RESEARCH LETTER

Geoscience Letters





Simulating 2-D magnetotelluric responses using vector-quantized temporal associative memory artificial neural network-based approaches

Phongphan Mukwachi¹, Banchar Arnonkijpanich² and Weerachai Sarakorn^{1*}

Abstract

In this research, we explore the application of artificial neural networks, specifically the vector-quantized temporal associative memory (VQTAM) and VQTAM coupled with locally linear embedding (VQTAM-LLE) techniques, for simulating 2-D magnetotelluric forward modeling. The study introduces the concepts of VQTAM and VQTAM-LLE in the context of simulating 2-D magnetotelluric responses, outlining their underlying principles. We rigorously evaluate the accuracy and efficiency of both VQTAM variants through extensive numerical experiments conducted on diverse benchmark resistivity and real-terrain models. The results demonstrate the remarkable capability of VQTAM and VQTAM-LLE in accurately and efficiently predicting apparent resistivity and impedance phases, surpassing the performance of traditional numerical methods. This study underscores the potential of VQTAM and VQTAM-LLE as valuable computational alternatives for simulating magnetotelluric responses, offering a viable choice alongside conventional methods.

Keywords Magnetotelluric modeling, Neural networks, VQTAM, VQTAM-LLE

Introduction

In the traditional approach, magnetotelluric (MT) modeling is described using boundary-valued problems (BVP) of partial differential equations (PDEs). The approximated solutions are derived through classical numerical methods, including finite difference method (FDM) (Mackie et al. 1994; Siripunvaraporn et al. 2002; Tan et al. 2003; Tong et al. 2018), finite element method (FEM) (Wannamaker et al. 1987; Mogi 1996; Franke et al. 2007; Sarakorn 2017; Sarakorn and Vachiratienchai 2018; da Conceição Batista and Sampaio 2019), finite volume method (FVM) (Jahandari and Farquharson 2015; Du et al. 2016; Han et al. 2018; Wang et al. 2019; Khampichit and Sarakorn 2021), boundary element method (BEM) (Xu and Zhou 1997; Ren et al. 2014; Wittke and Tezkan 2014; Yang et al. 2019), hybrid mixed finite element domain decomposition method (HMFEDDM) Zyserman et al. (1999), and hybrid finite difference-finite element method (HBFDFEM) (Sarakorn and Vachiratienchai 2018). Each method has its own set of advantages and disadvantages. FDM is known for its speed and accuracy in handling simple models, while FEM, although slower, provides superior accuracy for more realistic models. Similarly, FVM and BEM offer comparable accuracy but require higher computational resources. HBFDFEM combines the strengths of both FDM and FEM, but its speed diminishes compared to FDM.



© The Author(s) 2024. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

^{*}Correspondence:

Weerachai Sarakorn

wsarakorn@kku.ac.th

¹ Department of Mathematics, Faculty of Science, Khon Kaen University, Khon Kaen 40002, Thailand

² Department of Imaging and Printing Technology, Faculty of Science, Chulalongkorn University, Phayathai Road, Bangkok 10330, Thailand

In recent years, artificial neural networks (ANN), computational models inspired by the human nervous system, have witnessed significant advancements in solving, simulating, and optimizing problems across diverse fields. Their applications in electromagnetic problems have been notable. For instance, in 2003, ANN were utilized to analyze magnetotelluric time-series data Manoj and Nagarajan (2003). Subsequently, during 2005-2006, ANN employing different learning paradigms were applied to determine subsurface layer structures from magnetotelluric data Khalil et al. (2006) and inversion of geo-electrical resistivity sounding data Singh et al. (2005). Developing convolutional neural networks (CNN) led to their use in MT inversion (Puzyrev 2019; Puzyrev and Swidinsky 2021). Furthermore, in 2021, ANN with multitask learning were employed in 2-D MT forward modeling to predict apparent resistivity and phase (Shan et al. 2021). Recent work in 2023 introduced Deep Learning MT forward modeling and its application in inversion Deng et al. (2023). Despite these advancements, improving the accuracy and efficiency of these ANN models remains challenging.

Among these ANN techniques, the vector-quantized temporal associative memory (VQTAM) method (Barreto and Araujo 2004) is one of the most efficient supervised neural network models. It was developed based on the self-organizing map (SOM) architecture (Kohonen 2013), an unsupervised clustering method that allows VQTAM to perform both recognition and clustering simultaneously. Originally, SOM was an unsupervised neural algorithm utilizing competitive learning to represent spatial neighborhood relationships within unlabeled datasets, designed to uncover topological structures within multidimensional data spaces. SOM has found applications in diverse fields, such as pattern classification (Chetchotsak et al. 2015), machine vision (Wongsriworaphon et al. 2015), image compression (Arnonkijpanich et al. 2011), data visualization (Arnonkijpanich et al. 2010), and even kinematic modeling of parallel manipulators (Limtrakul and Arnonkijpanich 2019). Subsequently, an improved version of VQTAM, known as vector-quantized temporal associative memory with locally linear embedding (VQTAM-LLE) technique, was introduced to enhance its accuracy (Wongsriworaphon et al. 2015; Limtrakul and Arnonkijpanich 2019).

This study employed VQTAM and VQTAM-LLE techniques alongside the traditional FE method to simulate 2-D magnetotelluric responses, specifically apparent resistivity and phases. In addition, the possibility of our technique to generate additional reliable data for the restricted cases where the MT survey cannot be done due to some limitations, such as the topographic zone in the pre-inversion process, was presented. Various benchmark resistivity models and real-terrain models were selected. The training dataset was generated using the Finite Element (FE) approach (Sarakorn 2017), and another dataset of responses was estimated using these two neural networks. The accuracy and efficiency of both VQTAM and VQTAM-LLE were thoroughly examined, analyzed, and discussed.

2-D magnetotelluric modeling

In the traditional way, MT modeling describes the natural electromagnetic (EM) fields interacting with the Earth. The EM fields act like a plane wave with harmonic diffusion, the displacement currents is neglected, and the time-dependent is assumed to be $e^{-i\omega t}$, where ω is the angular frequency. In the case that the strike direction is *x*-direction, the electrical conductivity σ is therefore varied in only the *y* and *z*-directions, i.e., $\sigma = \sigma(y, z)$, the boundary-valued problem of the second-order partial equations for the 2-D MT problem is expressed by

$$\frac{\partial}{\partial y} \left(\mu^{-1} \alpha \frac{\partial G_x}{\partial y} \right) + \frac{\partial}{\partial z} \left(\mu^{-1} \alpha \frac{\partial G_x}{\partial z} \right) + \beta G_x, = 0.$$
(1)

The notations α , β and G for two polarizations: E- and H-polarizations are denoted as follows:

E-Polarization :
$$\alpha = 1$$
, $\beta = i\omega\sigma$, $G = E$, (2)

H-Polarization :
$$\alpha = \frac{1}{\sigma}, \quad \beta = i\omega, \quad G = H,$$
(3)

where E_x and H_x are the strike aligned electric and magnetic fields, respectively, μ is the magnetic permeability in free space and $\mu = \mu_0 = 4\pi \times 10^{-7} (Vs/Am)$. The computational domain for the 2-D case is considered as a region $\Omega = \Omega_1 \cup \Omega_2 \cup \Gamma \cup \Gamma^{int} \subset \mathbb{R}^2$.

