

Classificação de faltas em linhas de transmissão utilizando métodos de aprendizado de máquina

Gabriel Aparecido Fonseca* Danton Diego Ferreira**
Flávio Bezerra Costa*** Aryfrance Rocha Almeida****
Robson Rosserrani de Lima†

* Departamento de Engenharia – Universidade Federal Lavras, MG,
(e-mail: gabrielfonseca91@hotmail.com)

** Departamento de Engenharia – Universidade Federal Lavras, MG,
(e-mail: danton@ufla.br).

*** Escola de Ciências e Tecnologia – Universidade Federal do Rio
Grande do Norte, RN, (e-mail: flaviocosta@ect.ufrn.br)

**** Departamento de Engenharia Elétrica – Universidade Federal do
Piauí, PI, (e-mail: aryfrance@ufpi.edu.br)

† Faculdade de Engenharia - Universidade Federal de Juiz de Fora,
MG, (e-mail: rosserrani@gmail.com)

Abstract: Power transmission lines are components highly susceptible to faults. Several works have already explored the use of computational intelligence, signal processing and other techniques in the construction of protective methods for quick verification and action in the occurrence of transmission line fault. Many of these works focus on approaches using signal processing such as Fourier or wavelet transforms. With the advance of machine learning, some techniques began to be used in this area with success. This work focuses on the offline classification of ten (AG, BG, CG, AB, AC, BC, ABG, ACG, BCG and ABC) types of faults that arise when a short circuit occurs in the transmission line, investigating the use of classical techniques such as notch filter and random forests. For comparative purposes, recently created techniques, called random convolutional kernel transform (Rocket) and MiniRocket, were used to extract features in time series and good results were obtained in the identification of faults that occurred in the transmission line. As a result of this paper, accuracies greater than 93% were obtained considering up to 1/16 cycle post-fault. For signals with 1 and 1/2 cycle post-fault, accuracies higher than 97% were obtained.

Resumo: As linhas de transmissão de energia elétrica são componentes altamente susceptíveis a faltas. Diversos trabalhos já exploraram o uso de inteligência computacional, processamento de sinais e outras técnicas na construção de métodos protetivos para rápida verificação e atuação na ocorrência de faltas nas linhas de transmissão. Muitos desses trabalhos focam em abordagens utilizando processamento de sinais como transformada de Fourier ou *wavelet*. Com o avanço do aprendizado de máquina, algumas técnicas passaram a ser utilizadas nessa área obtendo êxito. Esse artigo foca na classificação *offline* de dez (AT, BT, CT, AB, AC, BC, ABT, ACT, BCT e ABC) tipos de faltas que surgem quando ocorre um curto-circuito na linha de transmissão, investigando o uso de técnicas clássicas como filtro *notch* e florestas aleatórias. Para efeito comparativo, foram utilizadas técnicas criadas recentemente, chamadas de *random convolutional kernel transform* (Rocket) e MiniRocket, para a extração de características em séries temporais e bons resultados foram obtidos na identificação das faltas ocorridas na linha de transmissão. Como resultado desse artigo, acurácias superiores a 93% foram alcançadas considerando-se até 1/16 ciclo pós falta. Para sinais com 1 e 1/2 ciclo pós falta foram obtidas acurácias superiores a 97%.

Keywords: Rocket; MiniRocket; fault classification; transmission lines; notch filter; random forests.

Palavras-chaves: Rocket; MiniRocket; classificação de faltas; linhas de transmissão; filtro notch; florestas aleatórias.

1. INTRODUÇÃO

Durante a operação do sistema elétrico em tempo real diversas perturbações podem ocorrer. Muitas delas sequer são percebidas pelo consumidor já que não provocam medidas severas como a interrupção do fornecimento de energia. Inúmeras causas podem levar a tais interrupções: defeitos de equipamentos, eventos meteorológicos (tempestades, ventos, chuvas etc), incêndios, falhas nos sistemas protetivos, erros humanos e operacionais, ação de animais, dentre diversas outras causas. Um dos componentes mais susceptíveis a faltas é a linha de transmissão. Dado que ela, em geral, está exposta ao ambiente externo os fatores anteriormente citados podem interferir no seu perfeito funcionamento fazendo que uma falta ocorra e o fornecimento de energia seja interrompido (de Souza, 2018).

Tendo em vista o que foi citado, existe uma clara necessidade da criação e implementação de sistemas protetivos eficientes para que danos graves sejam evitados ou minimizados. Tais sistemas são basicamente divididos em três etapas. A primeira delas é a detecção da falta que corresponde à identificação da ocorrência da falta e consequente amostragem do sinal. Em seguida é realizada a classificação da falta cuja função é identificar que tipo de falta ocorreu, ou seja, quais fases estão presentes na falta e se existe ou não presença do terra. Por fim, é executada a localização da falta que baseia-se em estimar a distância entre o ponto de ocorrência da falta e o local do relé (De Carvalho et al., 2014).

Existem dois grupos de faltas que podem ocorrer nas linhas de transmissão: as de circuito aberto (ou faltas do tipo série) e as de curto-circuito (também chamadas faltas *shunt*). As faltas de circuito aberto podem ser identificadas observando a tensão em cada fase, um aumento nos valores de tensão pode ser um indicativo de que ela ocorreu, apesar desse comportamento ocorrer em outros eventos elétricos. Elas podem ser classificadas em dois tipos: um circuito aberto e dois circuitos abertos. Essas faltas ocorrem raramente e são pouco abordadas na literatura. Por sua vez, as faltas de curto circuito podem ser identificadas de duas formas: pela diminuição no nível de tensão na fase ou elevação dos valores de corrente. Elas podem ser divididas em dois tipos: assimétricas e simétricas (Prasad et al., 2017b).

