

Metodologia de Análises Elétricas Integradas de Sistemas de Distribuição e Subtransmissão usando o Software OpenDSS

Israel Panazollo* Gustavo Cordeiro* Viviane L. S. de Lima*
Daniel P. Bernardon** Mauricio Sperandio** Rafael C. Ney***

* *Curso de Engenharia Elétrica, Universidade Federal de Santa Maria, RS, (e-mail: israelpanazollo@gmail.com).*

** *DESP/CT, Universidade Federal de Santa Maria, RS*

*** *CEEE-D | Grupo Equatorial Energia, Pelotas, RS*

Abstract: Faced with the various computational tools with their different functionalities and applications, some organizations or companies may choose to use different software. you can use OpenDSS. The work proposes a program or computational environment in which it would be possible to analyze in an integrated way these two areas of electrical networks, using Python programming. In this way, developing broader analyzes and understanding the impact of distribution networks on transmission networks and vice versa, such as power flow, losses, voltage limits, among other important values, in addition to providing a planning that would take into account these different areas.

Resumo: Diante das várias ferramentas computacionais com suas diferentes funcionalidades e aplicações, algumas organizações ou empresas podem optar por utilizar diferentes softwares. Quando se realiza análises no âmbito da transmissão de energia, pode-se usar o ANAREDE, enquanto para o âmbito da distribuição da energia, pode-se utilizar o OpenDSS. O trabalho propõe um programa ou ambiente no qual fosse possível analisar de maneira integrada esses dois âmbitos das redes elétricas, com a utilização da programação em Python. Dessa forma, desenvolver análises mais amplas e perceber o impacto das redes de distribuição nas redes de transmissão e vice-versa, como o fluxo de potências, perdas, limites de tensão, entre outros valores importantes, além de propiciar um planejamento que levaria em conta esses âmbitos distintos.

Keywords: OpenDSS; ANAREDE; ; Python; Network Integration; Energy Transmission; Energy distribution.

Palavras-chaves: OpenDSS; ANAREDE; Python; Integração de Redes; Transmissão de Energia; Distribuição de Energia.

1. INTRODUÇÃO

No Brasil, as primeiras linhas de transmissão e usinas serviam para alimentar cargas pontuais, hoje, porém, com mais de 100 anos de existência, o setor elétrico se tornou algo de proporções gigantescas, dada a missão de suprir energeticamente um país de tamanho continental, seja nos grandes centros urbanos, ou nas regiões mais interioresanas (1).

As redes elétricas do Brasil estão divididas em 4 subsistemas, a saber, Sul, Sudeste/Centro-Oeste, Nordeste e Norte compondo o Sistema Interligado Nacional (SIN) (2). O Brasil apresenta um sistema elétrico com um elevado nível de ramificações e interligação, formando um sistema complexo que abrange quase todo o país, sendo apenas uma parte da Região Norte não integrada (1). (2) comenta que com essas malhas de redes de transmissão é possível atender as demandas energéticas e transferir energia entre os subsistemas.

Para realizar diagnósticos, analisar e planejar a expansão de um sistema tão complexo e vasto são necessárias ferramentas avançadas. Ainda mais com a inserção de geração distribuída e produtores de energia independentes, tornando muito maior o desafio da operação do sistema elétrico (3).

As ferramentas utilizadas em redes elétricas passaram por diversas transformações, desde os diagramas de círculo, que permitiam a representação gráfica da variação da tensão, corrente e potência de uma rede, antes da computação, passando pela aplicação do método de Gauss Seidel em computadores a válvula em 1956, para a aplicação do método de Newton Raphson em computadores com transistores e circuitos integrados em 1967 até chegar nos desenvolvimentos mais avançados de hoje em dia (2).

Dentre as ferramentas na área de sistemas de potência que se destacam no Brasil, há o ANAREDE (Análise de Redes), um programa desenvolvido pela CEPEL (Centro de Pesquisa de Energia Elétrica), que permite fornecer soluções de fluxo de potência, equivalente de redes e aná-

lises como de sensibilidade, contingências e segurança (2). Outra ferramenta que tem ganhado espaço nas análises e planejamentos de sistemas distribuídos é o software OpenDSS (Open Distribution System Simulator) que permite a construção de redes fragmentadas e possibilita articular diferentes partes em uma simulação, algo muito útil quando se consideram as constantes alterações numa rede de distribuição (4).

Diante das várias ferramentas computacionais com suas diferentes funcionalidades e aplicações, as organizações ou empresas podem optar por utilizar diferentes softwares. Quando se realizam análises no âmbito da transmissão de energia, pode-se usar o ANAREDE, enquanto para o âmbito da distribuição da energia, pode-se utilizar o OpenDSS. O desafio é desenvolver um programa ou ambiente no qual é possível analisar de maneira integrada esses dois cenários das redes elétricas.

Assim seria possível desenvolver análises mais amplas e perceber o impacto das redes de distribuição nas redes de transmissão e vice-versa, como o fluxo de potências, perdas, limites de tensão, entre outros valores importantes, além de propiciar um planejamento que levaria em conta esses âmbitos distintos. Dessa forma, este trabalho objetiva desenvolver uma metodologia que permita a realização desta análise integrada.

