NDM-Net and Model Changes

Elena Gondyul(✉), Kirill Gadylshin, Vadim Lisitsa, and Dmitry Vishnevsky

Institute of Petroleum Geology and Geophysics SB RAS,
Koptug Avenue 3, Novosibirsk, Russia 630090
gondyulea@ipgg.sbras.ru

Abstract. The paper discusses the problem of suppressing the numerical error, which together represents the numerical dispersion and the error of the roughened velocity model. Such a numerical error is suppressed by using a NDM-net (Numerical Dispersion Mitigation neural network), which was originally proposed to map seismograms with numerical dispersion to accurate solutions calculated on a fine grid. This approach makes it possible to accelerate the seismic modeling by 6 times without significant changes in quality. Also, the paper discusses two-stage training of NDM-net, in which the numerical dispersion is first suppressed, and then the interface error.

Keywords: Seismic modelling · Deep learning · Numerical Dispersion

1 Introduction

Seismic modeling has an important role in elucidating the laws of seismic wave propagation and response mechanisms in complex media. A widely used method for modeling wave fields is the finite difference method [11,16,23,26] due to its simple implementation and the use of a small amount of memory. However, when using coarse grids to reduce computational costs, numerical dispersion occurs. Along with other classical grid methods, such as finite elements [32], spectral elements [25,33], discontinuous Galerkin [3,13,20] pseudo-spectral methods [4, 19] and others, moreover, require sufficiently large computational costs to obtain solutions without numerical errors. Thus, if the grid step has been halved, the number of grid points increases in 16 times.

One way to reduce the numerical error is to use an increased approximation order (the formal order of the basis function) [9,12,18]. However, this approach reduces the Courant number, which makes the stability condition more stringent. Also, the length of the stencil (for finite differences) or the size of the local mass and stiffness matrices for the discontinuous Galerkin method also increases, requiring more floating-point operations per grid node.

G. Gondyul performed numerical simulations on NDM-net application, V. Lisitsa suggested the approach to data suppression via FFT, K. Gadylshin developed the algorithm to apply the U-net to dispersion suppression and D.Vishnevsky performed seismic modelling under the support of Russian Science Foundation grant no. 22-11-00004.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2024
O. Gervasi et al. (Eds.): ICCSA 2024 Workshops, LNCS 14817, pp. 263–275, 2024.
https://doi.org/10.1007/978-3-031-65238-7_18

Another approach is to use numerical dispersion suppression schemes [2,6,14] in the finite difference method. However, this approach also has a large number of flops per grid node and a strict stability condition.

The presented methods are complemented by procedures prior to modeling wave fields. An alternative method is to use deep learning methods that are applied to seismic fields calculated on a coarse grid. Thus, we can reduce computational costs and get enough high-quality data, since neural networks are good approximators of the transition operator by extracting high-level features from the data.

Deep learning has become an integral part of seismic processing nowadays. Methods such as Neural Networks are used to interpolate missed traces [17,31], suppress white and ground noise [24,30], detect first arrival picking [22,29] and so on. The Numerical Dispersion Mitigation neural network (NDM-net) [5] has been proposed to suppress numerical dispersion in seismic fields. Such a network has performed well in suppressing the numerical dispersion associated with temporal and spatial coordinates.

In this paper, we explore the possibilities of NDM-net, particularly, whether it is possible to use NDM-net when changing not only the computational grid but also changing the sampling rate of the velocity model. Such a study will allow us to expand the functionality of the NDM-net.

2 Statement of the Problem

2.1 Seismic Modelling

Consider the elastic wave equation written in the Cartesian coordinates in terms of velocity and stress tensor components:

$$\begin{aligned}
\rho\frac{\partial u_x}{\partial t} &= \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xz}}{\partial z},\\
\rho\frac{\partial u_z}{\partial t} &= \frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{zz}}{\partial z},\\
\frac{\partial \sigma_{xx}}{\partial t} &= (\lambda+2\mu)\frac{\partial u_x}{\partial x} + \lambda\frac{\partial u_z}{\partial z} + \hat{f}_{xx},\\
\frac{\partial \sigma_{zz}}{\partial t} &= \lambda\frac{\partial u_x}{\partial x} + (\lambda+2\mu)\frac{\partial u_z}{\partial z} + \hat{f}_{zz},\\
\frac{\partial \sigma_{xz}}{\partial t} &= \mu\frac{\partial u_x}{\partial z} + \mu\frac{\partial u_z}{\partial x} + \hat{f}_{xz},
\end{aligned} \tag{1}$$

where $\vec{u} = (u_x, u_z)^T$ is the particle velocity vector, σ_{xx}, σ_{zz}, σ_{xz} are the stress tensor components, $\rho = \rho(x,z)$ is the mass density, $\lambda = \lambda(x,z)$ and $\mu = \mu(x,z)$ are the Lame's parameters, and $\hat{f}_{xx} = f_{xx}(t)\delta(x-x_s)\delta(z-z_s)$, $\hat{f}_{zz} = f_{zz}(t)\delta(x-x_s)\delta(z-z_s)$, $\hat{f}_{xz} = f_{xz}(t)\delta(x-x_s)\delta(z-z_s)$ are the seismic moments tensor. Typically, the point sources are considered, thus $\delta(x-x_s)$ is the Kronecker's delta function, where (x_s, z_s) is the source location coordinates. The time wavelets $f_{xx}(t)$, $f_{zz}(t)$, $f_{xz}(t)$ are defined by a band-limited impulse.