As shown in Fig. 1, the subregion Ω_1 is defined as the air layer, the subregion Ω_2 is defined as the Earth layer, the boundary Γ is the outer boundary, and Γ^{int} is the air–earth interface. Crossing Γ^{int} , the electrical conductivity σ (or the electrical resistivity ρ) is discontinuous. The partial differential equations (1) are subjected to the Dirichlet boundary conditions:

$$G = G_0(y, z) \text{ on } \Gamma, \tag{4}$$

where $G_0(y, z)$ is obtained analytically or numerically from 1-D MT forward problem. Equation (1) can be solved analytically when its model structure is simple. In contrast, solving (1) with various numerical techniques are more suitable (Wannamaker et al. 1987; Key and Weiss 2006; Franke et al. 2007; Sarakorn 2017; Tong et al.



Fig. 1 The computational domain $\Omega = \Omega_1 \cup \Omega_2 \cup \Gamma \cup \Gamma^{int}$ for 2-D MT case

2018; Sarakorn and Vachiratienchai 2018; Khampichit and Sarakorn 2021). When the electric and magnetic fields are obtained for each period, the impedances, the apparent resistivity, and phases at each sample station are calculated by (5)-(7), respectively,

$$Z_{ij} = \frac{E_i}{H_j}, \ Z_{ii} = Z_{jj} = 0,$$
 (5)

$$\rho_{ij}^a = \frac{1}{\omega\mu} |Z_{ij}|^2,\tag{6}$$

$$\phi_{ij} = \arg(Z_{ij}),\tag{7}$$

where i, j = x, y.

The VQTAM and VQTAM-LLE techniques VQTAM as SOM-based neural network

The self-organizing map (SOM) is a neural network based on unsupervised learning. SOM is categorized as an unsupervised version of a vector quantization technique with topology preservation among the neurons. A SOM consists of neurons arranged on a regular two-dimensional grid \mathcal{A} . Each neuron *i* on the grid corresponds to a location of the prototype vector in the input space, i.e., $w_i^{in} \in \mathbb{R}^n$. This feature is used to connect both modules. At each time of prototype adjustment during SOM training, a prototype that belongs to the winner neuron i^* and its neighbor neurons are updated to approach a fed input sample. Then, an elastic net derived from linking between prototypes is unfolded and spanned over input space. This way, SOM can represent data faithfully using a set of prototypes, in which these learned prototypes are considered as centers of subclusters in input space. Thus, SOM is suitable for the task of data clustering. Interestingly, the prototype distribution still preserves neighborhood-based topology and ordering in the 2D neural grid. In addition, another goal of SOM is to transform *n*-dimensional patterns in input space into a two-dimensional array of neurons. VQTAM was proposed to increase the efficiency of SOM at mapping patterns in input space into output space. The space of the outputs must be included as the third module in the structure of the classical SOM. Thereby, a weight vector in output space, $w_i^{out} \in \mathbb{R}^z$ is also attached to each w_i^{in} which belongs to neuron *i*. The underlying idea of VQTAM is that, during training on both spaces, the SOM algorithm adjusts the module of prototypes $\{w_i^{in}, w_i^{out}\}$ in order to represent the data manifold on both input and output spaces. Note that a prototype vector w_i^{in} corresponds to w_i^{out} , which is similar to the dataset in input space x^{in} that corresponds to the target set in output space x^{out} . Then, a mapping from w_i^{in} to w_i^{out} can be considered as forward mapping from the input space to the output space. Note that this mapping explicitly performs approximator/predictor like supervised ANN based on multilayer perceptron (MLP) Wongsriworaphon et al. (2015).

Because MT modeling can be considered as nonlinear system identification, the neural networks for nonlinear MT system can be constructed by training input/output data of the system. In this work, the input vector x^{in} contains values from periods and stations, while the output vector x^{out} consists of apparent resistivity and phases. VQTAM is used to map patterns in input space to desired outputs. The training vector x fed into the VQTAM and the module of prototypes w_i are defined as follows:

$$\mathbf{x} = \begin{pmatrix} x^{in} \\ x^{out} \end{pmatrix}$$
 and $\mathbf{w}_i = \begin{pmatrix} w_i^{in} \\ w_i^{out} \end{pmatrix}$. (8)

During training process, the winning neuron i^* is determined by using only the portion corresponding to x^{in} :

$$i^* = \underset{i \in \mathcal{A}}{\operatorname{argmin}} \Big\{ \Big\| x^{in} - w_i^{in} \Big\| \Big\}.$$
(9)

Note that, each neuron *i* is arranged on a 2D lattice structure A such that prototype vectors or weight vectors in input and output spaces, i.e., w_i^{in} and w_i^{out} , are associated to each neuron *i*. Then, the learning rules are

$$\Delta w_i^{in} = a(m)h(i^*, i; m) \left[x^{in} - w_i^{in} \right], \tag{10}$$

$$\Delta w_i^{out} = a(m)h(i^*, i; m) \left[x^{out} - w_i^{out} \right]. \tag{11}$$

Thus, the weight vectors on both input and output spaces are updated by

$$w_i^{in} \longleftarrow w_i^{in} + a(m)h(i^*, i; m) \left[x^{in} - w_i^{in} \right], \qquad (12)$$

$$w_i^{out} \longleftarrow w_i^{out} + a(m)h(i^*, i; m) [x^{out} - w_i^{out}], \quad (13)$$

Algorithm 1 Algorithm of VQTAM scheme

Training part

used for the testing stage, such that each vector of the test set, x_{test}^{in} , is used for the winner assignment: $i^* = \underset{i \in \mathcal{A}}{\operatorname{argmin}} \{ \|x_{test}^{in} - w_i^{in}\| \}$. Afterward, a weight vector $w_{i^*}^{out}$, which corresponds to the weight of the winner neuron $w_{i^*}^{in}$ can be used as an estimation of the outputs. This arrangement means that the approximate values of apparent resistivity and phase of x_{test}^{in} are obtained by using the following equation:

$$x_{test}^{out} = w_{i^*}^{out}.$$
 (15)

The algorithm of the VQTAM technique is summarized as Algorithm 1.

1: Set the input data X and the weight vector w_i as $\left(\begin{array}{c} x^{in} \end{array} \right) \qquad \left(\begin{array}{c} x^{in} \end{array} \right)$

 $\mathbf{x} = \begin{pmatrix} x^{in} \\ x^{out} \end{pmatrix} \quad \text{and} \quad \mathbf{w}_i = \begin{pmatrix} w^{in}_i \\ w^{out}_i \end{pmatrix}$

, respectively. 2: Find the index of winner neuron i^* according to x^{in} by

$$i^* = \operatorname{argmin}_{i \in \mathcal{A}} \{ \parallel x^{in} - w_i^{in} \parallel \}$$

3: Adjust the weight vectors of all neurons by

$$\begin{split} & w_i^{in} \Longleftarrow w_i^{in} + a(m)h\left(i^*, i; m\right) \left[x^{in} - w_i^{in}\right] \\ & v_i^{out} \twoheadleftarrow w_i^{out} + a(m)h\left(i^*, i; m\right) \left[x^{out} - w_i^{out}\right] \end{split}$$

4: Do until the termination condition is met.