2. REFERENCIAL TEÓRICO

2.1 Análise de Redes

Um sistema elétrico de potência (SEP), tem como função suprir com energia elétrica, no momento que for solicitado, aos consumidores na quantidade e qualidade adequadas. Um sistema é constituído basicamente por três grandes blocos: a geração, a transmissão e a distribuição. A geração compreende as fontes que transformam um tipo de energia em energia elétrica. A transmissão é representada pelo transporte dos pontos de geração para pontos de recepção em grandes centros consumidores. A distribuição trata do transporte de energia desses pontos de entrega para os consumidores finais (2).

A transmissão da energia pode ocorrer em corrente contínua (CC) ou corrente alternada (CA). O tipo que prevalece, pelo menos no Brasil, é em CA (6). Os sistemas elétricos podem possuir uma ou diversas fases (polifásicos), porém o tipo mais comum é o sistema trifásico (7). Devido ao comportamento senoidal da transmissão em CA, utilizam-se valores das grandezas elétricas como fasores, isto é, com módulo e ângulo, assim uma matemática com números complexos é necessária (6).

Em uma rede elétrica haverá diversos elementos com diferentes especificações. Primeiramente estão as barras onde estão conectadas os geradores, cargas, linhas, entre outros equipamentos. As barras são classificadas como barras de carga ou PQ, as barras de tensão contralada ou PV e as barras swing ou $V\theta$. Nas barras PQ, os valores das potências ativas e reativas são conhecidos. Nas barras PV, o valor de potência ativa e o valor do módulo da tensão. Nas barras $V\theta$, tanto o módulo quanto o ângulo da tensão são conhecidos como barras de referência (2).

2.2 ANAREDE

O programa de análise de redes (Anarede) consiste num conjunto de aplicações computacionais integradas. O programa visa fornecer ao setor elétrico técnicas, métodos e algoritmos que sejam adequados e eficientes para realizar estudos, tanto para planejamento quanto operação de sistema elétricos de potência (8).

As versões mais recentes do programa realizam o fluxo de potência, equivalentes de rede, análise de contingências, análises de sensibilidade para tensão ou fluxo, dentre outros. Os dados de entradas são definidos em códigos de execução, num arquivo com extensão .pwf. Além disso, o Anarede permite o uso de uma interface gráfica, para visualizar a rede em questão, sendo os arquivos desse tipo numa extensão .lst. Uma das principais funcionalidades é a obtenção do fluxo de potência. Os métodos disponíveis para solucionar as equações da rede elétrica em CA são o método desacoplado rápido e o método de Newton. As condições iniciais para os cálculos são dados pelos valores de entrada ou pela opção FLAT com valores padrão. Além disso, é possível solucionar um sistema CC, porém será enfatizado neste trabalho os cálculos para uma rede CA (8).

Os dados de entrada mais básicos para realizar o fluxo de potência são associados a topologia da rede e ao carregamento do sistema. Destacam-se os dados referentes aos códigos de execução DBAR e DLIN. É válido destacar que dados adicionais tanto para geradores, como cargas ou demais elementos podem ser adicionados no cartão para realizar análises mais complexas (8).

A modelagem das barras CA é realizada de maneira semelhante ao modelo completo do fluxo de potência. Os transformadores e linhas são representados por uma reatância em série entre os terminais. As cargas são modeladas, em geral, como potência constante, enquanto os geradores são tratados como potência ativa fixa com variação de potência reativa para manter a tensão constante (8). Os códigos de DBAR apresenta os dados das barras CA. No cartão, cada linha de DBAR informa os dados de uma barra. As colunas são agrupadas em campos que indicaram os dados da barra por tipo.

2.3 OpenDSS

O Open Distribution System Simulator (OpenDSS) é uma ferramenta de simulação para sistemas elétricos de distribuição. O software tem sido usado para o planejamento e análise das redes de distribuição, simulações de carga e gerações anuais, simulações com painéis fotovoltaicos, análise de interconexões com geração distribuída, estudos probabilísticos, entre outros usos. O programa possui diferentes tipos de soluções tais como fluxo de potência snapshot, diário ou anual, além de soluções dinâmicas, estudos de faltas entre outros tipos de solução (9).

Para utilizar o programa é possível trabalhar com um arquivo executável ou ainda DLL (Dynamic Link Library) seja com um servidor COM (Component Object Model) ou não. A versão executável, que será empregada neste trabalho, possui uma interface para usuário baseada em

texto. Nela será digitado o script da rede e operações desejadas, bem como serão visualizados os resultados(9).

Um dos usos mais comuns do OpenDSS é na resolução de problemas de fluxo de potência, mesmo que originalmente o programa tenha sido projetado para tratar de diversos aspectos do planejamento de sistemas de geração distribuída (GD) (9).

O OpenDSS utiliza modelos dos equipamentos presentes nos sistemas elétricos de distribuição. Os equipamentos são classificados em elementos de entrega de potência (Power Delivery - PD) e elementos de conversão de potência (Power Conversion - PC). Nos PDs se encontram transformadores, linhas, capacitores e reatores, enquanto nos PCs estão elementos como geradores, cargas, armazenamento, sistemas fotovoltaicos, entre outros (9).

Além disso, o programa conta com as barras (bus), onde os elementos do circuito serão conectados. Vale destacar que no OpenDSS as barras não são classificadas em PV, PQ ou V θ , sendo apenas locais onde para conectar os equipamentos. Para o OpenDSS, uma barra e um nó são conceitos distintos, no caso, uma barra contém nós, no qual se conectam objetos. É possível conectar objetos em um nó específico, por exemplo, ao adicionar uma carga monofásica a um sistema trifásico. Contudo o nó 0 será sempre a referência, isto é, o terra (ground), com uma tensão de 0 volts (9).

