

Remoção de múltiplas do dado sísmico por modelos de aprendizado de máquina

Seismic data multiples removal by machine learning models

Thiago Pacheco Carneiro* e Marcelo Portes de Albuquerque†

*Centro Brasileiro de Pesquisas Físicas
Rua Xavier Sigaud, 150, Urca,
CEP: 22290-180, Rio de Janeiro, Brazil*

Elisangela Lopes de Faria‡
*Centro Brasileiro de Pesquisas Físicas
Rua Xavier Sigaud, 150, Urca,
CEP: 22290-180, Rio de Janeiro, Brazil*

Submetido: 07/11/2022

Aceito: 20/03/2023

Resumo: Reflexões múltiplas são um ruído de tratamento complexo e caro no imageamento sísmico, o qual é ferramenta fundamental para diversas atividades econômicas, como a indústria de óleo e gás. No presente trabalho, é investigada a criação de modelos de aprendizado de máquina baseados na arquitetura RESUNET++ para remoção de múltiplas internas e de superfície livre de imagens sísmicas. Para tal é elaborado um simulador capaz de gerar imagens sísmicas sem múltiplas, bem como um conjunto de treinamento de 100.000 cenários geológicos. Após o treinamento da RESUNET++ com as imagens sísmicas com e sem múltiplas, a rede neural é avaliada em um conjunto de teste. Para avaliação da capacidade de generalização do modelo, ele também é executado em cenários geológicos selecionados. Foi identificada boa capacidade de remoção de múltiplas e de generalização da rede.

Palavras-chave: Aprendizado de Máquina. Geofísica. Múltiplas Sísmicas.

Abstract: Multiples are a kind of noise which is both complex and expensive to deal with in seismic imaging, a fundamental tool for various economic activities, such as the oil & gas industry. Here we investigate the creation of machine learning models based on RESUNET++ for internal and free surface multiples removal in seismic images. To that end, we develop a simulator capable of generating seismic images without multiples, as well as a training set of 100,000 geological scenarios. Trained with seismic images with and without multiples, the RESUNET++ is evaluated in a test set. To characterize the generalization capacity, the network in selected geological scenarios. We conclude that the neural network has good capacity for multiples removal and generalization.

Keywords: Machine Learning. Geophysics. Seismic Multiples.

1. INTRODUÇÃO

Muitos dos recursos naturais que a humanidade consome encontram-se no subsolo: metais, carvão, petróleo, gás, dentre outros. Também é no subsolo que é encontrada a chave de uma das maiores promessas para minorar os efeitos nocivos da nossa dependência de combustíveis fósseis: as técnicas para armazenamento de gás carbônico em subsolo avançam rapidamente. Logo, é inegável a importância de termos informações adequadas sobre a composição e a estrutura da subsuperfície.

Como a observação direta de cada trecho de grandes áreas de interesse do subsolo é inviável, em especial em alto mar onde o solo está muito distante para o emprego de técnicas de observação direta (perfuração), são utilizadas técnicas ge-

ofísicas para sondar essas regiões à distância. Uma das principais técnicas que empregamos é o chamado “método sísmico”, uma espécie de ultrassonografia do subsolo: emitimos ondas mecânicas e captamos a resposta do ambiente a elas. A análise cuidadosa dos dados obtidos permite descobrir uma enorme gama de informações. Na terceira parte deste trabalho iremos detalhar melhor o método.

Há uma série de desafios nessa técnica. Um desses desafios é a presença de múltiplas. Múltiplas são ondas que ao invés de fazerem os caminhos mais curtos da origem até o destino, passam parte de sua jornada reverberando dentro das muitas camadas do subsolo (Fig. 1). Essas ondas aparecem na imagem sísmica como repetições das reflexões primárias, que são as ondas que nos trazem as informações que sabemos interpretar corretamente, dificultando a leitura do sinal. Assim, é preciso conseguir remover as múltiplas dos dados adquiridos para extrair a melhor informação possível da subsuperfície [1].

Existem várias técnicas bem estabelecidas e de custo razoável para a remoção das múltiplas de superfície livre (aquelas nas quais a reverberação ocorre entre o assoalho

*Electronic address: thiago.carneiro@petrobras.com.br

†Electronic address: marcelo@cbrpf.br

‡Electronic address: elisangela.elf@gmail.com

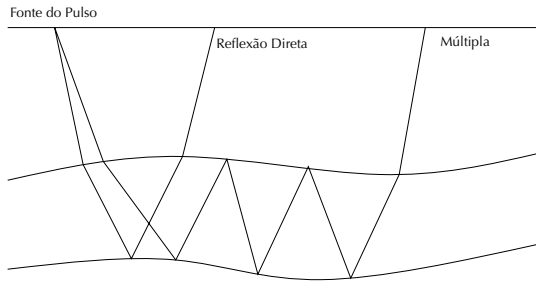


FIG. 1: Reflexão Direta e Reflexão Múltipla

oceânico e a superfície livre da água), tais como SRME (Surface-Related Multiple Elimination) [2], EPSI (Estimating Primaries by Sparse Inversion) [3] e ISS (Inverse Scattering Series) [4]. Mas a remoção de múltiplas internas (aquelas cuja reverberação ocorre nas camadas internas da subsuperfície) é um grande desafio para a indústria [5], devido à necessidade de um grande poder computacional, bem como de qualidade dos dados adquiridos e, ainda, de estimativas sobre as camadas presentes na subsuperfície. Recentemente, tem-se começado a investigar a remoção de múltiplas internas e de superfície por modelos de aprendizado de máquina. Os modelos são treinados a partir de exemplos reais ou sintéticos, dos quais foram removidas as múltiplas através do uso de técnicas tradicionais.

Dentre as investigações recentes sobre o tema, destacam-se três exemplos:

Wang et al. [6] treinaram uma RESUNET a partir de perfis reais, cujas múltiplas de superfície foram filtradas com técnicas tradicionais. Para isso, realizaram um processo de *data augmentation* utilizando um ruído de fundo e rotações dos dados. Em seu trabalho, porém, a rede treinada com dados reais de um local só classificou bem dados de locais próximos (geologia semelhante). Para locais diferentes, é preciso retrainar a rede.

Bugge et al. [7] tiveram uma proposta mais arrojada para o treinamento: criaram as reflexões primárias e múltiplas de superfície matematicamente, de forma aleatória. Tiveram bons resultados aplicando a rede assim treinada em dados reais, removendo as múltiplas de superfície livre com menos perda de sinal do que no método tradicional.

Bao et al. [8] usaram dados simulados e reais, filtrados com técnicas tradicionais de remoção de múltiplas de superfície livre e internas. Porém, assim como no trabalho de Wang et al., foi preciso retrainar a rede para cada geologia diferente.

Por fim, o objetivo do presente trabalho é realizar a remoção de múltiplas internas e de superfície livre em dados sintéticos, usando uma rede neural RESUNET++ treinada com dados gerados por um simulador capaz de gerar dados com e sem múltiplas, bem como verificar sua capacidade de generalização para novos cenários geológicos.

2. REDES NEURAI PROFUNDAS

Aprendizado de Máquina é um campo da Inteligência Artificial no qual não há um humano a programar regras, a ensinar ao computador uma lógica a ser seguida. O que se faz é

expor um algoritmo generalista a um conjunto de dados para deles extrair conhecimento. Em especial, no Aprendizado de Máquina Supervisionado é usada uma técnica na qual se treina um programa a partir de exemplos de entradas e de suas respectivas saídas esperadas para que ele possa aprender a inferir, a partir de uma entrada inédita, uma saída apropriada [9]. Por exemplo: fornecendo dados meteorológicos de um longo período ele pode conseguir estimar, a partir dos dados do instante atual, as condições meteorológicas aproximadas nas próximas horas. Nas palavras de Gilbert Strang, o objetivo do Aprendizado de Máquina é “construir uma função que classifique os dados de treinamento corretamente, para que possa generalizar para dados inéditos.” [10, pg. iii]¹ Um dos métodos de Aprendizado de Máquina que tem ganhado destaque no tratamento de dados sísmicos é o de Redes Neurais Profundas [11], conforme vimos nas referências [6], [7] e [8].

