RF de 10mW, onde apenas 3 unidades foram acionadas, a saída do SE é resultou na curva da Figura 20.

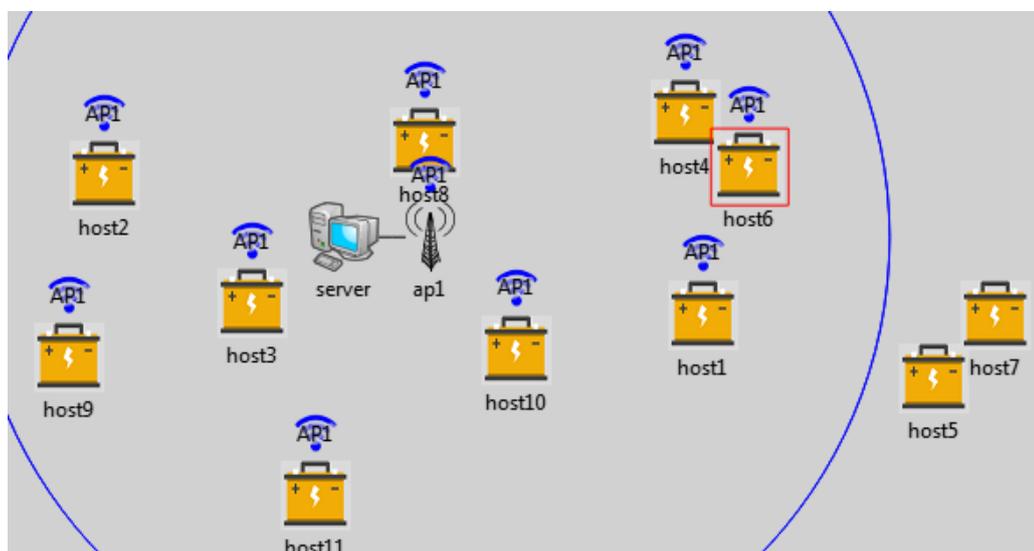
Figura 20: Nuvem transitória SE caso 10mW de sinal transmitido



FONTE: O Autor (2019)

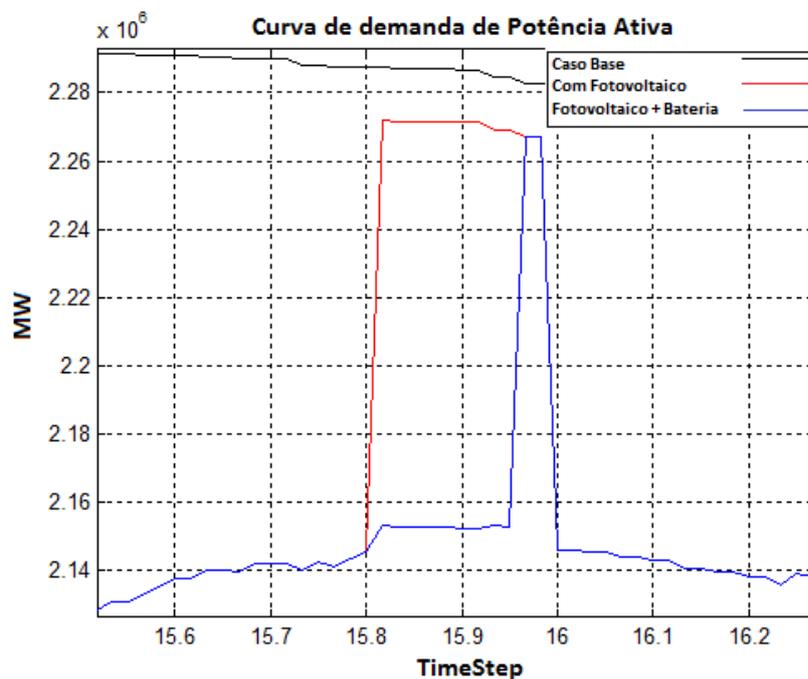
4.2 POTÊNCIA DE TRANSMISSÃO 30 mW

Figura 21: Transmissão RF 30mW



FONTE: O Autor (2019)