O programa ainda permite a plotagem de certos valores como a potência, tensão, corrente, perdas, capacidade restante, ressonância harmônica, entre outras informações. Além disso, possibilita uma série de opções seja no algoritmo de solução, como Normal ou Newton, ou período analisado, por exemplo instantâneo (snap), diário (daily) e anual (yearly), bem como no tipo de cálculo, como o fluxo de potência, estudo de falta (Faultstudy), solução harmônica (Harmonics), solução dinâmica (Dynamics), Monte-Carlo (M1,M2,M3,MF), entre outras diversas opções (9).

2.4 Python

O Python teve sua origem em 1980 e é uma linguagem com paradigma de programação do tipo orientado ao objeto, isto é, os dados são representados por objetos, que por sua vez, possuem três aspectos, sendo o identificador (o nome do objeto), o tipo, isto é natureza dos dados, por exemplo inteiro (int) ou lista (list), e o conteúdo, que é o valor armazenado (11).

Esta linguagem possui uma série de características distintas que a tornam uma boa escolha para utilizar na programação. Dentre as características que podem ser destacadas o fato de ser open source, isto é, pode ser distribuído e utilizado livremente. Além disso, o Python está disponível para várias plataformas, como Linux, Windows, macOS, entre outras, já que o código-fonte está na linguagem ANSI C e as bibliotecas-padrão são executadas da mesma maneira em qualquer destas plataformas (11). Salienta-se ainda que o Python é considerado uma linguagem mais fácil de aprender, sendo optada como a primeira linguagem de programação para ensino (10).

O Python consegue unir a simplicidade e robustez que permite que projetos, dos mais triviais aos grandes

e complexos, possam ser desenvolvidos em diversas áreas como ferramentas para engenharia, aplicações científicas, administração, jogos, entre outros, seja com grandes volumes de dados, podendo trabalhar com interfaces (11).

3. METODOLOGIA

A metodologia proposta neste trabalho possibilita a modelagem e a análise de redes de transmissão juntamente com redes de distribuição de maneira integrada utilizando o software OpenDSS.

O cartão .pwf do ANAREDE será lido por um algoritmo em Python. Os dados dessa conversão são aplicados em outro algoritmo em Python, que terá como resultado um arquivo no formato .dss. O fluxograma da Figura ?? apresenta um panorama dessas conversões.

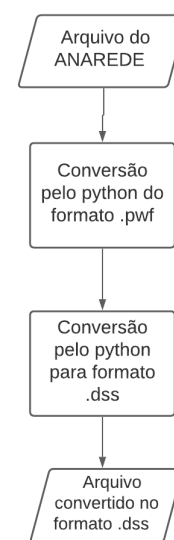


Figura 1. Fluxograma da conversão do arquivo .pwf para uso no OpenDSS. Fonte: Autores.

Em seguida, será apresentado um panorama geral da conversão do ANAREDE para OpenDSS utilizando o sistema do Litoral Norte do Rio Grande do Sul. Desta forma, o objetivo é integrar a rede do Litoral Norte com a de Tramandaí e realizar simulações no OpenDSS.

3.1 Conversão do ANAREDE para OpenDSS

A conversão de um cartão do ANAREDE para o OpenDSS será implementada utilizando o ambiente Python. O cartão .pwf do ANAREDE será lido por um algoritmo em Python denominado `conversor_pwf_para_py`. Os dados dessa conversão são aplicados em outro algoritmo chamado de `conversor_py_para_dss`, que terá como resultado um arquivo no formato .dss.

3.2 Conversão do ANAREDE para Python

Para que os dados do cartão do ANAREDE possam ser utilizados no ambiente Python é necessário realizar uma conversão inicial. Neste processo os dados que estão no formato .pwf serão convertidos em três matrizes de dados. A primeira das matrizes é referente aos dados de DBAR

(barras), a segunda está associada aos dados do DLIN (linhas) e a terceira aos dados de DGBT.

O algoritmo `conversor_pwf_para_py` utilizado nessa etapa já está pré-construído, necessitando apenas alterar alguns dados das linhas para realizar a conversão corretamente. A primeira informação que precisa ser adicionada é o nome do arquivo com sua localização numa pasta, alterando a linha 5 do código.

Realizando uma inspeção visual no cartão do ANAREDE é possível obter alguns dados que serão necessários para a conversão do arquivo, sendo estes as linhas para iniciar e terminar a conversão de DBAR e conversão de DLIN.

Este processo é feito manualmente devido a praticidade, uma vez que as linhas de início e término de conversão podem variar bastante entre os cartões do ANAREDE. É válido ressaltar que usa o valor da linha-1 para a linha inicial, uma vez que o Python começa a contagem de vetores em 0, depois 1 e assim por diante.

O processo que o `conversor_pwf_to_py` realizará será basicamente ler o cartão e colocar as informações em um vetor A. Em seguida, é realizada a transposição desse vetor A e o vetor B receberá esses valores. Na segunda etapa, então, será criada uma matriz E, que receberá cada caractere separado que havia em cada linha do cartão.

Na terceira etapa, serão construídas duas matrizes utilizando os valores da matriz E. Nesse momento, cada caractere será reunido conforme o formato dos dados que o ANAREDE orienta. Para DBAR serão obtidos 19 elementos por linha, cada um com informações das barras, como o número de identificação, se a barra está ligada ou não, grupo de tensão associada, se alguma geração ou carga na barra dentre tantas informações.