Rede Neurais Profundas são aquelas em que há um grande número de camadas de neurônios, nas quais a entrada é abstraída em passos incrementais. Das representações mais elementares (como identificação de bordas e texturas) nas primeiras camadas para as mais complexas e informativas (como reconhecimento de formas complexas: rostos, objetos, expressões) nas camadas mais profundas, a Rede Neural aprende quais as características mais importantes dos dados. [12, pgs. 7 e 17] Isso traz um grande trunfo para essas Redes: não há necessidade de realizar a chamada “engenharia de características”² pois a própria Rede se encarrega disto. Redes Neurais Profundas tem sido empregadas com sucesso no tratamento de imagens sísmicas. [13] [6] [14]

2.1. RESUNET++

A RESUNET++ [15] é uma rede baseada na RESUNET [16] [17] (a qual por sua vez é baseada na UNET [18] [19]), com a inclusão de camadas de *Squeeze and Excitation*, *Atrous Spatial Pyramidal Pooling* e de “Atenção”. Como toda arquitetura descendente da UNET, possui três partes: uma “descida” na qual a imagem é codificada, uma base na qual temos a representação mais abstrata da imagem, e uma “subida” na qual a imagem é decodificada, usando convoluções transpostas, de volta ao tamanho original.

Convolução transposta é um tipo de convolução que resulta numa imagem com o dobro do tamanho original. Num primeiro passo, pixels são inseridos entre cada pixel da imagem original, com valor 0, como visto na Figura 3. Então é aplicada uma convolução, a qual irá gerar uma representação da imagem em maior escala.

Os mecanismos de Atenção [20], [21] podem ser entendidas por uma analogia com a visão: assim como nossos olhos vêem uma imagem focalizada no centro de nossa visão e bor-

¹ No original: To construct a function that classifies the training data correctly, so it can generalize to unseen test data.

² A engenharia de características (*feature engineering*) é a criação de novas características a partir do dado original. Por exemplo, a partir de um valor inicial x , entrega-se ao modelo também o valor x^n para permitir que um modelo linear distinga características não lineares.

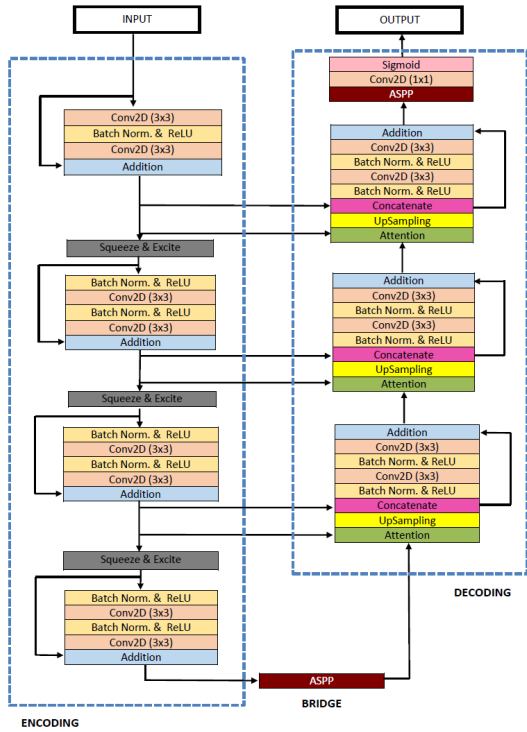


FIG. 2: RESUNET++. Imagem de [15]

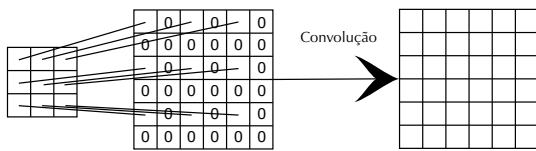


FIG. 3: Convolução Transposta: duplicação do tamanho da imagem usando 0s, seguida de convolução.

rada ao redor, os mecanismos de Atenção aprendem a identificar quais partes da imagem precisam de mais destaque, aumentando seu peso, enquanto diminuem o peso das partes menos importantes. Assim, melhoram a qualidade das características a serem tratadas pela Rede. São utilizados nos blocos de decodificação na RESUNET++, antes das Convoluções Transpostas, para identificar as regiões mais importantes nas representações da imagem.

As camadas de *Squeeze and Excitation* [22] identificam e amplificam os canais mais importantes na representação da imagem. Para cada canal é atribuído um peso, e os canais são multiplicados por esses pesos. A rede aprende a fazer essa distribuição de pesos de maneira que os canais mais importantes para o aprendizado são reforçados, enquanto que canais menos importantes tem seu sinal enfraquecido. De certa forma, a ação das camadas de *Squeeze and Excitation* (Fig. 4) é análoga à das camadas de atenção: ambas ajudam a rede a reforçar informação relevante e a ignorar informação irrelevante, porém atuam em visões diferentes da informação.

As camadas de *Atrous Spatial Pyramid Pooling* (ASPP) [23], são capazes de extrair informações em várias escalas. Em uma *Atrous Convolution*, também conhecida como “Convolução Dilatada”, a convolução é feita sobre pixels es-

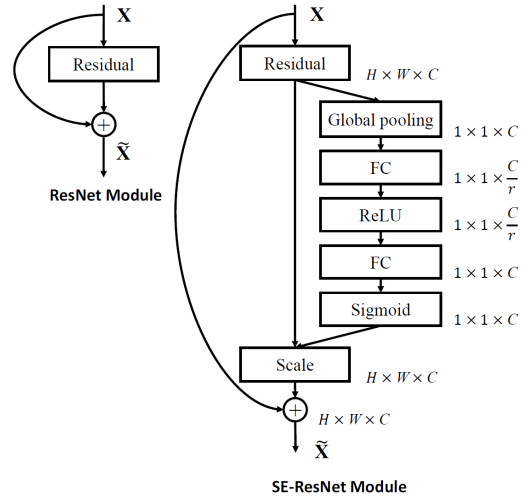


FIG. 4: Bloco residual com *Squeeze and Excitation*. Imagem de [22].

paçados, “enxergando” uma região maior da imagem sem incorrer em maior consumo de processamento ou memória do que uma convolução tradicional. A ASPP faz várias Convoluções Dilatadas (Fig. 5) em paralelo, cada uma com um espaçamento diferente entre os pixels, e concatena os resultados. Assim, pode detectar padrões relevantes em diferentes escalas. Na RESUNET++ a ASPP é usada na base do U, fazendo a “ponte” entre a codificação e a decodificação, bem como na última camada de decodificação.

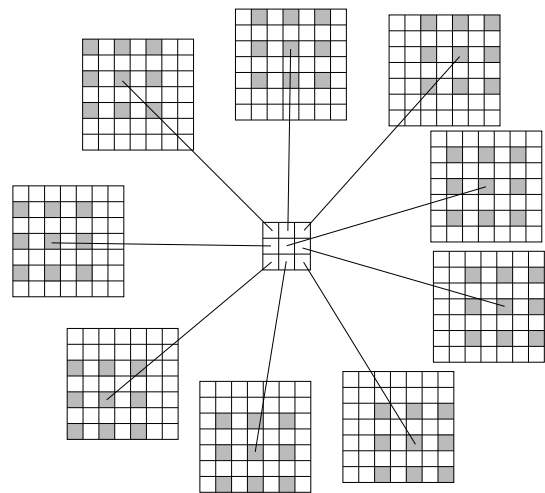


FIG. 5: Convolução Dilatada: pontos espaçados (cinza) ao invés de área contínua.

