

Desenvolvimento de uma Ferramenta Computacional para Aplicação num Estudo de Qualidade da Energia Elétrica Associada a Geração Solar Fotovoltaica

Sandy Aquino dos Santos* Matheus Neves Carvalho*
Daniel Barbosa* Renato José Pino de Araújo*
Kleber Freire da Silva*

* *Departamento de Engenharia Elétrica e de Computação, UFBA,
(e-mail: sandyufba@gmail.com, matheusvca@gmail.com,
dbarbosa@ufba.br, renato.pino@ufba.br, kfreire@ufba.br)*

Abstract: The study of new sources of electric power generation, especially those that meet the recent environmental and sustainable requirements is essential, since electric power is fundamental for economic and social development. In this context, the insertion of photovoltaic systems in the distribution network stands out since it does not require the installation of large substations or long transmission lines. However, due to its intermittent operational characteristic, it is necessary that there is a constant monitoring of the quality of the electrical energy to guarantee the continuity and reliability of the electrical energy in the established operational parameters. This work proposed to demonstrate the construction of a graphical interface in Python together with the OpenDSS software that could contribute to the analysis of solar photovoltaic systems' impact in the electrical power quality, especially on steady state voltage. As a preliminary, a simple study was carried out in order to demonstrate its potential.

Resumo: O estudo de novas fontes de geração de energia elétrica, em especial, aquelas que atendam as novas exigências ambientais e sustentáveis é essencial, uma vez que a energia elétrica é fundamental para o desenvolvimento econômico e social. Nesse contexto, a inserção de sistemas fotovoltaicos na rede de distribuição se destaca já que não requer a instalação de grandes subestações ou longas linhas de transmissão. Contudo, devido a sua característica operacional intermitente, é necessário que haja um monitoramento constante da qualidade da energia elétrica para garantir a continuidade e a confiabilidade da energia elétrica nos parâmetros operacionais estabelecidos. Este trabalho propôs demonstrar a construção de uma interface gráfica em python em conjunto com o *software* OpenDSS que contribua futuramente para análises de impacto dos sistemas solares fotovoltaicos na qualidade da energia elétrica, sobretudo tensão de regime permanente. Como prévia realizou-se um estudo simples de caráter demonstrativo de sua potencialidade.

Keywords: Python, OpenDSS, Electricity Quality, Photovoltaic System, Graphical User Interface.

Palavras-chaves: Python, OpenDSS, Qualidade da Energia Elétrica, Sistema Fotovoltaico, Interface Gráfica de Usuário.

1. INTRODUÇÃO

Segundo o Escritório Técnico de Estudos Econômicos do Nordeste (ETENE), a fonte solar tem elevado a sua competitividade no Brasil, mostrando-se como uma alternativa para diversificação da matriz energética (Bezerra, 2018). A utilização da Geração Solar Fotovoltaica (GSFV) tanto na forma de Geração Centralizada (GC), quanto na forma de Geração Distribuída (GD), tem sido bastante incentivada, pois alinha os propósitos de segurança energética com as preocupações ambientais (Bezerra, 2018).

A GSFV conectada aos Sistema de Distribuição (SD), introduz novas questões na operação dessas redes. Tipicamente, o circuito apresenta uma configuração radial e, portanto, grande parte de suas proteções operam somente observando uma direção no fluxo de carga. Além disso, a GSFV é uma fonte intermitente o que pode causar possíveis problemas em torno da Qualidade da Energia Elétrica (Lucchese et al., 2018).

Além disso, o avanço tecnológico contínuo impulsiona uma expansão progressiva do Sistema Elétrico de Potência (SEP), implicando na necessidade de planejamento por parte das concessionárias, pois as mesmas devem garantir

* Este trabalho contou com o apoio financeiro da Fundação de Amparo à Pesquisa do Estado da Bahia (FAPESB).

os níveis adequados de qualidade de serviço e do produto entregue aos consumidores finais (Barbosa, 2018).

Nesse cenário, o desenvolvimento de ferramentas computacionais que simulem o comportamento do sistema elétrico torna-se imprescindível, visto que o avanço em termos de metodologia, equipamentos e modelos devem ser disponibilizados aos profissionais do setor o mais rápido quanto possível (Agostini et al., 2002).

