

UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Davi Pavezi Wilke

**ANÁLISE DO IMPACTO DO CARREGAMENTO DE VEÍCULOS
ELÉTRICOS NA REDE ELÉTRICA**

Santa Maria, RS
2023

Davi Pavezi Wilke

**ANÁLISE DO IMPACTO DO CARREGAMENTO DE VEÍCULOS ELÉTRICOS NA
REDE ELÉTRICA**

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia Elétrica, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do título de **Engenheiro Eletricista**.

Orientador: Prof. Dr. Daniel Pinheiro Bernardon

Santa Maria, RS
2023

Davi Pavezi Wilke

**ANÁLISE DO IMPACTO DO CARREGAMENTO DE VEÍCULOS ELÉTRICOS NA
REDE ELÉTRICA:**

Trabalho de Conclusão de Curso apresentado
ao Curso de Engenharia Elétrica, da
Universidade Federal de Santa Maria (UFSM,
RS), como requisito parcial para obtenção do
título de **Engenheiro Eletricista**

Aprovado em 16 de novembro de 2023:

Prof. Dr. Daniel Pinheiro Bernardon (UFSM)
(Presidente/Orientador)

Marina Camponogara (UFSM)

Juliano Silva Damiani (UFSM)

Santa Maria, RS
2023

AGRADECIMENTOS

Primeiramente, quero agradecer ao Curso de Engenharia Elétrica da Universidade Federal de Santa Maria (UFSM) pelo ambiente de aprendizado excepcional e pelo compromisso com o ensino público, gratuito e de qualidade. A oportunidade de aprender e crescer academicamente ao longo desses anos foi fundamental para minha formação.

Um agradecimento especial vai para os meus pais, Alexandre Adalberto Wilke e Neiva Pavezi, pelo amor, pelo apoio incondicional e pelo incentivo durante toda a minha vida. Suas lições de vida e seus valores moldaram minha jornada acadêmica.

Agradeço também ao meu professor Daniel Pinheiro Bernardon, cuja orientação e conhecimento foram fundamentais para minha trajetória acadêmica. Agradeço também a todos demais professores: seus ensinamentos e apoio foram essenciais para o desenvolvimento deste trabalho.

Quero estender meus agradecimentos aos meus colegas Pedro Henrique Eisenkraemer e Julia Carla Madaloz, pelo auxílio na criação deste trabalho, e a todos os demais colegas de classe, que compartilharam comigo momentos de aprendizado e desafios ao longo desta jornada, tornando-a ainda mais enriquecedora.

Não poderia deixar de mencionar meus amigos Gabriel Giacomini de Freitas e Natália Dotto. A amizade e o suporte de vocês foram pilares importantes ao longo desta jornada acadêmica.

Este agradecimento também vai para ti! E a todos que não mencionei, mas que, de alguma forma, contribuíram para este trabalho e para minha formação: meu muito obrigado!

" No. Try not. Do... or do not. There is no try."

Yoda.

RESUMO

ANÁLISE DO IMPACTO DO CARREGAMENTO DE VEÍCULOS ELÉTRICOS NA REDE ELÉTRICA

AUTOR: Davi Pavezi Wilke
ORIENTADOR: Prof. Dr. Daniel Pinheiro Bernardon

A crescente adoção de Veículos Elétricos (VE) como alternativa de transporte sustentável tem impulsionado a necessidade de infraestruturas de recarga capazes de atender à demanda em expansão. No entanto, o aumento significativo no número de VE traz consigo desafios no gerenciamento de energia elétrica e na estabilidade da rede. Diante desse contexto, este trabalho tem como objetivo analisar o impacto do carregamento de VE na rede. Através da análise dos perfis de carga dos VE, serão investigadas as flutuações de demanda e a concentração de carga. Serão identificados os percentuais de penetração na rede que podem gerar sobrecargas nas linhas de distribuição e quedas de tensão excessivas. Para alcançar esses objetivos, utiliza-se uma metodologia de seleção de unidades consumidoras com VE para análise da simulação do carregamento de VE no *software* OpenDSS, com suporte de programação em *python*. Os resultados obtidos permitiram identificar os padrões de carregamentos críticos de demanda e propor adequadas estratégias de gerenciamento de energia. Espera-se que este estudo contribua para o avanço do conhecimento no campo do gerenciamento de energia em estações de recarga de VE, fornecendo informações relevantes para concessionárias de energia, empresas do setor e pesquisadores interessados pela integração eficiente e sustentável dos VE na rede elétrica.

Palavras-chave: Veículos elétricos. OpenDSS. Python. Gerenciamento de energia.

ABSTRACT

ANALYSIS OF ELECTRIC VEHICLE CHARGING IMPACT ON THE ELECTRICAL GRID:

AUTHOR: Davi Pavezi Wilke
ADVISOR: Daniel Pinheiro Bernardon.

The increasing adoption of electric vehicles (EVs) as a sustainable transportation alternative has driven the need for recharging infrastructure capable of meeting the expanding demand. However, the significant rise in the number of EVs brings challenges to electric energy management and network stability. In this context, this work aims to analyze the impact of EV charging on the grid. Through the analysis of EV load profiles, fluctuations in demand and load concentration will be investigated. We will identify penetration percentages in the network that can lead to overloads in distribution lines and excessive voltage drops. To achieve these goals, a methodology was used of selecting consumer units with EVs for the simulation analysis of EV charging in the OpenDSS software, supported by Python programming. The results obtained will allow us to identify critical demand charging patterns and propose appropriate energy management strategies. It is expected that this study will contribute to advancing knowledge in the field of energy management at EV charging stations, providing relevant information for energy utilities, industry companies, and researchers interested in the efficient and sustainable integration of EVs into the electrical grid.

Keywords: Electric Vehicles. OpenDSS. Python. Energy management.

LISTA DE FIGURAS

FIGURA Error! No text of specified style in document. – Volume total de vendas anuais de VE.....	11
FIGURA 2 – Diagrama simplificado da estrutura de um BEV.....	14
FIGURA 3 – Diagrama simplificado da estrutura de um HEV.....	15
FIGURA 4 – Diagrama simplificado da estrutura de um FCEV.....	16
FIGURA 5 – Carregador NEOCHARGE plug and play.....	17
FIGURA 6 – Fluxograma do processo do algoritmo.....	22
FIGURA 7 – Níveis de carregamento de transformadores sobrecarregados no cenário B.....	28
FIGURA 8 – Índice de perdas (%) no sistema no Cenário B.....	29
FIGURA 9 – Limites de Tensão (em pu) no cenário B.....	30
FIGURA 10 – Índice de perdas (%) no sistema no cenário C.....	32
FIGURA 11 – Nível de carregamento dos transformadores em sobrecarga no cenário C.....	33
FIGURA 12 – Limites de tensão (em pu) no cenário C.....	34
FIGURA 13 – Índice de perdas (%) no sistema no cenário D.....	35
FIGURA 14 – Limites de tensão (em pu) no cenário D.....	36
FIGURA 15 – Nível de carregamento dos transformadores sobrecarregados no cenário D....	37

LISTA DE TABELAS

TABELA 1 – Níveis de recarga de acordo com padrão SAE.....	17
TABELA 2 – Limites de operação da Rede de Distribuição.....	24
TABELA 3 – Resultados obtidos no caso base.....	27
TABELA 4 – Transformadores sobrecarregados no cenário B.....	31
TABELA 5 – Transformadores sobrecarregados no cenário C.....	34
TABELA 6 – Transformadores sobrecarregados no cenário D.....	37

SUMÁRIO

INTRODUÇÃO	10
1.1 OBJETIVOS	11
1.1.1 Objetivo geral.....	12
1.1.2 Objetivos específicos.....	12
2 REFERENCIAL TEÓRICO	13
2.1 O VEÍCULO ELÉTRICO	13
2.1.1 Veículo elétrico a bateria (BEV)	13
2.1.2 Veículo elétrico híbrido (HEV e PHEV).....	14
2.1.3 Veículo elétrico a célula de combustível (FCEV).....	15
2.2 A ESTRUTURA DE RECARGA	16
2.3 IMPACTOS NO SISTEMA DE DISTRIBUIÇÃO	18
2.4 MÉTODO DE MONTE CARLO	19
3 METODOLOGIA	21
3.1 CONSIDERAÇÕES INICIAIS	21
3.2 DESCRIÇÃO DO ALGORITMO.....	21
3.2.1 Curvas de carga do consumidor.....	22
3.2.2 Cenários.....	23
3.2.3 Alocação de cargas	23
3.2.4 Resultados dos impactos	24
4 OBTENÇÃO DOS RESULTADOS	26
4.1 CENÁRIO A	26
4.2 CENÁRIO B.....	27
4.3 CENÁRIO C.....	31
4.4 CENÁRIO D	34
5 CONCLUSÃO	38
REFERÊNCIAS	39
APENDICE A – Roteiro de Simulação.....	42

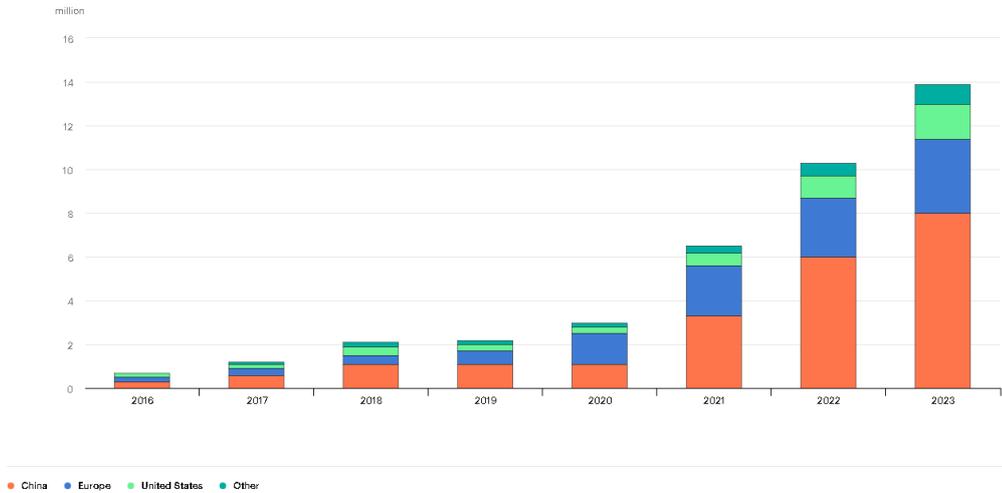
INTRODUÇÃO

Veículos pessoais são responsabilizados pelas emissões que causam poluição do ar nas áreas urbanas. Segundo a *International Energy Agency* (IEA), cerca de 15% das emissões totais de CO₂ do mundo todo correspondem a esses veículos (ELETRIC, 2023). Mais do que nunca, o setor dos transportes enfrenta um sério desafio no seu percurso de desenvolvimento sustentável. Sendo assim, nos últimos anos, os interesses em veículos elétricos (VE) foram motivados pela economia de emissões de carbono, com isso número crescente de VE ligados à rede teria tanto um impacto negativo (perda de qualidade da rede) quanto positivo (redução de poluição) nos sistemas energéticos existentes.

Apesar de não ser propriamente um produto novo no mercado — já que sua origem foi com a indústria automotiva —, seu amplo uso e maior acessibilidade são mais recentes, como é possível observar com as novas pautas ambientais, a exemplo da Agenda 2030, da Organização das Nações Unidas (ONU). Um dos objetivos da Agenda 2030 é “assegurar o acesso confiável, sustentável, moderno e a preço acessível à energia para todos”. São pautas cujo cumprimento contribui para a mitigação das mudanças climáticas, a redução da poluição do ar e a melhoria da saúde pública, impulsionando a inovação tecnológica e o alcance de um desenvolvimento global sustentável. Com isso, incentiva-se o desenvolvimento do setor de VE, junto de incentivos ao consumidor, como subsídios e isenção de impostos.

Conforme a Figura 1, torna-se evidente que a tendência é que esse mercado aumente cada vez mais. Ela mostra que, em 2019, havia apenas em torno de 2 milhões de VE; 4 anos depois, esse valor aumentou em 7 vezes, alcançando, em 2023, cerca de 14 milhões de volume de venda (ELETRIC, 2023).

Figura **Error! No text of specified style in document.** – Volume total de vendas anuais de VE.



Fonte: ELETRIC (2023).

Segundo Wijaya (2021), o aumento da procura por energia devido à grande penetração de VEs poderia proporcionar uma abordagem eficaz para o consumo da energia elétrica gerada pelas Fontes de Energia Renováveis (FER), estimulando a grande integração das energias renováveis. No entanto, a rede distribuída localizada será mais vulnerável com muitos veículos elétricos, uma vez que estão ligados, principalmente, às áreas residenciais locais, onde as redes são por vezes muito carregadas.

O fato de o Brasil ter uma matriz elétrica favorável para a difusão de VE, com grande participação de fontes renováveis de energia (EPE, 2021), permite a penetração de uma boa parcela de VE. Para lidar com os possíveis desafios e realizar o planejamento ideal do Sistema de Distribuição (SD) com grande penetração de VE, entender seu comportamento de carga e descarga é a tarefa principal (WANG *et al.*, 2012).

1.1 OBJETIVOS

Em relação ao Brasil, é difícil obter dados históricos que abrangem métricas como distância de deslocamento diário e duração de carregamento de VE. Como a implantação de VE no país ainda está na fase inicial de desenvolvimento, foi possível analisar esses dados com base em um método aleatório para modelar essas incertezas: o Método de Monte Carlo (MMC). Assim, este trabalho fez a análise de possíveis impactos causados pela inserção dos VE nos SD do Brasil através da interface do *software* OpenDSS e a linguagem de programação *python* para a sua manipulação.

O trabalho proposto apresenta algumas contribuições quanto às simulações e análises em relação ao trabalho de Pedro Eisenkraemer (2022), o qual foi utilizado como referência na elaboração deste Trabalho de Conclusão de Curso (TCC). O fator que leva à escolha do OpenDSS é a possibilidade de ser controlado por outros programas, como o *python*, neste caso, além de o OpenDSS vir se tornando muito popular no Brasil após ser adotado pela Agência Nacional de Energia Elétrica (FREITAS, 2020) conforme Nota Técnica nº 057/2014-SRD/ANEEL.