Figura 22: Nuvem transitória SE caso 30mW de sinal transmitido

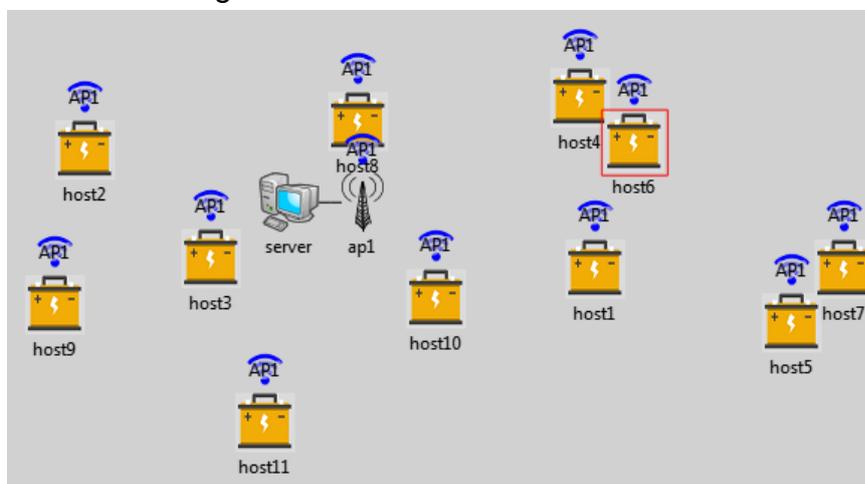


FONTE: O Autor (2019)

No segundo caso, com uma potência de 30mW, 9 unidades de armazenamento foram acionadas para despacho. As unidades host5 e host7 não conseguiram se associar ao AP da WLAN.

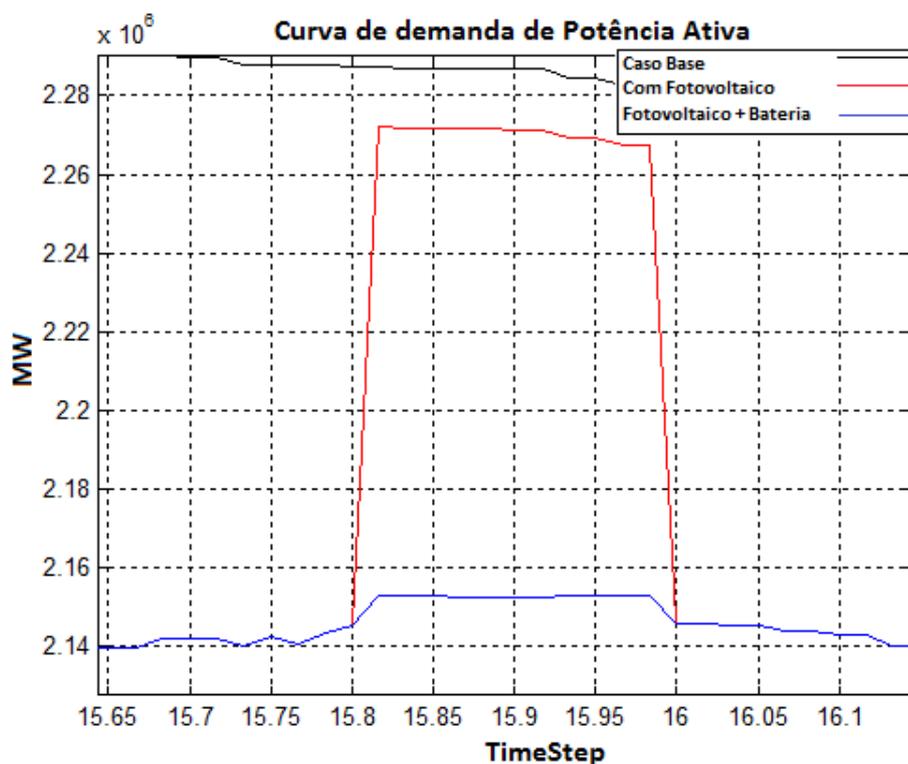
4.3 POTÊNCIA DE TRANSMISSÃO 150 mW

Figura 23: Transmissão RF 150mW



FONTE: O Autor (2019)

Figura 24: Nuvem transitória caso 150mW de sinal transmitido



FONTE: O Autor (2019)

Por fim, no terceiro caso, utilizando uma potência de transmissão de 150mW, todas as unidades de armazenamento foram associadas ao AP, deste modo o efeito da nuvem transitória foi totalmente amenizado pelo sistema de baterias.

5 CONCLUSÃO

Foi implementado uma metodologia de simulação de ambientes *smart grids* baseados na co-simulação de softwares especializados: GridLAB-D e OMNeT++, unidos por uma interface Matlab para realizar a comunicação e troca de dados entre ambos os softwares. Nela, um simulador é utilizado como “mestre” o GridLAB-D e sempre que necessita de dados que dependem de outro software especializado, a interface matlab pausa a simulação “mestre” e executa a simulação “escrava”, ou seja, o OMNeT++, dando rapidez, confiabilidade e liberdade na hora de realizar projetos.