As faltas mais comuns são aquelas que envolvem apenas uma fase e o terra (AT, BT, CT). Faltas bifásicas (AB, BC e CA) são um pouco mais raras e severas. Já as faltas que envolvem duas fases e o terra (ABT, BCT e CAT) são ainda menos comuns e mais nocivas. Por fim, faltas trifásicas (ABC) e trifásicas-terra (ABCT) quando ocorrem podem levar o sistema inteiro ao colapso, mas também possuem uma frequência de ocorrência menor (Prasad et al., 2017a). A partir de observações práticas verificou-se que entre 70% e 80% das faltas em linha de transmissão envolvem apenas uma fase (Stevenson, 1978).

Por sua vez, faltas nas três fases ocorrem em apenas 5% dos casos.

Ao realizar uma pesquisa sobre trabalhos desenvolvidos na área, verificou-se que diversos pesquisadores já propuseram a utilização de técnicas variadas em sistemas de proteção de faltas em linhas de transmissão. Pelos trabalhos analisados, na área de classificação de faltas são utilizados em sua maioria técnicas de processamento de sinais, que podem ser utilizadas isoladamente, ou em conjunto com métodos de reconhecimento de padrões e inteligência computacional. Dentre essas técnicas é possível citar estatísticas de ordem superior (De Carvalho et al., 2014), análise de componentes independentes (Almeida et al., 2017), redes neurais (Saravanan and Rathinam, 2012; Kumar et al., 2014; Maheshwari et al., 2019), máquina de aprendizado extremo de soma *wavelet* (*summation-wavelet extreme learning machine*) (Chen et al., 2018), etc.

Em geral, as técnicas mais utilizadas para pré-processamento de sinais são a transformada de Fourier e *wavelet*. Enquanto a primeira fornece informações sobre todas as frequências presentes no sinal ela não fornece nenhuma indicação de quando esses sinais estavam presentes no tempo. Por sua vez, a transformada *wavelet* fornece uma análise multi-resolução na frequência e no tempo (Das et al., 2005). No entanto, sabe-se que os seus parâmetros são muito sensíveis ao ruído, que é gerado devido a muitos fenômenos relacionados à linha de transmissão, isto é, campos eletromagnéticos, correntes de fuga em isoladores e condições ambientais (Almeida et al., 2017).

Dessa forma, é possível notar que alguns dos métodos apresentados nessa seção necessitam que se conheça bem os sinais utilizados. Como, por exemplo, como foram gerados, suas frequências, se possuem ou não ruído, suas estatísticas, distribuições e características, para que essas técnicas sejam melhor aproveitadas.

O objetivo geral deste trabalho foi realizar um estudo comparativo utilizando técnicas clássicas e modernas no processamento de sinais para a classificação de faltas em linhas de transmissão. Dessa forma, utilizou-se filtro *notch* e *random forest* que são amplamente utilizados na área de processamento de sinais e classificação. Além disso, *random convolutional kernel transform Rocket* (Dempster et al., 2020) e *MiniRocket* (Dempster et al., 2021) que são técnicas recentes de extração de características de séries temporais foram utilizadas junto com um classificador de regressão *ridge* (Scikit-learn, 2021) para efeito comparativo. Os objetivos específicos estão listados a seguir:

- analisar e processar os sinais disponíveis utilizando filtro *notch*, *MiniRocket* e *Rocket*;
- verificar o impacto da informação do sinal pós falta na classificação das faltas;
- investigar se ocorre diferença no resultado final a partir da variação das técnicas utilizadas no pré-processamento;

d) avaliar o desempenho da *random forest* e regressão *ridge* na identificação de faltas em linhas de transmissão.

Para facilitar a leitura e o entendimento do texto esse artigo foi subdividido em seções. Na sessão 2 são apresentados os materiais, métodos e metodologias utilizados na realização desse projeto. Na sessão 3 são apresentados os resultados alcançados na realização desse trabalho. Por fim, as conclusões obtidas a partir da análise dos resultados são apresentadas na seção 4.

2. MATERIAIS E MÉTODOS

Nessa seção serão apresentados os materiais e métodos utilizados no desenvolvimento de projeto. Ao final serão apresentadas as abordagens utilizadas para efeito comparativo e serão posteriormente apresentados os resultados obtidos e quais abordagens tiveram desempenho mais satisfatório.

2.1 Random Forest

Random Forest, ou florestas aleatórias, é um modelo de aprendizado de máquina amplamente utilizado em inúmeras aplicações em diversas áreas distintas. Esse método é baseado na combinação de árvores de decisão de tal forma que cada árvore depende dos valores de um vetor aleatório amostrado independentemente e com a mesma distribuição de todas as árvores na floresta (Breiman, 2001).

Random forest pode ser usada tanto em problemas de regressão quanto classificação e quando comparado com a utilização de árvores de decisão as florestas aleatórias são menos susceptíveis a sofrer *overfitting*. No entanto, dependendo da quantidade de árvores utilizadas e de suas profundidades, pode ser demorado proceder com o treinamento utilizando esse método.

2.2 Rocket

Com o sucesso do uso das redes neurais convolucionais em séries temporais, alguns métodos derivados surgiram, no entanto, a maioria possui alta complexidade computacional e pode se tornar inviável dependendo da aplicação. A partir da evolução desses métodos Dempster et al. (2020) apresentaram um novo modelo chamado *Rocket* (acrônimo de *RandOm Convolutional Kernel Transform*), que consegue obter resultados do estado da arte em acurácia na classificação de séries temporais. Este método transforma as séries temporais de entrada em um conjunto de *features* usando *kernels* (filtros) convolucionais aleatórios, após isso é possível realizar a classificação utilizando qualquer classificador linear simples.

Apesar de outros métodos baseados em *kernels* convolucionais terem sido utilizados eficientemente na classificação de séries temporais, geralmente tais modelos possuem um alto custo computacional devido o treinamento da rede. Como *Rocket* é baseado na utilização de vários *kernels* convolucionais aleatórios ele é capaz de obter acurácia de estado da arte mas com uma fração do tempo quando comparado com os métodos atualmente usados (Dempster et al., 2020).