1.1.1 Objetivo geral

O Presente trabalho propões uma rotina de simulação que possibilita a identificação e quantificação de unidades consumidoras (UCs) sob maior impacto da implementação de VE no SD.

1.1.2 Objetivos específicos

- 1- Desenvolver o código em *python* para diferentes casos de penetração de VE;
- 2- Apresentar os principais impactos causados pelo alto nível de penetração;
- 3- Propor diferentes estratégias de carregamento para validar menor impacto no SD;
- 4- Comparar as diferentes estratégias de carregamento com o caso base.

2 REFERENCIAL TEÓRICO

Neste capítulo, são apresentados os VE, sua classificação e quais características do VE utilizadas como modelo na presente pesquisa. Também foi abordado a infraestrutura de recarga a ser adotada, o modo de recarga e os modelos de carregadores. Posteriormente, apresentam-se os impactos adversos que a implementação de VE causa, os meios de mitigação e o método para identificar e quantizar os impactos. Para finalizar, explica-se o MMC e sua aplicação em estudos de VE.

2.1 O VEÍCULO ELÉTRICO

Em 2022, a Associação Brasileira de Normas Técnicas (ABNT) lançou uma coleção de normas sobre a infraestrutura elétrica para carregamento de VE, nas quais também especifica que

[...] os Veículos Elétricos e Híbridos (VEH) caracterizam os veículos que são propelidos a eletricidade em alguma situação. São muitas as possibilidades de eletrificação de veículos, total ou parcial. Pode ser um veículo elétrico puro movido exclusivamente por motor/bateria, elétrico híbrido, com alguma combinação do uso de motor/bateria e motor a combustão (ABNT, 2022, s. p.).

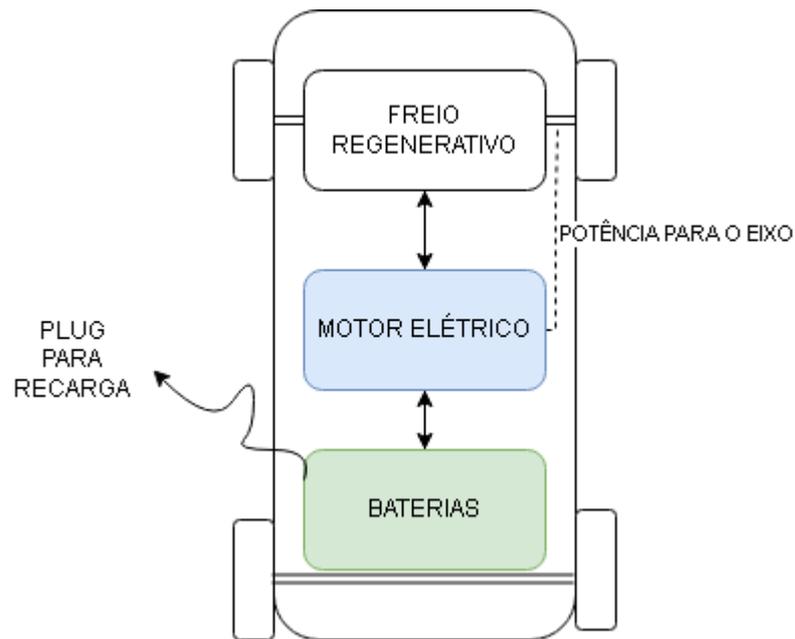
Os VE apresentam diferentes modelos de aplicação e classificação, podendo ser definidos entre quatro tipos, dependendo do aspecto do veículo (BAYRAM, 2017): veículo elétrico a bateria (BEV – *Battery Electrical Vehicle*); veículo elétrico híbrido (HEV – *Hybrid Electrical Vehicles*); veículo elétrico híbrido *plug-in* (PHEV – *Plug-in Hybrid Electrical Vehicle*); e veículo elétrico a célula de combustível (FCEV – *Fuel Cell Electrical Vehicle*). Entender o padrão de carga e descarga desses veículos é a primeira tarefa. Para isso, podemos considerar que o VE pode ser caracterizado como uma carga flexível e estocástica que muda espacial e temporariamente através da rede de distribuição. (ELNOHAZY; SALAMA, 2014).

2.1.1 Veículo elétrico a bateria (BEV)

Atualmente, nos BEVs, a bateria de íon de lítio é a mais utilizada, pois apresenta uma longevidade maior e é capaz de passar por mais ciclos de carga, que é o objetivo principal a ser aplicado em um VE atualmente (BOWELL, 2011). A Figura 2 ilustra o modelo simplificado da estrutura de um BEV. O BEV funciona a partir de um banco de baterias que alimenta o motor

elétrico, este distribui a potência para o eixo que faz com que as rodas girem, junto a este conjunto existe também o freio regenerativo, que serve para transformar a energia cinética absorvida no momento da freada em energia elétrica de volta para a bateria, aumentando a autonomia do veículo.

Figura **Error! No text of specified style in document.** – Diagrama simplificado da estrutura de um BEV.



Fonte: Elaborado pelo autor.

As vantagens desses veículos são que eles possuem zero emissão de poluentes durante o seu uso e são completamente independentes de combustível. Normalmente, o comportamento de utilização desses veículos é com recarga sendo feita na casa do proprietário. Modelos representativos desse tipo de VE à venda no Brasil e que apresentam um perfil de carga conhecido são o Renault Kwid e-tech (com bateria de capacidade de 26,7 kWh), o Hyundai Kona Electric (com bateria de capacidade de 39,2 kWh) e o Peugeot e-2008 (com bateria de capacidade de 50 kWh). Todos estes mencionados são de carregamento monofásico ou trifásico, dependendo apenas da Estação de Recarga (ER). Esse é o tipo de VE que será referenciado para os modelos de carga a serem simulados posteriormente.

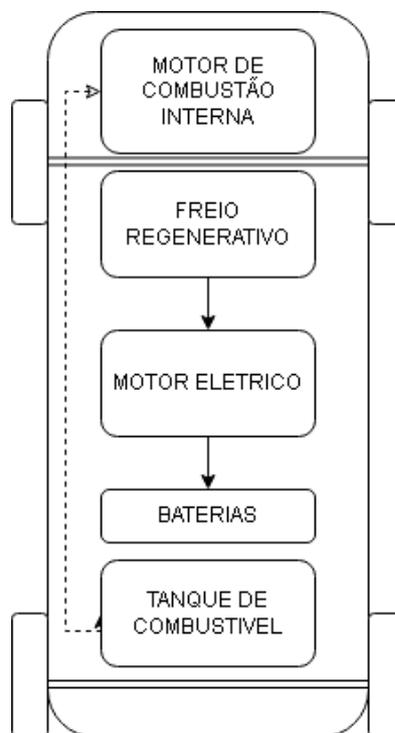
2.1.2 Veículo elétrico híbrido (HEV e PHEV)

Seguindo o mesmo princípio do BEV, o diferencial dos HEV e PHEV é que, junto com o motor elétrico, eles combinam um motor de combustão interna, o qual tem um minigerador

integrado capaz de carregar a bateria quando o veículo está em movimento. Normalmente, a dupla atuação dos motores aumenta a autonomia e a economia de combustível, como é possível ver na Figura 3.

O modo de atuação ocorre em duas possibilidades. A primeira é quando os níveis de energia armazenados na bateria chegam a um valor mínimo; então, o motor a combustão passa a ser utilizado. A segunda possibilidade é quando atinge um nível de velocidade muito elevado, e o objetivo passa a ser aumentar a eficiência do VE.

Figura 3 – Diagrama simplificado da estrutura de um HEV.

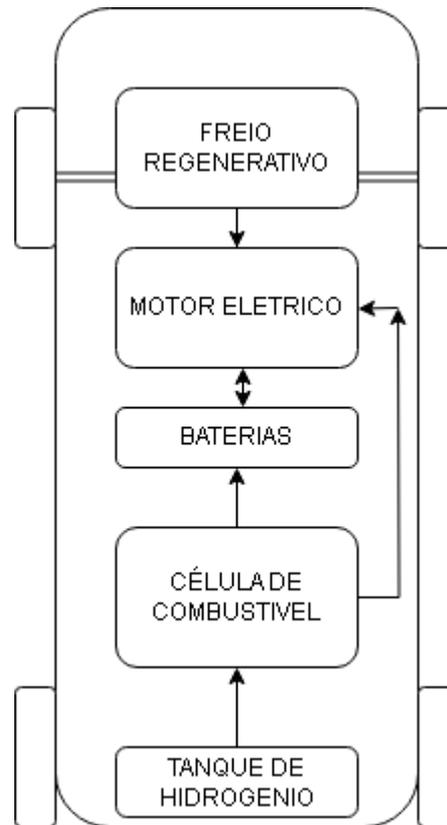


Fonte: Elaborado pelo autor.

2.1.3 Veículo elétrico a célula de combustível (FCEV)

O FCEV usa eletricidade para o funcionamento do motor. A diferença dele para o BEV é que, ao invés de usar somente sua bateria como fonte de energia, utiliza uma célula de combustível movida a hidrogênio. A maior vantagem do ponto de vista ambiental é que esse tipo de veículo não gera poluição enquanto está em uso, pois a queima do combustível gera água e calor. Apesar disso, esse tipo de VE não é aplicado em um volume tão grande quanto os BEV e HEV, pois não há tanto estudo de desenvolvimento e aperfeiçoamento dessa tecnologia, o que deixa o seu custo muito elevado em relação aos concorrentes. Na Figura 4, é ilustrado um diagrama simplificado da sua estrutura.

Figura 4 – Diagrama simplificado da estrutura de um FCEV.



Fonte: Autor

2.2 A ESTRUTURA DE RECARGA

A norma IEC (*International Electrotechnical Commission*) 61581-1:2010 define o padrão para três tipos de recarga em corrente alternada, que variam de acordo com o nível da corrente. O modo de carregamento devagar ou lento (do inglês, *slow charging*) é adotado em infraestruturas domésticas e públicas; já os modos *fast charging* (carregamento rápido) e *rapid charging* (outro modo de carregamento rápido) são encontrados somente em pontos de recarga públicos, onde é possível obter uma corrente muito mais alta de recarga (CHAN; CHAU, 2001). A padronização da SAE (*Society of Automobile Engineers*) (2017) é o que define os perfis de carregamento de acordo com o nível de corrente, conforme a Tabela 1.

Tabela 1 – Níveis de recarga de acordo com padrão SAE.

MÉTODO DE RECARGA	TENSÃO AC (V)	CORRENTE MAXIMA(A)	POTENCIA MAXIMA (kW)
AC 1	120 (1 fase)	12 - 16	1,44 – 1,92
AC 2	208-240 (1 fase)	24 - 80	5,0 – 19,2
AC 3	208-600 (3 fases)	80 - 200	22,7 - 166
DC 1	50 – 1000	80	80
DC 2	50 - 1000	400	400

Fonte: Adaptado de SAE (2017).

Considerando o atual cenário do Brasil, nota-se que os primeiros pontos de recarga que existirá serão domésticos — ou ainda trifásicos, se considerar estações de recarga em postos de trabalho. No presente trabalho, a definição dos perfis de recarga do VE é feita de acordo com o modo de carga lento.

Durante este trabalho, considera-se o carregador do fornecedor NEOCHARGE, exemplificado na Figura 5, sendo ele de uso doméstico (*wallbox*). Existem as versões de 7,4 kW e de 22kW (para recarga trifásica nessas simulações, será inserido um transformador de 30 kVA, que é instalado junto com esses carregadores). Mas é de conhecimento que no mercado existem estruturas de recarga com módulos fotovoltaicos e baterias para acúmulo de carga durante o dia, a fim de ter melhor eficiência energética.

Figura 5 – Carregador NEOCHARGE plug and play.



Fonte: SMART (2023).

A média de demanda de potência de 7,4 kW, que é a designada para um VE em carga lenta, será usada como padrão para os modelos. Além disso, cerca de 80% da energia armazenada na bateria do VE é de fato utilizada para fazer a roda girar (QIAN *et al.*, 2011), e esse será o valor usado para eficiência de recarga nos próximos cálculos.

2.3 IMPACTOS NO SISTEMA DE DISTRIBUIÇÃO

Diferentes estudos, tais como Gao, Meyer e Wang (2018) e Pothinun e Premrudeepreechacharn (2018), mostram que a incorporação de mais estações de recarga tem impactos negativos nos SD, reduzindo a qualidade de energia fornecida e a disponibilidade da rede. Vários fatores adversos podem vir junto com a implementação de VE, os quais variam conforme o seu grau de penetração aumenta, a distância percorrida diariamente por cada veículo, o hábito de carregamento dos usuários e as possíveis tarifas diferenciadas. Nesses trabalhos, foi utilizada uma análise de fluxo de potência através do *software* OpenDSS em combinação com o MATLAB, e os efeitos da recarga de VE foram visualizados com foco no impacto nas redes de distribuição.

Juntando as informações obtidas nos trabalhos mencionados e os dados desenvolvidos com a metodologia de Neto e Piotrowski (2019), onde é visto um modelo probabilístico proposto para diferentes tipos de veículos elétricos, há diferentes níveis de integração do sistema e perfis de recarga devido à natureza estocástica dos padrões de mobilidade. Este trabalho também usará esse método pelo OpenDSS, devido à velocidade, à confiança e à eficiência para resolver os fluxos de potência, que foram os principais pontos identificados por Gao, Meyer e Wang (2018) para esta escolha.

No trabalho de Boonraska *et al.* (2019), pelo método de Newton-Raphson, é estudado o impacto causado na rede pelo carregamento de um VE de transporte coletivo no sistema de teste (IEEE 33 barras). Nesse trabalho, torna-se evidente que a adição dessa carga é muito maior que a de um simples VE de uso particular, causando problemas de estabilidade e confiança, além de muito mais perdas no SD. As simulações mostram que o carregamento de VE causam uma sobrecarga nos transformadores devido ao aumento de pico de carga, variações no perfil de média tensão e até distorções harmônicas.