Testing part

- 1: Feed the vector x_{test}^{in}
- 2: Find the winner neuron i^* according to x_{test}^{in} by

$$i^* = \operatorname*{argmin}_{i \in \mathcal{A}} \left\{ \left\| x_{test}^{in} - w_i^{in} \right\| \right\}$$

3: The output x_{test}^{out} can be estimated by

$$x_{test}^{out} = w_{i^*}^{out}$$

where 0 < a(m) < 1 is the learning rate in epoch *m*. $h(i^*, i; m) = \exp\left(-\frac{\|r_i - r_{i^*}\|^2}{2b^2(m)}\right)$ is a Gaussian neighborhood function where r_i and r_{i^*} are locations of neurons *i* and i^* in the lateral lattice space. Note that r_i and r_{i^*} correspond to positions of w_i and w_i*, respectively. The variables a(m) and b(m) are the exponential decay functions, which are used to guarantee that the weight vectors converge to steady states. These functions are given by

$$a(m) = a_0 \left(\frac{a_M}{a_0}\right)^{\frac{m}{M}} \quad \text{and} \quad b(m) = b_0 \left(\frac{b_M}{b_0}\right)^{\frac{m}{M}},$$
(14)

where a_0 and b_0 are initial values, while a_M and b_M are final ones after training M epochs. After training, we obtain the set of weight vectors $\{w_i^{in}, w_i^{out}\}$ belonging to each neuron *i*. Then, a trained VQTAM network can be

Improvement of classical VQTAM

Note that the estimates of x_{test}^{out} produced by Eq. (15) have low reliability because of a high error value. The number of neurons must be sufficiently large to obtain a high-accuracy prediction. This arrangement leads to a time-consuming problem during the training process. An efficient strategy to defeat this restriction is to train VQTAM with a few neurons to get the data topology. Then, an autoregressive (AR) model and locally linear embedding (LLE) technique Roweis and Saul (2000) can be used to improve the accuracy of estimates derived from VQTAM. However, in practice, the AR model relying on the pseudo-inverse derived from the normal equations becomes problematic due to a pseudo-inverse matrix being close to singular, leading to a badly scaled optimization problem. In addition, the

AR model is based on linear mapping and used to fit the linearly distributed data. For the LLE algorithm, this method is one of the most widely used nonlinear dimensionality reduction techniques. LLE reduces the dimensionality of high-dimensional data while preserving local geometries in a low-dimensional representation of the original data. In contrast to the AR model, LLE preserves the local relationships of the data manifold. LLE is also categorized as nonlinear mapping, which is appropriate to represent the nonlinear data. Further, as suggested in Wongsriworaphon et al. (2015); Limtrakul and Arnonkijpanich (2019), VQTAM based on LLE gives an accurate and robust prediction better than the result of VQTAM combined with the AR model. Thus, in this work, only the LLE technique is chosen to improve classical VQTAM. The outline of the LLE application to our work is composed of three steps. In the first step, the k winner neurons of x_{test}^{in} are defined as $w_{i_1^*}^{i_n}, w_{i_2^*}^{i_n}, \ldots, w_{i_k^*}^{i_n}$, where

$$i_{1}^{*} = \underset{i \in \mathcal{A}}{\operatorname{argmin}} \left\{ \left\| x_{test}^{in} - w_{i}^{in} \right\| \right\},$$

$$i_{2}^{*} = \underset{i \in \mathcal{A}, i \notin \{i_{1}^{*}\}}{\operatorname{argmin}} \left\{ \left\| x_{test}^{in} - w_{i}^{in} \right\| \right\},$$

$$\vdots$$

$$i_{k}^{*} = \underset{i \in \mathcal{A}, i \notin \{i_{1}^{*} \dots i_{k-1}^{*}\}}{\operatorname{argmin}} \left\{ \left\| x_{test}^{in} - w_{i}^{in} \right\| \right\}.$$
(16)

Then, each vector x_{test}^{in} in the test set can be written as a linear combination of its *k* nearest neural weights:

$$x_{test}^{in} = c_1 w_{i_1^*}^{in} + c_2 w_{i_2^*}^{in} + \dots + c_k w_{i_k^*}^{in}.$$
 (17)

In the second step, for each vector x_{test}^{in} , the coefficients $C = \{c_l \mid l = 1, ..., k\}$ of the linear combination are calculated to obtain the best representation of the local geometries of the input space. In this step, the LLE attempts to reconstruct the *k* coefficients. This action corresponds to minimizing the reconstruction error that appears in the cost function

$$\phi(C) = \left\| x_{test}^{in} - \sum_{l=1}^{k} c_l w_{i_l^*}^{in} \right\|.$$
(18)

This representation function is intended to minimize the cost function under two constraints. In the first constraint, each vector x_{test}^{in} is reconstructed only from its k nearest winner neurons, thus $c_l = 0$, if $w_{l_l}^{in}$ is not a neighbor of x_{test}^{in} . The second condition is that the coefficients sum to one, i.e., $\sum_{l=1}^{k} c_l = 1$. The optimal matrix C can be computed by solving a constrained least-squares problem. In the third step, the estimates of x_{test}^{out} are improved by using the coefficients from the previous step:

$$x_{test}^{out} = c_1 w_{i_1}^{out} + c_2 w_{i_2}^{out} + \ldots + c_k w_{i_k}^{out}.$$
 (19)

The algorithm of VQTAM-LLE technique is summarized as Algorithm 2. We can see from Algorithms 1 and 2 that the VQTAM and VQTAM-LLE have the same training stage. Therefore, the number of iterations used to reach the stopping criteria is the same. However, both methods use a different number of iterations and consume different CPU time in the testing stage. Training part

1: Set the input data x and the weight vector w_i as

$$\mathbf{x} = \begin{pmatrix} x^{in} \\ x^{out} \end{pmatrix}$$
 and $\mathbf{w}_i = \begin{pmatrix} w^{in}_i \\ w^{out}_i \end{pmatrix}$

respectively.

2: Find the index of winner neuron i^* according to x^{in} by

 $i^* = \operatorname{argmin}_{i \in A} \{ \| x^{in} - w^{in}_i \| \}$

3: Adjust the weight vectors of all neurons by

$$w_i^{in} \Leftarrow w_i^{in} + a(m)h(i^*, i; m)[x^{in} - w_i^{in}]$$

$$w_i^{out} \leftarrow w_i^{out} + a(m)h(i^*, i; m)[x^{out} - w_i^{out}]$$

4: Do until the termination condition is met.