O único hiperparâmetro a ser ajustado no *Rocket* é o número de *kernels*, o que conduz a um *tradeoff* entre a

acurácia e o tempo computacional. Em geral, um número alto de *kernels* resulta em uma alta acurácia de classificação, mas à custa de uma computação proporcionalmente mais longa. Os criadores desse método indicaram o uso de 10000 *kernels* por padrão, e apesar de ser um número alto, *Rocket* mostrou ser extremamente rápido.

Os parâmetros dos *kernels* são obtidos da seguinte maneira:

- **Comprimento:** selecionado aleatoriamente do conjunto $\{7, 9, 11\}$ com probabilidade igual;
- **Pesos:** amostrados de uma distribuição normal, $\forall k \in \mathbf{K}$, $k \sim \mathcal{N}(0, 1)$ e são centralizados na média depois de ajustados, $k = \mathbf{K} - \bar{\mathbf{K}}$;
- **Bias:** amostrados de uma distribuição uniforme $b \sim \mathcal{U}(-1, 1)$, somente valores positivos no mapa de características são usados;
- **Dilatação:** é amostrada em uma escala exponencial $d = \lceil 2^x \rceil$, $x \sim \mathcal{U}(0, A)$ onde $A = \log_2 \frac{l_{input}-1}{l_{kernel}-1}$, que assegura o comprimento efetivo do *kernel*, incluindo a dilatação, até o comprimento da série temporal de entrada l_{input} ;
- **Padding:** quando cada *kernel* é gerado, o modelo decide (aleatoriamente) se *padding* será usado ou não. Em modelos de *deep learning* o termo *padding* basicamente indica linhas e colunas extras anexadas na borda de uma matriz. Caso seja usado, o valor de zero *padding* é anexado no início e no final de cada série temporal quando o *kernel* é aplicado. Dessa forma, o elemento central do *kernel* é centralizado em cada ponto da série temporal, ou seja, $((l_{kernel} - 1) \times 2) / 2$. Sem a utilização do *padding* os *kernels* não são centralizados no primeiro e último pontos da série temporal. O que significa que o “foco” se dará em padrões da região central dando menos importância para as regiões no início e final da série temporal.

Outro parâmetro importante desse método é o *stride*. Esse basicamente é um termo usado em *deep learning* para indicar o quanto o *kernel* se move de uma posição para a próxima. No caso, para o método *Rocket*, o *stride* é sempre um, indicando que o *kernel* move sempre uma posição.

Além de todos os parâmetros citados, diferentemente de como ocorre nas redes neurais convolucionais, no método *Rocket* não são aplicadas não linearidades aos *feature maps*. Importante observar que os parâmetros para pesos e biases são ajustados baseados na suposição que, como uma prática padrão, séries temporais de entrada foram normalizadas para ter média zero e desvio padrão igual a um.

Em resumo cada *kernel* é aplicado a cada entrada, produzindo um *feature map*. A operação de convolução envolve um produto interno entre um *kernel* e uma série temporal de entrada. Esse processo é descrito pela equação 1.

$$X_i * k = \sum_{j=0}^{l_{kernel}-1} X_{i+(j \times d)} \times k_j \quad (1)$$

onde k representa o *kernel*, d a dilatação, X a série temporal, i a posição na série.

A equação 1 gera vários mapas de características na qual são computadas 2 *features* para cada mapa, produzindo dois números com valor real como recursos por *kernel*: o valor máximo e a proporção de valores positivos (ppv). O ppv produz maior acurácia de classificação do que outras *features* como a média, por exemplo. Para k *kernels*, *Rocket* produz $2k$ *features* por série temporal.

Apesar do *Rocket* poder ser usado com qualquer classificador, os autores encontraram que ele é muito efetivo quando usado com classificadores lineares. Por esse motivo, em Dempster et al. (2020) é proposto o uso da regressão logística e regressão *ridge*.

2.3 MiniRocket

MiniRocket (MINImally RandOm Convolutional KERnel Transform) é uma reformulação do método *Rocket* 75 vezes mais rápido em grandes conjuntos de dados sendo quase totalmente determinístico, enquanto mantém essencialmente a mesma acurácia. Para isso, *MiniRocket* usa um conjunto pequeno e fixo de *kernels*, ele mantém os dois mais importantes aspectos do *Rocket*: dilatação e proporção de valores positivos (ppv). E, além disso, explora várias propriedades dos *kernels* e da ppv, para efetivamente reduzir o tempo necessário para a transformação. Enquanto *kernels* convolucionais aleatórios são altamente efetivos, é possível essencialmente obter a mesma acurácia usando um procedimento determinístico e muito mais rápido (Dempster et al., 2021).

MiniRocket faz mudanças cruciais para remover quase toda aleatoriedade do *Rocket*, e explora essas modificações para aumentar a velocidade da transformação. A Tabela 1 resume as principais diferenças entre os dois métodos.

Tabela 1. Comparativo entre métodos *Rocket* e *MiniRocket*

	<i>Rocket</i>	<i>MiniRocket</i>
comprimento	{7, 9, 11}	9
pesos	$\mathcal{N}(0, 1)$	{-1, 2}
biases	$\mathcal{U}(-1, 1)$	da saída da convolução
dilatação	aleatório	fixo
<i>padding</i>	aleatório	fixo
<i>features</i>	ppv + max	ppv
num. <i>features</i>	20.000	10.000

Visando maximizar a eficiência, o conjunto de *kernels* deve ser o menor possível. Em relação ao comprimento, no *MiniRocket* existem $2^9 = 512$ possíveis *kernels* (dois valores possíveis de peso e o comprimento de 9, como mostrado na Tabela 1), no entanto, os autores (Dempster et al., 2021) sugerem utilizar um subconjunto de 84 desses *kernels* de tal forma que é balanceada a acurácia com a baixa complexidade computacional.