System of Eq. (1) is solved in a limited area $D = [x_s - L_x, x_s + L_x] \times [0, Z]$, which is determined for each source x_s with maximal offset L_x and depth of the model Z.

The classical 2D acquisition system is considered. The system consists of N_s sources located on a surface with coordinates $(x_s, 0)$, and N_r receivers for each source with coordinates $(x_r^j(x_s), 0)$.

For convenience, we introduce the offset coordinates $x_o^j = x_r^j - x_s$, where $x_o \in [L_o, L_o]$ changes with fixed interval for each source positions with maximal offset $L_o \leq L_x$. We consider seismograms, denoted as $\vec{u}^{x_s}(t, x_o)$, as a solution to system (1).

2.2 Finite-Difference Scheme

The Lame coefficients are used as parameters for the system (1), which are obtained from knowledge about the velocity of seismic waves in elastic media. For each source, a velocity model is considered in a limited volume $M(x, z, x_s^j) \subset M(x, z)$, where $M(x, z)$ is the entire model. The model itself can be refined or not refined, depending on the discretization step of the model h_x^M, h_y^M, and is interpolated for further numerical modeling using the finite difference method.

We used staggered-grid finite differences [8,9,26,27] to solve system (1), which is described in detail in [5]. We will denote $\vec{u_h}(t, x_0, x_s^k)$ as a numerical solution of system (1) using a grid step $h = h_x = h_z$.

Second-order in time and fourth-order accurate in space scheme defined on spaced grids are used:

$$\begin{aligned}
&\rho D_t[u_x]^{n+1/2}_{i+1/2,j} = D_x[\sigma_{xx}]^{n+1/2}_{i+1/2,j} + D_z[\sigma_{xz}]^{n+1/2}_{i+1/2,j},\\
&\rho D_t[u_z]^{n+1/2}_{i+1/2,j} = D_x[\sigma_{xz}]^{n+1/2}_{i+1/2,j} + D_z[\sigma_{zz}]^{n+1/2}_{i+1/2,j},\\
&D_t[\sigma_{xx}]^{n}_{i,j} = (\lambda + 2\mu) D_x[u_x]^{n}_{i,j} + \lambda D_z[u_z]^{n}_{i,j} + (\hat{f}_{xx})^{n}_{i,j},\\
&D_t[\sigma_{zz}]^{n}_{i,j} = \lambda D_x[u_x]^{n}_{i,j} + (\lambda + 2\mu) D_z[u_z]^{n}_{i,j} + (\hat{f}_{zz})^{n}_{i,j},\\
&D_t[\sigma_{xz}]^{n}_{i+1/2,j+1/2} = \mu D_x[u_z]^{n}_{i+1/2,j+1/2} + \mu D[u_x]^{n}_{i+1/2,j+1/2} + (\hat{f}_{xz})^{n}_{i+1/2,j+1/2},
\end{aligned} \tag{2}$$

where finite-difference operators is defined as:

$$D_t[f]^N_{I,J} = \frac{f^{N+1/2}_{I,J} - f^{N-1/2}_{I,J}}{\tau} \approx \frac{\partial f}{\partial t}, \tag{3}$$

$$\begin{aligned}
D_x[f]^N_{I,J} &= \tfrac{9}{8} \tfrac{f^N_{I+1/2,J} - f^N_{I-1/2,J}}{h_x} - \tfrac{1}{24} \tfrac{f^N_{I+3/2,J} - f^N_{I-3/2,J}}{h_x} \approx \tfrac{\partial f}{\partial x},\\
D_z[f]^N_{I,J} &= \tfrac{9}{8} \tfrac{f^N_{I,J+1/2} - f^N_{I,J-1/2}}{h_z} - \tfrac{1}{24} \tfrac{f^N_{I,J+3/2} - f^N_{I,J-3/2}}{h_z} \approx \tfrac{\partial f}{\partial z},
\end{aligned} \tag{4}$$

where $f(t, x, z)$ is a smooth enough function.

Here, we do not specify the way to treat the coefficients of the Eq. $\lambda(x, z)$, $\mu(x, z)$, and $\rho_{x,z}$ it is discussed in [10,15,16,28].