Como vimos na introdução, vários artigos demonstram resultados positivos com o uso de redes derivadas da UNET para tratamento de dados sísmicos, assim como também demonstram [24] e [25]. A RESUNET++, por sua vez, já foi aplicada em imagens sísmicas com sucesso, mas para estimativa de porosidade [26]. Esta é a primeira vez que a RESUNET++ é usada para tratar a imagem sísmica em si.

3. IMAGEM SÍSMICA

Métodos sísmicos são utilizados para prospectar o subsolo a partir da análise da propagação de ondas mecânicas. Na aquisição sísmica terrestre de reflexão é usada uma série de detectores (geofones) alinhados, colocados em intervalos regulares entre si, e uma fonte que irá gerar as ondas mecânicas, colocada também alinhada, próxima ou entre esses detectores. As ondas geradas pela fonte se propagam tanto diretamente em direção aos geofones pela superfície do solo como também penetram nas camadas inferiores do solo, onde são refletidas de volta à superfície ao passar por um contraste de impedância, no encontro entre diferentes tipos de rocha. A análise do tempo de viagem e da amplitude dessas componentes refletidas é que permite a inferência da geometria e do modelo de velocidades subsuperfície.

Em um meio com densidade ρ em que ondas acústicas (paralelas à direção de propagação) viajam com velocidade v , a impedância Z é dada por $Z = \rho v$. Quando uma onda acústica atinge perpendicularmente a interface entre dois meios de impedâncias diferentes, Z_1 e Z_2 , a amplitude A_r da onda refletida é dada por $A_r = A_i (Z_2 - Z_1) / (Z_2 + Z_1)$, onde A_i é a amplitude da onda incidente. Ou seja, quanto maior a diferença entre as impedâncias, maior a parcela da onda que será refletida (se $Z_1 > Z_2$, a onda é refletida com uma diferença de fase π) [1].

As reflexões primárias, ou diretas, são assim chamadas porque são o produto de apenas uma reflexão (Figs. 6 e 7):

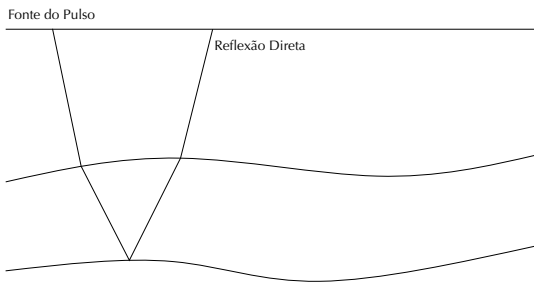


FIG. 6: Reflexão direta.

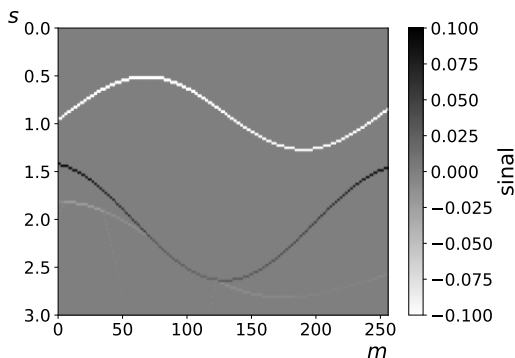


FIG. 7: Imagem sísmica com reflexões diretas. Imagem gerada no simulador recursivo elaborado para este artigo.

As reflexões múltiplas, por outro lado, são o resultado de reverberações entre as camadas (Figs. 8 e 9):

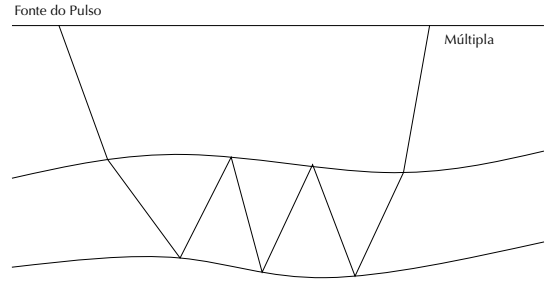


FIG. 8: Reflexão múltipla interna.

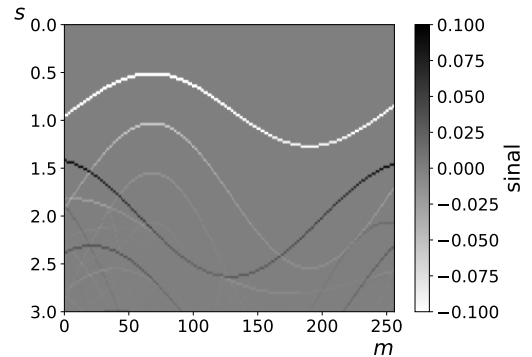


FIG. 9: Imagem sísmica com reflexões diretas e múltiplas internas e de superfície livre. Imagem gerada no simulador recursivo elaborado para este artigo.

Note que o sinal das múltiplas se mistura com o das primárias na imagem sísmica.

Na maioria das técnicas de análise da imagem sísmica, as múltiplas não são aproveitadas para a investigação de subsuperfície: apenas o sinal das reflexões primárias é considerado [27]. Desta forma, as múltiplas são consideradas um tipo de ruído e devem ser removidas para permitir análises corretas.

3.1. Simulação sísmica

A simulação de aquisição sísmica normalmente é feita pelo Método de Diferenças Finitas (MDF) [28], a partir da equação da onda. Entretanto, nesse método não é possível separar automaticamente as reflexões primárias das múltiplas internas, o que é necessário para o treinamento e avaliação quantitativa dos modelos de aprendizado de máquina. Assim, será preciso elaborar uma técnica que permita essa separação.

Isso é possível a partir de um método recursivo que simula a propagação unidimensional de um pulso. O pulso se desloca entre camadas do subsolo, as quais possuem velocidade de propagação de onda constante, e na interface entre elas é refletido e transmitido na direção vertical, gerando duas chamadas recursivas ao propagador de pulsos. Uma imagem sísmica é montada realizando centenas de simulações destas lado-a-lado. Quando se deseja eliminar as múltiplas, basta programar o algoritmo para que não haja reflexão (para baixo) das ondas que viajam para cima, apenas transmis-

são. Assim, não há reverberações. Para fins de identificação, aqui este simulador será denominado “Simulador Recursivo”. O resultado desta técnica é semelhante ao que seria atingindo usando aquisições de offset zero (ou seja, com a fonte do pulso sísmico e um único geofone tão próximos que podem ser considerados localizados no mesmo ponto) em cenários de camadas paralelas, a menos da convolução de uma wavelet. Propositalmente, não é empregada a convolução com uma wavelet no resultado do simulador recursivo: o uso de pulsos pontuais nos permite usar métricas de classificação, as quais nos permitem analisar pixel a pixel da imagem, e não de regressão. Modelos treinados com esses dados não são aplicáveis a dados reais.