Nos trabalhos de Carillo *et al.* (2017) e Al Johani, Saad e Mohammed (2020), são estudados os efeitos de carga no transformador e a tensão de saída em um período de 24 horas. O resultado apresenta que recargas sem um plano de controle geram uso muito superior ao projetado da rede, reforçando a ideia de que a rede elétrica não está preparada. O estudo de

Skok *et al.* (2020) confirma essa teoria e conclui que a mobilidade requer a construção de uma infraestrutura elétrica que suporte o SD e garanta a confiança e a segurança, pois adotar VE causará grandes problemas nas linhas de baixa tensão. Em suma, após as verificações dessas pesquisas, temos a informação de que os possíveis problemas que ocorrem são: sobrecarga da linha e transformadores; aumento das perdas; redução da qualidade de energia; aumento da demanda de pico; e instabilidade de tensão.

2.4 MÉTODO DE MONTE CARLO

Devido à natureza estocástica das cargas, a solução pela Análise do Fluxo de Potência (AFP) é proposta empregando o Método de Monte Carlo (MMC), que resolve problemas de cenários como consumo e produção de energia (BETANCUR *et al.*, 2021). Os cenários são aleatoriamente gerados a partir de uma função de densidade de probabilidade. Mesmo que o MMC obtenha resultados com alto grau de precisão, neste tipo de estudo ele acaba requerendo um alto número de interações para convergir, o que faz com que os cálculos tenham um aumento significativo de tempo computacional (DIOP; HENNEBEL, 2017).

Sucintamente, as etapas do MMC são: sortear variáveis de entrada; obter resultado computacional através do modelo matemático; obter os resultados de saída; e repetir o processo. Para isso, há a variável estimativas de valor esperado $E(x)$ e variância $V(x)$. Para essas, variáveis, é adotado uma simplificação baseada no teorema do limite central (RUBINSTEIN; KROESE, 2008). Nesse teorema, a média da amostra é usada como estimativa do valor esperado quando há muitas iterações no processo. É possível equacionar o valor esperado conforme a seguinte equação.

$$E(x) = \frac{1}{N} \sum x_i \quad (11)$$

$$V(x) = \frac{1}{N} \sum (x_i - E(x))^2 \quad (2)$$

O desvio padrão se dá pela forma

$$\sigma = \sqrt{\frac{V(x)}{N}} \quad (3)$$

3 METODOLOGIA

Neste capítulo, são descritos os métodos adotados para adquirir os dados dos impactos na rede. Além disso, são apresentados o cenário a ser analisado e o método de programação através do *python* junto do OpenDSS.

3.1 CONSIDERAÇÕES INICIAIS

A pesquisa segue uma abordagem quantitativa, utilizando o *software* OpenDSS para simulações de fluxo de potência que representem diferentes cenários de penetração de VE e padrões de carga dos consumidores. É considerados três cenários de penetração de VE: baixa, média e alta. Além disso, a técnica MMC é empregada para gerar amostras aleatórias de perfis de carga dos consumidores, permitindo análises probabilísticas. As métricas de avaliação incluem as perdas elétricas, os níveis de tensão na rede e a sobrecarga dos transformadores. É importante destacar que, embora o estudo forneça *insights* relevantes, suas conclusões estão sujeitas às limitações dos dados disponíveis e à simplicidade dos modelos adotados. A pesquisa busca contribuir para o planejamento eficiente e sustentável da infraestrutura de recarga de VE e para o gerenciamento da demanda elétrica, visando à transição para um sistema de energia mais limpo e resiliente.

3.2 DESCRIÇÃO DO ALGORITMO

Como visto no capítulo anterior, para as simulações do MMC terem menor margem de erro, é necessário ter maior quantidade de iterações. Sendo assim, observa-se o diagrama de blocos para uma descrição simples de como funciona o algoritmo em *python* e sua implementação.

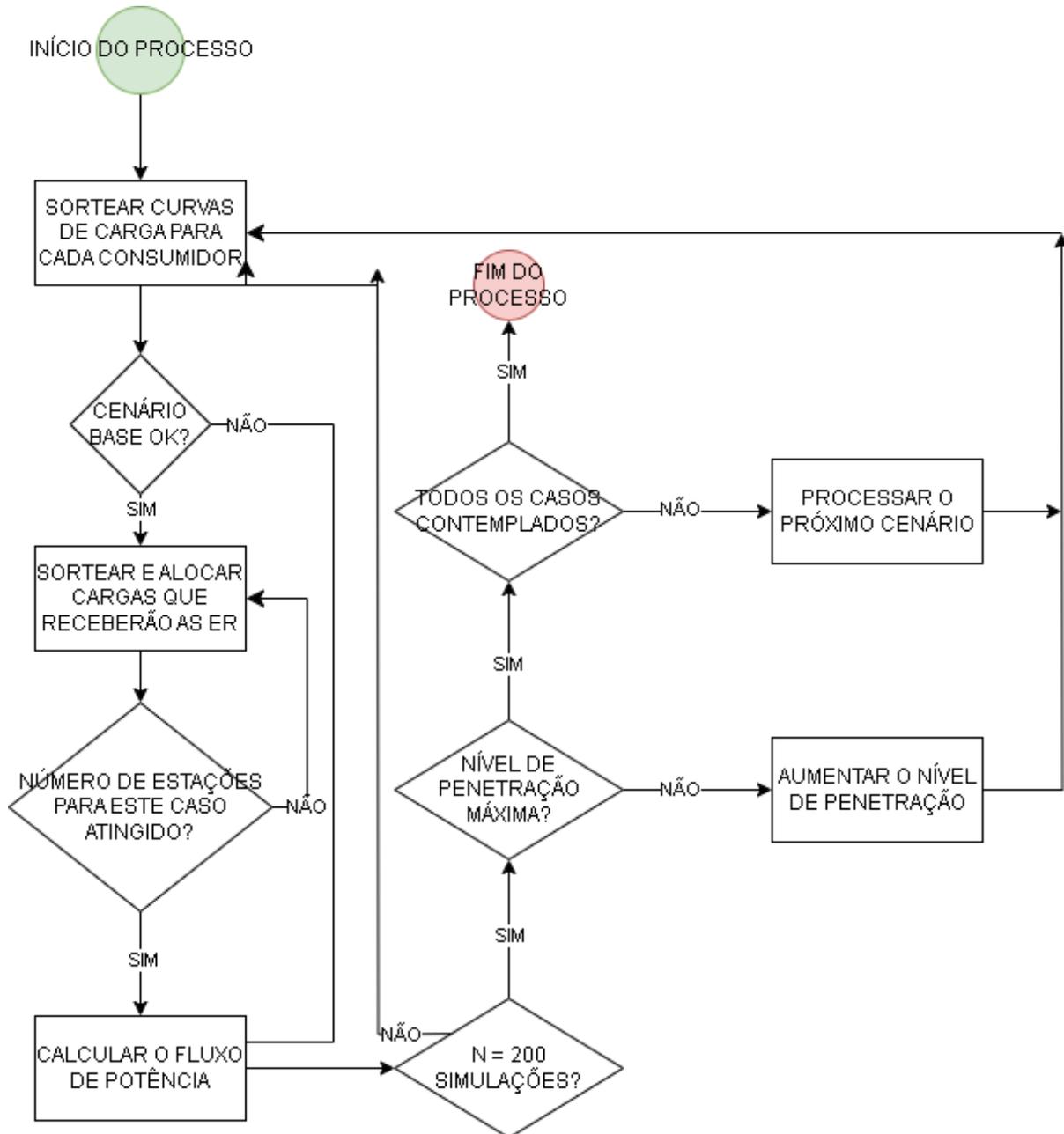
O código em é exposto em sequência no diagrama apresentado na Figura 6, de forma simplificada, como funciona o algoritmo, considerando o proposto por de Qian *et al.* (2019), que usa o MMC para estudar o impacto no SD com muitas iterações. Porém, neste trabalho, a quantidade de iterações será menos (150 simulações por cenário), devido ao limite computacional disponível as simulações levaram em torno de 7 horas para serem concluídas com as seguintes especificações do sistema:

CPU AMD Ryzen 7 3700U com Radeon Vega Mobile Gfx 2.3 GHz

GPU AMD Radeom RX Vega 10

10 GB RAM DDR4
Windows 11 (64 bit)

Figura 6 – Fluxograma do processo do algoritmo.



Fonte: Elaborado pelo autor.

3.2.1 Curvas de carga do consumidor

Para criar um modelo de carga probabilístico que modele o comportamento da carga do VE, uma função de densidade de probabilidade de distância percorrida diariamente foi gerada

através dos dados de pesquisa feita por Wang *et al.* (2012), o qual pode ser visto na equação (4), onde μ (média) e σ (desvio padrão) adotado como 0,5.

$$f(d) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{d-\mu}{\sigma}\right)^2} \quad (4)$$

Essa função é usada para simular diferentes cenários de demanda de energia, considerando a variação das curvas de cargas ao longo dessa distribuição de probabilidade. Ao final da execução da função, curvas de carga são criadas com essas novas variações, as quais, posteriormente, serão utilizadas nas simulações dos diferentes cenários, servindo como o fator de multiplicação dos valores de demanda para aumentar a carga.

3.2.2 Cenários

Seguindo para a segunda etapa do fluxo do processo, foram determinados 4 diferentes cenários.

- Cenário A: cenário base – para este caso, o índice de penetração de veículos elétricos é 0%.
- Cenário B: aqui, será estudada a alocação de 20% a 100%, a um passo de 20% de VE, com Estações de Recarga (ER) monofásicas distribuídas em todo o SD.
- Cenário C: neste caso, será alocado de 5% a 25%, a um passo de 5% de VE, com ER trifásicas na rede.
- Cenário D: neste caso, será considerado o caso de ER monofásicas e trifásicas simultâneas, com 20% a 100% de penetração de VE, a um passo de 20%.

3.2.3 Alocação de cargas

Para este caso, não é considerado o sistema de 33 barras como feito no trabalho de BOONRAKSA, et al. Será utilizado um sistema de rede de 3.669 barras com 847 UC, com tensão primária de 13,8 kV e operação em 60 Hz. Essa rede segue um modelo para simular a cidade de Barretos, no estado de São Paulo (SP).

Seguindo com referencial no trabalho de Qian *et al.* (2018), considera-se que o número total de UC da rede é a quantidade de pontos de ER para os VE, considerando 100% de

penetração, como se em todas as UC houvesse uma ER. Para as ER trifásicas, foram consideradas apenas barras de média tensão e, para ER monofásicas, somente barras de baixa tensão. A aleatoriedade da seleção das unidades foi realizada por meio aleatório, com a função *numpy random* no *python*.

Essa função de alocação de cargas atua de acordo com o cenário e o nível de penetração de VE fornecido. Primeiro, recebe o nome do cenário e o nível de penetração de VE como entrada; em seguida, com base no cenário e no nível de penetração, identifica quais as estações serão afetadas pela presença de VE. Para cada estação afetada, a função vai ajustar a curva de carga correspondente, a fim de refletir o aumento na demanda de energia devido à alocação da ER.

3.2.4 Resultados dos impactos

Após o processamento das etapas anteriores, os dados são coletados e organizados em listas específicas por cenário para, posteriormente, serem criados os gráficos para comparar o desempenho do SD em cada um desses cenários. Os dados a serem analisados são:

- a) Perdas: representa as perdas no SD de cada simulação para diferentes níveis de penetração dos VEs e para os diferentes cenários. As perdas são medidas em porcentagem da carga total do sistema. O índice de perdas elétricas (IPE) é dado conforme a equação (5):

$$IPE = \frac{\textit{perdas}}{\textit{consumo}} \times 100\% \quad (5)$$

- b) Níveis de tensão: são recebidos dados de tensão máxima e mínima observadas no SD em cada simulação para diferentes níveis de penetração de VE e para os diferentes cenários. Essas tensões são normalizadas em pu e devem seguir as faixas nominais de operação estabelecidas pela PRODIST, no módulo 8 – Qualidade da Energia Elétrica, conforme Tabela 2.

Tabela 2 – Limites de operação da Rede de Distribuição.

TENSÃO DE ATENDIMENTO (TA)	FAIXA DE VARIAÇÃO DA TENSÃO DE LEITURA (TL) EM RELAÇÃO A TENSÃO NOMINAL (TN)
ADEQUADA	$0,92 \text{ TN} \leq \text{TL} \leq 1,05 \text{ TN}$
PRECÁRIA	$0,87 \text{ TN} \leq \text{TL} \leq 0,92 \text{ TN}$ OU $1,05 \text{ TN} \leq \text{TL} \leq 1,06 \text{ TN}$
CRÍTICA	$\text{TL} < 0,87\text{TN}$ OU $\text{TL} > 1,06 \text{ TN}$

FONTE: (PRODIST, 2021)

- c) Sobrecarga de transformadores: é recebido o número de transformadores que estão sobrecarregados em cada simulação e o carregamento médio deles para diferentes níveis de penetração de VE e para todos os cenários pré-estabelecidos.

4 OBTENÇÃO DOS RESULTADOS

Como exposto no trabalho de Qian *et al.* (2018), o OpenDSS pode ser comandado por Matlab. Aqui, é usado o *python* junto do pacote disponível por Paulo Radatz de Freitas “py-dss-interface”, cujo acesso é permitido pela interface *Dynamic Link Library* (DLL)

Em *python*, é possível utilizar classes como forma de organizar dados e funcionalidades. Para isso, foi criado um “tipo” de objeto, permitindo que novas “instâncias” desse tipo sejam produzidas e cada instância da classe possa ter atributos anexados a ela. A classe criada pode ser vista conforme o segmento de código adaptado do trabalho feito por Eisenkraemer (2022) apresentado na sequência. Dessa forma, conseguimos salvar os dados de modo que, posteriormente, ficará mais fácil o seu acesso para manobrar no código principal.

```

Class Rede(object):
    def __init__(self, losses, vmax, vmin, overload_transformers, overload_mean, cenario,
penetration_level):
        self.losses = losses
        self.vmax = vmax
        self.vmin = vmin
        self.overload_transformers = overload_transformers
        self.overload_mean = overload_mean
        self.cenario = cenario
        self.penetration_level = penetration_level

```

Para a sequência, foram criadas funções para melhor leitura e posterior manipulação do código, indexadas no apêndice A deste trabalho

4.1 CENÁRIO A

Com o grupo de controle, primeiramente foi feito a simulação para o caso base. Com o índice zero de penetração de VE, é obtido o relatório da simulação na Tabela 3. Com o caso de controle, verifica-se que a média de tensão máxima e mínima estão dentro dos limites da PRODIST, vistos previamente na Tabela 2.