Testing part

- 1: Feed the vector $\boldsymbol{x}_{test}^{in}$
- 2: Find the k winner neurons $w_{i_1^n}^{in}, w_{i_2^n}^{in}, \cdots, w_{i_k^n}^{in}$ according to x_{test}^{in} by

$$\begin{split} i_1^* &= \operatorname{argmin}_{i \in \mathcal{A}} \{ \parallel x_{test}^{in} - w_i^{in} \parallel \}, \\ i_2^* &= \operatorname{argmin}_{i \in \mathcal{A}, i \notin \{i^*\}} \{ \parallel x_{test}^{in} - w_i^{in} \parallel \}, \end{split}$$

$$i_k^* = \operatorname{argmin}_{i \in \mathcal{A}, i \notin \{i_1^*, \cdots, i_{k-1}^*\}} \{ \parallel x_{test}^{in} - w_i^{in} \parallel \}$$

3: Find the coefficients $C = \{c_l | l = 1, 2, ..., k\}$ by minimization of the representation function

$$\phi(C) = \| x_{test}^{in} - \sum_{l=1}^{k} c_l w_{i_l^*}^{in} \|$$

where $c_l=0$ if $w_{i_l^*}^{in}$ is not a neighbor of x_{test}^{in} and $\sum_{l=1}^k c_l=1$

4: The output x_{test}^{out} can be estimated by

$$x_{test}^{out} = c_1 w_{i_1}^{out} + c_2 w_{i_2}^{out} + \dots + c_k w_{i_k}^{out}$$

Applications of VQTAM and VQTAM-LLE techniques for simulating MT responses

As described above, both VQTAM and VQTAM-LLE have been applied to a wide range of tasks such as function approximation Barreto and Araujo (2004), computer vision Wongsriworaphon et al. (2015), robot modeling Limtrakul and Arnonkijpanich (2019). These algorithms have shown promising results in various applications and are considered to be effective methods for dynamic modeling and pattern recognition. In the MT problems, ANN have been employed for both forward and inverse MT problems, as evidenced by previous studies (Singh et al. 2005; Khalil et al. 2006; Puzyrev 2019; Shan et al. 2021; Puzyrev and Swidinsky 2021; Deng et al. 2023). However, the application of VQTAM and VQTAM-LLE to MT modeling still needs to be explored. Exploring the potential of these VQTAM techniques in the context of MT problems presents an intriguing and challenging avenue of research. Despite their successful track record in other domains, their adaptation and performance in the specific domain of MT modeling have yet to be thoroughly investigated.

The process of utilizing VQTAM approaches to estimate 2-D magnetotelluric (MT) responses begins with the preparation of responses for the training dataset. Let us consider the sampling of electromagnetic (EM) periods $T_i \in [A, B]$ for $i = 1, 2, ..., N_T$, where N_T represents the total number of sampling periods during the training stage. Typically, the EM period interval for MT falls $[A, B] = [10^{-1}, 10^4]$ seconds. within Additionally, sampling stations on the Earth's surface are denoted as $S_i = (y_i, z_i) \in [Y_l, Y_r] \times [Z_a, Z_b] \subset \mathbb{R}^2$ within the training domain, where $j = 1, 2, ..., N_S$ and N_S signifies the total number of sampling stations. It is important to note that the scale for (y_i, z_i) is in meters. In this context, the z-direction is considered as positive downward, meaning that $z_i = 0$ represents sea level. z_i on the Air–Earth interface can be positive or negative depending on the presence of terrain.

These data, along with the resistivity structure model, serve as inputs for the finite element (FE) approach utilizing quadrilateral elements Sarakorn (2017). The outputs consist of matrices representing apparent resistivity

$$\rho_{xy}^{a} = [\rho_{ij}^{a_{xy}}]_{N_{S} \times N_{T}}, \quad \rho_{yx}^{a} = [\rho_{ij}^{a_{yx}}]_{N_{S} \times N_{T}}, \tag{20}$$

and phase

$$\phi_{xy} = [\phi_{ij}^{xy}]_{N_S \times N_T}, \quad \phi_{yx} = [\phi_{ij}^{yx}]_{N_S \times N_T}.$$
 (21)

The input vector $x^{in} = [x_{pq}^{in}]_{N_d \times 3}$, where $N_d = N_T \cdot N_S$, is constructed as

$$x^{in} = \begin{bmatrix} T_1 & y_1 & z_1 \\ T_1 & y_2 & z_2 \\ \vdots & \vdots & \vdots \\ T_1 & y_{N_S} & z_{N_S} \\ T_2 & y_1 & z_1 \\ T_2 & y_2 & z_2 \\ \vdots & \vdots & \vdots \\ T_2 & y_{N_S} & z_{N_S} \\ \vdots & \vdots & \vdots \\ T_{N_T-1} & y_{N_S} & z_{N_S} \\ T_{N_T} & y_1 & z_1 \\ T_{N_T} & y_2 & z_2 \\ \vdots & \vdots & \vdots \\ T_{N_T} & y_{N_S} & z_{N_S} \end{bmatrix},$$
(22)

 $x^{out} = \begin{bmatrix} \vdots & \vdots \\ \rho_{N_{S1}}^{a_{xy}} & \phi_{N_{S1}}^{xy} \\ \rho_{12}^{a_{xy}} & \phi_{12}^{xy} \\ \rho_{22}^{a_{xy}} & \phi_{22}^{xy} \\ \vdots & \vdots \\ \rho_{N_{S2}}^{a_{xy}} & \phi_{N_{S2}}^{xy} \\ \vdots & \vdots \\ \rho_{1N_{T}}^{a_{xy}} & \phi_{1N_{T}}^{xy} \\ \rho_{2N_{T}}^{a_{xy}} & \phi_{2N_{T}}^{xy} \\ \vdots & \vdots \\ \rho_{N_{SN_{T}}}^{a_{xy}} & \phi_{N_{SN_{T}}}^{xy} \end{bmatrix}$

 $\begin{array}{c}\rho_{11}^{a_{xy}}\\\rho_{21}^{a_{xy}}\end{array}$

 $\phi_{11}^{xy}\\\phi_{21}^{xy}$

for *E*-polarization and

and the vector $x^{out} = [x_{pq}^{out}]_{N_d \times 2}$



Fig. 2 The resistivity structures of COMMEMI2D-2 model. The red triangles are the locations of $N_S = 21$ stations

(23)

$$x^{out} = \begin{bmatrix} \rho_{11}^{a_{yx}} & \phi_{11}^{yx} \\ \rho_{21}^{a_{yx}} & \phi_{21}^{yx} \\ \vdots & \vdots \\ \rho_{N_{S1}}^{a_{yx}} & \phi_{N_{S1}}^{yx} \\ \rho_{12}^{a_{yx}} & \phi_{12}^{yx} \\ \rho_{22}^{a_{yx}} & \phi_{22}^{yx} \\ \vdots & \vdots \\ \rho_{N_{S2}}^{a_{yx}} & \phi_{N_{S2}}^{yx} \\ \vdots & \vdots \\ \rho_{N_{S2}}^{a_{yx}} & \phi_{N_{S2}}^{yx} \\ \vdots & \vdots \\ \rho_{2N_{T}}^{a_{yx}} & \phi_{2N_{T}}^{yx} \\ \vdots & \vdots \\ \rho_{N_{SNT}}^{a_{yx}} & \phi_{N_{SNT}}^{yx} \end{bmatrix}$$
(24)