A função recursiva utilizada no nosso simulador segue a seguinte lógica:



FIG. 10: Aquisição de Offset Zero

```

FUNCAO PULSAR(camada, tempo, pulso, sentido, com_multiplas):
  SE sentido == descida:
    ida = PULSAR(PROXIMA_CAMADA(camada),
      tempo+TEMPO_PERCURSO(camada),
      pulso*DISSIPACAO(camada)*
        COEFICIENTE_TRANSMISSAO_DESCENDO(camada),
      descida,
      com_multiplas)
    vinda = PULSAR(camada,
      tempo+TEMPO_PERCURSO(camada),
      pulso*DISSIPACAO(camada)*
        COEFICIENTE_REFLEXAO_DESCENDO(camada),
      subida,
      com_multiplas)
  SE sentido == subida:
    SE com_multiplas == verdadeiro:
      ida = PULSAR(camada,
        tempo+TEMPO_PERCURSO(camada),
        pulso*DISSIPACAO(camada)*
          COEFICIENTE_REFLEXAO_SUBINDO(camada),
        descida,
        com_multiplas)
    SE camada == primeira:
      RETORNAR (tempo+TEMPO_PERCURSO(camada),
        pulso*DISSIPACAO(camada))
    SENAO:
      vinda = PULSAR(CAMADA_ANTERIOR(camada),
        tempo+TEMPO_PERCURSO(camada),
        pulso*DISSIPACAO(camada)*
          COEFICIENTE_TRANSMISSAO_SUBINDO(camada),
        subida,
        com_multiplas)
  RETORNAR ida + vinda

```

Podemos comparar uma simulação de tiro de offset zero realizada com a biblioteca Devito e a mesma situação simulada no nosso simulador recursivo, considerando um modelo de velocidades de camadas paralelas de $100m$, mesma densidade e velocidades alternadas de 1000 e $2000m/s$:

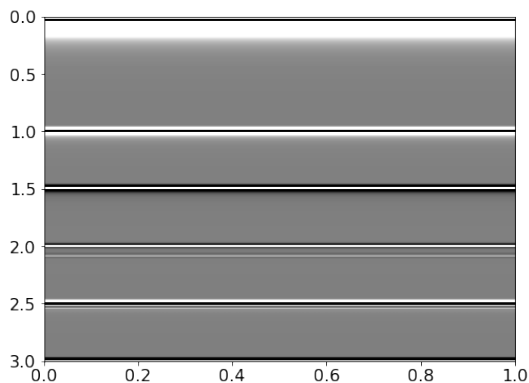


FIG. 11: Imagem sísmica construída por aquisições de offset zero com a biblioteca Devito.

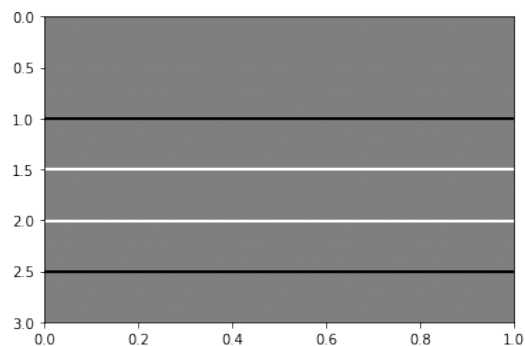


FIG. 12: Imagem sísmica gerada no simulador recursivo.

Desconsiderando pequenas diferenças, como a assinatura da wavelet e imprecisões das simulações (por exemplo, o atraso no tempo de chegada da múltipla de superfície livre na Fig. 12, a qual chega um pouco após os 2s teóricos), vemos que os tempos dos eventos são coincidentes entre as duas simulações, validando o resultado do simulador recursivo.

4. METODOLOGIA

O presente artigo propõe investigar a remoção de múltiplas internas e de superfície por uma Rede Neural Profunda – mais especificamente por uma RESUNET++ – treinada com dados gerados por um simulador capaz de gerar imagens sísmicas com e sem múltiplas.

Primeiramente é preciso elaborar um simulador capaz de gerar dados sintéticos com e sem múltiplas. Como vimos na literatura [7], não é necessário que o simulador seja fisicamente realista. Basta que sejam gerados dados com semelhança suficiente com uma imagem sísmica real (ou seja, que pareça realista) para que possa ser treinada uma Rede Neural Profunda capaz de tratar dados sísmicos.

Em seguida é preciso elaborar centenas de milhares de sistemas de sub-superfície parametrizados aleatoriamente, a fim de obter ampla cobertura de cenários. Escolhemos uma parametrização utilizando senóides, em que o período, fase, amplitude e profundidade de cada senóide determinam a interface entre diferentes camadas, as quais tem velocidade e densidade também aleatórias. A partir desses sistemas, aplica-se a simulação sísmica, gerando dois conjuntos de imagens sísmicas: o primeiro é o conjunto de sísmicas que incluem tanto as reflexões primárias como as múltiplas internas e de superfície livre. O segundo é o conjunto de imagens sísmicas que apresentam apenas as reflexões primárias.

Os conjuntos são divididos em subconjuntos de treino, validação e teste com 100.000, 30.000 e 30.000 amostras, respectivamente. Com os conjuntos de treino são alimentados diferentes modelos de Rede Neural. Os conjuntos de validação são usados a cada época³ do treinamento para avaliar a capacidade de generalização dos modelos em treinamento. O estado de cada modelo que melhor generalizar para o conjunto de validação será escolhido como seu estado ótimo.

Devem ser treinados diferentes modelos de Rede Neural Profunda, baseados na RESUNET++. Em particular, deve ser investigado o efeito da adição de uma camada de Batch Normalization no início da Rede, o número de filtros usados, o uso de diferentes métricas (MSE [29], Binary CrossEntropy [30], Focal Loss [31]) e o tamanho de conjunto de treinamento. Também faremos alguns testes com a UNET e a RESUNET para comparação dos resultados.

Como medida objetiva, deverá ser utilizada a métrica Dice [32], sendo também calculadas a Interseção sobre União (IoU em inglês) [33], Sensibilidade (*recall*) e Precisão [34], para fins de comparação.

Para avaliação da capacidade de generalização das Redes Neurais, poderão ser avaliadas suas performances em 3 cenários de sub-superfície típicos na investigação sísmica, em especial de hidrocarbonetos: camadas paralelas, domo de sal homogêneo e domo de sal heterogêneo sobre reservatório.

5. EXPERIMENTOS COMPUTACIONAIS E RESULTADOS

Para obter um modelo adequado de remoção de múltiplas é preciso, em primeiro lugar, possuir dados que possam servir de exemplo. Além disso, é preciso definir a arquitetura do modelo a ser treinado.

5.1. Redes Neurais

Para avaliação serão utilizadas 10 diferentes arquiteturas, baseadas na RESUNET++, RESUNET e UNET:

³ Uma época de treinamento significa que cada elemento do conjunto de treinamento foi usado uma vez para treinar o modelo. Todo o processo de treinamento consiste em várias épocas: a Rede Neural revisita os dados de treinamento diversas vezes, aprendendo iterativamente.

RESUNET++ 32: A arquitetura RESUNET++ com 4 camadas de codificação e decodificação, sendo a primeira camada de codificação com 32 filtros. Usa a *Focal Loss* como métrica de treinamento.

RESUNET++ 32BN: A mesma arquitetura, mas com uma camada extra de *Batch Normalization* na entrada da rede. Usa a *Focal Loss* como métrica de treinamento.

RESUNET++ 32BN MSE: A mesma arquitetura da RESUNET++ 32BN, mas usando o erro médio quadrado (MSE em inglês) como métrica de treinamento.

RESUNET++ 32BN BCE: A mesma arquitetura da RESUNET++ 32BN, mas usando a entropia binária cruzada (BCE em inglês) como métrica de treinamento.

RESUNET++ 32BN 10K: A mesma arquitetura da RESUNET++ 32BN, mas usando 10.000 exemplos no treinamento, ao invés de 100.000. Usa a *Focal Loss* como métrica de treinamento.

RESUNET++ 16BN: A mesma arquitetura, mas com apenas a metade do número de filtros em suas camadas. Usa a *Focal Loss* como métrica de treinamento.

RESUNET 32: A arquitetura RESUNET com 6 camadas de codificação e decodificação, sendo a primeira camada de codificação com 32 filtros. Usa a *Focal Loss* como métrica de treinamento.

RESUNET 32BN: A mesma arquitetura, mas com uma camada extra de *Batch Normalization* na entrada da rede. Usa a *Focal Loss* como métrica de treinamento.