É observável também que já existem dois transformadores sobrecarregados nesse SD, o que é indicador de que, naturalmente, o sistema já atua sobrecarregado em seu estado inicial. Isso pode ser um indicativo de que esses transformadores já estão localizados em uma área em que há maior aglomeração de unidades, o que é um ponto crítico para o sistema conforme o aumento das cargas que serão implementadas. O que pode ser observado em alguns pontos na figura 7, onde transformadores específicos apresentam uma sobrecarga muito maior que outros.

Tabela 3 – Resultados obtidos no caso base.

INFORMAÇÕES	VALOR
MÉDIA DE PERDAS	4.383%
MÉDIA DA TENSÃO MÁXIMA	1.0352 pu
MÉDIA DE TRANSFORMADORES SOBRECARRREGADOS	2
MÉDIA DE TENSÃO MÍNIMA DOS TRANSFORMADORES	0.9632 pu

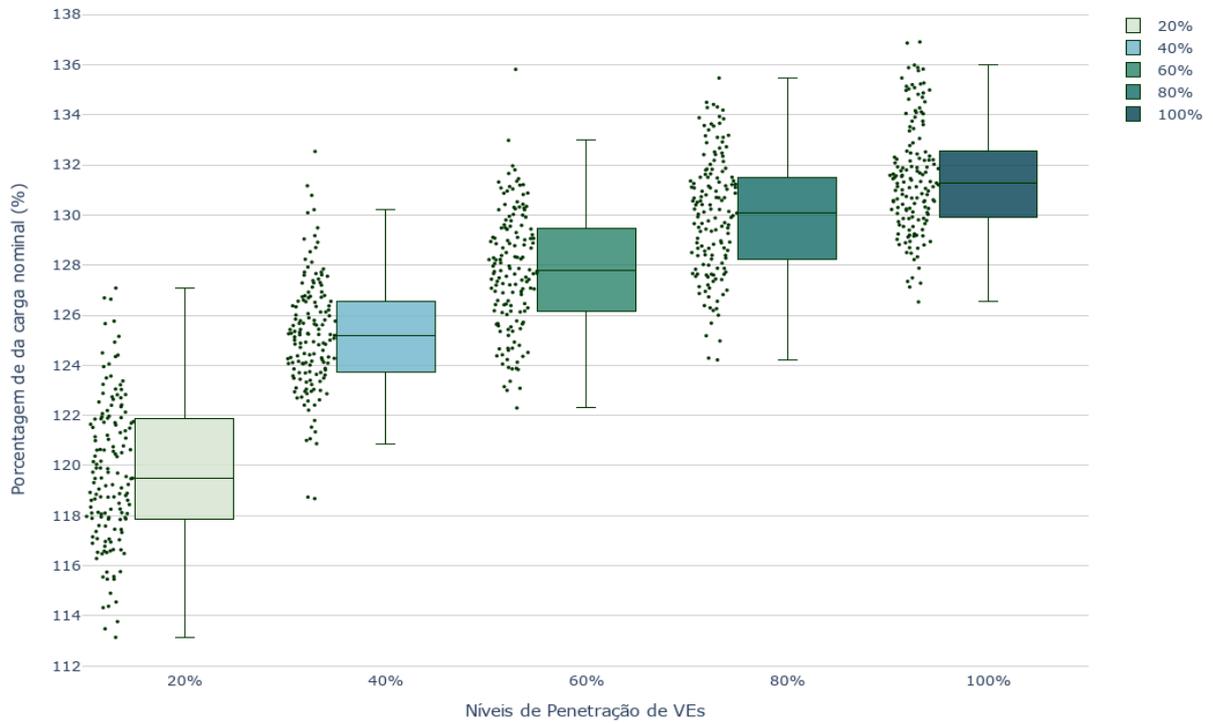
Fonte: Elaborado pelo autor.

A média de perdas elétricas para este caso base está em torno de 4,4%. Segundo relatórios disponíveis pela ANEEL (2021), esse valor é bem abaixo da média do Brasil inteiro, que está em torno de 7,5% de acordo com os estudos realizados de perdas regulatórias.

4.2 CENÁRIO B

No cenário B, caso de penetração de apenas unidades monofásicas inseridas no sistema, é visto na Figura 7, que logo com um índice de 20% de penetração na rede já é visível transformadores carregados em mais de 126%, dependendo do sorteio do algoritmo. O cenário fica pior para a sobrecarga do transformador conforme aumenta o índice de penetração. Também é possível observar que, em alguns pontos da rede onde foi alocado, as cargas acabam afetando de menor modo a rede, fato observado pelo caso base, em que dois transformadores já operavam previamente em sobrecarga. Esses pontos podem ser mais bem estudados, de modo que sejam selecionados pelo SD como apto a receber uma ER.

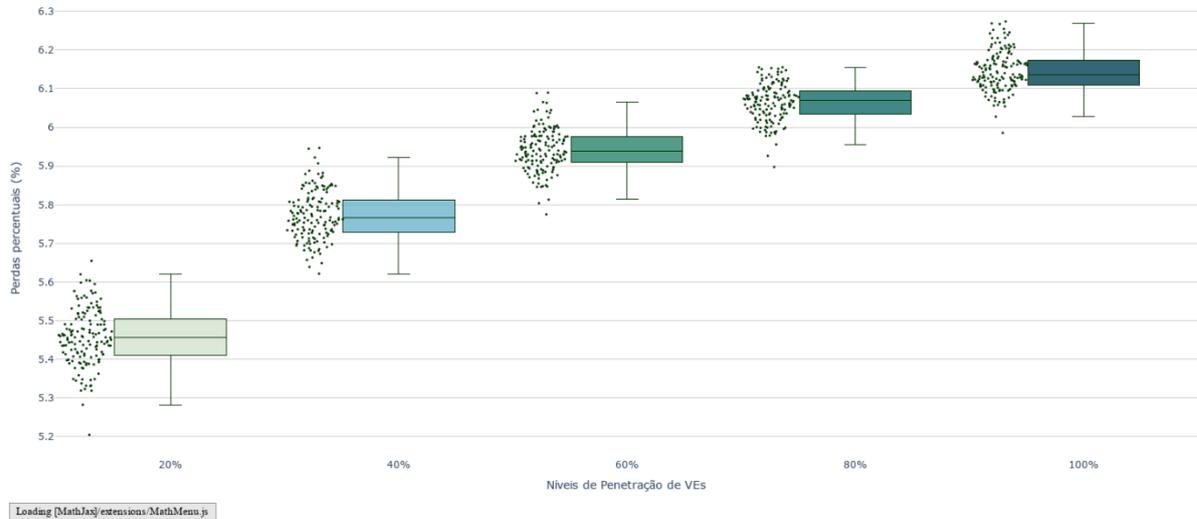
Figura 7 – Níveis de carregamento de transformadores sobrecarregados no cenário B.



Fonte: Elaborado pelo autor.

Para o caso das perdas, podemos ver, na Figura 8, que o aumento do fluxo de energia no sistema devido à demanda da carga dos VE resulta num aumento gradual das perdas no SD, ultrapassando os 6% quando a penetração atinge 100%

Figura 8 – Índice de perdas (%) no sistema no cenário B.

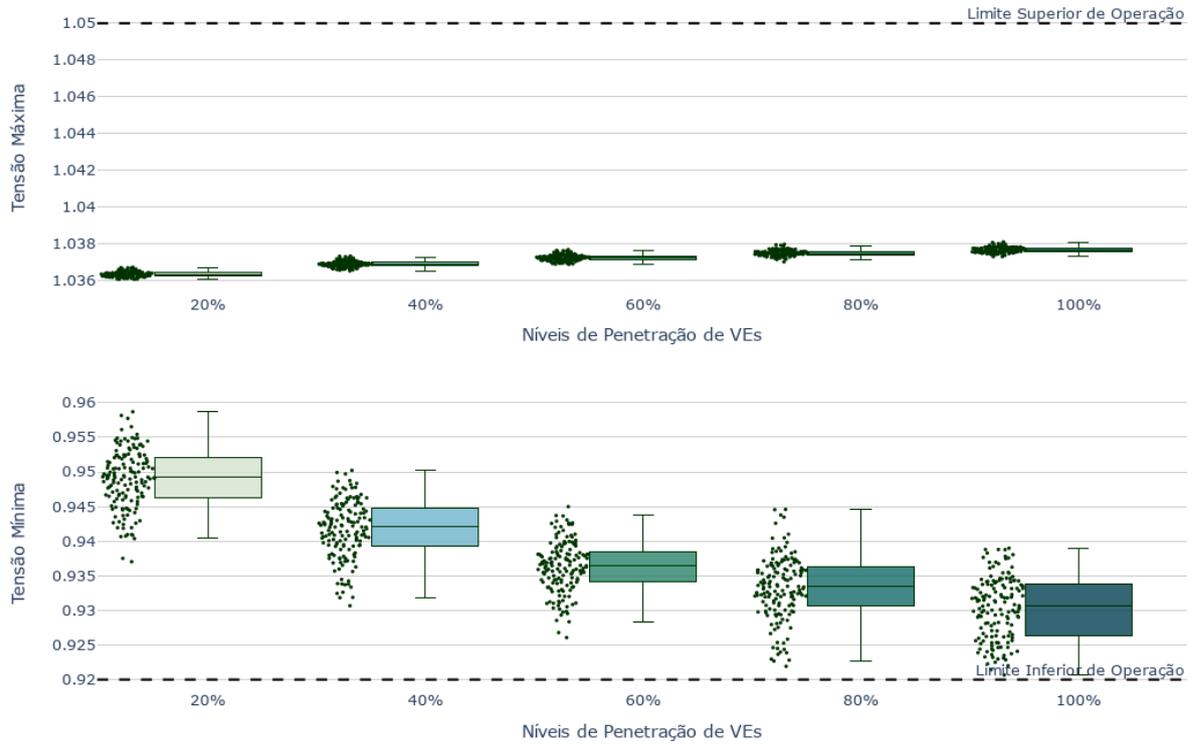


Fonte: Elaborado pelo autor.

Os limites de tensão máxima e mínima no SD observados durante a simulação para todos os níveis de penetração de VE no cenário B acabam por dentro da norma da PRODIST nos limites aceitáveis, conforme Figura 9.

É de se notar que a tensão mínima atingida no SD cai bruscamente conforme o aumento da penetração dos VE. Isso se deve ao fato de a geração para esse modelo de rede não estar corretamente dimensionada para essa grande quantidade de carga nova implementada.

Figura 9 – Limites de Tensão (em pu) no cenário B.



Fonte: Elaborado pelo autor.

O aumento de transformadores sobrecarregados conforme o aumento do nível de penetração de VE no SD também é visível na tabela 4. Apenas ao acrescentar 20% das UC com carregadores monofásicos, já é visível que os transformadores da rede passarão a operar com sobrecarregamento. Isso é evidente porque a estrutura da rede distribuição ainda não está preparada para um aumento tão significativo de cargas.

Mesmo com a inserção de 100% de VE nas unidades e o limite de tensão máxima ainda esteja abaixo do valor máximo adequado, o limite inferior de operação atinge o seu limite, deixando à beira da criticidade da tensão de atendimento da concessionária.

Tabela **Error! No text of specified style in document.** – Transformadores sobrecarregados no Cenário B

Nível de penetração 20%	7
Nível de penetração 40%	8
Nível de penetração 60%	8
Nível de penetração 80%	8
Nível de penetração 100%	9

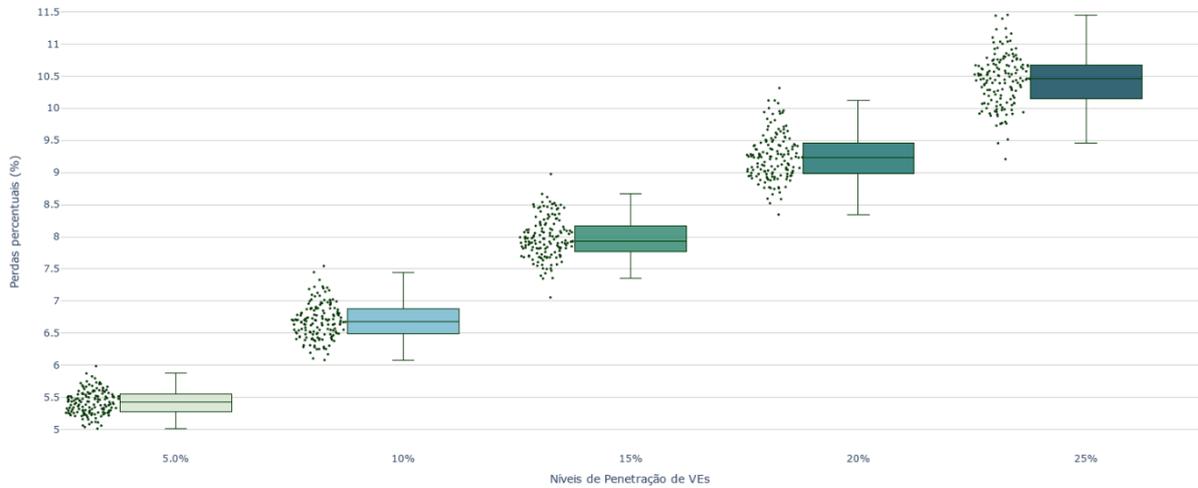
Fonte: Autor

4.3 CENÁRIO C

Para o cenário C, cujo caso foi apenas ER trifásicas, foi observada a mesma reação para as perdas conforme o aumento da penetração. Porém, desta vez, apenas com 25% de penetração, já se atingia um nível de perdas em torno de 10,5%, o que já ultrapassa a média geral do Brasil (ANEEL, 2021). Nesse cenário, foi simulado apenas até 25% de penetração, pois foi verificado que, mais que isso, já ultrapassava muito o valor de limite inferior de operação da tensão da rede, não sendo capaz de o alimentador do SD suprir toda a demanda.