Os pesos usados nos *kernels* foram definidos de forma arbitrária para serem -1 ou 2, a escala não é importante já que o bias é extraído da saída da convolução e assim, por definição, coincide com a escala dos pesos e a escala da entrada. No entanto, é importante que a soma dos pesos seja zero, isso garante que a saída da convolução seja invariante à adição ou subtração de qualquer constante na entrada.

Como apresentado por Dempster et al. (2021), o conjunto de 84 *kernels* de tamanho 9, com três pesos com valor 2

e seis com o valor -1 tem as propriedades desejadas de ser relativamente pequeno e produzir alta acurácia. No entanto, subconjuntos de *kernels* de tamanho 9, ou outros tamanhos fornecerem uma acurácia similar. Os autores, a partir de seus testes, apenas decidiram utilizar por padrão esse subconjunto de 84 *kernels*.

Os valores de bias, como citado, são extraídos da saída da convolução e são usados para computar a ppv, conforme a equação 2.

$$ppv(X * k) = \frac{1}{n} \sum [X * k > b] \quad (2)$$

onde, X representa a série temporal, k o *kernel*, e b o *bias*.

A transformação da série temporal usando *MiniRocket* é otimizada devido alguns fatores. Dentre eles, cabe citar que a saída da operação de convolução é reusada para calcular múltiplas *features*, por exemplo para vários valores de *bias* diferentes. Isso reduz o custo computacional de computar a ppv para cada valor de *bias*.

Outro fator que otimiza a transformação é a restrição dos pesos dos *kernels* a dois valores apenas. Isso permite com que a operação de convolução seja executada usando apenas adição. Para uma série temporal $X = [x_0, x_1, \dots, x_{n-1}]$ e *kernel* $k = [k_0, k_1, \dots, k_{m-1}]$, com dilatação d , a convolução é calculada usando a equação 3.

$$X * k = \sum_{j=0}^{m-1} x_{i - (\frac{m}{2} \cdot d) + (j \cdot d)} \cdot k_j, \forall i \in \{0, 1, \dots, n-1\} \quad (3)$$

De forma equivalente, a operação de convolução pode ser considerada como a soma das colunas de uma matriz \hat{C} na qual cada linha corresponde a multiplicação da série temporal de entrada pelo peso apropriado do *kernel*, e o alinhamento das linhas corresponde à dilatação.

O que foi mostrado nessa seção contribui para que o *MiniRocket* produza uma acurácia semelhante ao *Rocket* mas com menor custo computacional. Isso faz com que ele seja atualmente o estado da arte em classificação de séries temporais (Dempster et al., 2021).

2.4 Base de dados

Nesse trabalho foram utilizados dois conjuntos de dados que, apesar de terem parâmetros similares, possuem uma pequena diferença. O primeiro banco de dados é composto por 940 arquivos que representam falhas ocorridas em linhas de transmissão com diversas variações de parâmetro, e foi proposto no trabalho de Costa et al. (2010). Estes dados foram gerados a partir de simulações no *software* Matlab/Simulink, com as seguintes características:

- (1) Tensão: 500 kV.
- (2) Extensão da linha: 300 km.
- (3) Tipo de falta: AT, BT, CT, AB, BC, CA, ABT, BCT, CAT e ABC.
- (4) Quantidade de amostras por tipo de falta: 94.
- (5) Distância na qual a falta ocorreu em relação ao barramento local: 20 km, 150 km e 280 km.
- (6) Ângulo de aplicação da falta: varia entre 0° e 180° com passo de 10°.

- (7) Resistência da ocorrência da falta: 1 Ω , 50 Ω e 100 Ω .
- (8) Frequência fundamental: 60 Hz.
- (9) Amostras por ciclo: 256.
- (10) Frequência de amostragem: 15360 Hz.
- (11) Os sinais originalmente tem tamanho de 3842 pontos amostrais.

O diagrama unifilar do sistema é apresentado na Figura 1 onde os parâmetros principais da linha de transmissão e as fontes são descritas. O comprimento total da linha é 300 km ($L_1 + L_2 = 300$ km).

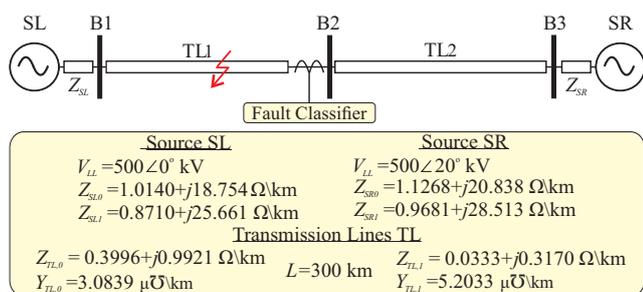


Figura 1. Diagrama unifilar para o sistema de energia trifásico de 500 kV.

Nesse trabalho focou-se exclusivamente na classificação das faltas, por esse motivo, foram utilizados os sinais após a detecção da falta não se dando enfoque na forma que a mesma foi realizada. Tendo em vista isso, foi incorporado um ruído branco gaussiano aos sinais de falta após a ocorrência da falta. Muitos pesquisadores utilizam esse tipo de ruído em seus estudos e em vários artigos é possível verificar que o nível do ruído utilizado varia entre 75 e 20 dB (Mukherjee et al., 2021). Por fim, decidiu-se utilizar a relação sinal-ruído (*signal-noise-ratio* - SNR) de 60 dB nos sinais utilizados nesse projeto.