UNET 32: A arquitetura UNET com 6 camadas de codificação e decodificação, sendo a primeira camada de codificação com 32 filtros. Usa a *Focal Loss* como métrica de treinamento.

UNET 32BN: A mesma arquitetura, mas com uma camada extra de *Batch Normalization* na entrada da rede. Usa a *Focal Loss* como métrica de treinamento.

Devido a limitações do hardware utilizado, é necessário limitar a resolução das imagens a 256x128 (LxA). Cada pixel tem seu valor representado por um ponto-flutuante de dupla precisão.

Inicialmente, há o desafio de elaborar, para o treinamento de um modelo de aprendizado de máquina, uma grande massa de sistemas de subsuperfície, a qual deveria ser representativa do máximo de cenários possível. Para atender a este desafio, pode ser utilizada uma parametrização de sistemas baseada em senóides. Com uma grande faixa de valores aleatórios para o período da onda, fase, amplitude e profundidade, bem como com a escolha aleatória da velocidade e densidade das camadas, há uma rica variedade de sistemas configuráveis, bastante distintos entre si (Fig. 13).

Com o conjunto de teste podemos avaliar e comparar os modelos treinados. Durante a produção dos dados, são gerados 160.000 sistemas de subsuperfície, os quais posteriormente são separados em um conjunto de treinamento (100.000), teste (30.000) e validação (30.000) (Fig. 14).

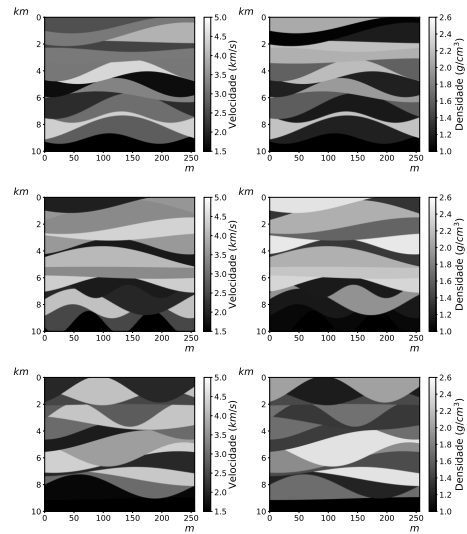


FIG. 13: Gráficos de velocidade e densidade de sistemas de subsuperfície de treinamento gerados aleatoriamente.

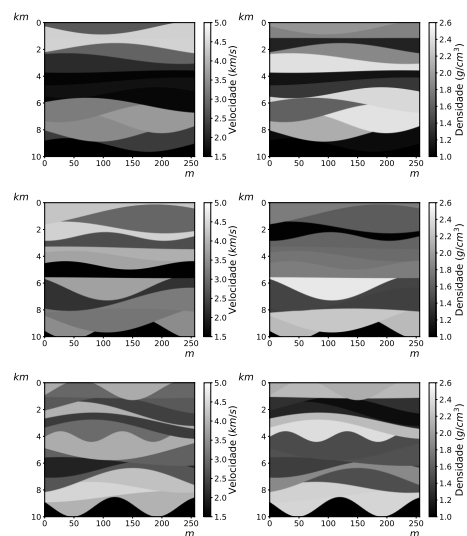


FIG. 14: Alguns dos sistemas subsuperfície geradores do conjunto de teste.

Como exemplo, apresentamos (Fig. 15) um exemplo retirado dos sistemas subsuperfície (densidade e velocidade) com sua respectiva imagem sísmica, além de uma máscara binária identificando os pontos de ocorrência de sinal, independentemente da amplitude do mesmo. Em seguida (Fig. 16 a Fig. 18), temos o resultado da aplicação de diferentes modelos de aprendizado de máquina sobre este exemplo. A máscara binária na Fig. 15 mostra toda a complexidade do sinal recebido, independente da ordem de magnitude. Na imagem sísmica podemos observar a presença das múltiplas, como esperado, e notar como um número muito reduzido dos sinais é composto por reflexões diretas (que podem ser identificadas em y_{teste} nas Figs. 16 a 18). Observando as figuras, podemos verificar visualmente as diferentes performances dos modelos (por exemplo, é possível observar o menor número de erros na Fig. 16, referente ao modelo mel-

Modelo	Recall	Precision	IoU	DC
RESUNET++ 32BN	0,9820	0,9939	0,9761	0,9879
RESUNET++ 16BN	0,9801	0,9908	0,9713	0,9854
UNET 32BN	0,9758	0,9934	0,9696	0,9845
RESUNET 32BN	0,9711	0,9956	0,9670	0,9832
RESUNET++ 32BN MSE	0,9657	0,9381	0,9079	0,9517
UNET 32	0,8687	0,9919	0,8627	0,9263
RESUNET 32	0,8393	0,9905	0,8326	0,9087
RESUNET++ 32	0,8090	0,9927	0,8042	0,8915
RESUNET++ 32BN BCE	0,8125	0,8632	0,7198	0,8371
RESUNET++ 32BN 10K	0,5902	0,9481	0,5718	0,7275

TABLE I: Tabela de resultados do conjunto de teste, ordenada pelo *Dice Coefficient*. Devido ao longo tempo necessário para cada treinamento, não foi possível realizar múltiplos treinamentos e assim estimar a variância das medidas.

hor classificado nos testes):

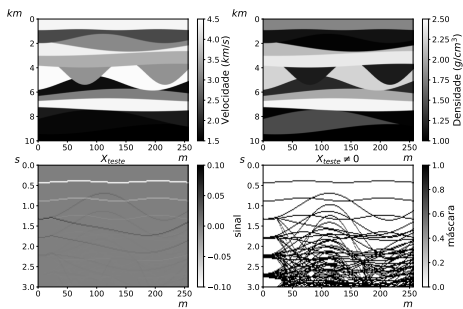


FIG. 15: Sistema subsuperfície do conjunto de teste e sua imagem sísmica.

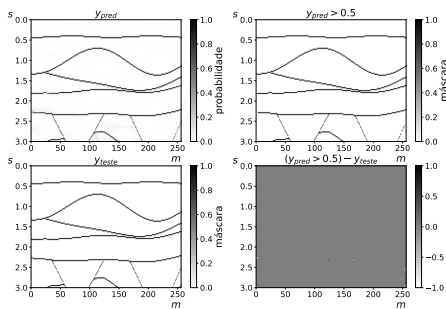


FIG. 16: RESUNET++ 32BN aplicada em elemento do conjunto de teste.

Por simplicidade, optamos por incluir neste artigo apenas algumas das imagens geradas pelas 10 redes neurais.

5.2. Generalização

A fim de avaliar a capacidade de generalização dos diferentes modelos treinados, além do conjunto de teste serão utilizados 3 cenários geológicos diferentes: camadas paralelas

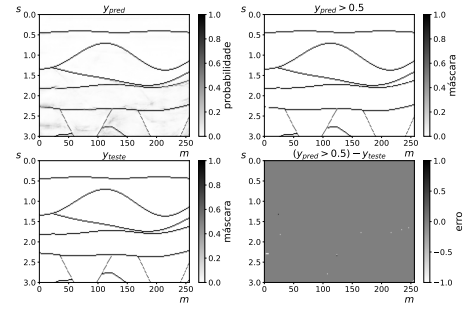


FIG. 17: UNET 32BN aplicada em elemento do conjunto de teste.

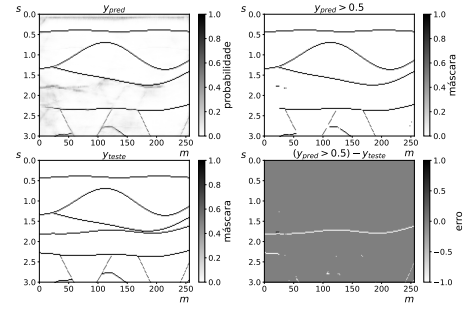


FIG. 18: UNET 32 aplicada em elemento do conjunto de teste.