Na Figura 10 e na Figura 11, vemos que, dependendo do ponto de alocação da carga da UC trifásica, acaba-se tendo um impacto significativo no índice de perdas e um impacto muito maior no nível de carregamento dos transformadores conforme o aumento de penetração.

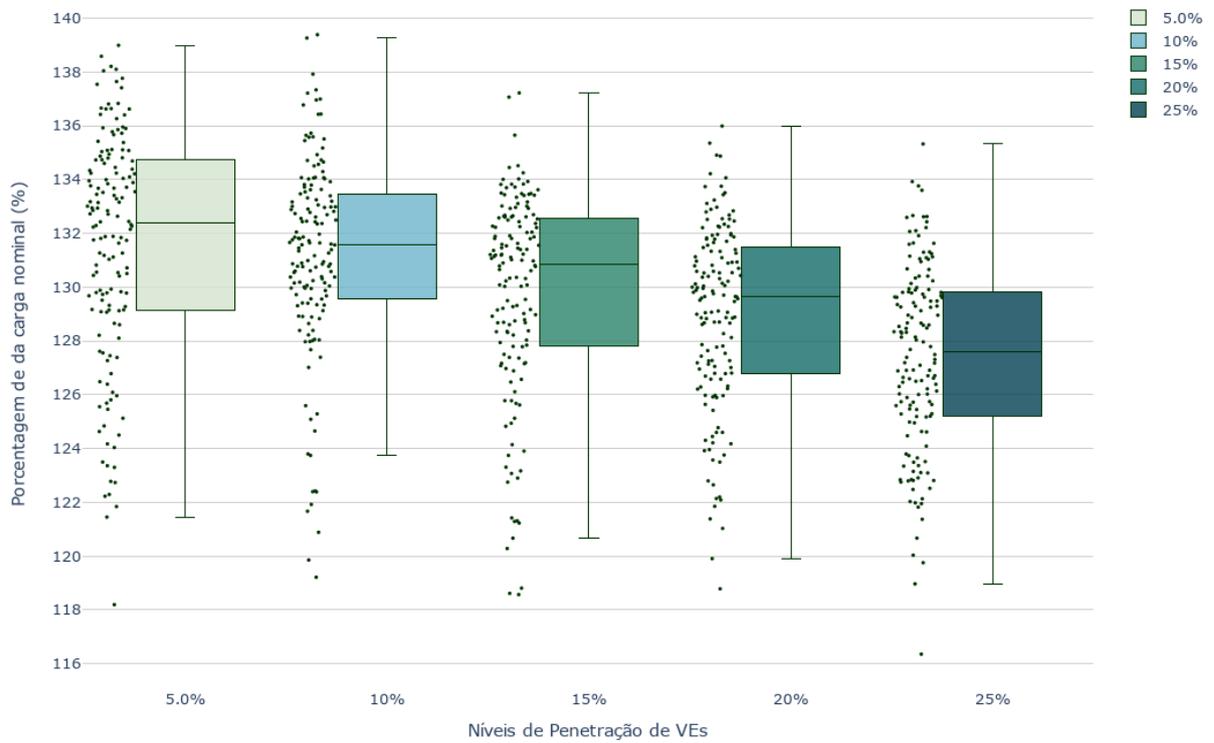
Figura 10 – Índice de perdas (%) no sistema no cenário C.



Fonte: Elaborado pelo autor.

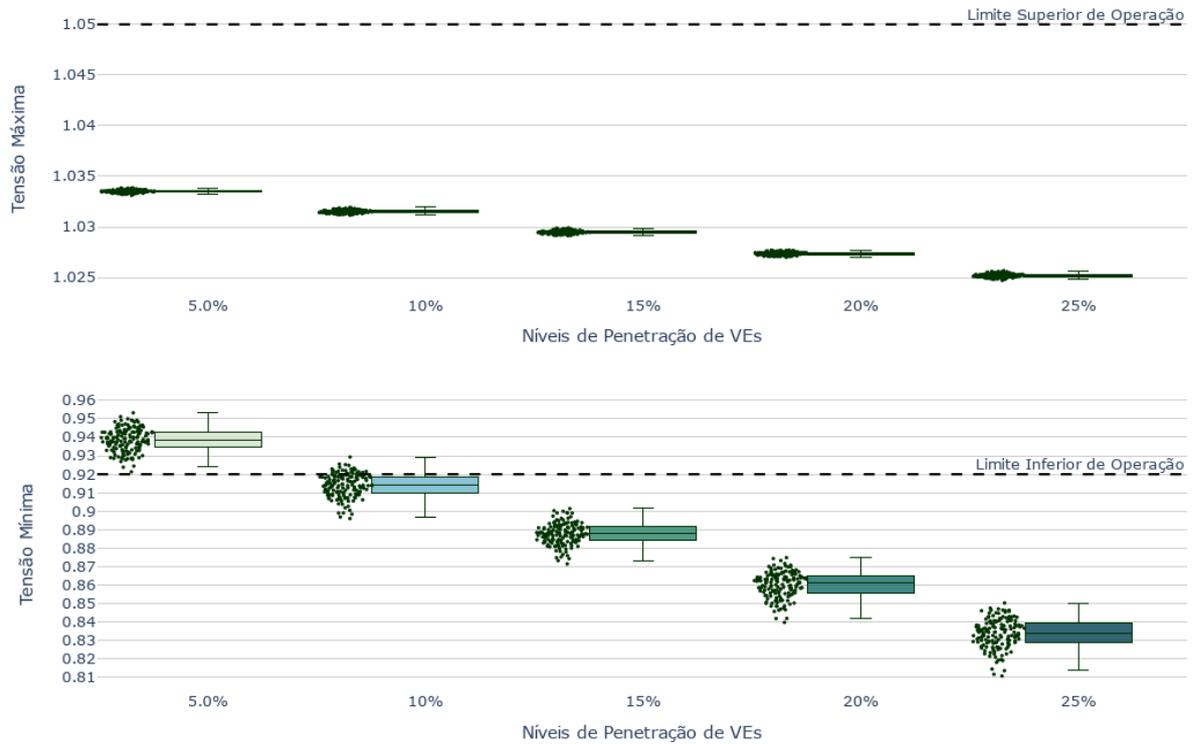
Por outro lado, o nível de carregamento dos transformadores que operam sobrecarregados acaba diminuindo conforme aumento do nível de penetração. Isso ocorre porque, na ER trifásica deste caso, já consta um transformador de 30kVa, normalmente instalado junto a essas estações para atender à demanda. Vale atentar que, se esse transformador instalado fosse de potência ainda maior, essa queda na sobrecarga dos transformadores previamente instalados no SD seria ainda menor. É importante considerar que, conforme a Tabela 5, o número de transformadores em sobrecarga neste caso é o mesmo do caso base, o que se deve ao fato de as cargas terem sido aplicadas apenas em barras trifásicas, o que acaba não impactando os transformadores de distribuição que foi visto no caso base.

Figura 11 – Nível de carregamento dos transformadores em sobrecarga no cenário C.



Fonte: Elaborado pelo autor.

Figura 12 – Limites de tensão (em pu) no cenário C.



Fonte: Elaborado pelo autor.

Tabela 5 – Transformadores sobrecarregados no cenário C.

Nível de penetração 5%	2
Nível de penetração 10%	2
Nível de penetração 15%	2
Nível de penetração 20%	2
Nível de penetração 25%	2

Fonte: Elaborado pelo autor.

4.4 CENÁRIO D

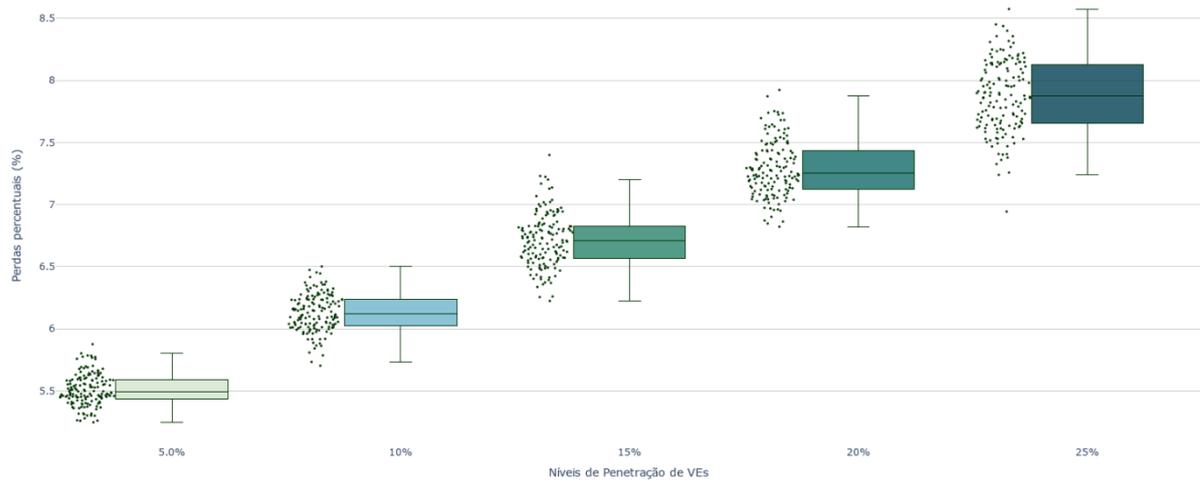
No cenário D, com a implementação de ER monofásicas e trifásicas, conseguimos observar na Figura 14 que, a partir de um índice de penetração de 15%, é ultrapassado o limite mínimo de tensão da rede de 0,92 pu. Não sendo assim também uma solução adequada para o início da implementação de VE no SD proposto.

As ER trifásicas afetam de forma muito intensa os valores de tensão na rede, e as ER monofásicas fazem que com o número de transformadores sobrecarregados aumente. Ambas

causam o aumento do IPE no SD (Figura 13), e a combinação de ambas acaba deixando a rede em um estado crítico muito rapidamente. Em resumo, esse cenário acaba pegando apenas os pontos negativos dos dois cenários anteriores e juntando-os, revelando um quadro muito pior para a rede quando aplicado simultaneamente.

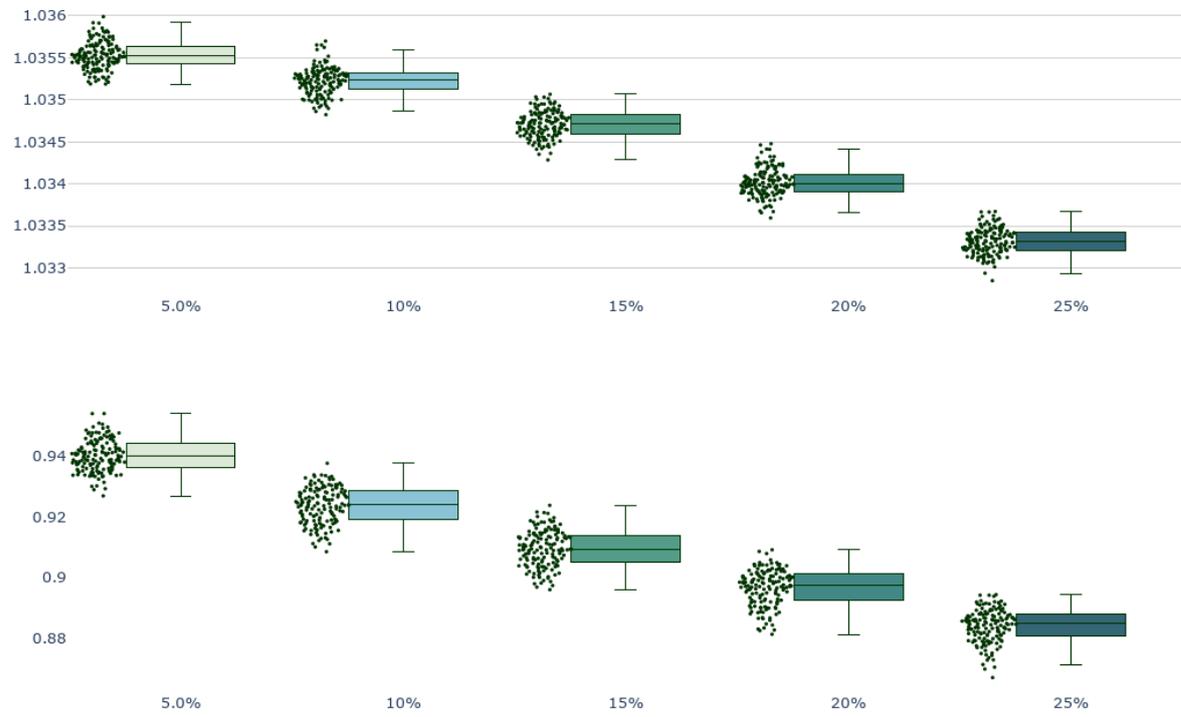
Na Figura 15, é possível ver que o nível de carregamento dos transformadores sobrecarregados é um pouco menor se comparado com o cenário do caso B. Porém, de acordo com a Tabela 6, referente à quantidade de transformadores sobrecarregados, vemos que logo com o baixo nível de penetração já supera em quantidade o número de transformadores operando em sobrecarga.

Figura 13 – Índice de perdas (%) no sistema no cenário D.