For the testing phase, a different set of periods $T_i^{test} \in [A, B]$ for $i = 1, 2, ..., N_T^{test}$, where N_T^{test} represents the total number of testing periods, and $S_j^{test} = \left(y_j^{test}, z_j^{test}\right) \in [Y_l, Y_r] \times [Z_a, Z_b] \text{ for } j = 1, 2, ..., N_S^{test},$ where N_S^{test} is the total number of testing stations, are utilized as input. The construction of the input vector x_{test}^{in} mirrors the training stage, although the size of matrices may vary. Ultimately, the estimated MT responses obtained through the VQTAM or VQTAM-LLE approaches are stored in x_{test}^{out} with dimensions $(N_T^{test} \cdot N_S^{test}) \times 2$ for each polarization. Note that for all experiments in this study, we keep $S_i^{test} = S_i$ and $N_S^{test} = N_S$. This consistency is crucial for making valid comparisons among the MT sites within the specified domain.

Numerical experiments

In this section, we present a comprehensive evaluation of the VQTAM and VQTAM-LLE approaches, focusing on their performance, accuracy, and reliability. For these experiments, we employ three distinct models. The initial benchmark model is derived from the COMMEMI project (Zhdanov et al. 1997). The second model represents a modification of another original COMMEMI project, while the final model incorporates topographic zone. To assess the accuracy of the developed VQTAM and VQTAM-LLE methods, the mean absolute percentage error (MAPE) that measures the prediction accuracy of a predicting method defined by



Fig. 3 The graph between num. of iterations during the training stage of both VQTAM and VQTAM-LLE methods and various num. of neurons

$$MAPE = \frac{1}{N_T^{test}N_S^{test}} \sum_{j=1}^{N_T^{test}} \sum_{i=1}^{N_S^{test}} \left| \frac{d_{ij} - Q_{ij}}{d_{ij}} \right| \times, 100, \quad (25)$$

where d_{ij} is the referenced response obtained from the FEM scheme (Sarakorn 2017) and \hat{d}_{ij} is the predicted data obtained by the VQTAM or VQTAM-LLE methods is utilized. This evaluation involves the apparent resistivity and phase data for both E- and H-polarizations. Furthermore, we gauge the efficiency of our approaches by analyzing overall CPU times and the number of iterations during the training stage. All experiments are made on MacBook Pro(M2) with memory 16GB and CPU 10 cores.

COMMEMI2D-2 model

The resistivity structures of the COMMEMI2D-2 model (Zhdanov et al. 1997) are illustrated in Fig. 2. The Earth's surface is flat in this representation, with z = 0 denoting the air–earth interface and sea level. We have positioned $N_S = 21$ stations at intervals of 4 km, covering a range of [-50, 50] km on the Earth's surface. Each station is represented as $S_i = (y_j, z_j) = (-50 + 4(j - 1) \text{ km}, 0 \text{ km})$, where *j* ranges from 1 to N_S . These station locations are marked by red triangles in Fig. 2. For our testing purposes, we have selected 61 periods of electromagnetic (EM) waves within the range of [1, 1000] seconds, spaced at intervals of 0.05 on a logarithmic scale.

Several critical parameters influence the performance of our neural networks including the number of training datasets, the number of neurons, and the stopping criteria used to determine the stability of the objective



Fig. 4 The resistivity distributions of COMMEMI2D-1-like model. The red triangles are the locations of 21 stations



Fig. 5 The graph displays MAPE values with respect to the ratios between training and testing data for VQTAM (left) and VQTAM-LLE (right) approaches

functions. The number of training datasets plays a pivotal role in the accuracy and efficiency of the neural networks. While a more extensive training dataset typically results in higher algorithm accuracy, an extensive dataset can reduce overall performance. Note that the parameter k in Algorithm 2 is fixed as k = 4 in all experiments.

In this model, we utilize $N_T = 31$ periods selected from a total of 61 periods within the specified interval, defined as $T_i = 10^{0.1(i-1)}$ s, where *i* ranges from 1 to N_T . These periods are employed for simulating the magnetotelluric (MT)

data and as input training data for our algorithms. It is important to note that these period selections remain consistent across all cases for this model. The impact of varying training data will be explored in the subsequent model.

The second parameter we consider is the number of neurons allocated for each input dataset. Utilizing the SOM architecture, the neural network structure forms a 2D lattice, denoted as $N_n \times N_n$ neurons. In our experimentation, we configure the design with N_n set to 10, 15, 20, 25, 30, 35, and 40.



Fig. 6 The graph on the left illustrates the number of iterations, while the graph on the right depicts CPU time, both with respect to the ratios between training and testing data used by VQTAM and VQTAM-LLE approaches

Lastly, the third factor involves setting the stopping criteria to determine the stability of objective functions a(m) and b(m) in equation (14), which dictate the termination of the training step for the VQTAM and VQTAM-LLE approaches before the commencement of the testing phase. Specifically, the stability of the objective function is achieved when the values of these functions do not fluctuate beyond 3% to 0.05%. By adjusting these key parameters, we assess the accuracy of both methods in terms of MAPE. The resulting MAPE values for the VQTAM and VQTAM-LLE approaches, employing a stopping criterion for objective functions set at 1%, are summarized in Table 1.

In our numerical experiments, it is evident that increasing the number of neurons leads to a decrease in MAPE across all MT responses, while a reduction in the number of neurons significantly raises MAPE levels. Specifically, employing 40×40 square grid configuration ensures that the MAPE for all MT responses remains below 5%. The VQTAM-LLE method displays a smaller MAPE than the conventional VQTAM approach in this scenario. This trend persists when the stopping criterion is reduced to 0.5%, as illustrated in Table 2.

Similar trends are observed in cases where the stopping criteria are set at 0.1% and 0.05% (refer to Tables 3 and 4, respectively). In these instances, MAPE levels below 5% are consistently achieved with neuron configurations ranging from 30×30 to 40×40 . Notably, the choice of stopping criteria does not significantly impact the MAPE levels within this range.

To consider efficiency, the relationship between the total number of iterations during the training stage and varying numbers of neurons is depicted in Fig. 3. Decreasing the stopping criteria increases the number

of iterations, while an increase in the number of neurons results in higher iteration counts. To regard CPU time, Table 5 demonstrates that a higher number of neurons leads to increased CPU usage for both VQTAM and VQTAM-LLE approaches across all stopping criteria. The CPU time increases when the stopping criteria decrease across all neural network scenarios. Additionally, when employing the same number of neurons, VQTAM scheme exhibits shorter CPU times than VQTAM-LLE scheme.