(Fig. 19), domo de sal homogêneo (Fig. 20) e domo de sal heterogêneo com um reservatório abaixo do mesmo (Fig. 21). Estes cenários serão preparados no nosso Simulador Recursivo para avaliar nossos modelos de remoção de múltiplas.

Abaixo temos os sistemas geológicos dos 3 cenários (Figs. 19, 20 e 21).

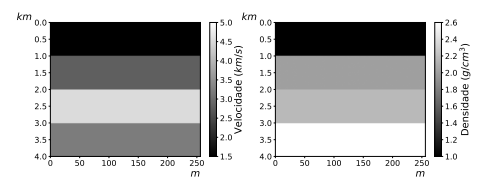


FIG. 19: Sistema de camadas paralelas

Para as diferentes camadas de subsuperfície foram consideradas velocidades entre 2.800 e 4.500m/s e densidades entre 2 e 2.6g/cm³. Para o sal, foram consideradas velocidades de 4.500 a 5.000m/s e densidades de 2.16 a 2.5g/cm³. Para o reservatório, usamos a velocidade de 2.400m/s e densidade de 1.6g/cm³.

A simulação recursiva destes sistemas geológicos gera as sísmicas vistas adiante (Figs. 22, 26 e 30).

Abaixo temos, para cada cenário e para alguns modelos de Rede Neural Profunda: a sísmica simulada sem múltiplas, a sísmica simulada com múltiplas internas e de superfície, a máscara da sísmica simulada sem múltiplas (Figs. 22, 26 e 30). Em seguida, vemos a máscara sísmica prevista pelos

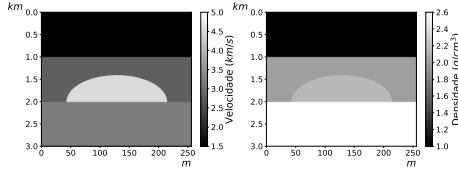


FIG. 20: Sistema de domo de sal homogêneo

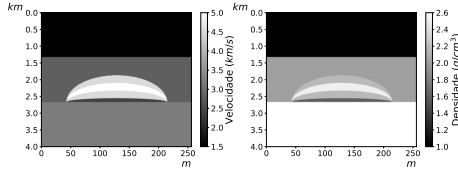


FIG. 21: Sistema de domo de sal heterogêneo sobre reservatório

ossos modelos e a diferença entre as duas (demais Figs. 23 a 33).

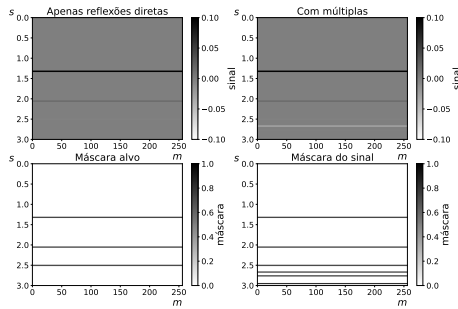


FIG. 22: Sísmica com e sem múltiplas do sistema de camadas paralelas

Modelo	Recall	Precision	IoU	DC
RESUNET++ 32BN	1,0000	1,0000	1,0000	1,0000
RESUNET++ 32BN MSE	1,0000	1,0000	1,0000	1,0000
RESUNET++ 16BN	1,0000	1,0000	1,0000	1,0000
RESUNET++ 32	0,9974	1,0000	0,9974	0,9987
RESUNET 32BN	1,0000	0,9624	0,9624	0,9808
UNET 32BN	1,0000	0,9505	0,9505	0,9746
RESUNET 32	0,7708	1,0000	0,7708	0,8706
UNET 32	0,7122	1,0000	0,7122	0,8319
RESUNET++ 32BN BCE	0,6693	0,6853	0,5120	0,6772
RESUNET++ 32BN 10K	0,6667	0,6667	0,5000	0,6667

TABLE II: Paralelas - Tabela de resultados

5.3. Análise dos Resultados

A performance em todas as métricas dos modelos no conjunto de teste (Tab. I) deixa clara a importância de usarmos uma camada de *Batch Normalization* na entrada da rede para os dados da simulação. As vantagens da RESUNET++

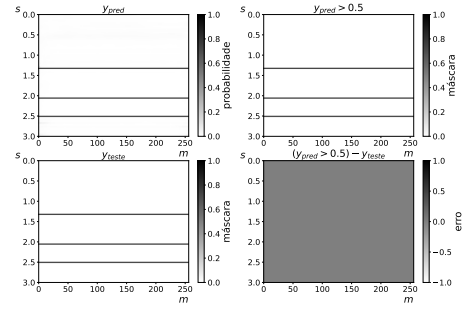


FIG. 23: Camadas paralelas - RESUNET++ 32BN

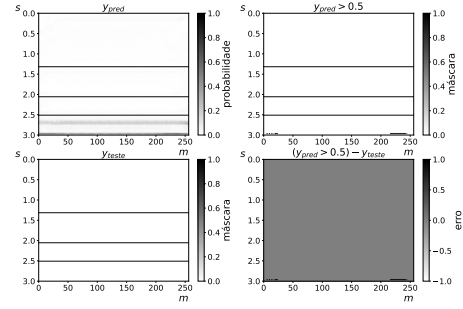


FIG. 24: Camadas paralelas - UNET 32BN

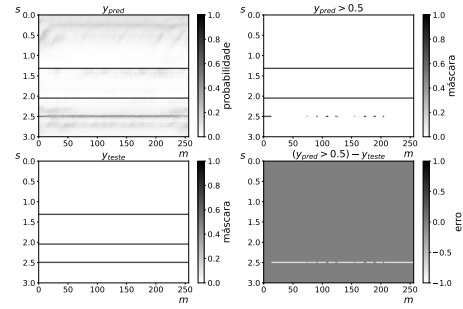


FIG. 25: Camadas paralelas - UNET 32

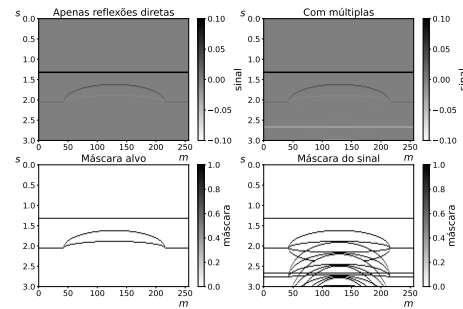


FIG. 26: Sísmica com e sem múltiplas do sistema de domo de sal homogêneo

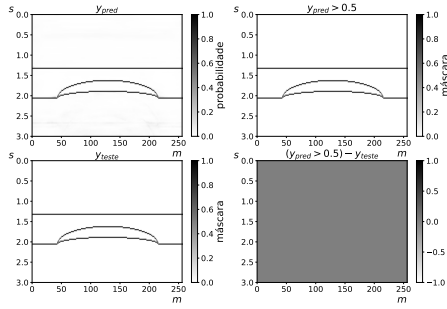


FIG. 27: Domo de sal homogêneo - RESUNET++ 32BN

Modelo	Recall	Precision	IoU	DC
RESUNET++ 32BN	1,0000	1,0000	1,0000	1,0000
UNET 32	1,0000	1,0000	1,0000	1,0000
RESUNET++ 32	0,9956	1,0000	0,9956	0,9978
RESUNET 32	1,0000	0,9870	0,9870	0,9935
RESUNET 32BN	1,0000	0,9856	0,9856	0,9927
UNET 32BN	1,0000	0,9421	0,9421	0,9702
RESUNET++ 32BN MSE	1,0000	0,9119	0,9119	0,9539
RESUNET++ 16BN	1,0000	0,8836	0,8836	0,9382
RESUNET++ 32BN BCE	0,9985	0,8187	0,8177	0,8997
RESUNET++ 32BN 10K	0,6955	0,6498	0,5059	0,6719