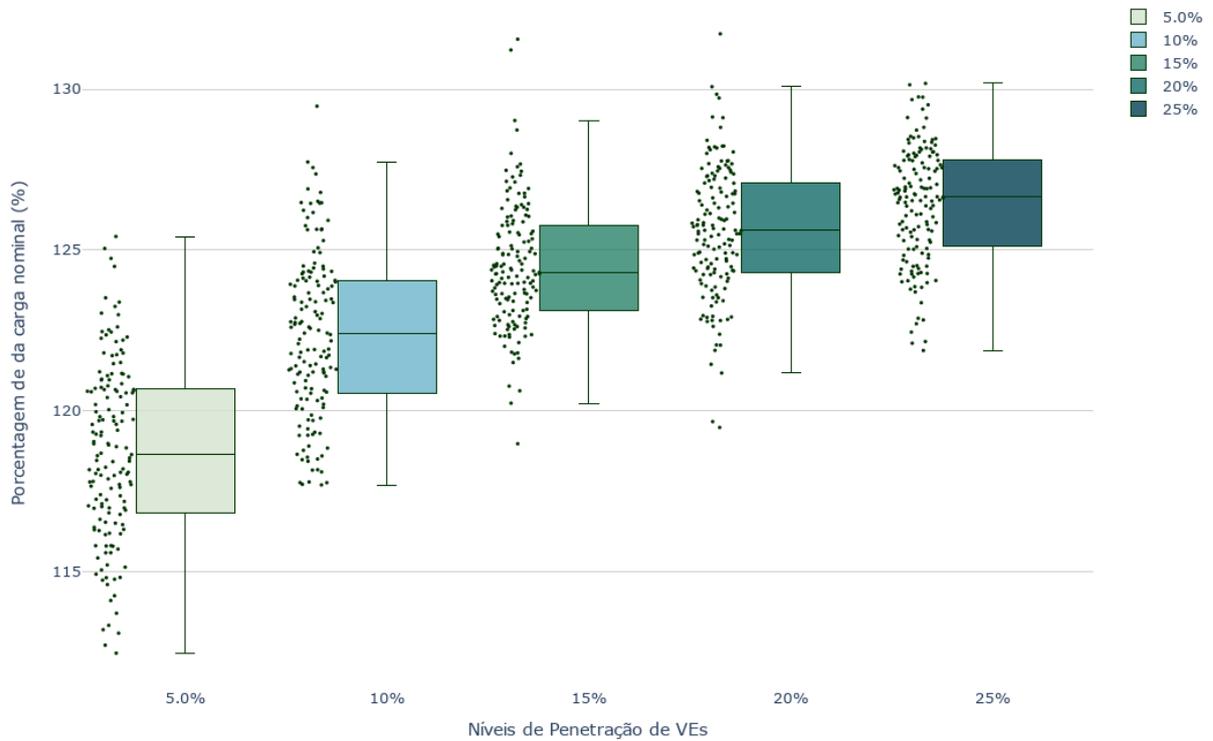
Fonte: Elaborado pelo autor.

Figura 14 – Limites de tensão (em pu) no cenário D.



Fonte: Elaborado pelo autor.

Figura 15 – Nível de carregamento dos transformadores sobrecarregados no cenário D.



Fonte: Elaborado pelo autor.

Tabela 6 – Transformadores sobrecarregados no cenário D.

Nível de penetração 5%	7
Nível de penetração 10%	8
Nível de penetração 15%	8
Nível de penetração 20%	8
Nível de penetração 25%	8

Fonte: Autor

5 CONCLUSÃO

Neste estudo, conduziu-se uma abrangente análise comparativa dos impactos associados à introdução de VE nos sistemas de distribuição de energia elétrica (SDEE). A principal abordagem utilizada foi a implementação de uma metodologia baseada no Método de Monte Carlo para modelar diferentes cenários de inserção de VEs. Para atingir o objetivo, integrou-se ferramentas computacionais por meio da linguagem de programação *python* e do software de código aberto OpenDSS. Realizamos simulações que levaram em consideração a variabilidade dos perfis de consumo dos consumidores e a alocação de estações de carregamento de VE.

O estudo incluiu três modelos de estações de carregamento de VE na rede, representando estações monofásicas e estações trifásicas convencionais e a combinação das duas. Explorou-se diferentes cenários que combinaram essas estações com diversos níveis de penetração na rede. Os resultados indicam uma relação significativa entre o aumento da penetração e o crescimento das perdas no SD, o que se torna uma preocupação relevante para os operadores de sistemas elétricos, especialmente à medida que a mobilidade elétrica ganha força. Além disso, notou-se impactos substanciais nos níveis de tensão da rede — exceto no cenário com estações monofásicas residenciais. Os transformadores também sofreram sobrecarga em certos cenários, o que afeta seu desempenho.

Conclui-se que é essencial desenvolver novas técnicas e metodologias para avaliar os impactos da operação de VE nos sistemas de distribuição. À medida que a mobilidade elétrica cresce, surgem desafios complexos para o planejamento e a operação de sistemas elétricos. O investimento em inovação e em projetos de pesquisa e desenvolvimento é fundamental para enfrentar esses desafios e prosperar em um cenário de mobilidade elétrica em crescimento.

Este estudo representa um passo importante na compreensão dos impactos da mobilidade elétrica nos sistemas elétricos e destaca a necessidade contínua de adaptação e inovação nesse setor. Para estudos futuros, sugere-se considerar estações integradas com fontes de energia renováveis, as quais, segundo estudos referenciados durante este trabalho, mostraram menor impacto em termos de perdas e níveis de tensão, devido à geração distribuída incorporada a elas. Isso abre oportunidades para aplicações e modelos de negócios inovadores, que promovam a integração eficaz da mobilidade elétrica com o sistema de energia.

REFERÊNCIAS

AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA – ANEEL. **Anexo VIII da Resolução Normativa ANEEL nº 956, de 7 de dezembro de 2021.** Estabelece Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional – PRODIST, revoga as Resoluções Normativas nº 395, de 15 de dezembro de 2009; nº 424, de 17 de dezembro de 2010; nº 432, de 5 de abril de 2011 e dá outras providências. Brasília, 2021. Disponível em: https://www2.aneel.gov.br/cedoc/aren2021956_2_7.pdf. Acesso em: 9 de setembro de 2023.

AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA – ANEEL. **Perdas de energia elétrica na distribuição.** 1. ed. Brasília, 2021. Disponível em: https://antigo.aneel.gov.br/documents/654800/18766993/Relat%C3%B3rio+Perdas+de+Energia_+Edi%C3%A7%C3%A3o+1-2021.pdf/143904c4-3e1d-a4d6-c6f0-94af77bac02a#:~:text=As%20perdas%20totais%20na%20distribui%C3%A7%C3%A3o,do%20mercado%20consumidor%20em%202020. Acesso em: 9 de setembro de 2023.

ALJOHANI, T.; SAAD, A.; MOHAMMED, O. On the real-time modeling of voltage drop and grid congestion due to the presence of electric vehicles on residential feeders. *In: PROCEEDINGS OF THE 2020 IEEE INTERNATIONAL CONFERENCE ON ENVIRONMENT AND ELECTRICAL ENGINEERING AND 2020 IEEE INDUSTRIAL AND COMMERCIAL POWER SYSTEMS EUROPE (EEEIC/I&CPS Europe)*, 2020, Madrid, Spain, **Anais...** 9-12 de junho de 2020. p. 1-7

BAYRAM, A. T. I. S. **Plug-in electric vehicle grid integration.** [S. l.; s. n.], 2017. 276 p.

BETANCUR, D.; DUARTE, L. F.; REVOLLO, J.; RESTREPO, C.; DÍEZ, A. E.; ISAAC, I. A.; LÓPEZ, G. J.; GONZÁLEZ, J. W. Methodology to evaluate the impact of electric vehicles on electrical networks using Monte Carlo. **Energies**, v. 14, n. 5, 2021, 14, 1300

BOONRAKSA, T.; PAUDEL, A.; DAWAN, P.; MARUNGSRI, B. Impact of electric bus charging on the power distribution system a case study IEEE 33 bus test system. *In: PROCEEDINGS OF THE 2019 IEEE PES GTD GRAND INTERNATIONAL CONFERENCE AND EXPOSITION ASIA (GTD Asia)*, 2019, Bangkok, Thailand, **Anais...** 19-23 de março de 2019. p. 819-823.

BOXWELL, M.; BOXWELL, M. **The 2011 electric car guide.** Ryton on Dunsmore, Warwickshire, U.K.: Greenstream Pub., 2011.

CARRILLO, A.; PIMENTEL, A.; RAMIREZ, E.; LE, H. T. Mitigating Impacts of Plug-in Electric Vehicles on Local Distribution Feeders Using a Charging Management System. *In: PROCEEDINGS OF THE 2017 IEEE TRANSPORTATION ELECTRIFICATION CONFERENCE AND EXPO (ITEC)*, 2017, Chicago, IL, USA, **Anais...** 22-24 de junho de 2017. p. 174-179.

CHAN, C.; CHAU, K. **Modern electric vehicle technology.** Oxford: Oxford University Press, 2001.

CONTI, S.; RAITI, S. Probabilistic load flow for distribution networks with photovoltaic generators part 1: theoretical concepts and models. *In: PROCEEDINGS OF THE 2007*

DIOP, F.; HENNEBEL, M. Probabilistic Load Flow Methods to Estimate Impacts of Distributed Generators on a LV Unbalanced Distribution Grid. *In: PROCEEDINGS OF THE 2017 IEEE MANCHESTER POWERTECH*, 2017, Manchester, UK, **Anais...** 18-22 de junho de 2017. p. 1-6

EHSANI, M.; SINGH, K. V.; BANSAL, H. O.; MEHRJARDI, R. M. State of the art and trends in electric and hybrid electric vehicles. **Proceedings of the IEEE**, v. 109, n. 6, p. 967-984, 2021.

EISENKRAEMER, P. H. **METODOLOGIA PARA ANÁLISE DE IMPACTO DE VEÍCULOS ELÉTRICOS EM REDES DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA**. Trabalho de Conclusão de Curso, Universidade Federal de Santa Maria, Rio Grande do Sul, 2022.

ELECTRIC car sales, 2016-2023, **IEA**, Paris, 2023. Disponível em: <https://www.iea.org/data-and-statistics/charts/electric-car-sales-2016-2023>. Acesso em: 9 de setembro de 2023.

ELNOZAHY, M. S.; SALAMA M. M. A. A comprehensive study of the impacts of PHEVs on residential distribution networks. **IEEE Transactions on Sustainable Energy**, v. 5, n. 1, p. 332-342, 2014.

FREITAS, P. R. R. **Impacto de funções inteligentes de inversores de sistemas fotovoltaicos na operação de redes de distribuição de energia elétrica**. 2019. 191 f. Dissertação (Mestrado em Engenharia Elétrica) — Programa de Pós-Graduação em Engenharia Elétrica, Universidade de São Paulo, São Paulo, 2019.

GAO, Z.; MEYER, D.; WANG, J. Visualizing the impact of pev charging on the power grid. *In: PROCEEDINGS OF THE 2018 IEEE TRANSPORTATION ELECTRIFICATION CONFERENCE AND EXPO (ITEC)*, 2018, Long Beach, CA. **Anais...** 13-15 de junho de 2018. p. 737-743

INTERNATIONAL CONFERENCE ON CLEAN ELECTRICAL POWER, 2007, Capri, Italy, **Anais...** 21-23 de maio de 2007. p. 132-136.

IEC/ISO 61581-1:2010. Electric vehicle conductive charging system - Part 1: General requirements.

JARAMILLO, P.; SAMARAS, C.; WAKELEY, H.; MEISTERLING, K. Greenhouse gas implications of using coal for transportation: Life cycle as-sessment of coal-to-liquids, plug-in hybrids, and hydrogen pathway. **Energy Policy**, v. 37, n. 7, p. 2689-2695, jul., 2009.

NETO, N. K.; PIOTROWSKI, L. Methodology for analysis of the impact of residential charging of electric vehicles. **IEEE Latin America Transactions**, v. 17, n. 6, p. 953-961, jun., 2019.

ONU. Objetivos de Desenvolvimento Sustentável | As Nações Unidas no Brasil. Disponível em: <https://brasil.un.org/pt-br/sdgs>. Acessado em 16 de novembro de 2023.

POTHINUN, T.; PREMRUDEEPREECHACHARN, S. Power Quality Impact of Charging Station on MV Distribution Networks: A Case Study in PEA Electrical Power System. *In: PROCEEDINGS OF THE 2018 53rd INTERNATIONAL UNIVERSITIES POWER ENGINEERING CONFERENCE (UPEC)*, 2018, Glasgow, UK, **Anais...** 4-7 de setembro de 2018. p. 1-6.

RUBINSTEIN, R. Y., KROESE, D. P., 2008. Simulation and the Monte Carlo Method. 2^a Ed. John Wiley & Sons, Inc.

QIAN, K.; ZHOU, C.; ALLAN, M.; YUAN, Y. Modeling of load demand due to EV battery charging in distribution systems. **IEEE Transactions on Power Systems**, v. 26, n. 2, p. 802-810, 2011.

SAE electric vehicle and plug in hybrid electric vehicle conductive charge coupler. **SAE International**, 13 de outubro de 2017.

SAMARAS, C.; MEISTERLING, K. Life cycle assessment of greenhouse gas emissions from plug-in hybrid vehicles: Implications for policy. **Environmental Science & Technology**, Washington, v. 42, n. 9, p. 3170-3176, abr., 2008.

SKOK, S.; HAVAŠ, L.; RADOSEVIC, V.; CVITANOVIC, M. Impact of electromobility to the power distribution system. *IN PROCEEDINGS OF THE 2020 IEEE PES/IAS PowerAfrica*, 2020, Nairobi, Kenya, **Anais...** 25-28 de agosto de 2020. p. 1-5.

SMART Wallbox Neocharge. **NeoCharge**. [S. l.]: s. d. Disponível em: <https://www.neocharge.com.br/carregador-carro-eletrico/smart-wallbox-neocharge>. Acesso em 9 de setembro de 2023.

TEHRANI, N.; WANG, P. Probabilistic estimation of plug-in electric vehicles charging load profile. **Electric Power Systems Research**. n. 124, p. 133-143, 2015.

UNTORO, A. A., GUNAWAN, T., HIDAYANTO, B. D., PERKSA, F. A., & WIJAYA, A. A. (2021). Study of Economic Viability of Floating Photovoltaic Electric Power in Indonesia. *IOP Conference Series: Materials Science and Engineering*.