Para DLIN haverá 23 elementos por linha, cada um representando dados das linhas, como a barra de origem e de destino, se a linha de fato está em operação ou não, os valores de reatância e resistência, a presença de transformadores ou capacitores dentre outras informações. Para DGBT existem 2 elementos por linha, que indicam a sigla para uma base de tensão e o valor que esta representa.

Se deve levar em consideração que, alguns cartões podem não possuir todos os caracteres, o que impediria a conversão, por isso, nesses casos são adicionados valores padrões.

Após obter cada um dos termos de uma linha, uma matriz K receberá esta linha e o processo recomençará até que todas as linhas de dados sejam escritas. Cabe destacar que as conversões do python para OpenDSS envolverão matrizes quadradas. Esta é uma das razões, pelas quais serão convertidos apenas as linhas com os dados de DBAR, e posteriormente de DLIN, já que elas terão o mesmo número de caracteres, ao invés das linhas com textos que não possuem relevância para esta conversão.

3.3 Conversão Python para OpenDss

Para realizar a conversão do ambiente python para OpenDSS utilizam-se as matrizes advindas do `conversor_pwf_para_py` e são aplicados no código desta etapa da conversão, o `conversor_py_para_dss`. Neste passo, serão obtidos os dados relevantes para o código no formato `.dss`.

Alguns desses dados exigirão operações com os valores presentes nas matrizes. Com os dados um arquivo do tipo `.dss` será construído como resultado.

Os elementos constituintes do sistema, foram levantados previamente, conforme a matrizes geradas, previamente abordadas, na seção anterior. Uma vez que, os elementos e seus dados foram obtidos é escrito um arquivo que será usado no OpenDSS. É possível que esse arquivo tenha a extensão `.dss`, assim pode ser aberto diretamente pelo programa, ou então pode ter a extensão `.txt` para que possa ser aberto por um leitor de texto. Este arquivo será criado na pasta onde se encontra o código de conversão.

O documento produzido pelo algoritmo começará pelo "circuit", com as informações oriundas das matrizes como o nome da barra, a base de potência, a base de tensão e o valor da tensão em módulo, em p.u., e ângulo. Os dados que não indicados, assumirão um valor padrão, definido pelo OpenDSS. Durante a obtenção das informações de cada um dos elementos, é verificada sua existência e caso haja, quantos existem.

Para escrever o número de elementos de cada tipo, utiliza-se um loop com o número de iterações igual ao número daquele tipo de elemento. Cada um dos elementos será nomeado conforme a convenção de acordo com a iteração na qual este se encontra. Por exemplo um transformador será nomeados com a convenção TR_i, sendo i o número da iteração da escrita.

Depois do circuit, o elemento a ser escrito será o transformador. Adicionam-se as informações da resistência, reatância e número de taps e define-se que serão transformadores de 2 enrolamentos, o valor padrão do OpenDSS. Para o primeiro enrolamento, coloca-se a barra de onde o transformador saiu, junto com o tap, a tensão e a potência, a conexão será do tipo estrela, sendo este o padrão do OpenDSS. Para o segundo enrolamento ocorre o processo análogo.

Escrevem-se os dados do "linecode", colocando a resistência e a reatância do cabo, nesse caso, apenas os valores de sequência positiva. Então define-se a unidade como quilômetros. Cada linecode será nomeado como Cabo_i, com i sendo o número da iteração da escrita.

Na escrita das linhas, serão registradas as barras onde a linha está conectada. O tipo da barra, dado pelo linecode, será definido pelo tipo de cabo escrito previamente. O comprimento será unitário e a unidade será a mesma que o linecode, neste caso serão quilômetros.

A razão de criar uma linha de valor unitário, é que esta seja uma linha equivalente com os valores advindos do ANAREDE. Uma vez que, ANAREDE oferece os valores em p.u. e não oferece a distância das barras, dessa forma, não seria possível fazer um linecode usual que tivesse uma impedância por unidade de distância (Ex: 10 Ω/m).

Então são geradas as linhas de informações das cargas. A carga é nomeada, se coloca os dados da barra onde está conectada, da tensão da barra, o valor de potência ativa, da potência reativa e do fator de potência. Os demais valores serão atribuídos as pre-definições do OpenDSS.

```

Results for Actor ID # 1
CPU selected : 0
Status = SOLVED
Solution Mode = Daily
Number = 24
Load Mult = 1.000
Devices = 26811
Buses = 9775
Nodes = 31962
Control Mode =STATIC
Total Iterations = 4
Control Iterations = 1
Max Sol Iter = 4

- Circuit Summary -

Year = 0
Hour = 24
Max pu. voltage = 1
Min pu. voltage = 0.86747
Total Active Power: 2.79753 MW
Total Reactive Power: 1.25369 Mvar
Total Active Losses: 0.119531 MW, (4.273 %)
Total Reactive Losses: 0.16756 Mvar
Frequency = 60 Hz
Mode = Daily
Control Model = STATIC
Load Model = PowerFlow
    
```

Figura 2. Resultado da simulação da rede de Tramandaí no OpenDSS.