O segundo banco de dados utilizado foi apresentado em um trabalho prévio reportado em Fonseca et al. (2022) e apresenta as mesmas características mostradas no início dessa sessão para o primeiro *dataset*. Ou seja, mesma tensão, extensão da linha, tipos de falta, distâncias da ocorrência, ângulo de incidência, resistência, frequência fundamental e de amostragem. A diferença dele para o primeiro reside no fato de ter sido incluído um banco de capacitores conectados em série na linha. Devido essa variação, para esse segundo *dataset* é gerada uma quantidade bem maior de dados totalizando 6840 amostras, ou seja, 684 para cada tipo de falta. Para efeitos de referência esse banco de dados será referenciado como *dataset 2*. Essa configuração define um sistema compensado em série, isso é responsável por melhorar os sistemas elétricos conectando um capacitor em série com a linha de transmissão. Dessa forma, a potência reativa é inserida em série com a linha para melhorar a impedância do sistema. Isso melhora a capacidade de transferência de potência, sendo usada principalmente em linhas de tensão extra e ultra-alta (Circuit Globe, 2021).

A compensação em série tem várias vantagens, como aumentar a capacidade de transferência de potência, melhorar a estabilidade do sistema, controlar a regulação da tensão e garantir a divisão de carga adequada entre os alimentadores paralelos (Circuit Globe, 2021). No entanto, como será mostrado nesse artigo, ao utilizá-la é possível

que dificulte a classificação de faltas dependendo da quantidade de ciclo pós falta considerada.

2.5 Metodologia

Nesse projeto foram implementadas abordagens distintas, para se ter um resultado comparativo entre métodos clássicos de aprendizado de máquina com metodologias que surgiram recentemente. Além disso, foram usados os sinais de corrente como sinais de entrada para treinamento dos métodos implementados. Um resumo dessas abordagens é apresentado na Tabela 2.

Cabe destacar ainda que um dos objetivos desse trabalho foi observar o comportamento dos modelos quando uma menor quantidade de sinal era utilizado na entrada. Dessa maneira, foram criadas variações distintas dos sinais contendo 1/4 de ciclo pré falta e com a quantidade de ciclo pós falta variando entre 1, 1/2, 1/4, 1/8, 1/16, 1/32, 1/64 e 1/128.

Como é possível notar nas abordagens A1 e A2, apresentadas na Tabela 2, são utilizadas as mesmas técnicas variando-se apenas o conjunto de dados utilizado. Da mesma forma isso foi realizado para as abordagens B1 e B2.

O treinamento dos métodos foram realizados utilizando-se a validação cruzada *k-fold* com 10 folds. De tal forma que foram utilizadas 676 amostras para treinamento, 76 amostras para teste utilizadas durante o processo de *k-fold* e 188 para validação. Os resultados apresentados na Tabela 3 e Tabela 5 foram obtidos utilizando esse último conjunto.

3. RESULTADOS E DISCUSSÕES

Na subseção 2.4 foram apresentados os bancos de dados usados nesse projeto. Apesar de terem muitos parâmetros iguais entre eles, apenas a inclusão do banco de capacitores no *dataset 2* é capaz de alterar o comportamento dos sinais.

Foi possível observar que com a inclusão do *dataset 2* os modelos treinados perdem a capacidade de generalização quando existem poucos pontos amostrais no sinal pós falta. Para efeito de comparação, utilizou-se a técnica t-SNE para visualizar como os dados usados se distribuem em um espaço bidimensional possibilitando observar a relação entre as classes e como as amostras estão separadas ou agrupadas nesse espaço.

A Figura 2 apresenta o t-SNE em duas dimensões para um sinal com um ciclo pós falta. Na imagem é possível notar que existem *clusters* relativamente separados no espaço. Os dados originais tem cerca de cem *features* então deve-se destacar que existe uma perda considerável de informação ao realizar essa redução de dimensionalidade. O que é interessante notar pelo gráfico é quais tipos de faltas estão próximas entre si, indicando que existe um grau de semelhança e que um possível classificador poderia se confundir entre elas.

Ao analisar os gráficos gerados pelo t-SNE notou-se que os *clusters* dos dados de acordo com o tipo de falta eram homogêneos ao utilizar o *dataset 2* até 1/16 ciclo pós falta. No entanto, com a diminuição da quantidade de ciclo pós

Tabela 2. Abordagens adotadas na realização desse projeto

Abordagem	Conjunto de dados	Ciclo pós falta	Processamento	Classificador
A1	dataset 1	1 a 1/128	Filtro notch	Random forest
A2	dataset 1 + dataset 2	1 a 1/128	Filtro notch	Random forest
B1	dataset 1	1 a 1/128	MiniRocket e Rocket	Regressão ridge
B2	dataset 1 + dataset 2	1 a 1/128	MiniRocket e Rocket	Regressão ridge

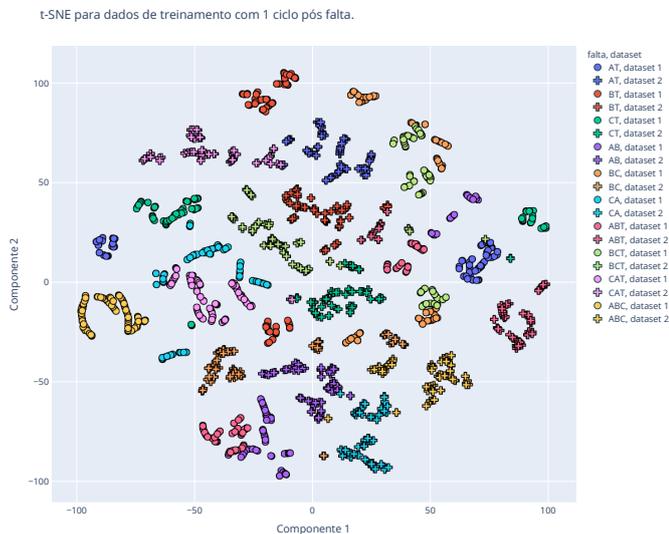


Figura 2. t-SNE para dados de treinamento com 1 ciclo pós falta.

falta, ou seja, utilizando-se 1/32, 1/64 e 1/128 ciclo pós falta, esses *clusters* tendem a se misturar. Dessa maneira, os tipos de falta passam a ser classificados incorretamente. A Figura 3 mostra como ocorre essa mistura entre *clusters* para os dados de treinamento com 1/128 ciclo pós falta.