Contudo, a maioria dos *softwares* conhecidos voltados para sistemas elétricos de potência estão baseados na primeira geração das linguagens de alto nível, o que traz desafios quanto à manutenção e às atualizações, sobretudo quando aumentam as suas dimensões e complexidade (Manzoni, 2005).

Destaca-se também, que a maioria dos programas consolidados no setor elétrico são de caráter comercial, os chamados *softwares* proprietários, que só podem ser obtidos por meio de licenças com um alto custo de aquisição, inviabilizando muitas vezes a sua utilização, sobretudo para usos com fins educacionais (Oliveira et al., 2019).

Como alternativa a esse segmento de *softwares* com licenças comerciais ou protetoras, surgiu o conceito de programas gratuitos e de código aberto, conhecido como *softwares* open-sources (FOSS, do inglês *Free and Open Source Software*). Esse conceito traz como principais características o acesso livre à informação bem como à decisões realizadas em domínio público, promovendo a descentralização, simultaneidade e colaboração, permitindo ao programador a possibilidade de uma completa modificação e adaptação do programa para adequação aos problemas estudados (Sarkinen, 2007).

Um exemplo de software livre é o simulador OpenDSS destinado a estudos aplicados a sistemas de distribuição. O programa é mantido e atualizado atualmente por um grupo de engenheiros do EPRI (Electric Power Research Institute). Desta forma, objetivo deste trabalho foi desenvolver uma ferramenta para análise de sistemas elétricos de potência em linguagem de programação Python utilizando o software OpenDSS como processador de cálculo.

O objetivo deste trabalho é mostrar o desenvolvimento de uma ferramenta computacional educacional que auxilie nos estudos em sistemas elétricos de potência, utilizando como resultado um estudo prévio aplicado a QEE numa rede simples, demonstrando assim as possibilidades da ferramenta.

2. O SOFTWARE OPENDSS

O OpenDSS é um *software* de código aberto utilizado como uma ferramenta de simulação no domínio da frequência. O mesmo pode ser utilizado para criar modelos de sistemas de distribuição de energia elétrica e partir daí realizar análises relacionadas ao planejamento de distribuição, tais como geração distribuída, análise de harmônicos e qualidade de energia. Entretanto, o simulador não realiza simulações de transitórios eletromagnéticos (domínio do tempo) (Dugan, 2016).

O OpenDSS foi implementado em duas versões, a primeira corresponde a um programa executável autônomo e a segunda a um servidor COM (*Component Object Model*) im-

plementado a partir de uma *Dynamic-link library* (DLL). Como os seus desenvolvedores já reconheciam a imprevisibilidade de todas as necessidades que os usuários teriam, isso fez com que uma interface COM fosse implementada através da versão DLL do programa para que usuários que se aprofundassem no conhecimento sobre o *software* pudessem realizar novos estudos de maneira aprofundada. De forma simplificada, a interface COM, através da versão DLL, permite que o OpenDSS possa ser controlado por outros programas como Python ou MATrix LABoratory (MATLAB), por exemplo.

3. INTERFACE GRÁFICA

A construção de uma *Graphical User Interface* (GUI) é importante para propiciar uma interação mais amigável com o usuário facilitando o entendimento dos resultados que podem ser explorados. Para otimização do desenvolvimento de uma GUI, é fundamental a escolha de *frameworks*, pois há um maior aproveitamento do tempo de concepção, visto que as bibliotecas já dispõem de recursos necessários para a caracterização dos elementos que compõem a GUI, como botões, barra de ferramentas, *menus*, dentre outros.

Desta forma, construiu-se uma interface gráfica que permite ao usuário simular redes de distribuição. É possível escolher tanto redes padrões IEEE, quanto redes geradas a partir da Base de Dados Geográficas das Distribuidoras, o BDGD (ANEEL, 2018). Um dos módulos construídos realiza a conversão das informações presentes no BDGD para o formato reconhecido pelo OpenDSS, permitindo que sejam realizados estudos específicos em alimentadores reais, como pode ser observado na Fig. 2.

Na construção da interface gráfica utilizou-se o Qt, um *framework* multiplataforma, criado originalmente na linguagem “C++” para desenvolver aplicações *desktop*. Entretanto, como a Python foi a linguagem de programação utilizada para o desenvolvimento deste plano de trabalho e por haver potencial de portabilidade do Qt, utilizou-se “*bindings*” do Qt em Python, o PyQt. A interface gráfica é sucintamente apresentada pela Fig. 1.