WANG, H.; SONG, Q.; ZHANG, L.; WEN, F.; HUANG, J. Load characteristics of electric vehicles in charging and discharging states and impacts on distribution systems. *In: INTERNATIONAL CONFERENCE ON SUSTAINABLE POWER GENERATION AND SUPPLY (SUPERGEN 2012)*, 2012, Hangzhou. **Anais...** Hangzhou, 2012. p. 144-144.

APENDICE A – Roteiro de Simulação

Script em *python* para a simulação, adaptado de Eisenkraemer (2022).

Função para compilar o OpenDSS

```
# Função para compilar o OpenDSS
def compile_dss(dss_file, datapath, casename):
    dss.dss_clear_all()
    dss.text("compile {}".format(dss_file))
    dss.text("Set datapath={}".format(datapath))
    dss.text("Set casename={}".format(casename))
    dss.text("Set Mode=yearly ControlMode=Time time=(0,0) Number=288 stepsize=5m") #aqui foi
    #definido desta maneira pois o Sistema estava com problemas para fazer o modo diário, foi feito
    #um modo anual porém dividido pelos 288 dias do ano.
    dss.text("Set demandinterval=true DIVerbose=true")
    dss.text("Set overloadreport=true voltexcept=true")
    dss.text("Set normvminpu=0.92 normvmaxpu=1.05")
```

Função para rodar as simulações

```
# Função para rodar simulações
def solve_dss():
    dss.solution_solve_all()
    dss.text("closed")
def get_loads_total_power():
    total_power = 0
    dss.loads_first()
    for i in range(dss.loads_count()):
        load_kw = dss.loads_read_kw()
        total_power = total_power + load_kw
    return total_power
```

Função para coletar os dados de tensão

```
# Função para coletar dados de tensão da simulação
def get_voltages(datapath, casename):
    voltage_csv = r"\{}\DI_yr_0\DI_VoltExceptions_1.CSV".format(casename)
    folder = datapath + voltage_csv
    df = pd.read_csv(folder, engine='python')
    vmax = df["Max Voltage"].max()
    vmin = df["Min Voltage"].min()
    return vmax, vmin
```

Função para coletar dados de carregamento

```
# Função para coletar dados de carregamento da simulação
def get_loading(datapath, casename):
    overload_csv = r"\{}\DI_yr_0\DI_Overloads_1.CSV".format(casename)
    folder = datapath + overload_csv
    df = pd.read_csv(folder, engine='python')

    overload = []
    overload_m = []
    transformers = df[df["Element"].str.contains("Transformer")]
    transformers_filter = transformers.groupby("Element", sort=False)["%
Normal"].apply(list).tolist()
    for transformer in transformers_filter:
        overload.append(np.mean(transformer))
        overload_m.append(np.max(transformer))

    overload_transformers = len(transformers_filter)
    overload_mean = np.mean(overload)
    overlaod_max = np.max(overload_m)
    return overload_transformers, overload_mean
```

Função para coletar os dados de perdas

```
# Função para coletar dados de perdas da simulação
def get_losses(datapath, casename):
    losses_csv = r"\{}\\DI_yr_0\SystemMeter_1.CSV".format(casename)
    folder = datapath + losses_csv
    df = pd.read_csv(folder, engine='python')

    energia_kwh = df[' kWh'].max()
    perdas_kwh = df[' "Losses kWh"'].max()
    perdaspercentuais = (perdas_kwh/energia_kwh)*100
    return perdaspercentuais
```

Função para gerar a estação de recarga monofásica

```
# Função para gerar estação de recarga monofásica
def create_monophasic_station(name, bus, kv):
    dss.text("New line.mono_{} phases=1 bus1={} bus2=mono_sec_{} switch=yes".format(name, bus,
    bus))
    dss.text("makebuslist")
    dss.text("setkvBase bus=mono_sec_{} kVLL={}".format(bus, kv))
    dss.text("New Load.station_mono_{} bus1=mono_sec_{} phases=1 conn=wye model=1 kv=0.127
kw=7.4 pf=0.98"
    " yearly=Flat_Curve".format(name, bus))
```

Função para gerar a estação de recarga trifásica

```
# Função para gerar estação de recarga trifásica
def create_triphasic_station(name, bus, kv):
    phase_kv = kv * np.sqrt(3)

    dss.text("New line.tri_{} phases=3 bus1={} bus2=tri_sec_{} switch=yes".format(name, bus,
    bus))
    dss.text("New transformer.tri_{} phases=3 windings=2 buses=(tri_sec_{}, tri_ter_{}
conns=(wye, wye) "
    "kVs=({},0.38) xhl=3.78 %R=1.82 kVAs=(30.0, 30.0)".format(name, bus, bus,
    phase_kv))

    dss.text("makebuslist")
    dss.text("setkvBase bus=tri_sec_{} kVLL={}".format(bus, phase_kv))
    dss.text("setkvBase bus=tri_ter_{} kVLL=0.38".format(bus))

    dss.text("New Load.station_tri_{} bus1=tri_ter_{} phases=3 conn=wye model=1 kv=0.38
kw=22.0 pf=0.98"
    " yearly=Flat_Curve".format(name, bus))
```

Função para aleatorização da curva de cargas de consumidores

```
# função para gerar variações nas curvas de carga dos consumidores
def cdf(media):
    sigma = 0.5

    sorteio = np.random.normal(loc=media, scale=sigma, size=1)
    while sorteio[0] < 0:
        sorteio[0] = np.random.normal(loc=media, scale=sigma, size=1)

    return sorteio[0]
```

Função para criar curva de cargas de consumidores

```
# Função para criar novas variações das curvas de carga dos consumidores
def create_loadshapes(datapath, variations):
    archive = r"\new_loadshapes.dss"
    folder = datapath + archive
```

```

with open("new_loadshapes.dss") as f: # Verifica se o arquivo já existe e se
    contem todas as variações
    firstline = f.readlines()[0].rstrip()
    description = "!Curve_Variations = {}".format(variations)

if os.path.exists(folder) and description in firstline:
    pass
else:
    arquivo = open("new_loadshapes.dss", "w") # cria um dss na pasta onde está o código
    arquivo.truncate(0)

    df_1 = pd.read_csv(r"D:\TCC\Teste_DLL\Teste_DLL\Curva_Flat.csv")
    df_irrad = pd.read_csv(r"D:\TCC\Teste_DLL\Teste_DLL\Irradiacao.csv")
    df_4 = pd.read_csv(r"D:\TCC\Teste_DLL\Teste_DLL\Curva_Estacao_banco.csv")

    loadshape_flat = str(list(df_1['Curva']))[1:-1]
    loadshape_irrad = str(list(df_irrad['Curva']))[1:-1]
    loadshape_banco = str(list(df_4['Curva']))[1:-1]
    hora = str(list(df_1['Hora']))[1:-1]

    arquivo.write(str("!Curve_Variations = {} {}".format(variations, "\n\n")))
    arquivo.write(str("New LoadShape.Flat_Curve Npts=96 Hour=({}) Mult=({})
    {}".format(hora, loadshape_flat, "\n\n")))
    arquivo.write(str("New LoadShape.Station_Bank Npts=96 Hour=({}) Mult=({})
    {}".format(hora, loadshape_banco, "\n\n")))
    arquivo.write(str("New LoadShape.Myirrad Npts=96 Hour=({}) Mult=({}) {}".format(hora,
    loadshape_irrad, "\n\n")))
    arquivo.write(str("New tshape.Mytemp npts=24 interval=1 temp=[25, 25, 25, 25, 25, 25,
    25, 25, 35, 40, 45, 50,"
    " 60, 60, 55, 40, 35, 30, 25, 25, 25, 25, 25]
    {}".format("\n\n")))
    arquivo.write(str("New XYCurve.Myeff npts=4 xarray=[.1 .2 .4 1.0] yarray=[.86 .9
    .93 .97] {}".format("\n\n")))
    arquivo.write(str("New XYCurve.Mypvst npts=4 xarray=[0 25 75 100] yarray=[1.2 1.0
    0.8 0.6] {}".format("\n\n")))

    new_loadshape = list()
    dss.loadshapes_first()

    for p in range(dss.loadshapes_count()):
        k = 0
        while k < variations:
            k = k + 1
            loadshape_name = dss.loadshapes_read_name()
            p_curve = dss.loadshapes_read_p_mult()
            h_curve = dss.loadshapes_read_time_array()

            if "default" not in loadshape_name:
                for i in range(len(p_curve)):
                    new_loadshape.append(cdf(p_curve[i]))
                    arquivo.write(str("New LoadShape.{}_{}_ Npts=288 mult=({}) hour=({}) {}".
                    .format(str(loadshape_name), str(k),
                    str(new_loadshape)[1:-1], str(h_curve)[1:-1], "\n\n")))

            new_loadshape = []

            dss.loadshapes_next()

    arquivo.close()

```

Função para sortear as curvas criadas para cada consumidor

```

# Função para sortear novas curvas de carga para os consumidores
def sample_loadshapes(numcurv):
    dss.text("Redirect new_loadshapes.dss")
    dss.loads_first()
    for i in range(dss.loads_count()):
        sorteio = random.randint(1, numcurv)
        nome = dss.loads_read_name()
        loadshape = dss.loads_read_daily()
        if "station" not in nome:
            dss.text("Load." + nome + ".yearly=" + loadshape + "_" + str(sorteio))

```

```
dss.loads_next()
```

Função para lista de barras MT trifásicas

```
# Função para elecionar lista de barras de média tensão trifásicas
def make_3phase_buslist():
    buses = dss.circuit_all_bus_names()
    my_buses = list()
    my_bus_voltage_dict = dict()

    for bus in buses:
        dss.circuit_set_active_bus(bus)
        if bus == "sourcebus":
            pass
        elif len(dss.bus_nodes()) == 3 and dss.bus_kv_base() >= 1.0:
            my_buses.append(bus)
            my_bus_voltage_dict[bus] = dss.bus_kv_base()

    return my_buses, my_bus_voltage_dict
```

Função para lista de barras BT monofásicas

```
# Função para selecionar lista de barras de baixa tensão
def make_1phase_buslist():
    buses = dss.circuit_all_bus_names()
    my_buses = list()
    my_bus_voltage_dict = dict()

    for bus in buses:
        dss.circuit_set_active_bus(bus)
        if bus == "sourcebus":
            pass
        elif dss.bus_kv_base() <= 1.0:
            my_buses.append(bus)
            my_bus_voltage_dict[bus] = dss.bus_kv_base()

    return my_buses, my_bus_voltage_dict
```

Função para selecionar as barras de acordo com o cenário

```
# Função para sortear a barra das estações de acordo com o cenário
def sample_stations(cenario, penetration_level):
    bus_list_3phase, my_bus_voltage_dict_3phase = make_3phase_buslist() # Cria lista de
barras trifásicas para sorteio
    bus_list_1phase, my_bus_voltage_dict_1phase = make_1phase_buslist() # Cria lista de
barras monofásicas para sorteio
    total_consumers = dss.loads_count()

    if cenario == 'base_case':
        pass

    elif cenario == 'monophasic_station':
        kv = my_bus_voltage_dict_1phase
        for i in range(int(round(total_consumers * penetration_level))):
            bus = random.sample(bus_list_1phase, 1)
            create_monophasic_station(str(i), bus[0], kv[bus[0]])

    elif cenario == 'triphasic_station':
        kv = my_bus_voltage_dict_3phase
        for i in range(int(round(total_consumers * (penetration_level/4)))):
            bus = random.sample(bus_list_3phase, 1)
            create_triphasic_station(str(i), bus[0], kv[bus[0]])

    else:
        kv_mono = my_bus_voltage_dict_1phase
        kv_tri = my_bus_voltage_dict_3phase
        for i in range(int((round(total_consumers * penetration_level * 0.85)))):
```

```

bus = random.sample(bus_list_1phase, 1)
create_monophasic_station(str(i), bus[0], kv_mono[bus[0]])

for i in range(int(round(total_consumers * penetration_level * 0.1))):
    bus = random.sample(bus_list_3phase, 1)
    create_triphasic_station(str(i), bus[0], kv_tri[bus[0]])

```

Função para plotar as imagens em forma de bloxplot

```

# Função para plotar boxplot das perdas
def plot_boxplot(data, cenario, variable):

    if cenario == 'monophasic_station' or cenario == 'stations_mix':
        x_data = ['20%', '40%', '60%', '80%', '100%']
    else:
        x_data = ['5.0%', '10%', '15%', '20%', '25%']
    y_data = data

    colors = ['rgba(213, 228, 207, 0.8)', 'rgba(110, 181, 205, 0.8)',
              'rgba(42, 132, 107, 0.8)', 'rgba(18, 107, 105, 0.8)', 'rgba(2, 65, 83, 0.8)']
    marker_colors = ['rgb(0,51,0)', 'rgb(0,51,0)', 'rgb(0,51,0)', 'rgb(0,51,0)',
                    'rgb(0,51,0)']

    fig = plt.Figure()

    for xd, yd, cls, marker in zip(x_data, y_data, colors, marker_colors):
        fig.add_trace(plt.Box(
            y=yd, name=xd,
            boxpoints='all',
            marker_color=marker, line_color=marker,
            jitter=0.5, whiskerwidth=0.2,
            fillcolor=cls,
            marker_size=3, line_width=1))

    if variable == 'losses':
        fig.update_layout(
            title='Perdas no sistema de distribuição no cenário ' + cenario,
            yaxis_title='Perdas percentuais (%)', xaxis_title='Níveis de Penetração de VEs',
            width=1500, height=750,
            yaxis=dict(
                autorange=True, showgrid=True, zeroline=False,
                gridcolor='rgb(208, 206, 206)', gridwidth=1,
                zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2),
            margin=dict(l=40, r=30, b=80, t=100),
            paper_bgcolor='rgb(255, 255, 255)', plot_bgcolor='rgb(255, 255, 255)',
            showlegend=False)

        if cenario == 'monophasic_station':
            fig.update_yaxes(
                autorange=True, showgrid=True, zeroline=False,
                dtick=0.1,
                gridcolor='rgb(208, 206, 206)', gridwidth=1,
                zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

        if cenario == 'triphasic_station':
            fig.update_yaxes(
                autorange=True, showgrid=True, zeroline=False,
                dtick=0.5,
                gridcolor='rgb(208, 206, 206)', gridwidth=1,
                zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

        if cenario == 'stations_mix':
            fig.update_yaxes(
                autorange=True, showgrid=True, zeroline=False,
                dtick=0.5,
                gridcolor='rgb(208, 206, 206)', gridwidth=1,
                zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

    elif variable == 'loading':
        fig.update_layout(
            title='Nível de carregamento dos Trafos em Sobrecarga no cenário ' + cenario,
            yaxis_title='Porcentagem de da carga nominal (%)', xaxis_title='Níveis de
            Penetração de VEs',
            width=1000, height=750,
            yaxis=dict(autorange=True, showgrid=True, zeroline=False,
                gridcolor='rgb(208, 206, 206)', gridwidth=1,