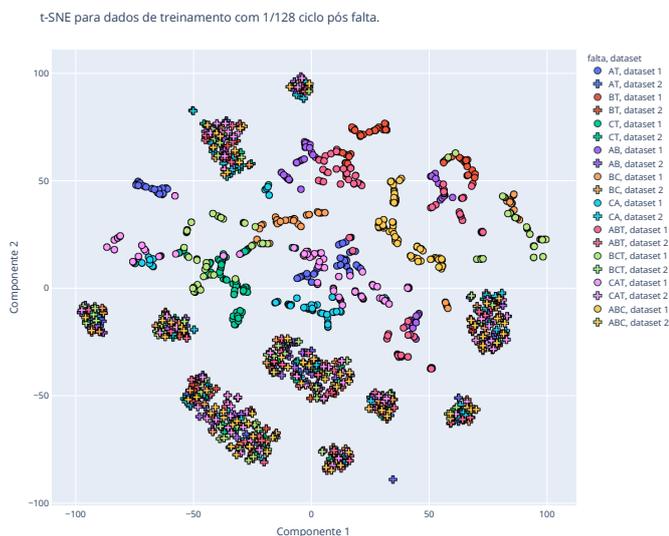


Figura 3. t-SNE para dados de treinamento com 1/128 ciclo pós falta.

3.1 Discussões

Conforme foi anteriormente citado, um dos objetivos desse projeto foi comparar a utilização de técnicas clássicas

(filtro *notch* com *random forest*) e técnicas modernas (*Rocket* e *MiniRocket*) no processamento e classificação de sinais elétricos em falta. Assim, tendo em vista que os modelos das abordagens A1 e B1 foram treinados com o mesmo conjunto de dados, é possível comparar os resultados obtidos e averiguar se teve algum ganho ao utilizar um ou o outro.

O desempenho dos modelos foram analisados observando-se três critérios, a acurácia no conjunto de validação, o tempo necessário para realizar a predição desse conjunto e o tempo operacional (tempo necessário para classificar uma única amostra). Assim, observou-se que utilizando-se a abordagem B1 foi obtida a acurácia de 100% para 1, 1/2, 1/4 e 1/8 de ciclo pós falta, enquanto na abordagem A1 obteve-se 99,74%, 100%, 97,87% e 97,87%, respectivamente. Por sua vez, para 1/16, 1/32, 1/64 e 1/128 ciclo pós falta a abordagem B1 novamente se destaca. Obtendo-se acurácias entre 97,87% e 99,47% com *MiniRocket* e entre 95,68% e 98,40% com *Rocket*. Enquanto para essa mesma quantidade de ciclo pós falta na abordagem A1 foram obtidas acurácias entre 90,96% e 95,75%. A comparação entre esses métodos é mostrada na Tabela 3. Deve-se destacar que há um ganho considerável no tempo operacional ao se utilizar *MiniRocket* na abordagem B1 em relação às técnicas usadas na abordagem A1, como mostrado na Tabela 4.

Tabela 3. Comparação entre os resultados das abordagens A1 e B1.

Ciclo	Abordagem B1						Abordagem A1	
	<i>Rocket</i>		<i>MiniRocket</i>		Acurácia val.	Tempo val.	Acurácia val.	Tempo val.
	Acurácia val.	Tempo val.	Acurácia val.	Tempo val.				
1	100,00%	0,040s	100,00%	0,011s	99,74%	0,045s		
1/2	100,00%	0,048s	100,00%	0,020s	100,00%	0,032s		
1/4	100,00%	0,041s	100,00%	0,045s	97,87%	0,018s		
1/8	100,00%	0,052s	100,00%	0,034s	97,87%	0,019s		
1/16	98,40%	0,076s	99,47%	0,020s	95,74%	0,012s		
1/32	98,40%	0,106s	98,94%	0,037s	95,21%	0,021s		
1/64	96,81%	0,077s	97,87%	0,017s	93,62%	0,015s		
1/128	94,68%	0,067s	97,87%	0,054s	90,96%	0,032s		

Tabela 4. Comparação entre os tempos operacionais mínimo e máximo para as abordagens A1 e B1.

Ciclo	Abordagem B1				Abordagem A1	
	<i>Rocket</i>		<i>MiniRocket</i>		Tempo oper. mín.	Tempo máx.
	Tempo mín.	Tempo máx.	Tempo mín.	Tempo máx.		
1	0,014s	0,015s	0,012s	0,014s	0,069s	0,070s
1/2	0,014s	0,014s	0,012s	0,012s	0,045s	0,046s
1/4	0,013s	0,015s	0,012s	0,012s	0,030s	0,030s
1/8	0,014s	0,030s	0,012s	0,012s	0,025s	0,026s
1/16	0,018s	0,019s	0,011s	0,011s	0,020s	0,020s
1/32	0,019s	0,021s	0,012s	0,012s	0,022s	0,022s
1/64	0,021s	0,026s	0,011s	0,011s	0,019s	0,019s
1/128	0,022s	0,023s	0,014s	0,014s	0,027s	0,027s

De maneira similar, foi realizada a comparação entre as abordagens A2 e B2 que utilizam como dados de entrada tanto o *dataset 1* quanto o *dataset 2*. Nesse caso se observou que existe uma diferença mínima, mas não muito significativa, obtendo-se maiores acurácias na abordagem B2 variando entre 94,15% e 100% usando *Rocket* e 93,09% e 100% com *MiniRocket* para 1, 1/2, 1/4, 1/8, 1/16 de ciclo pós falta. Enquanto para a abordagem A2 foram obtidas

acurácias entre 92,82% e 99,73%. Notou-se também que para 1/32, 1/64 e 1/128 de ciclo pós falta foram obtidas acurácias significativamente baixas utilizando ambas abordagens, a comparação entre esses métodos encontra-se na Tabela 5. Além disso, como é possível notar pela Tabela 6, novamente se obtém um ganho no tempo operacional com a utilização da técnica *MiniRocket* e classificador de regressão *ridge* em comparação ao filtro *notch* e *random forest*.