2.3 NDM-Net

Here, we will briefly describe the Numerical Dispersion Mitigation neural network (NDM-net), which was suggested in [5]. To begin with, we need to calculate seismograms $\vec{u_{h_2}}(t, x_0, x_s^k)$ for all source positions x_s^k using a coarse grid with

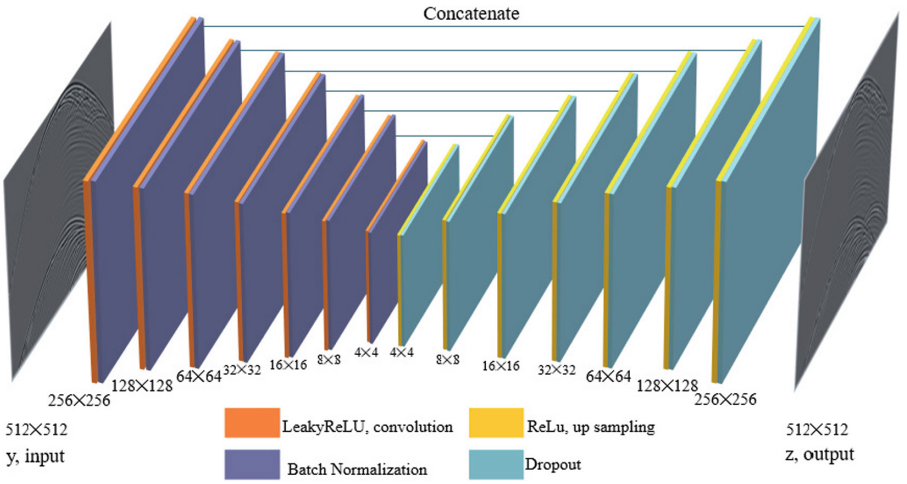


Fig. 1. NDM-net architecture

characteristic grid step h_2. This allows to obtain an inaccurate solution with less computational costs. At the same time, it is possible to compute a small number of seismograms $\vec{u_{h_1}}(t, x_o, x_s^l)$, where $l \in S_t \subset \{1, ..., N_s\}$ is the index of source included in the training sample, using a fine enough mesh with characteristic grid step $h_1 < h_2$ to obtain an accurate solution. After that neural network is trained by optimizing the loss function:

$$L(\Theta) = \mathbf{E}_{l \in S_t} \left[\left\| \vec{u}_{h_1}(t, x_o, x_s^l) - F\left[\vec{u}_{h_2}(t, x_o, x_s^l), \Theta\right] \right\|_1 \right],$$

where $F : \vec{u}_{h_2} \rightarrow \vec{u}_{h_1}$ is the operator represented by a multilayer perceptron, $\Theta = \{\mathbf{W}, \mathbf{b}\}$ is the network parameters, which includes the weight matrix $\mathbf{W}$ and bias $\mathbf{b}$, $\mathbf{E}_{l \in S_t}$ is a mean over the training dataset.

We used U-shaped network based on U-net architecture [21] as shown in Fig. 1. And includes 8 downsampling layers with convolution operations, activation function $ReLU(\cdot) = \max(0, \cdot)$ and 8 upsampling layers with deconvolution operations and $LeakyReLU(\cdot) = \max(1, \cdot) + c\min(1, \cdot)$ as an activation function with slope factor $c = 0.2$. The training parameters are given in Table 1. They were selected in the course of numerous experiments during the study.

It is obvious that a different network architecture can be promising in seismic data processing. For example, [7] suggests using a cascade of dense U-net for denoising images, [30] uses a deep convolutional neural network for denoising and ground-roll attenuation of seismic data, and [17] uses cGan for interpolation of seismic data. However, our task requires fairly fast learning and good generalizability, which is provided by U-net.

We used two NVIDIA GeForce RTX 3090 cards with 8 GB of memory and Intel Core CPU with 256 GB RAM to train NDM-net. Calculations were performed using the Pytorch framework.

Table 1. Hyperparameters and training parameters

Parameter	Value
Learning rate	from 0.0002 to 0.0001 during epoch
Momentum parameters	$\beta_1 = 0.5, \beta_2 = 0.999$
Batch size	1
The number of epochs	750
Optimization algorithm	Adaptive Momentum
Optimality criterion	Mean Absolute Error

3 Numerical Experiments

In this section, we provide the analysis of numerical experiments with regularly located sources as training samples for a neural network. Firstly, we show the input data that is used for training and testing the NDM-net. Secondly, we train the NDM-net on this data, and then we calculate the initial error and the error between the generated and correct data on the test dataset (previously unseen by the NDM-net).

3.1 Input Data

In this section, we describe Marmousi model and acquisition systems that is used for subsequent experiments.

Initially, the Marmousi model was created based on a profile passing through the North Quenguela and included 160 layers [1]. The size of the model is 3501.25 m $\times$ 17001.25 m in vertical and horizontal directions, respectively. The P-wave velocity varies greatly from 1500 m/s to 4500 m/s and is shown in Fig. 2.

The acquisition system consists of 171 sources with a distance of 100 m between each other and receivers that are located at a distance of 25 m from each other. For each source position, the wave field is recorded by 513 receivers with an offset of about 6.5 km. The source is a Ricker wavelet with a central frequency of 30 Hz.

We simulated 6 sets of seismograms that differ from each other in the sampling steps in elastic medium models h^M and the sampling steps of the computational grid h. Let the seismogram be denoted as d^{M_*,S_*}, where M_* and S_* denote the model discretization and computational grid respectively. The description of the calculated grids for each type of seismogram is given in Table 2. The sampling rate was 1 ms, and the recording length was 5 s.