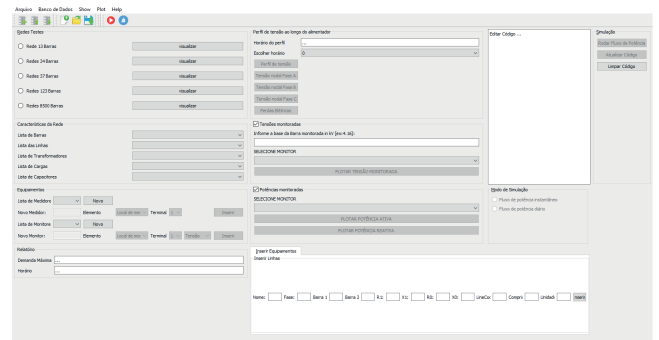


Figura 1. Interface gráfica desenvolvida e utilizada para obtenção dos resultados.

A linguagem Python foi escolhida dentre outras linguagens de programação conhecidas na literatura pelo seu potencial crescimento de usabilidade. Junta-se a isso, o fato de ser de código aberto, ser de alto nível e possuir uma sintaxe

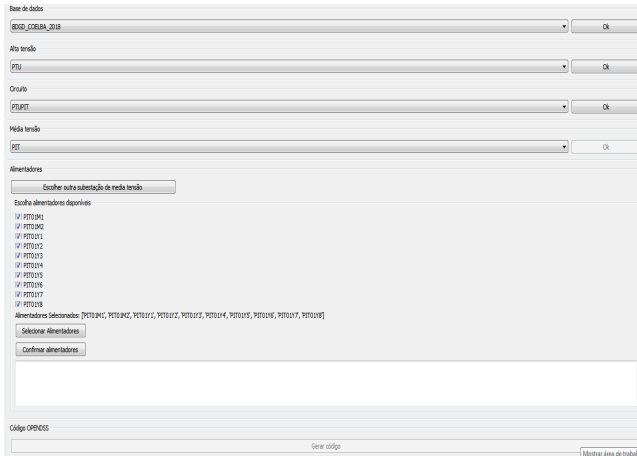


Figura 2. Módulo Conversor BDGD/OpenDSS.

clara e concisa que favorece a legibilidade do código fonte, tornando a linguagem mais produtiva (Borges, 2014).

Como o OpenDSS é um *software* de código aberto, foi possível sua implementação diretamente na linguagem de programação Python, que disponibiliza um conjunto de funções da biblioteca OpenDSS. Tornar o OpenDSS nativo em Python permite que entusiastas da cultura livre criem interfaces específicas para suas aplicações utilizando a linguagem. Chamado de *OpenDSSDirect.py* o pacote mencionado é multiplataforma e está disponível para Windows, Mac e Linux (PYPI, 2019).

Como os dados gerados pelo OpenDSS são fornecidos em arquivos de texto, ou arquivos .csv, é possível manipular tais dados de forma conveniente a obter-se resultados desejados, filtrando informações que a priori podem ser descartadas para alguma aplicação específica. Dessa forma, podem ser gerados gráficos, tabelas e análises conforme desejado, sendo, portanto, versátil à necessidade do programador. Destaca-se ainda que todos os resultados gerados nessa pesquisa são provenientes da interface gráfica desenvolvida.

A linguagem Python, em contraponto à metodologia de construção de *softwares* da primeira geração, apresenta-se como uma opção promissora para enfrentar os novos desafios da produção de ferramentas computacionais para a indústria da energia elétrica. Trata-se de uma linguagem de programação interpretada, orientada a objetos, utilizando-se da filosofia de Modelagem Orientada a Objetos (MOO) (Manzoni, 2005).

A MOO visa a construção de uma estrutura de dados sólida e consistente, a partir da qual são implementados softwares flexíveis, com alto grau de reutilização dos códigos e com facilidades para manutenções e atualizações (Manzoni, 2005).

A versão alpha do *software* proposta neste trabalho está definida em 7 módulos que serão descritos no decorrer do mesmo e todos foram desenvolvidos utilizando-se do paradigma da Programação Orientada a Objetos.