Para o gerador é utilizado a definição deste equipamento levando o model=1 do OpenDSS, ou seja, com uma injeção de potência ativa e reativa constante. Esta opção foi preferida, pelos valores das simulações resultarem a valores mais próximos das simulações do Anarede. O capacitor é último elemento escrito. Este será nomeado e então colocase a barra onde o capacitor shunt se encontra, adicionam-se a tensão da barra e potência reativa.

Por fim, são escritas algumas linhas sobre o tipo de solução, que nesse caso será no modo snapshot. Então, são escritos os valores das bases de tensão para o cálculo. E finalmente o comando para solucionar a rede construída, isto é, coloca-se o comando solve e assim encerra-se o arquivo gerado.

4. RESULTADOS

4.1 Rede de Tramandaí

A rede de Tramandaí, será o sistema em média e baixa tensão utilizado na integração. Esta rede foi modelada para OpenDSS e quando é simulada no programa apresenta os resultados indicados na Figura 2. Destacam-se os valores de potência ativa total, 2,7975 MW e potência reativa total 1,2537 Mvar.

4.2 Validação da conversão do Litoral Norte

Para avaliar se a conversão de um cartão do ANAREDE para o modelo utilizado no OpenDSS foi realizada de maneira adequada, basicamente se comparam os valores de fluxo de potência obtidos nos dois softwares e caso estes valores forem semelhantes, temos um indicativo de que a conversão foi realizada de maneira satisfatória.

Primeiramente, o cartão da rede do litoral norte do ANAREDE, na Figura 3, é convertido para o formato da modelagem do OpenDSS, Figura 4. Nas Figuras 5 e 6 é possível verificar que os valores do fluxo de potência, em especial da barra Vtheta do ANAREDE e da potência ativa e reativas totais da simulação do OpenDSS correspondem.

```

1 TITU
2 Litoral_Norte
3 DBAS
4 (Num)OETG( nome )GI ( V ) ( A ) ( Fg ) ( Qg ) ( Qm ) ( Bc ) ( F1 ) ( Q1 ) ( Sb)Are (Vθ)M(1) (2) (3) (4) (5) (6) (7) (8) (9) (10)
5 1 L26Atlantic-69 51020-58. 0.-.119-999999999
6 50048 L 69ENT936---69 1020-58. 11000
7 50052 L 69ENT904C---69 1020-58. 11000
8 50058 L 69ENT904B---69 1020-58. 11000
9 50066 L 69ENT904A---69 1020-58. 11000
10 99999
11 DGLT
12 (De)jd 0 d(Pa) HcEP ( Rk ) ( Xk ) (Hvar) (Tsp) (Tmm) (Tms) (Pns) (Bc ) (Cn) (Cs)Hs (Cq) (1) (2) (3) (4) (5) (6) (7) (8) (9) (10)
13 1 50048 1 3.5510.317 .175 75. 90. 75.
14 1 50052 1 .372 1.06 .018 75. 90. 75.
15 1 50058 1 .372 1.06 .018 75. 90. 75.
16 1 50066 1 .372 1.06 .018 75. 90. 75.
17 99999
18 DGLT
19 (G (Vmn) (Vmx) (Vme) (Vmax)
20 5 .95 1.05 .95 1.05
21 0 .95 1.05 .95 1.048
22 99999
23 DARE
24 (Ar (Xchg) ( Identificacao da area ) (Xmin) (Xmax)
25 1 0. 0. 3000.
26 99999
27 DOBT
28 (G ( KV)
29 69 69.
30 99999
31 FIM
    
```

Figura 3. Cartão da rede Litoral Norte do ANAREDE.

```

Clear
// ----- Novo Circuito -----
New circuit.Atlantic-69 bus1=1 baseVA=100.0 basekv=69. angle=-58.0 pu=1.02 phases=3 frequency=60
// ----- Linhas -----
New LineCode.Cabo0 R1=1.69815 X1=4.91192 Units=km
New LineCode.Cabo1 R1=0.177189 X1=0.504666 Units=km
New LineCode.Cabo2 R1=0.177189 X1=0.504666 Units=km
New LineCode.Cabo3 R1=0.177189 X1=0.504666 Units=km
// ----- Linhas -----
New Line.L0 bus1=1 bus2=50048 linecode=Cabo0 length=1 units=km
New Line.L1 bus1=1 bus2=50052 linecode=Cabo1 length=1 units=km
New Line.L2 bus1=1 bus2=50058 linecode=Cabo2 length=1 units=km
New Line.L3 bus1=1 bus2=50066 linecode=Cabo3 length=1 units=km
// ----- Capacitor -----
New capacitor.CAP0 bus1=1 kv=69. kvar=175.0 phases=3
New capacitor.CAP1 bus1=1 kv=69. kvar=18.0 phases=3
New capacitor.CAP2 bus1=1 kv=69. kvar=18.0 phases=3
New capacitor.CAP3 bus1=1 kv=69. kvar=18.0 phases=3
// ----- Tipo de Solução -----
Set mode=snapshot
// ----- Valores bases de tensão -----
Set voltagebases=[ 69.]
calcVoltagebases
// ----- Solucionar -----
Solve
    
```

Figura 4. Cartão da rede Litoral Norte convertida para OpenDSS.