The Fast Fourier Transform (FFT) is used to compress seismograms and translate them into the frequency domain. Such a procedure allows us to reduce the data dimension from 512×5000 to 512^2 by taking only the positive part of the spectrum and throwing out high-frequency information, which contains mainly noise. The non-trivial part of the FFT of the one seismogram is shown

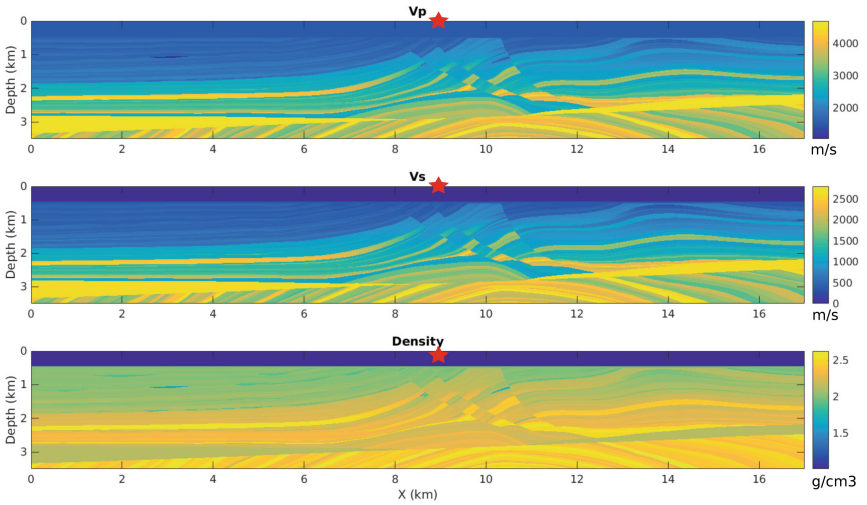


Fig. 2. Marmousi velocity model

Table 2. Sampling steps in an elastic medium model and a computational grid for calculating datasets

Data	$h_x^M = h_z^M$, m	$h_x = h_z$, m
d^{M_1,S_1}	1.25	1.25
d^{M_2,S_1}	2.5	1.25
d^{M_2,S_2}	2.5	2.5
d^{M_5,S_1}	5	1.25
d^{M_5,S_2}	5	2.5
d^{M_5,S_5}	5	5

in Fig. 3. It is limited by the frequency interval $[0, 102.4]$ Hz with the frequency resolution of $1/5$ Hz.

Thus, we have formed datasets to investigate the effect of errors in models on the results of using a NDM-net.

4 Training Results

We trained a total of 12 NDM-net on different training datasets with a size of 60% of the total number of seismograms N_s. Consider d as the seismogram with numerical dispersion as the target data and d' as the labeled data (clear seismogram) and d_{gen} as the generated by NDM-net seismogram. A training dataset is a pair consisting of $\{d, d'\}_{i=1}^{N}$, where N is the size of the dataset.

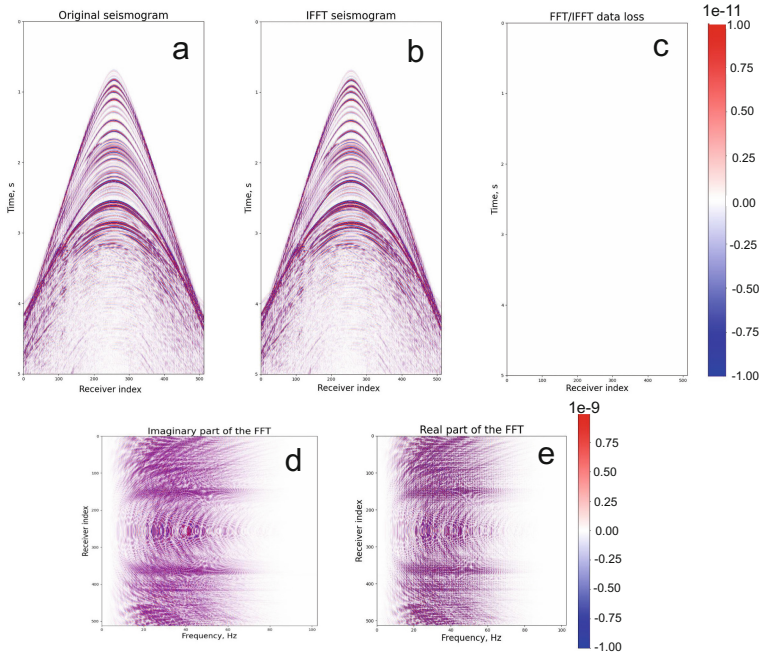


Fig. 3. Transfer of seismic data to the frequency domain and back to the time domain: a) the original seismogram; b) the inverse Fourier transform; c) the absolute difference between the original and inverse Fourier transform; d), e) the Fourier transform (real and imaginary part).

The relative L_2 norm to estimate the error between the data and verify the performance of the NDM-net is used:

$$L_2^{rel}(d', d_{gen}) = \frac{||d' - d_{gen}||_2}{||d'||_2} \times 100\%.$$

All types of training samples, as well as the average of relative error, are shown in Table 3.