Tabela 5. Comparação entre os resultados das abordagens A2 e B2.

Ciclo	Abordagem B2				Abordagem A2	
	<i>Rocket</i>		<i>MiniRocket</i>		Acurácia val.	Tempo val.
	Acurácia val.	Tempo val.	Acurácia val.	Tempo val.		
1	100,00%	0,023s	100,00%	0,044s	99,73%	0,055s
1/2	100,00%	0,051s	100,00%	0,031s	99,73%	0,035s
1/4	100,00%	0,072s	99,73%	0,048s	99,73%	0,039s
1/8	98,40%	0,072s	99,47%	0,066s	98,67%	0,033s
1/16	94,15%	0,102s	93,09%	0,060s	92,82%	0,025s
1/32	66,49%	0,105s	65,96%	0,051s	67,82%	0,043s
1/64	56,12%	0,092s	51,71%	0,057s	57,98%	0,026s
1/128	50,80%	0,103s	55,59%	0,043s	50,80%	0,010s

Tabela 6. Comparação entre os tempos operacionais mínimo e máximo para as abordagens A2 e B2.

Ciclo	Abordagem B2				Abordagem A2	
	<i>Rocket</i>		<i>MiniRocket</i>		Tempo oper. mín.	Tempo máx.
	Tempo mín.	Tempo máx.	Tempo mín.	Tempo máx.		
1	0,013s	0,014s	0,014s	0,014s	0,071s	0,072s
1/2	0,016s	0,016s	0,013s	0,013s	0,046s	0,047s
1/4	0,016s	0,020s	0,012s	0,013s	0,037s	0,037s
1/8	0,019s	0,021s	0,013s	0,014s	0,029s	0,029s
1/16	0,024s	0,027s	0,013s	0,013s	0,024s	0,024s
1/32	0,024s	0,026s	0,012s	0,013s	0,030s	0,030s
1/64	0,023s	0,025s	0,012s	0,013s	0,024s	0,025s
1/128	0,027s	0,029s	0,011s	0,013s	0,017s	0,018s

Por fim, é possível notar que bons resultados foram obtidos ao se utilizar *Rocket* e *MiniRocket*. Essas técnicas se destacam principalmente por não exigirem um conhecimento prévio em relação aos sinais, suas formas, frequências e estatísticas. Isso garante um bom funcionamento do sistema de classificação independente do sinal de entrada.

4. CONCLUSÕES

Esse trabalho visou estudar a classificação de faltas em linhas de transmissão utilizando abordagens distintas. Para isso utilizou-se duas técnicas baseadas em *deep learning* chamadas *MiniRocket* e *Rocket* para a extração de características dos sinais de corrente, em conjunto com a regressão *ridge* para a classificação do tipo de falta. Para comparação utilizou-se o método de florestas aleatórias e filtro *notch* em uma segunda abordagem. O resultado de ambas abordagens foi extraído, analisado e apresentado nesse artigo. Conforme foi mostrado as abordagens utilizando *MiniRocket* e *Rocket* apresentaram melhores resultados de acurácia e tempo operacional na maioria dos casos estudados.

É possível concluir com esse projeto que ao incluir os dados de linhas com compensação (*dataset 2*) existe uma queda de desempenho nos modelos (menor acurácia) quando se utiliza uma menor quantidade de ciclo pós falta, principalmente 1/32, 1/64 e 1/128. Cabe destacar ainda que dois testes foram realizados, mas como a princípio não se obteve uma melhora considerável não foi dada continuidade a eles. O teste inicial foi realizar somente o treinamento com o *dataset 1* e validar os modelos com o *dataset 2*, mas a acurácia nesse caso ficou entre 40% e 50%. Como foi apresentado nos gráficos t-SNE na seção 3, existem diferenças

consideráveis entre os *datasets* no espaço bidimensional. Para o *dataset 2* a diminuição da quantidade de ciclo pós falta também induz a uma aglomeração em *clusters* de diferentes tipos de faltas o que dificulta o treinamento e generalização dos modelos. O segundo teste conduzido foi realizar o treinamento de modelos específicos somente com os dados do *dataset 2*, mas ainda assim não se obteve alguma melhoria significativa. Por esses motivos decidiu-se em proceder com o treinamento dos métodos utilizando ambos conjuntos de dados, o que conferiu uma melhoria no desempenho para uma maior quantidade de ciclo pós falta, mas para 1/32, 1/64, 1/128 os modelos ainda não foram capazes de generalizar.

Por fim, comparando-se *Rocket* e *MiniRocket* com *random forest* pouca diferença foi notada. Para a abordagem B1 que utiliza somente o *dataset 1* houve uma leve diferença na acurácia, sendo que o modelo de florestas aleatórias foi levemente inferior. Em relação à abordagem B2 que utiliza os dois *datasets* praticamente não existe diferença no nível da acurácia obtida. No entanto, deve-se destacar que considerando o tempo operacional (tempo para classificar uma única amostra), em geral melhores resultados foram obtidos utilizando-se *MiniRocket*, isso pode ser uma vantagem considerando-se aplicações em tempo real.