COMMEMI2D-1-like model

For this model, we configured it according to the COM-MEMI2D-1 model (Zhdanov et al. 1997), as illustrated in Fig. 4. Originally, the subsurface anomaly size was 0.5 km \times 2 km. However, it was enlarged to 10 km \times 2 km for a larger scale. Twenty-one stations were strategically placed for measuring responses, spanning a range of [- 10, 10] km on the Earth's surface with 1 km spacing. A red triangle denotes each station. The EM wave is comprised of 81 periods ranging from [0.01, 100] s, spaced logarithmically at 0.05 intervals.

During the training process, the stopping criteria for changing the levels of a(m) and b(m) in equations (14) were set at 0.1%. The total number of neurons was specified as 1600 neurons. The purpose of employing this model was to assess the efficiency of VQTAM and VQTAM-LLE concerning the variation in the ratio between training data sampling periods (N_T) and testing data sampling periods (N_T), in which the total number of stations for two phases, N_S and N_S^{test} , were set to be equal. The ratios considered in this experiment were 14 : 67, 21 : 60, 41 : 40, and 58 : 23, respectively.



Fig. 7 The profiles that pass through some part of Northeast (P1: left bottom) and Kanchanaburi province (P2: right bottom) of Thailand

The MAPE obtained by VQTAM and VQTAM-LLE for each response type concerning these ratios is depicted in Fig. 5. An increase in training data led to a reduction in MAPE for both approaches. The highest MAPE occurred at the smallest ratio of 14 : 67, whereas the minimum MAPE was at the most significant ratio of 58 : 23. VQTAM consistently exhibited lower MAPE compared to VQTAM-LLE for all response types. Notably, the



Fig. 8 The resistivity models for the P1 (left) and the P2 (right) profiles. The red triangles are assumed as the MT sites for modeling



Fig. 9 The generated MT responses using FE method of the P1 (left) and the P2 (right) profiles for training and testing data



Fig. 10 The MAPE obtained by the VQTAM and VQTAM-LEE schemes for P1 (left) and P2 (right) profiles

Num. of neurons	Mean absolute percentage error (MAPE)								
	VQTAM				VQTAM-LLE				
	ρ^a_{xy}	$ ho_{yx}^a$	φ _{xy}	φ _{yx}	ρ^a_{xy}	$ ho_{yx}^a$	φ _{xy}	ф _{yx}	
10 × 10	92.55	148.30	26.69	43.12	231.71	738.93	110.63	52.74	
15 × 15	80.73	113.26	25.48	41.87	101.59	314.23	51.80	41.05	
20 × 20	64.59	160.79	20.91	33.97	128.26	369.00	21.09	29.86	
25 × 25	56.86	89.80	23.29	37.18	66.54	262.35	23.48	35.49	
30 × 30	38.51	30.71	7.24	6.72	38.51	39.93	6.16	5.61	
35 × 35	26.65	32.48	4.62	3.68	32.82	47.60	4.25	3.65	
40 × 40	3.40	2.90	0.54	0.55	2.53	2.51	0.38	0.43	

Table 1 The mean absolute percentage error (MAPE) of MT was obtained through the VQTAM and VQTAM-LLE approaches, with a stopping criterion set at 1% for changing the objective function

Table 2 The mean absolute percentage error (MAPE) of MT was obtained through the VQTAM and VQTAM-LLE approaches, with a stopping criterion set at 0.5% for changing the objective function

Num. of neurons	Mean absolute percentage error (MAPE)								
	VQTAM				VQTAM-LLE				
	ρ^a_{xy}	$ ho_{yx}^a$	φ _{xy}	φ _{yx}	ρ^a_{xy}	$ ho_{yx}^a$	φ _{xy}	ф _{yx}	
10 × 10	92.55	148.30	26.69	43.12	231.71	738.93	110.63	52.74	
15 × 15	33.02	25.36	7.23	6.14	47.44	33.84	7.15	5.77	
20 × 20	11.63	10.61	3.29	2.67	18.75	16.14	4.44	3.47	
25 × 25	6.57	6.49	1.45	1.37	6.97	6.71	1.63	1.66	
30 × 30	5.11	5.32	1.03	1.00	4.99	7.20	0.93	0.97	
35 × 35	3.48	3.39	0.70	0.72	3.44	7.17	0.53	0.62	
40×40	2.66	2.47	0.51	0.52	2.10	1.91	0.36	0.39	

Table 3 The mean absolute percentage error (MAPE) of MT was obtained through the VQTAM and VQTAM-LLE approaches, with a stopping criterion set at 0.1% for changing the objective function

Num. of neurons	Mean absolute percentage error (MAPE)								
	VQTAM				VQTAM-LLE				
	ρ^a_{xy}	$ ho_{yx}^a$	φ _{xy}	φ _{yx}	ρ^a_{xy}	$ ho_{yx}^a$	φ _{xy}	φ _{yx}	
10 × 10	92.55	148.30	26.69	43.12	231.71	738.93	110.63	52.74	
15 × 15	36.63	26.19	5.44	4.58	52.27	35.39	6.30	5.19	
20 × 20	10.64	10.01	2.89	2.35	15.55	12.99	3.79	3.06	
25 × 25	7.07	6.66	1.33	1.30	7.42	6.65	1.48	1.59	
30 × 30	4.24	4.08	0.80	0.80	4.28	3.98	0.73	0.83	
35 × 35	3.88	4.40	0.71	0.73	3.60	3.43	0.52	0.60	
40×40	2.88	2.95	0.55	0.58	2.21	2.09	0.36	0.41	

MAPE values for apparent resistivity were consistently higher than those for phase across all ratios. However, this trade-off was observed regarding the number of iterations and CPU time, with increased training data leading to longer computation times, as illustrated in Fig. 6.

Topographic models

For the topographic models, we selected two profiles denoted as P1 and P2, passing through distinct terrains and locations in Thailand. Sample points along these profiles were collected and utilized as input for the

Num. of neurons	Mean absolute percentage error (MAPE)								
	VQTAM				VQTAM-LLE				
	$\overline{ ho^a_{xy}}$	$ ho_{yx}^a$	φ _{xy}	φ _{yx}	ρ^a_{xy}	$ ho_{yx}^a$	φ _{xy}	φ _{yx}	
10 × 10	92.55	148.30	26.69	43.12	231.71	738.93	110.63	52.74	
15 × 15	43.06	30.02	6.13	5.06	61.65	40.44	7.01	5.50	
20 × 20	11.41	10.31	2.66	2.22	15.93	12.52	3.97	3.22	
25 × 25	6.34	6.64	1.44	1.30	6.60	6.38	1.54	1.47	
30 × 30	4.45	4.44	0.87	0.87	4.87	4.46	0.72	0.80	
35 × 35	3.63	3.56	0.66	0.69	3.32	3.14	0.50	0.58	
40 × 40	2.50	2.85	0.56	0.57	1.83	1.85	0.35	0.39	

Table 4 The mean absolute percentage error (MAPE) of MT was obtained through the VQTAM and VQTAM-LLE approaches, with a stopping criterion set at 0.05% for changing the objective function

Table 5 The CPU time (seconds) against different numbers of neurons utilized by the VQTAM and VQTAM-LLE approaches

Num. of neurons	Stopping o	Stopping criteria of objective functions									
	1.0%		0.5%		0.1%		0.05%				
	VQTAM	VQTAM-LLE	VQTAM	VQTAM-LLE	VQTAM	VQTAM-LLE	VQTAM	VQTAM-LLE			
10 × 10	5	6	5	6	5	5	5	5			
15 × 15	8	8	83	84	124	124	146	147			
20 × 20	23	23	134	134	153	154	170	171			
25 × 25	21	22	222	223	238	239	251	252			
30 × 30	118	119	338	339	387	388	401	402			
35 × 35	165	166	509	510	613	615	651	653			
40×40	743	745	801	803	955	957	1096	1098			

FE method to generate training and testing data. The locations of these profiles are depicted in Fig. 7 (top).