The error is reduced by the largest number of times, in the case when the initial error is large ($d^{M_5,S_5} \rightarrow d^{M_5,S_2}$, $d^{M_5,S_5} \rightarrow d^{M_5,S_1}$, $d^{M_5,S_5} \rightarrow d^{M_2,S_2}$, $d^{M_5,S_5} \rightarrow d^{M_1,S_1}$). Herewith, it is well suppressed both in the case of a fixed discretization of the model and a variation in the step of the computational grid and when both discretizations are changed. In the case when initial error is small, the best options are those in which the discretization by the model h^M is fixed and the step of the calculation grid h varies ($d^{M_2,S_1} \rightarrow d^{M_1,S_1}$), as well as with a fixed h and the changes of the h^M ($d^{M_2,S_2} \rightarrow d^{M_2,S_1}$).

A possible explanation for this is the introduction of unnecessary noise when approximating the transition operator by NDM-net $G : d \rightarrow d'$. Then, in the case of a large initial error, processes suppressing numerical dispersion prevail,

and in the case when the initial error is small, the artifacts introduced by the neural network begin to dominate.

The worst result (a ratio of about one) was obtained when transition from seismograms calculated with relatively coarse sampling of the model and a fine computational grid to more accurate one ($d^{M_5 S_1} \rightarrow d^{M_2,S_1}$, $d^{M_5,S_1} \rightarrow D^{M_1,S_1}$).

Table 3. Training results on different type of the training dataset with size of 60% of the total sources number

d	d'	$L_2^{rel}(d,d')$ (Init)	$L_2^{rel}(d_{gen},d')$	$\frac{L_2^{rel}(d,d')}{L_2^{rel}(d_{gen},d')}$
d^{M_5,S_5}	d^{M_5,S_2}	65.83	27.42	**2.4**
	d^{M_5,S_1}	60.31	25.5	**2.36**
	d^{M_2,S_2}	51.58	28.33	1.82
	d^{M_1,S_1}	55.32	30.96	1.78
d^{M_5,S_2}	d^{M_5,S_1}	21.18	17.14	1.24
	d^{M_2,S_2}	33.76	19.18	1.76
	d^{M_2,S_1}	26.06	23.04	1.13
d^{M_5,S_1}	d^{M_2,S_1}	19.58	18.46	1.06
	d^{M_1,S_1}	25.26	20.18	1.25
d^{M_2,S_2}	d^{M_2,S_1}	27.49	17.19	1.59
	d^{M_1,S_1}	20.62	17.72	1.16
d^{M_2,S_1}	d^{M_1,S_1}	18.54	11.01	1.68

4.1 Two Stage of Training NDM-Net

In order to reduce the calculation time of seismograms, we implemented two stages of neural network training. The procedure was as follows: first, the NDM-net is trained to translate the most rough data d to more accurate d', and then using the generated data d_{gen}, we formed the new training dataset. Next, the NDM-net is trained for transition the generated data d^{gen} to the data that is the most free from numerical dispersion available d''.

For example, the performance of trained NDM-net corresponding to transition $d^{M_5,S_5} \rightarrow d^{M_5,S_1} \rightarrow d^{M_1,S_1}$ was $< \frac{L_2^{rel}(d^{M_5,S_5},d^{M_1,S_1})}{L_2^{rel}(d_{gen}^{M_1,S_1},d^{M_1,S_1})} >= 1.97$. This indicator showed that a single-stage transition provides lower performance than a two-stage one (Fig. 4).

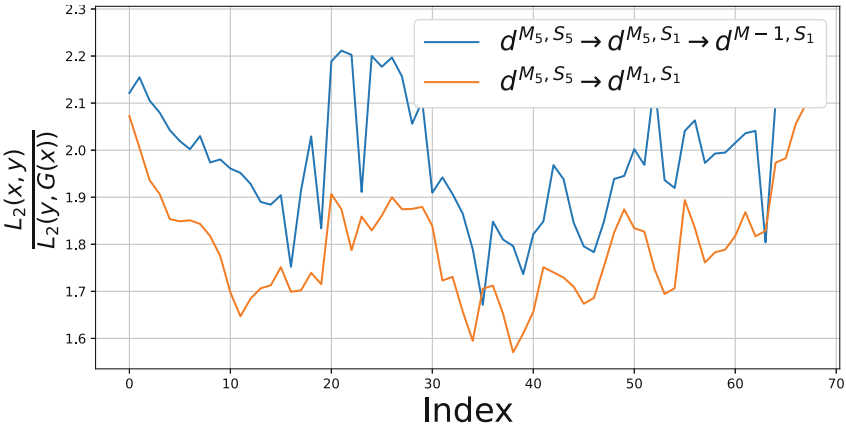


Fig. 4. Perfomance of Two-Stage NDM-net and Onet-stage NDM-net

4.2 Zero Offset

It is also important to build a zero-offset to verify our algorithm and visualize the profit of the NDM-net. Consider the transitions $d^{M_5,S_5} \rightarrow d^{M_1,S_1}$, $d^{M_5,S_5} \rightarrow d^{M_5,S_1}$ and $d^{M_5,S_1} \rightarrow d^{M_1,S_1}$ and illustrate in Fig. 5(a,b,c) the Initial Zero-Offset I^{M_5,S_5}, Zero Offset with coarse sampling of the model and a fine computational grid I^{M_5,S_1} and the accurate Zero Offset I^{M_1,S_1}. We also provided Zero Offsets in Fig. 5(d,e,f) obtained by applying a corresponding NDM-net to