4.1 Propostas de trabalhos futuros

Tendo em vista o que foi feito nesse trabalho, é possível realizar algumas melhorias e modificações que serão consideradas em trabalhos futuros. Dentre elas pode-se citar:

- (1) Utilização de outros métodos de processamento de sinais. Por exemplo, a análise de componentes independentes (*independent component analysis* - ICA) já foi utilizada em problemas de localização de faltas e talvez possa ser usada também em problemas de classificação.
- (2) Retreinar os modelos utilizando conjuntamente dados de corrente e tensão para verificar como eles se comportam e se ocorre algum ganho.
- (3) Utilizar alguma metodologia que foque em linhas de transmissão com compensação, especialmente quando há uma menor quantidade de ciclo pós falta. Nesse trabalho observou-se que o comportamento desses sinais com 1/32, 1/64 e 1/128 ciclo pós falta é muito divergente dificultando o treinamento e generalização dos modelos.
- (4) Variar a faixa de ruído para verificar se há impacto no modelo se a relação sinal-ruído é muito baixa, para esse artigo fixou-se o valor do ruído em 60 dB.
- (5) Realização do treinamento e validação dos modelos com dados reais.
- (6) Implementação dos modelos em tempo real.

AGRADECIMENTOS

Os autores agradecem ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) e à FAPEMIG (Fundação de Amparo à Pesquisa do Estado de Minas Gerais) por apoiarem este trabalho.

REFERÊNCIAS

Almeida, A.R., Almeida, O.M., Junior, B.F., Barreto, L.H., and Barros, A.K. (2017). ICA feature extraction

- for the location and classification of faults in high-voltage transmission lines. doi:10.1016/j.epsr.2017.03.030.
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. doi:10.1023/A:1010933404324. URL <https://doi.org/10.1023/A:1010933404324>.
- Chen, Y.Q., Fink, O., and Sansavini, G. (2018). Combined fault location and classification for power transmission lines fault diagnosis with integrated feature extraction. *IEEE Transactions on Industrial Electronics*, 65(1), 561–569. doi:10.1109/TIE.2017.2721922.
- Circuit Globe (2021). Series compensation. URL <https://circuitglobe.com/series-compensation.html>.
- Costa, F., Souza, B., and Brito, N. (2010). Real-time detection of fault-induced transients in transmission lines. *Electronics Letters*, 46, 753 – 755. doi:10.1049/el.2010.0812.
- Das, D., Singh, N.K., and Sinha, A.K. (2005). A comparison of fourier transform and wavelet transform methods for detection and classification of faults on transmission lines. *2006 IEEE Power India Conference*, 2005, 620–626. doi:10.1109/POWERI.2006.1632580.
- De Carvalho, J.R., Coury, D.V., Duque, C.A., and Paula, B.F. (2014). A new transmission line protection approach using cumulants and artificial neural networks. *Journal of Control, Automation and Electrical Systems*, 25(2), 237–251. doi:10.1007/s40313-013-0101-7.
- de Souza, D.P.M. (2018). *Localização de Falhas em Linha de Transmissão de Energia pela Análise da Função de Autocorrelação*. dissertation, Universidade Federal Fluminense.
- Dempster, A., Petitjean, F., and Webb, G.I. (2020). Rocket: exceptionally fast and accurate time series classification using random convolutional kernels. *Data Mining and Knowledge Discovery*, 34(5), 1454–1495. doi:10.1007/s10618-020-00701-z. URL <http://dx.doi.org/10.1007/s10618-020-00701-z>.
- Dempster, A., Schmidt, D.F., and Webb, G.I. (2021). Minirocket. *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. doi:10.1145/3447548.3467231. URL <http://dx.doi.org/10.1145/3447548.3467231>.
- Fonseca, G.A., Ferreira, D.D., Costa, F.B., and Almeida, A.R. (2022). Fault classification in transmission lines using random forest and notch filter. *Journal of Control, Automation and Electrical Systems*, (33), 598–609. doi:10.1007/s40313-021-00844-4.
- Kumar, S.K., Swamy, R. S., and Venkatesh, V. (2014). Artificial Neural Network Based Method for Location and Classification of Faults on a Transmission Lines. 4(1), 2250–3153. URL www.ijsrp.org.
- Maheshwari, A., Agarwal, V., and Sharma, S.K. (2019). Comparative Analysis of ANN-Based FL and Travelling Wave-Based FL for Location of Fault on Transmission Lines. *Journal of The Institution of Engineers (India): Series B*. doi:10.1007/s40031-019-00370-7.
- Mukherjee, A., Kundu, P.K., and Das, A. (2021). Transmission line fault classification under high noise in signal: A direct pca-threshold-based approach. *Journal of The Institution of Engineers (India): Series B*. doi:10.1007/s40031-021-00601-w. URL <https://doi.org/10.1007/s40031-021-00601-w>.
- Prasad, A., Belwin Edward, J., and Ravi, K. (2017a). A review on fault classification methodologies in power transmission systems: Part-I. *Journal of Electrical Systems and Information Technology*, 5(1), 48–60. doi:10.1016/j.jesit.2017.01.004. URL <http://dx.doi.org/10.1016/j.jesit.2017.01.004>.
- Prasad, A., Belwin Edward, J., and Ravi, K. (2017b). A review on fault classification methodologies in power transmission systems: Part-II. *Journal of Electrical Systems and Information Technology*, 5(1), 48–60. doi:10.1016/j.jesit.2017.01.004. URL <http://dx.doi.org/10.1016/j.jesit.2017.01.004>.
- Saravanan, N. and Rathinam, A. (2012). A comparative study on ANN based fault location and classification technique for double circuit transmission line. *Proceedings - 4th International Conference on Computational Intelligence and Communication Networks, CICN 2012*, 824–830. doi:10.1109/CICN.2012.15.
- Scikit-learn, D. (2021). Ridge regression and classification. URL https://scikit-learn.org/stable/modules/linear_model.html#ridge-regression-and-classification.
- Stevenson, W. (1978). *Elementos de análise de sistemas de potência*. McGraw Hill. URL <https://books.google.com.br/books?id=cvGIHAAACAAJ>.