3.1 Módulo 1: Seleção de redes testes

O módulo indicado pela Fig. 3 está configurado para seleção da rede que o usuário pretende simular. Essa rede

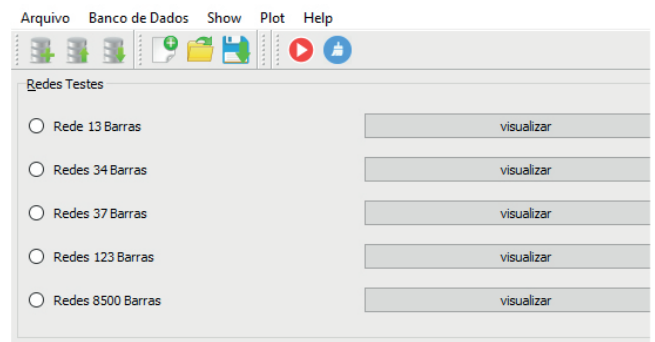


Figura 3. Módulo responsável pela seleção de redes pré definidas pelo software.

pode ser uma das redes testes pré definidas pelo software, sendo selecionada por um *radiobutton* indicado na imagem ou através de uma seleção de um arquivo .dss escolhido pelo usuário utilizando o botão vermelho da barra de ferramentas.

3.2 Módulo 2: Seleção do modo de simulação do fluxo de potência

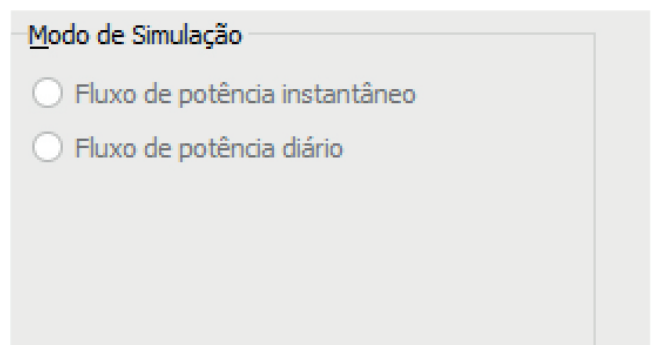


Figura 4. Módulo responsável pela seleção do modo de simulação do fluxo de potência.

Neste modulo, Fig. 4, o usuário escolhe o tipo de fluxo de potência que ele deseja realizar na simulação: O fluxo de potência instantâneo que realiza apenas um fluxo de potência considerando uma única condição de carga específica ou o Fluxo de potência diário que realiza 24 fluxos de potência, simulando cada hora do dia, associando a condição de carga a uma curva de carga pré definida e também os dados de irradiação solar nestes períodos de tempo.

3.3 Módulo 3: Realizar fluxo de potência, atualizar ou limpar código

O módulo 3, indicado pela Fig. 5, é responsável por realizar o fluxo de potência no modo selecionado e limpar a simulação anterior para realização de uma nova.

Após essa fase inicial da pré-configuração de simulação, onde o usuário indica qual rede pretende simular, seleciona qual tipo de fluxo de potência e realiza o mesmo, visualizando as principais características do sistema, ficam disponíveis os outros 2 módulos subsequentes que estão associados aos resultados e análises.

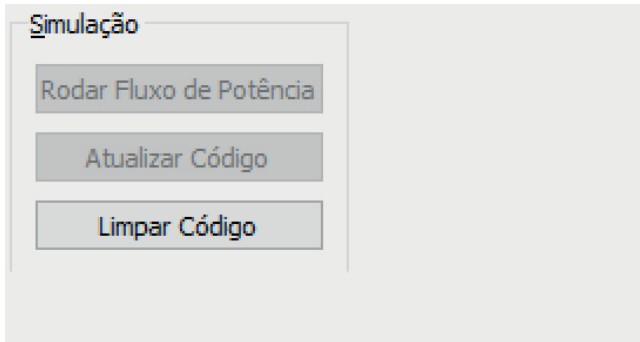


Figura 5. Módulo responsável pela realização do fluxo de potência.