The left-bottom part of Fig. 7 illustrates the P1 profile and its corresponding elevation, covering a length of 335 km and situated on the Khorat Plateau in northeastern Thailand. This region, as depicted in the figure, forms a lengthy basin. Geological evidence below the Khorat plateau indicates surface rock types such as sandstone, shale, and rock salt Bunkanpai et al. (2023). For this study, we create the simplified resistivity model with the top layer set at 100 Ω m, reaching depths of approximately 4 km. The bottom layer is defined at 5000 Ω m, as shown in Fig. 8 (left). There were $N_S = 21$ sites in total, and the data were generated over 41 periods ranging from 1 to 1000 s using the FE method for the P1 profile, as illustrated in Fig. 9 (left), showcasing significant distortions in the ρ_{vx}^{α} responses.

The right-bottom section of Fig. 7 displays the P2 profile, located in the Kanchanaburi province of Thailand. This 123 km long profile is positioned very close to the P4 profile studied by Boonchaisuk et al., 2013 Boonchaisuk et al. (2013), for investigating Miocene dual subduction zones beneath the Shan-Thai terrane in western Thailand.

The resistivity model for this profile was simplified based on the inversion results from the study mentioned above, as depicted in Fig. 8 (right). For the P2 profile, $N_S = 41$ sites were considered, and data were generated over 31 periods ranging [0.1, 100] s using the FE method, as shown in Fig. 9 (right).

In both cases, we set the acceptable average MAPE level at 5% for each response type, with 1600 neurons and a stopping criterion of 0.1% for the change in the objective function in both VQTAM and VQTAM-LLE approaches. The ratio between training and testing data was 21:20 for the first P1 model and 16:15 for the second P2 profile. Figure 10 depicts the calculated MAPE values. For the P1 profile, both approaches managed to reduce the average MAPE to the desired level within 151 iterations, demonstrating similar accuracy. Similarly, for the P2 profile, these methods required 141 iterations to achieve the desired MAPE level. The accuracy of both VQTAM and VQTAM-LLE appeared comparable, although the phase MAPE values were significantly lower than those of the apparent resistivity.

Discussion and conclusions

This study aims to develop and implement the VQTAM and VQTAM-LLE approaches for simulating 2-D magnetotelluric responses. Training and testing data, essential for evaluating the effectiveness of these methods, are generated through FE techniques. Both neural networks require input data, precisely the coordinates of magnetotelluric (MT) sites and their corresponding responses, including apparent resistivity and phase at selected electromagnetic (EM) periods. Various parameters, such as the number of neurons, stopping criteria for objective functions, the ratio between training and testing data, and the type of errors considered, are investigated to determine their impact on the accuracy and efficiency of the neural networks.

Benchmark models and real-world scenarios are employed to assess the potential of both VQTAM and VQTAM-LLE approaches. The results indicate that sufficient neurons are crucial for achieving high accuracy. Setting the stopping criteria for the objective function at an appropriate level is essential. Moreover, the quantity of training data should be substantial compared to testing data, and these data points need to be distributed within the specified range. When these parameters are appropriately configured, the VQTAM and VQTAM-LLE schemes demonstrate excellent performance for benchmark and confirmed cases, achieving the desired error levels.

to their impressive performance, Due these techniques may be helpful as alternative forward operators when incorporated alongside some traditional methods or as hybrid neural networkstraditional methods. Furthermore, the neural networks' role in the pre-inversion process may generate additional MT data for better input data sets or for the restricted stations or periods where the MT survey cannot be done with some limitations. However, enhancing the speed of these approaches for practical applications is crucial. Achieving this goal may require potential enhancements, redesigns, or re-architecture of these methods.

Acknowledgements

The authors express their gratitude to Dr. Chatchai Vachiratienchai from Curl-E Geophysics Co. Ltd. for their technical guidance and valuable discussions throughout this research. Additionally, the authors appreciate the reviewers for their helpful feedback, which greatly enhanced the quality of this manuscript.

Author contributions

Phongphan Mukwachi is credited with the study conception, design, data collection, analysis, and initial drafting of the manuscript. Banchar Arnonkijpanich is acknowledged for contributing to study design and substantial revisions of the manuscript. Weerachai Sarakorn supervised the entire research project, provided guidance on methodology, reviewed and interpreted the data, and played a significant role in manuscript writing and revisions.

Page 15 of 16

Funding

This research was supported by Department of Mathematics, Faculty of Science, Khon Kaen University, Fiscal Year 2022.

Availability of data and materials

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 10 October 2023 Accepted: 20 February 2024 Published online: 02 March 2024

References

- Arnonkijpanich B, Hasenfuss A, Hammer B (2010) Local matrix learning in clustering and applications for manifold visualization. Neural networks 23(4):476–486. https://doi.org/10.1016/j.neunet.2009.12.003
- Arnonkijpanich B, Hasenfuss A, Hammer B (2011) Local matrix adaptation in topographic neural maps. Neurocomputing 74(4):522–539. https://doi. org/10.1016/j.neucom.2010.08.016
- Barreto GA, Araujo AFR (2004) Identification and control of dynamical systems using the self-organizing map. IEEE Transactions on Neural Networks 15(5):1244–1259. https://doi.org/10.1109/TNN.2004.832825
- Boonchaisuk S, Siripunvaraporn W, Ogawa Y (2013) Evidence for middle triassic to miocene dual subduction zones beneath the shan-thai terrane, western thailand from magnetotelluric data. Gondwana Research 23(4):1607–1616. https://doi.org/10.1016/j.gr.2012.08.009
- Bunkanpai N, Suksribunampai B, Noisomsri E (2023) Korat formation in the boundary of the korat plateau[in thai]. Academic report, Department Of Mineral Resources, Thailand (October). https://www.dmr.go.th/
- Chetchotsak D, Pattanapairoj S, Arnonkijpanich B (2015) Integrating new data balancing technique with committee networks for imbalanced data: GRSOM approach. Cognitive neurodynamics 9(6):627–638. https://doi. org/10.1007/s11571-015-9350-4

da Conceição Batista J, Sampaio EES (2019) Magnetotelluric inversion of one-and two-dimensional synthetic data based on hybrid genetic algorithms. Acta Geophysica 67(5):1365–1377. https://doi.org/10.1007/ s11600-019-00325-y