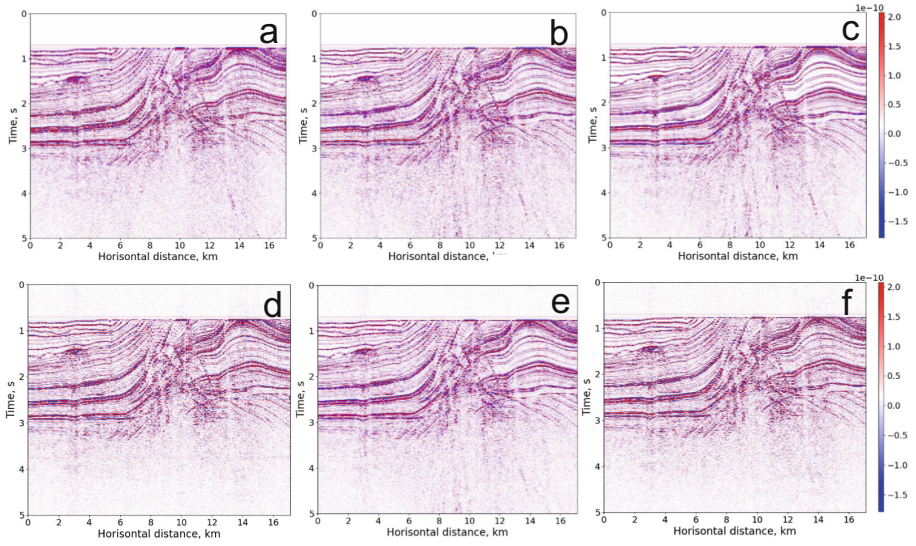


Fig. 5. Zero-Offset corresponding to seismograms modelled using a different discretizations (top row): a) I^{M_5,S_5}; b) I^{M_5,S_1}; c) I^{M_5,S_1} and the NDM-net prediction (bottom row): d) $I_{gen}^{M_5,S_5 \rightarrow M_1,S_1}$; e) $I_{gen}^{M_5,S_1 \rightarrow M_1,S_1}$; f) $I_{gen}^{M_5,S_5 \rightarrow M_5,S_1}$

translate from the roughest to the accurate seismograms $I_{gen}^{M_5,S_5 \to M_1,S_1}$, from seismograms devoid of numerical dispersion, but not the interface error, to the accurate seismograms $I_{gen}^{M_5,S_1 \to M_1,S_1}$ and from the roughest to seismograms without numerical dispersion, but with an interface error $I_{gen}^{M_5,S_5 \to M_5,S_1}$. The Fig. 6