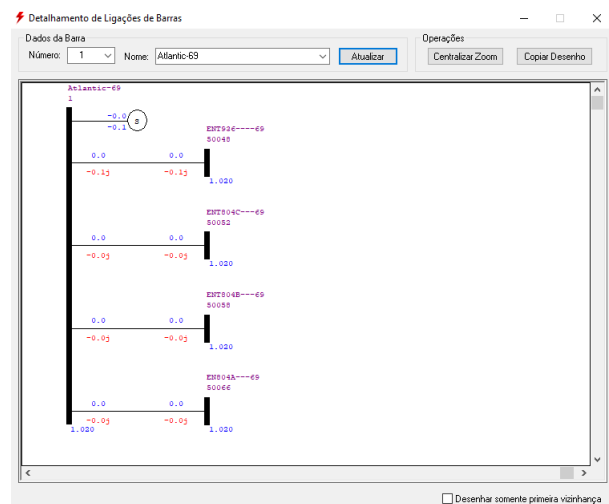


Figura 5. Simulação do Litoral Norte sem carga no ANAREDE.

4.3 Litoral Norte com uma carga e com três cargas

Deseja-se adicionar na rede Litoral Norte uma carga, nesse caso de 90 MW e 20 Mvar, em uma das barras no arquivo .pwf, converter o arquivo, simular o fluxo de potência com OpenDss e comparar com a simulação do Anarede. Depois o mesmo processo é realizado, porém com três cargas na rede, sendo estas de 10 MW e 2 Mvar, 5 MW e 1 Mvar e,

```

Results for Actor ID # 1
CPU selected : 0
Status = SOLVED
Solution Mode = Snap
Number = 1
Load Mult = 1.000
Devices = 9
Buses = 5
Nodes = 15
Control Mode =STATIC
Total Iterations = 2
Control Iterations = 1
Max Sol Iter = 2

- Circuit Summary -

Year = 0
Hour = 0
Max pu. voltage = 1.0201
Min pu. voltage = 1.0201
Total Active Power: 4.52195E-
009 MW
Total Reactive Power: -0.263713
Mvar
Total Active Losses: 4.51766E-
009 MW, (99.91 %)
Total Reactive Losses: -0.0254024
Mvar
Frequency = 60 Hz
Mode = Snap
Control Mode = STATIC
Load Model = PowerFlow
    
```

Figura 6. Simulação do Litoral Norte sem carga no OpenDSS.

```

Results for Actor ID # 1
CPU selected : 0
Status = SOLVED
Solution Mode = Snap
Number = 1
Load Mult = 1.000
Devices = 10
Buses = 5
Nodes = 15
Control Mode =STATIC
Total Iterations = 2
Control Iterations = 1
Max Sol Iter = 2

- Circuit Summary -

Year = 0
Hour = 0
Max pu. voltage = 0.99424
Min pu. voltage = 0.93584
Total Active Power: 90.4259 MW
Total Reactive Power: 28.5268
Mvar
Total Active Losses: 3.23382
MW, (3.576 %)
Total Reactive Losses: 9.37436
Mvar
Frequency = 60 Hz
Mode = Snap
Control Mode = STATIC
Load Model = PowerFlow
    
```

Figura 8. Simulação do Litoral Norte com uma carga no OpenDSS.

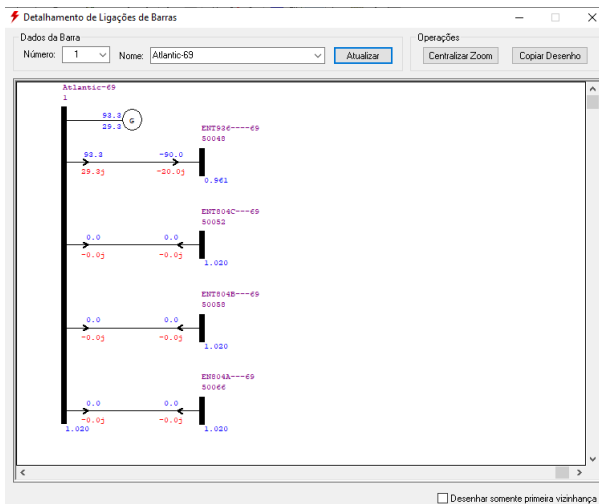


Figura 7. Simulação do Litoral Norte com uma carga no ANAREDE.

por fim, 6 MW e 1 Mvar . Os resultados das simulações são apresentados nas Figuras 7, 8, 9 e 10.

4.4 Processo de integração das redes

Para realizar a integração das redes de transmissão e distribuição altera-se o arquivo "circuit" utilizado pelo código mestre da rede de Tramandaí no OpenDSS. O arquivo "CircuitoMT_5707_TRA_7_1.dss" é modificado de modo que o circuit agora se tornará a rede do Litoral Norte. Vale destacar que para realizar a integração foi necessário adicionar um transformador e renomear as barras para que a integração funcione corretamente. Nas Figuras 11 e 12 estão apresentados os códigos do circuit antes e depois da modificação, respectivamente.