TABLE III: Domo de sal homogêneo - Tabela de resultados

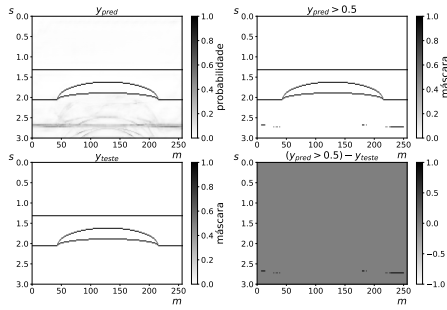


FIG. 28: Domo de sal homogêneo - UNET 32BN

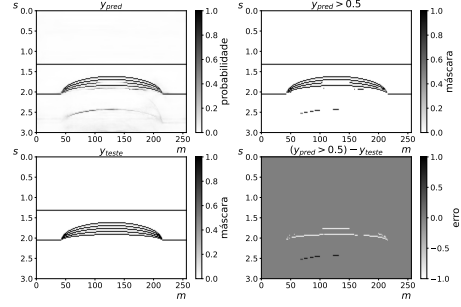


FIG. 31: Domo de sal heterogêneo sobre reservatório - RESUNET++ 32BN

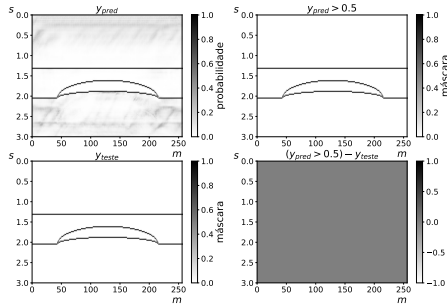


FIG. 29: Domo de sal homogêneo - UNET 32

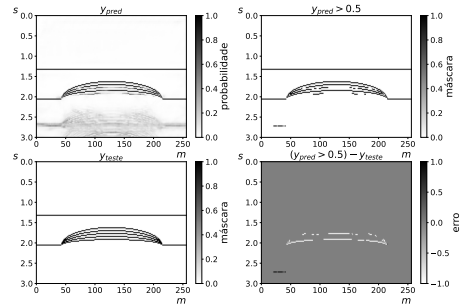


FIG. 32: Domo de sal heterogêneo sobre reservatório - UNET 32BN

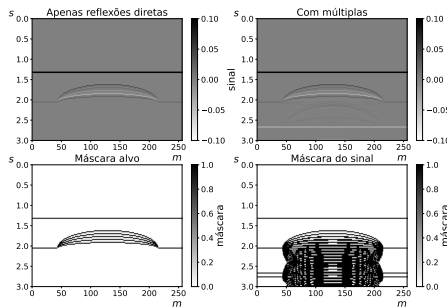


FIG. 30: Sísmica com e sem múltiplas do sistema de domo de sal heterogêneo sobre reservatório

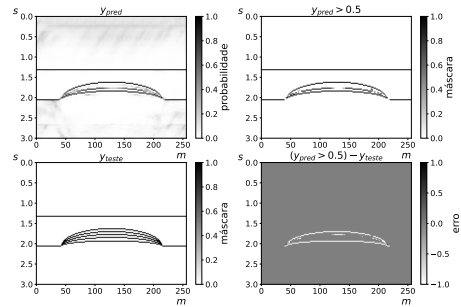


FIG. 33: Domo de sal heterogêneo sobre reservatório - UNET 32

Modelo	Recall	Precision	IoU	DC
RESUNET++ 32BN	0,8122	0,9897	0,8054	0,8922
UNET 32BN	0,8164	0,9807	0,8035	0,8910
RESUNET++ 32BN	0,8190	0,9661	0,7961	0,8864
RESUNET++ 32BN MSE	0,8206	0,9282	0,7717	0,8711
RESUNET 32BN	0,7453	0,9800	0,7342	0,8467
RESUNET++ 32	0,7250	0,9977	0,7238	0,8398
RESUNET++ 16BN	0,7648	0,9085	0,7101	0,8305
RESUNET 32	0,7064	0,9988	0,7058	0,8276
UNET 32	0,6548	0,9974	0,6537	0,7906
RESUNET++ 32BN 10K	0,4391	0,7119	0,3728	0,5432

TABLE IV: Domo de sal heterogêneo sobre reservatório - Tabela de resultados

32BN (*Dice Coeficient* 0,9879) nas métricas do modelo, embora numericamente pequenas, demonstram a superioridade da arquitetura, inclusive frente à própria RESUNET++16BN (*Dice Coeficient* 0,9854). Ou seja, a presença de um maior número de filtros na rede teve impacto positivo apreciável. As imagens da saída das redes mostram como o resultado da RESUNET++ 32BN mostra uma menor incerteza na classificação dos pontos da imagem. Isso também evidencia como a rede conseguiu processar melhor as sísmicas do que as outras arquiteturas.

Analisando os resultados dos 2 primeiros cenários da verificação de generalização (Figs. 22 a 25 e 26 a 29, respectivamente), podemos observar que o modelo RESUNET++ 32BN generalizou muito bem para esses cenários inéditos usando o simulador recursivo. No terceiro cenário (Figs. 30 a 33), mais complexo e de difícil tratamento, o modelo não conseguiu preservar totalmente a terceira e a última reflexões centrais (referentes a uma das camadas internas do sal e à base do reservatório, respectivamente), mas ainda assim teve boa performance (*Dice Coeficient* 0,9093), distintamente acima dos demais modelos.

O conjunto dos resultados indica que a RESUNET, em todas as suas versões, não demonstra superioridade frente à UNET na aplicação proposta. Da mesma maneira, verificamos que a métrica de treinamento *Focal Loss* é a mais adequada para esta tarefa e que 100.000 exemplos de treinamento são de fato necessários.

6. CONCLUSÃO

Nessa investigação foi possível criar o Simulador Recursivo, confirmar as hipóteses de que os produtos do Simulador Recursivo possuem informação suficiente para que uma Rede Neural aprenda a distinguir múltiplas de reflexões primárias e de que a Rede treinada com os cenários geológicos senoidais consegue ter um bom desempenho em cenários distintos do treinamento.

Como ponto positivo, destaca-se o ganho nas métricas de erro com a adoção da RESUNET++, que indica possibilidade de melhorias ainda maiores conforme novas arquiteturas de Redes Neurais venham a ser adotadas. Também observamos os ganhos de usar uma camada de *Batch Normalization* no

início da rede. O principal, porém, é que nossa investigação indica que é possível treinar a RESUNET++ para remoção das múltiplas, inclusive as internas, se houver o volume de dados de treinamento adequado. A capacidade de generalização da rede mostra que o uso de um conjunto de treinamento parametrizado por ondas senoidais promove uma boa independência dos modelos treinados em relação a diferentes cenários geológicos, que é uma limitação frequentemente presente em outros trabalhos.

Infelizmente nosso simulador não é capaz de considerar a real complexidade de propagação do pulso mecânico em um plano, limitando-se a simular a propagação em linhas verticais. Apesar disso precisamos usá-lo pois as técnicas para simulações existentes (MDF, por exemplo) não são capazes de gerar sísmicas sem múltiplas internas, necessárias para o treinamento da rede neural. Também observamos a necessidade de uma grande quantidade (100.000) de exemplos de treinamento. Isso por sua vez, impactou a nossa capacidade de treinar a rede para imagens maiores do que 256x128. Mesmo dispondo de uma GPU de 24GB e de 256GB de RAM, tivemos erros de falta de memória quando tentamos utilizar imagens de tamanho maior.