```

```

        zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2, ),
    margin=dict(l=40, r=30, b=80, t=100),
    paper_bgcolor='rgb(255, 255, 255)', plot_bgcolor='rgb(255, 255, 255)',
    showlegend=True)

if cenario == 'monophasic_station':
    fig.update_yaxes(
        autorange=True, showgrid=True, zeroline=False,
        dtick=2,
        gridcolor='rgb(208, 206, 206)', gridwidth=1,
        zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

if cenario == 'triphasic_station':
    fig.update_yaxes(
        autorange=True, showgrid=True, zeroline=False,
        dtick=2,
        gridcolor='rgb(208, 206, 206)', gridwidth=1,
        zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

if cenario == 'stations_mix':
    fig.update_yaxes(
        autorange=True, showgrid=True, zeroline=False,
        dtick=2,
        gridcolor='rgb(208, 206, 206)', gridwidth=1,
        zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

fig.write_image(r"D:\TCC\Teste_DLL\Teste_DLL\images\fig_" + cenario + "_" + variable +
".pdf")

# Função para plotar boxplot das tensões e dados de carregamento
def plot_double_boxplot(data1, data2, cenario, variable):

    if cenario == 'monophasic_station' or cenario == 'stations_mix':
        x_data = ['20%', '40%', '60%', '80%', '100%']
    else:
        x_data = ['5.0%', '10%', '15%', '20%', '25%']

    y_data1 = data1
    y_data2 = data2

    colors = ['rgba(213, 228, 207, 0.8)', 'rgba(110, 181, 205, 0.8)',
              'rgba(42, 132, 107, 0.8)', 'rgba(18, 107, 105, 0.8)', 'rgba(2, 65, 83, 0.8)']
    marker_colors = ['rgb(0,51,0)', 'rgb(0,51,0)', 'rgb(0,51,0)', 'rgb(0,51,0)',
'rgb(0,51,0)']

    fig = make_subplots(rows=2, cols=1)

    for xd, yd, cls, marker in zip(x_data, y_data1, colors, marker_colors):
        fig.add_trace(plt.Box(
            y=yd, name=xd, boxpoints='all',
            marker_color=marker, line_color=marker,
            jitter=0.5, whiskerwidth=0.2,
            fillcolor=cls, marker_size=3, line_width=1, ),
            row=1, col=1)

    for xd, yd, cls, marker in zip(x_data, y_data2, colors, marker_colors):
        fig.add_trace(plt.Box(
            y=yd, name=xd, boxpoints='all',
            marker_color=marker, line_color=marker,
            jitter=0.5, whiskerwidth=0.2,
            fillcolor=cls, marker_size=3, line_width=1),
            row=2, col=1)

    fig.update_layout(
        title='Limites de tensão observados no SD em (p.u.) no cenário ' + cenario,
        width=1000, height=750,
        yaxis=dict(autorange=True, showgrid=True, zeroline=False,
            gridcolor='rgb(208, 206, 206)', gridwidth=1,
            zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2, ),
        margin=dict(l=40, r=30, b=80, t=100),
        paper_bgcolor='rgb(255, 255, 255)', plot_bgcolor='rgb(255, 255, 255)',
        showlegend=False)

    if cenario == 'monophasic_station':
        fig.update_yaxes(
            title_text="Tensão Mínima",

```

```

        row=2, col=1,
        autorange=True, showgrid=True, zeroline=False,
        dtick=0.005,
        gridcolor='rgb(208, 206, 206)', gridwidth=1,
        zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

    fig.update_xaxes(title='Níveis de Penetração de VEs', row=1, col=1)

    fig.add_hline(y=1.05, line_dash="dash", name='teste',
                  line=dict(color='black', width=2),
                  annotation_text="Limite Superior de Operação",
                  row=1, col=1)

    fig.update_yaxes(
        title_text="Tensão Máxima",
        row=1, col=1,
        autorange=True, showgrid=True, zeroline=False,
        dtick=0.002,
        gridcolor='rgb(208, 206, 206)', gridwidth=1,
        zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

    fig.update_xaxes(title='Níveis de Penetração de VEs', row=2, col=1)

    fig.add_hline(y=0.92, line_dash="dash", name='teste',
                  line=dict(color='black', width=2),
                  annotation_text="Limite Inferior de Operação",
                  row=2, col=1)

elif cenario == 'triphasic_station':
    fig.update_yaxes(
        title_text="Tensão Mínima",
        row=2, col=1,
        autorange=True, showgrid=True, zeroline=False,
        dtick=0.01,
        gridcolor='rgb(208, 206, 206)', gridwidth=1,
        zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

    fig.update_xaxes(title='Níveis de Penetração de VEs', row=1, col=1)

    fig.add_hline(y=1.05, line_dash="dash", name='teste',
                  line=dict(color='black', width=2),
                  annotation_text="Limite Superior de Operação",
                  row=1, col=1)

    fig.update_yaxes(
        title_text="Tensão Máxima",
        row=1, col=1,
        autorange=True, showgrid=True, zeroline=False,
        dtick=0.005,
        gridcolor='rgb(208, 206, 206)', gridwidth=1,
        zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

    fig.update_xaxes(title='Níveis de Penetração de VEs', row=2, col=1)

    fig.add_hline(y=0.92, line_dash="dash", name='teste',
                  line=dict(color='black', width=2),
                  annotation_text="Limite Inferior de Operação",
                  row=2, col=1)

elif cenario == 'stations_mix':
    fig.update_yaxes(
        title_text="Tensão Mínima",
        row=2, col=1,
        autorange=True, showgrid=True, zeroline=False,
        dtick=0.01,
        gridcolor='rgb(208, 206, 206)', gridwidth=1,
        zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

    fig.update_xaxes(title='Níveis de Penetração de VEs', row=1, col=1)

    fig.add_hline(y=1.05, line_dash="dash", name='teste',
                  line=dict(color='black', width=2),
                  annotation_text="Limite Superior de Operação",
                  row=1, col=1)

    fig.update_yaxes(
        title_text="Tensão Máxima",
        row=1, col=1,

```

```

        autorange=True, showgrid=True, zeroline=False,
        dtick=0.005,
        gridcolor='rgb(208, 206, 206)', gridwidth=1,
        zerolinecolor='rgb(0, 0, 0)', zerolinewidth=2)

fig.update_xaxes(title='Níveis de Penetração de VEs', row=2, col=1)

fig.add_hline(y=0.92, line_dash="dash", name='teste',
              line=dict(color='black', width=2),
              annotation_text="Limite Inferior de Operação",
              row=2, col=1)

fig.write_image(r"D:\TCC\Teste_DLL\Teste_DLL\images\fig_" + cenario + "_" + variable +
               ".pdf")

```

Após definidas essas funções, o código principal fica mais facilmente visualizado assim.

```

Master_dss = r"D:\TCC\DAVI\Teste_DLL\Rede CPFL\Master.dss" # Arquivo Master.dss
datapath = r"D:\TCC\DAVI\Teste_DLL\Rede_CPFL" # Diretório dos resultados
-----

# Definição dos parâmetros da simulacao

loadshape_variations = 30 # Número de variações de
loadshapes desejado
max_iterations = 150 # Seleciona o numero de
iterações por cenário
random.seed(11) # Número inicial do
gerador de números aleatórios
np.random.seed(11) # Número inicial do
gerador de números aleatórios
cenarios = ['base_case', 'monophasic_station', 'triphasic_station', 'station_mix' # Cenários
de sorteios para o MMC 'monophasic_station', 'triphasic_station' 'stations_mix' , 'base_case'
]
fields = ['losses', 'vmax', 'vmin', 'overload_transformers', 'overload_mean']

# Início do processo de simulação

clear_folder(datapath) # Limpa o diretorio
compile_dss(Master_dss, datapath, 'base_case') # Compila o sistema
create_loadshapes(datapath, loadshape_variations) # Cria arquivo com
variações das curvas de carga
Results = [] # Vetor para coleta de
resultados

for i in range(len(cenarios)):
    cenarios_list = []
    for penetration_level in np.arange(0, 1.2, 0.2):
        level_list = []
        if cenarios[i] == 'base_case' and penetration_level > 0:
            break
        elif cenarios[i] != 'base_case' and penetration_level == 0:
            continue
        else:
            for simulation in range(max_iterations):
                compile_dss(Master_dss, datapath, cenarios[i])
                sample_loadshapes(loadshape_variations)
                sample_stations(cenarios[i], penetration_level)
                solve_dss()
                losses = get_losses(datapath, cenarios[i])
                vmax, vmin = get_voltages(datapath, cenarios[i])
                overload_transformers, overload_mean = get_loading(datapath, cenarios[i])
                level_list.append(Rede(losses, vmax, vmin, overload_transformers,
overload_mean, cenarios[i], penetration_level))

            cenarios_list.append(level_list)
    Results.append(cenarios_list)

# Análise dos dados para geração dos gráficos

```

```

losses = []
vmax = []
vmin = []
overload_transformers = []
overload_means = []

for i in range(len(cenarios)):
    cenario = Results[i]
    if cenarios[i] == 'base_case':
        for level in cenario:
            for simulation in level:
                losses.append(simulation.losses)
                vmax.append(simulation.vmax)
                vmin.append(simulation.vmin)
                overload_transformers.append(simulation.overload_transformers)
                overload_means.append(simulation.overload_mean)

                with open(cenarios[i] + '_Results.csv', 'w', newline='') as f:
                    csv_writer = csv.writer(f)
                    for line in range(len(losses)):
                        if line == 0:
                            csv_writer.writerow(fields)
                            csv_writer.writerow([losses[line], vmax[line], vmin[line],
overload_transformers[line],
                                                overload_means[line]])

                        f.close()

                print('Média das perdas no caso base = ', np.mean(losses))
                print('Média da tensão máxima no caso base = ', np.mean(vmax))
                print('Média da tensão mínima no caso base = ', np.mean(vmin))
                print('Média de transformadores sobrecarregados no caso base = ',
np.mean(overload_transformers))
                print('Média de carregamento dos transformadores sobrecarregados no caso base = ',
np.mean(overload_means))

            else:
                level_list_losses = []
                level_list_vmax = []
                level_list_vmin = []
                level_list_overload_transformers = []
                level_list_overload_means = []

                with open(cenarios[i] + ' Results.csv', 'w', newline='') as f:
                    csv_writer = csv.writer(f)

                    for level in cenario:
                        losses = []
                        vmax = []
                        vmin = []
                        overload_transformers = []
                        overload_means = []

                        for simulation in level:
                            losses.append(simulation.losses)
                            vmax.append(simulation.vmax)
                            vmin.append(simulation.vmin)
                            overload_transformers.append(simulation.overload_transformers)
                            overload_means.append(simulation.overload_mean)

                            if cenario.index(level) == 0 and level.index(simulation) == 0:
                                csv_writer.writerow(fields)
                                csv_writer.writerow([simulation.losses, simulation.vmax, simulation.vmin,
simulation.overload_transformers,
                                                    simulation.overload_mean])

                                level_list_losses.append(losses)
                                level_list_vmax.append(vmax)
                                level_list_vmin.append(vmin)
                                level_list_overload_transformers.append(overload_transformers)
                                level_list_overload_means.append(overload_means)

                        f.close()
                        t = 0
                        print('Média de Transformadores sobrecarregados no cenário ' + cenarios[i] + ':')
                        for transformer in level_list_overload_transformers:
                            print('No nível de penetração ' + str(t) + ': ', round(int(np.mean(transformer))))

```

```
t = t + 1  
  
plot_boxplot(level_list_losses, cenarios[i], 'losses')  
plot_boxplot(level_list_overload_means, cenarios[i], 'loading')  
plot_double_boxplot(level_list_vmax, level_list_vmin, cenarios[i], 'voltage')
```

NUP: 23081.146080/2023-96

Prioridade: Normal

Homologação de ata de defesa de TCC e estágio de graduação

125.322 - Bancas examinadoras de TCC: indicação e atuação

COMPONENTE

Ordem	Descrição	Nome do arquivo
5	Trabalho de conclusão de curso (TCC) (125.32)	TCC.pdf

Assinaturas

16/11/2023 21:19:44

DANIEL PINHEIRO BERNARDON (PROFESSOR DO MAGISTÉRIO SUPERIOR (Ativo))
01.28.00.00.0.0 - PRÓ-REITORIA DE INOVAÇÃO E EMPREENDEDORISMO - PROINOVA

16/11/2023 23:45:20

JULIANO SILVA DAMIANI (Aluno de Pós-Graduação - Aluno Regular)
07.10.01.01.0.0 - PG Engenharia Elétrica - Mestrado - 42002010003M8

22/11/2023 14:39:48

MARINA CAMPONOOGARA (Aluno de Pós-Graduação - Aluno Regular)
07.10.01.02.0.0 - PG Engenharia Elétrica - Doutorado - 42002010003D9

23/11/2023 13:25:55

DIEGO BERLEZI RAMOS (Coordenador(a) de Curso)
07.09.02.00.0.0 - CURSO DE ENGENHARIA ELÉTRICA - CEELE



Código Verificador: 3535077

Código CRC: f62c7f41

Consulte em: <https://portal.ufsm.br/documentos/publico/autenticacao/assinaturas.html>