4.5 Resultados das redes integradas

Haverá três casos a serem analisados de integração da rede Litoral Norte com a rede de Tramandaí. O primeiro caso é com a rede do Litoral Norte sem cargas, o segundo caso

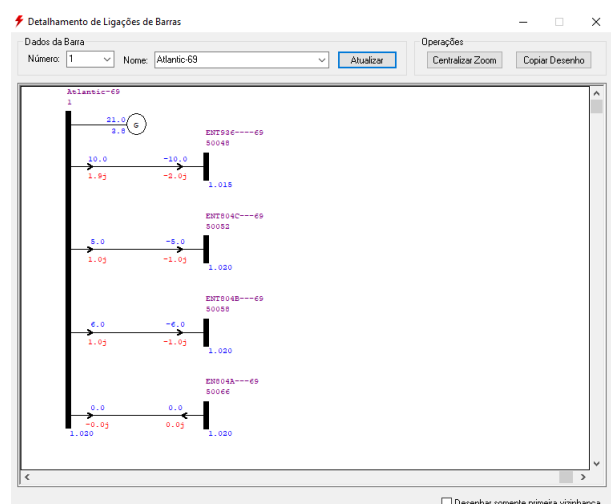


Figura 9. Simulação do Litoral Norte com três cargas no ANAREDE.

```

Results for Actor ID # 1
CPU selected : 0
Status = SOLVED
Solution Mode = Snap
Number = 1
Load Mult = 1.000
Devices = 12
Buses = 5
Nodes = 15
Control Mode =STATIC
Total Iterations = 2
Control Iterations = 1
Max Sol Iter = 2

- Circuit Summary -

Year = 0
Hour = 0
Max pu. voltage = 1.0156
Min pu. voltage = 1.01
Total Active Power: 21.0386 MW
Total Reactive Power: 3.85033
Mvar
Total Active Losses: 0.0384614
MW, (0.1828 %)
Total Reactive Losses: 0.0865084
Mvar
Frequency = 60 Hz
Mode = Snap
Control Mode = STATIC
Load Model = PowerFlow
    
```

Figura 10. Simulação do Litoral Norte com três cargas no OpenDSS.

```
1 ! Criação da seção de circuitos
2 New "Circuit.TRA_7" basekv=13.8 pu=1 bus1="53934253A" r1=0 x1=0.0001
```

Figura 11. Arquivo do circuit original.

```
1 Clear
2 // ----- Novo Circuito -----
3
4
5 New circuit.Atlantic-69 bus1=1 baseMVA=100.0 basekv=69. angle=-59.0 pu=1.02 phases=3 frequency=60
6 // ----- Transformador -----
7
8
9 New transformer.TR0 XHL=15. %r=0.05 numtaps=0 windings=2
10 ~ wdg=1 bus=1 cap=1.0 conn=nyye kv=69 kva=100000
11 ~ wdg=2 bus=53934253A conn=nyye kv=13.8 kva=100000
```

Figura 12. Arquivo do circuit alterado.

```
Results for Actor ID # 1
CPU selected : 0
Status = SOLVED
Solution Mode = Daily
Number = 24
Load Mult = 1.000
Devices = 26820
Buses = 9780
Nodes = 31977
Control Mode =STATIC
Total Iterations = 4
Control Iterations = 1
Max Sol Iter = 4

- Circuit Summary -

Year = 0
Hour = 24
Max pu. voltage = 5.0974
Min pu. voltage = 0.88306
Total Active Power: 2.8445 MW
Total Reactive Power: 1.06942 Mvar
Total Active Losses: 0.121538 MW, (4.273 %)
Total Reactive Losses: 0.182824 Mvar
Frequency = 60 Hz
Mode = Daily
Control Mode = STATIC
Load Model = PowerFlow
```

Figura 13. Simulação do Litoral Norte sem cargas integrado a rede de Tramandaí no OpenDSS.

```
Results for Actor ID # 1
CPU selected : 0
Status = SOLVED
Solution Mode = Daily
Number = 24
Load Mult = 1.000
Devices = 26821
Buses = 9780
Nodes = 31977
Control Mode =STATIC
Total Iterations = 4
Control Iterations = 1
Max Sol Iter = 4

- Circuit Summary -

Year = 0
Hour = 24
Max pu. voltage = 4.9679
Min pu. voltage = 0.85768
Total Active Power: 93.0141 MW
Total Reactive Power: 29.7369 Mvar
Total Active Losses: 3.34598 MW, (3.597 %)
Total Reactive Losses: 9.5601 Mvar
Frequency = 60 Hz
Mode = Daily
Control Mode = STATIC
Load Model = PowerFlow
```

Figura 14. Simulação do Litoral Norte com uma carga integrado a rede de Tramandaí no OpenDSS.

é com a rede do Litoral Norte com uma carga e o terceiro caso é com a rede do Litoral Norte com três cargas. Os resultados são apresentados nas Figuras 13, 14 e 15.

4.6 Discussão dos resultados

Primeiramente, o processo de conversão da rede Litoral Norte do Anarede para OpenDSS nos três casos apresentou valores quase idênticos com respeito aos valores de potências ativa e reativa totais. Diante desses resultados, comparou-se os valores obtidos nos casos de integração da

```
Results for Actor ID # 1
CPU selected : 0
Status = SOLVED
Solution Mode = Daily
Number = 24
Load Mult = 1.000
Devices = 26823
Buses = 9780
Nodes = 31977
Control Mode =STATIC
Total Iterations = 4
Control Iterations = 1
Max Sol Iter = 4

- Circuit Summary -

Year = 0
Hour = 24
Max pu. voltage = 5.0747
Min pu. voltage = 0.87862
Total Active Power: 23.8706 MW
Total Reactive Power: 5.173 Mvar
Total Active Losses: 0.15964 MW, (0.668 %)
Total Reactive Losses: 0.294566 Mvar
Frequency = 60 Hz
Mode = Daily
Control Mode = STATIC
Load Model = PowerFlow
```

Figura 15. Simulação do Litoral Norte com 3 cargas integrado a rede de Tramandaí no OpenDSS.

rede do Litoral Norte (na tabela, abreviado LN) e da rede de Tramandaí.