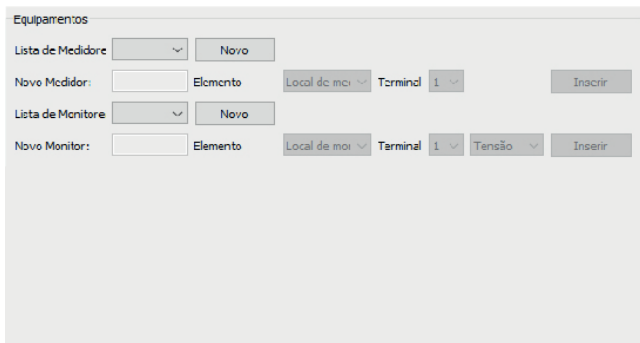


Figura 6. Módulo responsável por inserir equipamentos medidores e monitores no sistema simulado.

3.4 Módulo 4: Equipamentos medidores e monitores

Para registrar grandezas elétricas como tensão e potência é necessário que o usuário insira na rede objetos medidores de energia. Essa opção é construída pelo módulo 4 composto das seguintes etapas:

- Criar novo medidor;
- Nomear medidor;
- Selecionar na rede onde deseja-se inserir o medidor;
- Finalizar o processo de inserção de medidor.

Para mostrar de forma gráfica as grandezas registradas pelos medidores inseridos é necessário que o usuário insira objetos monitores. Essa opção é construída pelo módulo 5 composto das seguintes etapas:

- Criar novo monitor;
- Nomear monitor;
- Selecionar na rede onde deseja-se inserir o monitor;
- Selecionar na rede a opção de monitoramento (tensão ou potência);
- Finalizar o processo de inserção de monitor.

3.5 Módulo 5: Visualização de Gráficos

3.6 Módulo 6: Visualização de Gráficos de Potência e Tensão Monitoradas.

Os módulos 5 e 6 são responsáveis por dinamizar as análises, pois oferecem ao usuário a opção de visualizar gráficos associados ao sistema, colaborando para que o mesmo entenda o comportamento da rede sob determinada condição de operação auxiliando na tomada de decisão.

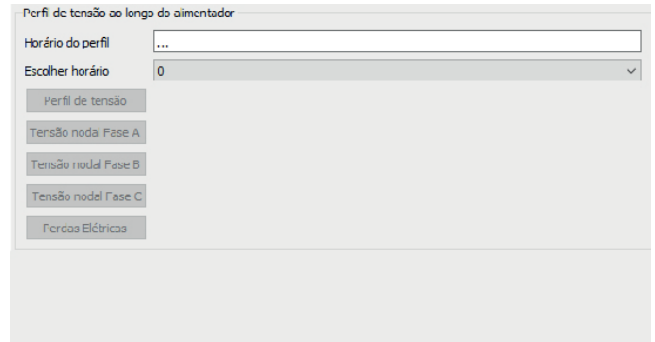


Figura 7. Módulo responsável por visualizar gráficos gerais - Perfil de Tensão - Tensões Nodais- Perdas Elétricas.

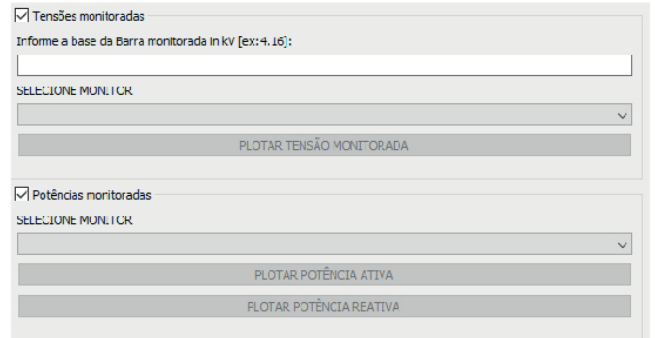


Figura 8. Módulo responsável por visualizar grandezas associadas aos monitores inseridos no sistema simulado.

Para demonstrar a potencialidade da *Graphical User Interface* (GUI) desenvolvida foi realizado um estudo de caso na área da qualidade da energia elétrica associado ao contexto da Geração Solar Fotovoltaica.