O sucesso da rede, especialmente em generalizar o treinamento, indica que temos uma boa base para novos desenvolvimentos: muitas melhorias podem ser aplicadas ao presente trabalho, mas elas estarão fundamentadas em uma base robusta, com a cobertura de diferentes modelos geológicos proporcionada pelo conjunto de treinamento construído por senoidais parametrizadas, gerado com baixo custo computacional. Uma das principais dificuldades que vemos em outros trabalhos, como os que citamos na introdução, é a generalização para além dos modelos geológicos presentes no treinamento.

Como investigação futura, seria interessante o desenvolvimento de uma versão do Simulador Recursivo em ray-tracing, pois poderia simular mais corretamente a propagação do pulso mecânico no plano. Também poderia-se investigar tratamentos nos produtos do Simulador Recursivo, como os efeitos da convolução com diferentes wavelets ou adição de ruídos diversos. Caso fosse aplicada uma convolução, entretanto, não seria possível analisar os resultados como um problema de classificação, pois não teríamos mais como alvo uma máscara binária, e sim um contínuo de valores. Consideramos importante, nesta avaliação inicial, poder utilizar métricas de classificação. Elas permitem identificarmos o quanto a rede é capaz de remover múltiplas e preservar as reflexões primárias independentemente da ordem de grandeza do sinal em cada ponto da imagem sísmica, o que não seria verdade com uma métrica como Mean Square Error, por exemplo. Seria interessante uma investigação da aplicação dos modelos aqui apresentados em simulações de Métodos de Diferenças Finitas, inclusive para comparação direta com imagens sísmicas que tenham suas múltiplas removidas pelos métodos tradicionais. Por fim, também há espaço para análises de treinamentos em resoluções maiores do que as que utilizamos, em equipamentos com maior capacidade de memória.

- [1] P. Kearey, M. Brooks, and I. Hill, *An Introduction to Geophysical Exploration* (Wiley-Blackwell, 2013), 3rd ed.
- [2] D. J. Verschuur, A. J. Berkhout, and C. P. A. Wapenaar, *GEOPHYSICS* **57**, 1166 (1992), URL <https://doi.org/10.1190/1.1443330>.
- [3] G. J. van Groenestijn and D. J. Verschuur, *GEOPHYSICS* **74**, A23 (2009), URL <https://doi.org/10.1190/1.3111115>.
- [4] F. M. Carvalho, A. B. Weglein, and R. H. Stolt, *Examples of a nonlinear inversion method based on the t matrix of scattering theory: Application to multiple suppression* (2005), URL <https://library.seg.org/doi/abs/10.1190/1.1889114>.
- [5] CGG, *Cgg: Seismic internal multiple attenuation (ima)* (2022), URL <https://www.cgg.com/geoscience/subsurface-imaging/internal-multiple-attenuation>.
- [6] X. Wang, Y. Sui, W. Wang, and J. Ma, *Mathematical Geosciences* (2022), ISSN 1874-8953, URL <https://doi.org/10.1007/s11004-022-10032-y>.
- [7] A. J. Bugge, A. K. Evensen, J. E. Lie, and E. H. Nilsen, *Leading Edge* **40**, 831 (2021), ISSN 19383789.
- [8] P. Bao, Y. Shi, W. Wang, J. Xu, and X. Guo, *Energies* **15**, 3883 (2022).
- [9] S. Russel and P. Norvig, *Artificial Intelligence: A Modern Approach* (2021).
- [10] G. Strang, *Linear Algebra and Learning from Data* (Wellesley - Cambridge Press, 2019), ISBN 978-0-692-19638-0.
- [11] E. Stevens, L. Antiga, and T. Viehmann, *Deep Learning with PyTorch* (Manning, 2020).
- [12] F. Chollet, *Deep Learning with Python* (Manning, 2021), 2nd ed., ISBN 9781617296864.
- [13] Z. York, Z. Qie, Y. Anar, and S. Yunzhi, *Leading Edge* (Tulsa, OK) **38**, 526 (2019), ISSN 1070-485X, URL <https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,sso&db=edsgsw&AN=edsgsw.2020.007098&lang=pt-br&site=eds-live&authtype=ip,sso&custid=s7173344&groupid=main&profile=eds>.
- [14] J. Li, X. Wu, and Z. Hu, *IEEE Transactions on Geoscience and Remote Sensing* **60**, 1 (2022).
- [15] D. Jha, P. H. Smedsrud, M. A. Riegler, D. Johansen, T. de Lange, P. Halvorsen, and H. D. Johansen (2019), URL <http://arxiv.org/abs/1911.07067>.
- [16] Z. Zhang, Q. Liu, and Y. Wang (2017), URL <http://arxiv.org/abs/1711.10684> <http://dx.doi.org/10.1109/LGRS.2018.2802944>.
- [17] H. Qian, X. Wang, X. Chen, and Z. Yang, *Energies* **15** (2022), ISSN 1996-1073, URL <https://www.mdpi.com/1996-1073/15/9/3362>.
- [18] O. Ronneberger, P. Fischer, and T. Brox (2015), URL <http://arxiv.org/abs/1505.04597>.
- [19] W. Wang, F. Yang, and J. Ma, **2018**, 1 (2018), ISSN 2214-4609, URL <https://www.earthdoc.org/content/papers/10.3997/2214-4609.201800917>.
- [20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, *CoRR* **abs/1706.03762** (2017), URL <http://arxiv.org/abs/1706.03762>.
- [21] H. Li, P. Xiong, J. An, and L. Wang (2018), URL <http://arxiv.org/abs/1805.10180>.
- [22] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu (2017), URL <http://arxiv.org/abs/1709.01507>.
- [23] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille (2016), URL <http://arxiv.org/abs/1606.00915>.
- [24] A. P. O. Muller, J. C. Costa, C. R. Bom, E. L. Faria, M. Klatt, G. Teixeira, M. P. de Albuquerque, and M. P. de Albuquerque, *GEOPHYSICS* pp. 1–50 (2022), ISSN 0016-8033.
- [25] M. Klatt, E. L. Faria, A. P. O. Muller, J. M. Coelho, J. L. González, M. P. de Albuquerque, M. P. de Albuquerque, C. R. Bom, and M. D. Correia, *GEOPHYSICS* pp. 1–51 (2022), ISSN 0016-8033.
- [26] H. Jo, Y. Cho, M. Pyrcz, H. Tang, and P. Fu, *GEOPHYSICS* **87**, M217 (2022), ISSN 0016-8033.
- [27] A. B. Weglein, *The Leading Edge* **34**, 808 (2015), URL <https://doi.org/10.1190/tle34070808.1>.
- [28] H. P. Langtangen and S. Linge, *Finite Difference Computing with PDEs* (Springer, 2017).
- [29] F. M. Dekking, C. Kraaikamp, H. P. Lopuhaä, and L. E. Meester, *A Modern Introduction to Probability and Statistics: Understanding Why and How* (Springer, 2005), ISBN 9781852338961, URL https://books.google.com.br/books?id=odn7_auSAnEC.
- [30] G. P. Miriyala, A. K. Sinha, D. N. K. Jayakody, and A. Sharma (2021), pp. 269–274.
- [31] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár (2017), URL <http://arxiv.org/abs/1708.02002>.
- [32] L. R. Dice, *Ecology* **26**, 297 (1945), URL <https://esajournals.onlinelibrary.wiley.com/doi/abs/10.2307/1932409>.
- [33] A. H. Murphy, *Weather and Forecasting* **11**, 3 (1996).
- [34] D. Powers, *Mach. Learn. Technol.* **2** (2008).