A metodologia de co-simulação utilizando os softwares GridLAB-D e OMNeT++ com auxílio do Framework INET para utilização dos protocolos de comunicação mostrou-se uma alternativa viável para casos deste gênero. Através deste método, é dado ao usuário grande liberdade na hora de criar situações que envolvam os dois contextos estudados, além de fornecer a confiabilidade de simuladores já consolidados no ambiente acadêmico e profissional. Os resultados demonstraram uma aplicação prática do problema que envolve o sistema de potência e sistema de comunicação.

Essa metodologia, fornece ao usuário grande liberdade e confiabilidade para desenvolvimento de projetos que envolvam redes de potência e redes de comunicação, o que não seria alcançado em softwares de redes inteligentes específicos, que demandariam grandes investimentos para a sua implementação.

O estudo de caso abordado ainda pode ser objeto de muitos outros estudos, principalmente no que tange redes de comunicação. Maior aprofundamento no tráfego de dados, perda de pacotes e diferentes protocolos de comunicação podem ser estudados para obter resultados precisos do projeto.

Para trabalhos futuros, pretende-se aprimorar o sistema de potência simulado, por exemplo, no estudo de caso em questão, houveram algumas limitações com o número de dados de irradiação que sobrecarregaram o *buffer* do software de potência GridLAB-D, o que impediu uma análise completa do caso, porém, ainda assim, foram obtidos resultados eficazes na amostra de tempo estudada.

REFERÊNCIAS

- [1] Godfrey, T., Mullen, S., Griffith, D. W., Golmie, N., Dugan, R. C., & Rodine, C. (2010). **Modeling Smart Grid Applications with Co-Simulation. 2010 First IEEE International Conference on Smart Grid Communications**
- [2] Lin, H., Sambamoorthy, S., Shukla, S., Thorp, J., & Mili, L. (2011). **Power system and communication network co-simulation for smart grid applications. IEEE PES Innovative Smart Grid Technologies Conference Europe, ISGT Europe, 1–6**
- [3] Liberatore, V., & Al-Hammouri, A. (2011). **Smart grid communication and co-simulation. IEEE 2011 EnergyTech, ENERGYTECH 2011, (1), 1–5**
- [4] Dede, J., Kuladinithi, K., Förster, A., Nannen, O., & Lehnhoff, S. (2015). **OMNeT++ and mosaik: Enabling Simulation of Smart Grid Communications. CoRR, abs/1509.0(September), 1–4.**
- [5] Troiano, G. O., Ferreira, H. S., Trindade, F. C. L., & Ochoa, L. F. (2016). **Co-simulator of power and communication networks using OpenDSS and OMNeT++.** In IEEE PES Innovative Smart Grid Technologies Conference Europe (pp. 1094–1099).
- [6] ANEEL. **Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional (PRODIST): Módulo 3 – Acesso ao Sistema de Distribuição.** 1. ed. rev. e atual. 2017a.
- [7] F.J. Lachovicz, A. N. Tabata, F. S. Retorta, G. M. S. Pereira, A. R. Aoki, L. D. Toso, M. M. C. Dias **“Electrical Evaluation of Battery Energy Storage System use in Distribution Feeder for Peak”**, “International Seminar on Policies, Incentives, Technology and Regulation os Smart Grids., 2017.

- [8] M. E. Baran and F. F. Wu, “**Optimal capacitor placement on radial distribution systems**,” IEEE Trans. Power Deliv., vol. 4, no. 1, pp. 725–734, 1989.
- [9] BALFOUR, John. **INTRODUÇÃO AO PROJETO DE SISTEMAS FOTOVOLTAICOS**. [s.l]: Ltc, 2016.
- [10] CENTRO DE GESTÃO E ESTUDOS ESTRATÉGICOS (CGEE). **Redes Elétricas Inteligentes: Contexto Nacional**. 2012.
- [11] T. Godfray, S. Mullen, R. C. Dugan, C. Rodine, D. W. Griffith, N. Golmie “**Modeling Smart Grid Applications with Co-Simulation**,” 2010.
- [12] JOÃO TAVARES PINHO (Org.). **MANUAL DE ENGENHARIA PARA SISTEMAS FOTOVOLTAICOS**. Rio de Janeiro: Cepel, 2014.
- [13] A. M. A. K. Abeygunawardana and G. Ledwich, “**Estimating benefits of energy storage for aggregate storage applications in electricity distribution networks in Queensland BT** - 2013 IEEE Power and Energy Society General Meeting, PES 2013, July 21, 2013 - July 25, 2013,” 2013
- [14] ZILLES, Roberto et al. **SISTEMAS FOTOVOLTAICOS CONECTADOS À REDE ELÉTRICA**. [s.l]: Oficina de Textos, 2012.