4. APLICAÇÃO DA FERRAMENTA NUM ESTUDO DE CASO ASSOCIADO A QUALIDADE DA ENERGIA ELÉTRICA

4.1 Análise para rede com Geração Solar Fotovoltaica Concentrada

Com o objetivo de avaliar o funcionamento da interface criada, foi realizado um estudo de caso que analisa o comportamento da rede IEEE 13 barras, apresentada na Fig. 9. A rede é bastante conhecida na literatura, possuindo tensão nominal 4,16kV, demanda total de aproximadamente 4MVA e fator de potência 0,85 adiantado (Carvalho et al., 2012). É possível testar várias configurações de rede, neste caso será realizada a análise da variação da tensão em regime permanente ao longo do dia, comparando cenários com e sem unidades de geração solar fotovoltaica concentrada no final do alimentador

Os cenários simulados são apresentados na Tabela 1 e a análise realizada será baseada nos critérios: acompanhamento da potência ativa fornecida pela subestação, das perdas técnicas do sistema e na variação de tensão de regime permanente durante o período analisado no ponto de acoplamento.

Ressalta-se que o nível de penetração de geração fotovoltaica é de 40%, 50%, 60%, sendo índice de penetração definido como a razão entre a potência nominal da geração

fotovoltaica e a potência aparente de pico da carga total do alimentador, conforme indica (Lopes et al., 2015).

Um exemplo da curva de potência para um dia típico ensolarado do SFV considerado nas análises pode ser observada na Fig. 10 (Carvalho et al., 2012).

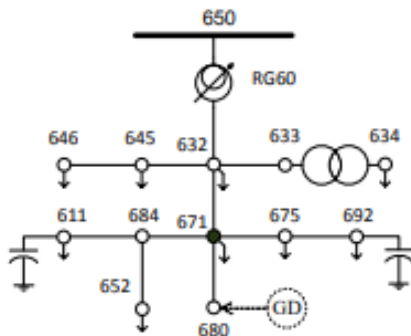


Figura 9. Redes de teste 13 barras, predefinida pelo software.

Comparação da Potência exportada pela subestação. Ao analisar a Fig. 11, nota-se que a potência exportada pela subestação ao longo do dia diminuiu proporcionalmente, em relação ao caso base, a medida que aumenta-se o índice de penetração do SFV na rede. Contudo, a potência no horário da demanda máxima, em torno das 21h, permanece inalterada em todos os cenários, como esperado, pois neste instante não há geração do SFV.

Comparação das Perdas Elétricas. Conforme observado na Fig. 12 a medida que o índice de penetração aumenta a perda elétrica média reduz seu valor em 9,2% , 11,0% e 12,5% respectivamente em relação ao caso base apresentado no primeiro gráfico da figura. O fato explica-se pois com a presença do SFV menos corrente elétrica percorre o alimentador, visto que parte da carga é suprida localmente.

Comparação da variação de tensão no Ponto de Acoplamento do SFV. Verifica-se por meio da Fig. 13 que ocorre uma elevação de tensão no ponto de acoplamento do SFV no horário onde há inserção de potência pelo sistema fotovoltaico. Além disso, nota-se que a elevação de tensão é proporcional ao índice de penetração. Contudo, o perfil da curva permanece inalterado nas horas onde não há geração solar fotovoltaica, o que indica que as alterações percebidas são decorrentes de fato da influência do SFV.

Ao finalizar as simulações descritas acima ressalta-se que o grande diferencial da interface proposta é a potencialidade de adaptações que ela oferece, além de sua interação mais intuitiva e amigável com o usuário. Por exemplo, na interface do software OpenDSS não é possível visualizar os gráficos de tensão em valores por unidade, também não é possível visualizar gráficos de perdas elétricas diárias. Ou seja, a união da linguagem de programação Python e o *opendssdirect.py* tem um grande potencial para desenvolvimento de aplicações específicas, como as citadas e o módulo construído para conversão do BDGD.

5. CONCLUSÃO

A crescente necessidade e preocupação com a QEE exige um maior esforço nas análises e nos estudos da mesma. No

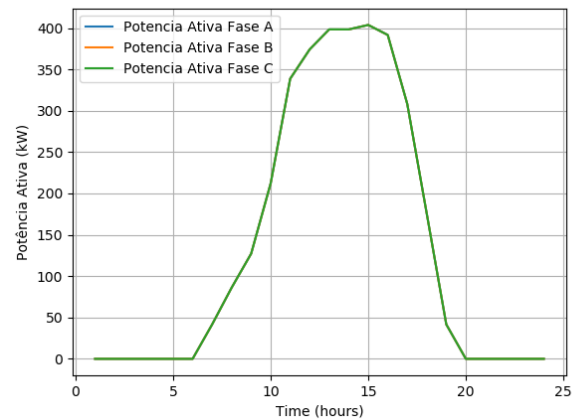


Figura 10. Curva de potência típica de um dia ensolarado.