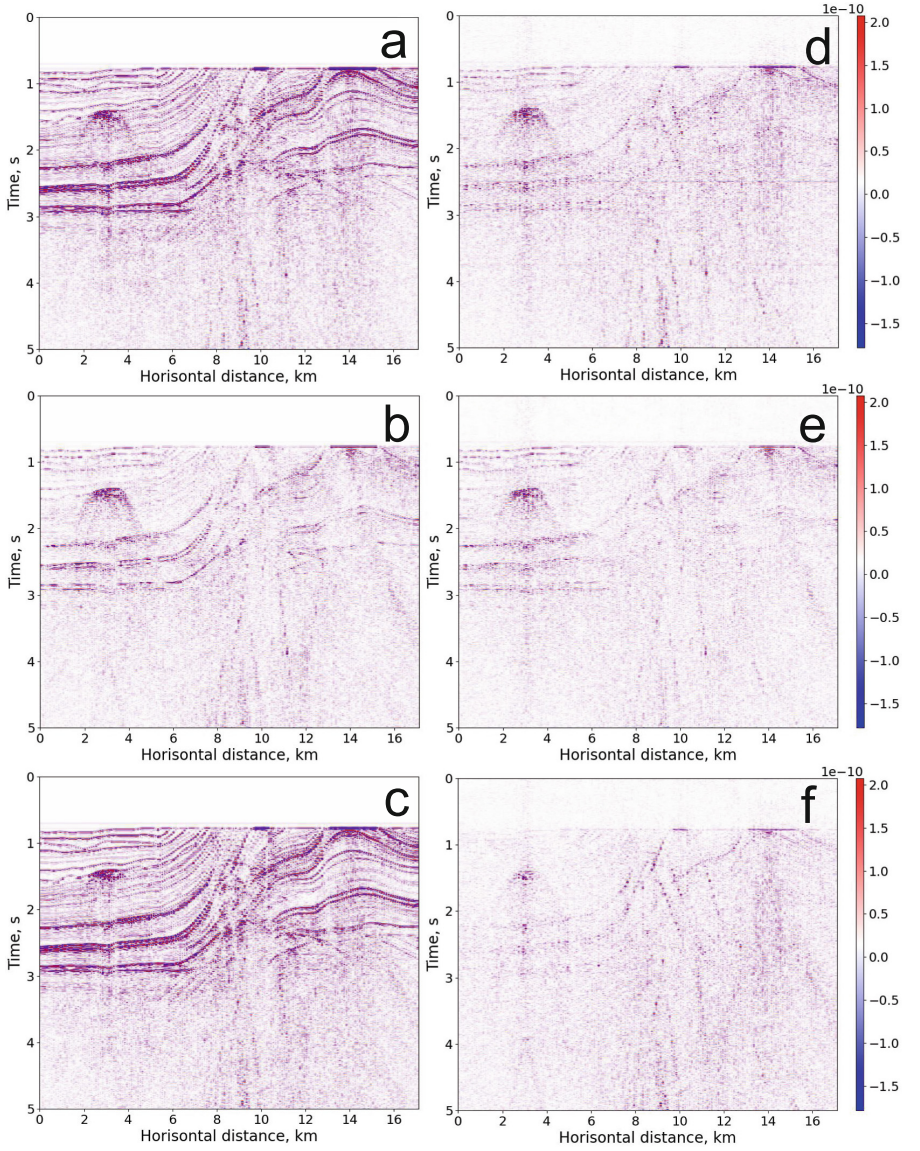


Fig. 6. Absolute error in the Zero-Offset corresponding to seismograms modelled using a different discretizations (left): a) $|I^{M_5,S_5} - I^{M_1,S_1}|$; b) $|I^{M_5,S_1} -^{M_1,S_1}|$; c) $|I^{M_5,S_5} - I^{M_5,S_1}|$ and the difference between the fine-grid data and NDM-net prediction (right): d) $|I_{gen}^{M_5,S_5 \to M_1,S_1} - I^{M-1,S_1}|$; e) $|I_{gen}^{M_5,S_1 \to M_1,S_1} - I^{M_1,S_1}|$; f) $|I_{gen}^{M_5,S_5 \to M_5,S_1} - I^{M_5,S_1}|$

also shows the absolute initial difference and the difference between the predicted data and fine-grid data.

The initial difference $L_2^{rel}(I^{M_5,S_5}, I^{M_1,S_1})$ is 79.7%, while the error between Zero-Offset obtained from generated seismograms by NDM-net $L_2^{rel}(I^{M_1,S_1}, I_{gen}^{M_1,S_1}) = 37.7\%$. In the case where only the error associated with the coarse sampling of the velocity model is suppressed, the final error was $L_2^{rel}(I^{M_5,S_1}, I_{gen}^{M_1,S_1}) = 28.9\%$, while the initial error was $L_2^{rel}(I^{M_5,S_1}, I^{M_1,S_1}) = 36.7\%$. When only the numerical dispersion is suppressed, the initial error was $L_2^{rel}(I^{M_5,S_5}, I^{M_5,S_1}) = 87.4\%$, while the error between the Zero Offset, which is obtained from predicted seismograms, and fine-grid seismograms was $L_2^{rel}(I^{M_5,S_5}, I_{gen}^{M_5,S_1}) = 31.9\%$.

This only confirms the conclusion that the NDM-net is more willing to suppress an error that has a large value, that is, a numerical dispersion. However, even with the largest variation in the model grids and computational grids, it is possible to mitigate the numerical dispersion by a factor of 2 for such a representation.

5 Conclusion

The NDM-net is presented, which was initially implemented to suppress numerical dispersion in seismic waves associated with a change in the computational grid. The paper expands the functionality of the NDM-net in case the grid according to the velocity model has also been changed. Thus, we have two types of numerical errors: numerical dispersion and interface error. It is shown that NDM-net effectively reduces both errors. However, the NDM-net suppresses the error well only if it is large enough (most often with a difference in the computational grids). Nevertheless, the results indicate the applicability of the network in the case of a coarsening of the model interface.

References

1. Brougois, A., Bourget, M., Lailly, P., Poulet, M., Ricarte, P., Versteeg, R.: Marmousi, model and data (05 1990). https://doi.org/10.3997/2214-4609.201411190
2. Chen, J.B.: A 27-point scheme for a 3D frequency-domain scalar wave equation based on an average-derivative method. Geophys. Prospect. **62**(2), 258–277 (2014)
3. Dumbser, M., Kaser, M.: An arbitrary high-order discontinuous galerkin method for elastic waves on unstructured meshes - ii. the three-dimensional isotropic case. Geophys. J. Int. **167**(1), 319–336 (2006). https://doi.org/10.1111/j.1365-246X.2006.03120.x
4. Fornberg, B.: The pseudospectral method: accurate representation of interfaces in elastic wave calculations. Geophysics **53**(5), 625–637 (1988)
5. Gadylshin, K., Vishnevsky, D., Gadylshina, K., Lisitsa, V.: Numerical dispersion mitigation neural network for seismic modeling. Geophysics **87**(3), T237–T249 (2022)
6. He, Z., Zhang, J., Yao, Z.: Determining the optimal coefficients of the explicit finite-difference scheme using the remez exchange algorithm. Geophysics **84**(3), S137–S147 (2019). https://doi.org/10.1190/geo2018-0446.1