Tabela 1. Resultados das simulações.

Caso	Pot. Ativa (MW)	Pot. Reativa (Mvar)
Rede Tramandaí	2,7975	1,2537
LN sem carga	0	-0,2637
LN com 1 carga	90,4259	28,5268
LN com 3 cargas	21,0386	3,8503
Primeiro caso	2,8445	1,0694
Segundo caso	93,0141	29,7369
Terceiro caso	23,8706	5,173

Nota-se que as potências ativas e reativas da rede de Tramandaí são somadas às potências do caso que a rede do Litoral Norte se encontra, algo que indica que há trânsito de potência entre as redes. No primeiro caso é possível visualizar que a potência reativa total é modificada, indo para 1,0694 Mvar já que havia 1,2537 Mvar na rede de Tramandaí e -0,2637 Mvar na rede do Litoral Norte. No segundo caso é válido destacar que foi preciso solicitar mais potência do circuit, no caso 93,0141 MW, dado que a carga há a carga integrada com a rede de Tramandaí que já solicitava 2,7975 MW. No terceiro caso, encontra-se algo semelhante aos casos anteriores, onde o valor de potência ativa total bem como o valor de potência reativa um pouco maiores.

Observa-se ainda que nas simulações o valor "max PU voltage" assumiu um valor em torno de 5 p.u. Isto ocorre porque o OpenDSS está encontrando um valor perto de 69 kV para o alimentador, porém numa base de 13,8 kV. Como 69 é 5 vezes maior que 13,8 este valor em torno de 5 p.u. é compreensível. A não alteração nas bases de cálculo foi optada para modificar minimamente o arquivo da rede de Tramandaí nesta integração.

5. CONSIDERAÇÕES FINAIS

Diante dos resultados é possível constatar que a integração das redes de fato está ocorrendo de modo satisfatório. Isto implica na possibilidade de integrar outras redes, bem como na exploração dessa metodologia de integração como um recurso para planejamento e análise de sistemas de

potência que envolvem distribuição e subtransmissão, no software OpenDSS, incluindo informações de redes que vem do software ANAREDE.

Uma possibilidade de aplicação dessa metodologia inclui a análise de geração de distribuída, que vêm crescendo nos últimos anos (12), e tem sido realizada no OpenDss, sendo que agora poderia ser feita levando em conta o impacto na subtransmissão. Outra possibilidade é a análise de Montantes de Uso de Sistema de Transmissão (MUST), que é do que a quantidade de energia que cada distribuidora deve contratar para suprir sua demanda em um determinado tempo. Esta contratação hoje é feita com informações volúveis e sem nenhum tipo de algoritmo reproduzível. Este projeto trás o desenvolvimento de uma metodologia que visa a implementação de um programa que calcule estes montantes a partir de dados de entrada, utilizando a integração das redes para determinação dos fluxos de potência trocados em cada ponto de conexão com a rede básica .

Quanto aos trabalhos futuros, sugere-se que esta metodologia seja testada em validada em diferentes configurações de rede, para aumentar a confiabilidade dos resultados desta proposta. Além disso, pode-se aplicar esta metodologia nos casos de interesse citados como a análise de GD ou MUST.

AGRADECIMENTOS

Os autores agradecem ao Programa de PD ANEEL, e a CEEE-D | Grupo Equatorial Energia, pelo financiamento deste trabalho via Contrato N^o 5000003925.

REFERÊNCIAS

- Barros, B. et al. (2018), **Geração, Transmissão e Consumo de Energia Elétrica**, v. 01, Ed. Erica, São Paulo, 2018.
- Robba, E. et al. (2020) **Análise de Sistemas de Transmissão de Energia Elétrica**, v.01, Ed. Blucher, São Paulo, 2020.
- Ned, M. et al. (2016) **Sistemas Elétricos de Potência: curso introdutório**, v.01, Ed. LTC, Rio de Janeiro, 2016.
- Andrade, B. (2020) **Modelagem de um sistema de distribuição real desbalanceado e análise do impacto da geração distribuída utilizando o software OpenDSS**, Journal Sociedade Brasileira de Automática, 2020.
- Halliday, D (2020) **Fundamento da Física**, v.03, Ed. LTC, Rio de Janeiro, 2020.
- Pinto, M. (2018) **Energia Elétrica: geração, transmissão e sistemas interligados**, v.01, Ed. LTC, Rio de Janeiro, 2018.
- Barioni, C et al. (2000) **Introdução a Sistemas Elétricos de Potência - Componentes Simétricas**, v.02 Ed. Blucher, São Paulo, 2000.
- Anarede (2022) **Programa de Análise de Redes: Manual do Usuário**, v.11.00.01, Ed.DRE, Rio de Janeiro, 2022.
- Dugan, R. et al (2020) **Reference Guide: The Open Distribution System Simulator**, Ed. OpenDSS, 2020.
- Perkovic, L. (2016) **Introdução à computação usando Python: um foco no desenvolvimento de aplicações**, v.01, Ed. LTC, Rio de Janeiro, 2016.
- Banin, P (2018) **Python 3: Conceitos e aplicações: uma abordagem didática**, v.01, Ed. Érica, São Paulo, 2018.
- Miranda, R (2013) **Análise da Inserção de Geração Distribuída de Energia Solar Fotovoltaica no Setor Residencial Brasileiro**, Dissertação, Universidade Federal de Rio de Janeiro, 2013.