contexto das fontes de energias intermitentes, ferramentas computacionais, por meio da sua alta capacidade e velocidade de processamento, agilizam resultados e tornam-os mais precisos. Sendo assim, a utilização e desenvolvimento de *softwares* que facilitam e aproximam os resultados do usuário, desempenha um importante papel na melhoria da análise da QEE. A interface desenvolvida, embora na versão embrionária, além de ser uma potencial ferramenta para essas análises propostas, durante seu processo de desenvolvimento foram apreendidos diversos conceitos de programação por partes dos autores.

AGRADECIMENTOS

Agradecemos a Fundação de Amparo à Pesquisa do Estado da Bahia, FAPESB, pelo financiamento fornecido para desenvolvimento da pesquisa, bem como a Liga Acadêmica de Sistemas Elétricos de Potência da Escola de Engenharia Elétrica da Universidade Federal da Bahia pela infraestrutura fornecida durante o desenvolvimento do projeto.

REFERÊNCIAS

- Agostini, M.N., Decker, I.C., and Silva, A.S. (2002). Nova filosofia para o projeto de softwares na área de sistemas de energia elétrica utilizando modelagem orientada a objetos. *CBA. Natal*, 1200–1205.
- ANEEL (2018). Prodlist: Módulo 10 – sistema de informação geográfico regulatório.
- Barbosa, M.d.F.L. (2018). Avaliação Probabilística do Impacto da Microgeração Fotovoltaica Distribuída em Redes de Distribuição de Energia Elétrica. Monografia (Bacharel em Engenharia Elétrica), URCAMP (Universidade Federal do Rio de Janeiro), Rio de Janeiro, Brasil.
- Bezerra, F.D. (2018). Energia Solar Fotovoltaica. Caderno Setorial, ETENE (Escritório Técnico de Estudos Econômicos do Nordeste), Ceará, Brasil.
- Borges, L.E. (2014). *Python para desenvolvedores: aborda Python 3.3*. Novatec Editora.
- Carvalho, R.M.d. et al. (2012). Reguladores de tensão e geração distribuída em uma implementação de fluxo de potência a três e a quatro fios.
- Dugan, R.C. (2016). *Opendss manual*. Electric Power Research Institute.

Tabela 1. Cenários.

Condição	cenário base	cenário 1	cenário 2	cenário 3
Período de análise	diário	diário	diário	diário
Carregamento do alimentador	100%	100%	100%	100%
Curva de carga	14 Res + 1 industrial	14 Res + 1 industrial	14 Res + 1 industrial	14 Res + 1 industrial
Tipo de curva carga	Curvas típicas	Curvas típicas	Curvas típicas	Curvas típicas
Índice de penetração	Sem SFV	40%	50%	60%

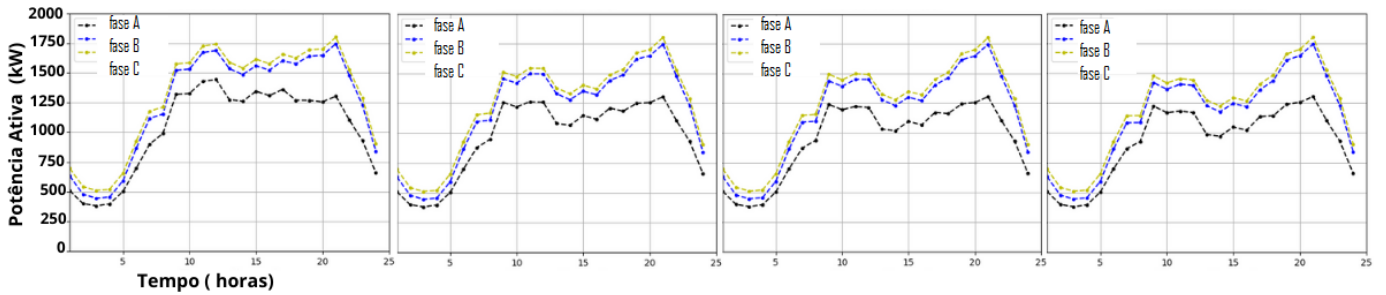


Figura 11. Potência exportada pela subestação - Cenários: base, 1, 2, e 3 respectivamente.