7. Jia, F., Wong, W.H., Zeng, T.: Ddunet: dense dense u-net with applications in image denoising. IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), pp. 354–364 (2021)
8. Kostin, V., Lisitsa, V., Reshetova, G., Tcheverda, V.: Local time-space mesh refinement for simulation of elastic wave propagation in multi-scale media. J. Comput. Phys. **281**, 669–689 (2015)
9. Levander, A.: Fourth-order finite-difference p-s. Geophysics **53**, 1425–1436 (11 1988). https://doi.org/10.1190/1.1442422
10. Lisitsa, V., Podgornova, O., Tcheverda, V.: On the interface error analysis for finite difference wave simulation. Comput. Geosci. **14**(4), 769–778 (2010)
11. Lisitsa, V., Vishnevskiy, D.: Lebedev scheme for the numerical simulation of wave propagation in 3D anisotropic elasticity. Geophys. Prospect. **58**(4), 619–635 (2010). https://doi.org/10.1111/j.1365-2478.2009.00862.x
12. Lisitsa, V.: Dispersion analysis of discontinuous Galerkin method on triangular mesh for elastic wave equation. Appl. Math. Modell. **40**, 5077–5095 (2016). https://doi.org/10.1016/j.apm.2015.12.039
13. Lisitsa, V., Tcheverda, V., Botter, C.: Combination of the discontinuous Galerkin method with finite differences for simulation of seismic wave propagation. J. Comput. Phys. **311**, 142–157 (2016)
14. Liu, Y., Sen, M.K.: Acoustic VTI modeling with a time-space domain dispersion-relation-based finite-difference scheme. Geophysics **75**(3), A11–A17 (2010)
15. Moczo, P., Kristek, J., Vavrycuk, V., Archuleta, R.J., Halada, L.: 3D heterogeneous staggered-grid finite-difference modeling of seismic motion with volume harmonic and arithmetic averagigng of elastic moduli and densities. Bull. Seismol. Soc. Am. **92**(8), 3042–3066 (2002)
16. Moczo, P., Kristek, J., Galis, M.: The finite-difference modelling of earthquake motion: Waves and Ruptures. Cambridge University Press (2014)
17. Oliveira, D., Ferreira, R., Silva, R., Brazil, E.V.: Interpolating seismic data with conditional generative adversarial networks. IEEE Geosci. Remote Sens. Lett. **15**(12), 1952–1956 (2018)
18. O'Reilly, O., Lundquist, T., Dunham, E.M., Nordstrom, J.: Energy stable and high-order-accurate finite difference methods on staggered grids. J. Comput. Phys. **346**, 572–589 (2017). https://doi.org/10.1016/j.jcp.2017.06.030
19. Pleshkevich, A., Vishnevskiy, D., Lisitsa, V.: Sixth-order accurate pseudo-spectral method for solving one-way wave equation. Appl. Math. Comput. **359**, 34–51 (2019)
20. Puente, J., Kaser, M., Dumbser, M., Igel, H.: An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes iv. anisotropy. Geophys. J. Int. **169**(3), 1210–1228 (2007). https://doi.org/10.1111/j.1365-246X.2007.03381.x
21. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III, pp. 234–241. Springer International Publishing, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
22. Saad, O.M., Chen, Y.: Capsphase: capsule neural network for seismic phase classification and picking. IEEE Trans. Geosci. Remote Sens. **60**, 1–11 (2022)
23. Saenger, E.H., Gold, N., Shapiro, S.A.: Modeling the propagation of the elastic waves using a modified finite-difference grid. Wave Motion **31**, 77–92 (2000)

24. Sang, Y., Peng, Y., Lu, M., Li., L.: Seisdenet: an intelligent seismic data denoising network for the internet of things. J. Cloud Comp. **12**, 34 (2023)
25. Tromp, J., Komatitsch, D., Liu, Q.: Spectral-element and adjoint methods in seismology. Commun. Comput. Phys. **3**(1), 1–32 (2008)
26. Virieux, J.: P-SV wave propagation in heterogeneous media: velocity-stress finite-difference method. Geophysics **51**(4), 889–901 (1986)
27. Virieux, J., Calandra, H., Plessix, R.E.: A review of the spectral, pseudo-spectral, finite-difference and finite-element modelling techniques for geophysical imaging. Geophys. Prospect. **59**(5), 794–813 (2011). https://doi.org/10.1111/j.1365-2478.2011.00967.x
28. Vishnevsky, D., Lisitsa, V., Tcheverda, V., Reshetova, G.: Numerical study of the interface errors of finite-difference simulations of seismic waves. Geophysics **79**(4), T219–T232 (2014)
29. Wamriew, D., Pevzner, R., Maltsev, E., Pissarenko, D.: Deep neural networks for detection and location of microseismic events and velocity model inversion from microseismic data acquired by distributed acoustic sensing array. Sensors **21** (2021)
30. Yu, S., Ma, J., Wang, W.: Deep learning for denoising. Geophysics **84**, 1–107 (2019)
31. Yu, S., Ma, J., Zhang, X., Sacchi, M.D.: Interpolation and denoising of high-dimensional seismic data by learning a tight frame. Geophysics **80**, V119–V132 (2015)
32. Zhang, J., Verschuur, D.J.: Elastic wave propagation in heterogeneous anisotropic media using the lumped finite-element method. Geophysics **67**(2), 625–638 (2002)
33. Zhang, Y., Fisser, J., Gerritsma, M.: A hybrid mimetic spectral element method for three-dimensional linear elasticity problems. J. Comput. Phys. **433**, 110179 (2021). https://doi.org/10.1016/j.jcp.2021.110179