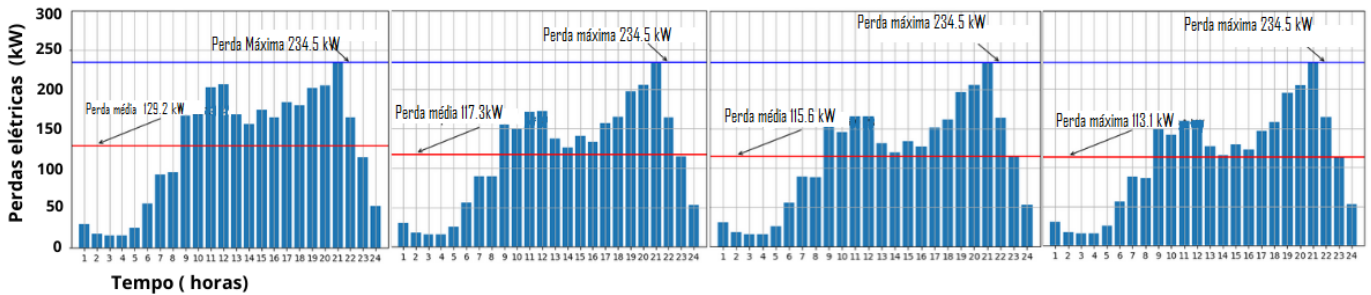


Figura 12. Perdas elétricas - Cenários: base, 1, 2, e 3 respectivamente.

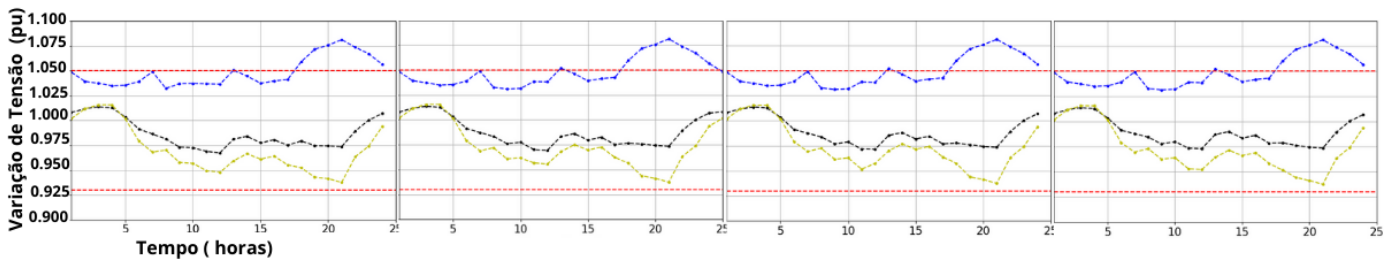


Figura 13. Variação de tensão em regime permanente na barra 680 - Cenários: base, 1, 2, e 3 respectivamente.

Lopes, M.G., Ferreira, T.S.D., Trindade, F.C.L., and Freitas, W. (2015). Análise dos impactos técnicos resultantes da intermitência de geração de sistemas fotovoltaicos na qualidade da tensão. In *Anais Da XI Conferência Brasileira Sobre Qualidade de Energia Elétrica*.

Lucchese, F.C., Anselmini, F., Martins, C.C., and Sperandio, M. (2018). Energia e mudanças climáticas: impactos socioambientais das hidrelétricas e diversificação da matriz energética brasileira.

Manzoni, A. (2005). Desenvolvimento de um sistema computacional orientado a objetos para sistemas elétricos de potência: aplicação a simulação rápida e análise da estabilidade de tensão. *Rio de Janeiro*.

Oliveira, T.L. et al. (2019). Desenvolvimento de um programa computacional livre, gráfico, e multiplataforma para analisar sistemas elétricos de potência em regime permanente e dinâmico.

PYPI (2019). *Opendsdirect.py*. Disponível em: <https://pypi.org/project/OpenDSSDirect.py/>. Acesso em: 05 de novembro. de 2019.

Sarkinen, J. (2007). An open source (d) controller. In *INTELEC 07-29th International Telecommunications Energy Conference*, 761–768. IEEE.