- Deng F, Hu J, Wang X, Yu S, Zhang B, Li S, Li X (2023) Magnetotelluric deep learning forward modeling and its application in inversion. Remote Sensing 15(14). https://doi.org/10.3390/rs15143667
- Du H-K, Ren Z-Y, Tang J-T (2016) A finite-volume approach for 2D magnetotellurics modeling with arbitrary topographies. Studia Geophysica et Geodaetica 60(2):332–347. https://doi.org/10.1007/ s11200-014-1041-9
- Franke A, Börner R-U, Spitzer K (2007) Adaptive unstructured grid finite element simulation of two-dimensional magnetotelluric fields for arbitrary surface and seafloor topography. Geophysical Journal International 171(1):71–86. https://doi.org/10.1111/j.1365-246X.2007. 03481.x
- Han B, Li Y, Li G (2018) 3D forward modeling of magnetotelluric fields in general anisotropic media and its numerical implementation in julia. Geophysics 83(4):29–40. https://doi.org/10.1190/geo2017-0515.1
- Jahandari H, Farquharson C (2015) Finite-volume modelling of geophysical electromagnetic data on unstructured grids using potentials. Geophysical Journal International 202(3):1859–1876. https://doi.org/10. 1093/gji/ggv257
- Key K, Weiss C (2006) Adaptive finite-element modeling using unstructured grids: The 2D magnetotelluric example. Geophysics 71(6):291–299. https://doi.org/10.1190/1.2348091
- Khalil A, Mekkawi M, El Bohoty MES (2006) Application of artificial neural network to magnetotelluric data. NRIAG Journal of Geophysics Special Issue, 95–111

Khampichit T, Sarakorn W (2021) Application of finite volume method with unstructured quadrilateral mesh for solving 2-D magnetotelluric modeling. Thai Journal of Mathematics 19(3):1082–1095

- Kohonen T (2013) Essentials of the self-organizing map. Neural networks 37:52–65. https://doi.org/10.1016/j.neunet.2012.09.018
- Limtrakul S, Arnonkijpanich B (2019) Supervised learning based on the selforganizing maps for forward kinematic modeling of stewart platform. Neural Computing and Applications 31(2):619–635. https://doi.org/10. 1007/s00521-017-3095-4
- Mackie RL, Smith JT, Madden TR (1994) Three-dimensional electromagnetic modeling using finite difference equations: The magnetotelluric example. Radio Science 29(4):923–935. https://doi.org/10.1029/94RS00326
- Manoj C, Nagarajan N (2003) The application of artificial neural networks to magnetotelluric time-series analysis. Geophysical Journal International 153(2):409–423. https://doi.org/10.1046/j.1365-246X.2003.01902.x

Mogi T (1996) Three-dimensional modeling of magnetotelluric data using finite element method. Journal of Applied Geophysics 35(2–3):185–189. https://doi.org/10.1016/0926-9851(96)00020-1

- Puzyrev V (2019) Deep learning electromagnetic inversion with convolutional neural networks. Geophysical Journal International 218(2):817–832. https://doi.org/10.1093/gji/ggz204
- Puzyrev V, Swidinsky A (2021) Inversion of 1D frequency- and time-domain electromagnetic data with convolutional neural networks. Computers & Geosciences 149:104681. https://doi.org/10.1016/j.cageo.2020.104681
- Ren Z, Kalscheuer T, Greenhalgh S, Maurer H (2014) A hybrid boundary element-finite element approach to modeling plane wave 3D electromagnetic induction responses in the earth. Journal of Computational Physics 258:705–717
- Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500):2323–2326. https://doi.org/10.1126/scien ce.290.5500.2323
- Sarakorn W (2017) 2-D magnetotelluric modeling using finite element method incorporating unstructured quadrilateral elements. Journal of Applied Geophysics 139:16–24. https://doi.org/10.1016/j.jappgeo.2017.02.005
- Sarakorn W, Vachiratienchai C (2018) Hybrid finite difference-finite element method to incorporate topography and bathymetry for two-dimensional magnetotelluric modeling. Earth, Planets and Space 70(1):1–14. https:// doi.org/10.1186/s40623-018-0876-7
- Shan T, Guo R, Li M, Yang F, Xu S, Liang L (2021) Application of multitask learning for 2-D modeling of magnetotelluric surveys: Te case. IEEE Transactions on Geoscience and Remote Sensing 60:1–9. https://doi.org/ 10.1109/TGRS.2021.3101119
- Singh UK, Tiwari RK, Singh S (2005) One-dimensional inversion of geoelectrical resistivity sounding data using artificial neural networks–a case study. Computers & geosciences 31(1):99–108. https://doi.org/10.1016/j. cageo.2004.09.014
- Siripunvaraporn W, Egbert G, Lenbury Y (2002) Numerical accuracy of magnetotelluric modeling: a comparison of finite difference approximations. Earth, planets and space 54(6):721–725. https://doi.org/ 10.1186/BF03351724
- Tan H, Yu Q, John B, Wei W (2003) Three-dimensional magnetotelluric modeling using the staggered-grid finite difference method. Chinese Journal of Geophysics 46(5):1011–1020. https://doi.org/10.1002/cjq2.420
- Tong X, Guo Y, Xie W (2018) Finite difference algorithm on non-uniform meshes for modeling 2D magnetotelluric responses. Algorithms 11(12):203. https://doi.org/10.3390/a11120203
- Wang N, Tang J, Ren Z, Xiao X, HUANG X (2019) Two-dimensional magnetotelluric anisotropic forward modeling using finite-volume method. Chinese Journal of Geophysics 62(10):3912–3922. https://doi. org/10.6038/cjg2019M0498
- Wannamaker PE, Stodt JA, Rijo L (1987) A stable finite element solution for two-dimensional magnetotelluric modelling. Geophysical Journal of the Royal Astronomical Society 88(1):277–296. https://doi.org/10.1111/j.1365-246X.1987.tb01380.x
- Wittke J, Tezkan B (2014) Meshfree magnetotelluric modelling. Geophysical Journal International 198(2):1255–1268. https://doi.org/10.1093/gji/ ggu207
- Wongsriworaphon A, Arnonkijpanich B, Pathumnakul S (2015) An approach based on digital image analysis to estimate the live weights of pigs in farm environments. Computers and Electronics in Agriculture 115:26–33. https://doi.org/10.1016/j.compag.2015.05.004

- Xu S-Z, Zhou H (1997) Modelling the 2D terrain effect on mt by the boundaryelement method. Geophysical prospecting 45(6):931–943. https://doi. org/10.1046/j.1365-2478.1997.610301.x
- Yang Z-W, Li X-B, Liu Y-G, ZHAO Q-f (2019) Study of 2D magnetotelluric forward based on multigrid finite element method. Progress in Geophysics 34(3):1132–1138. https://doi.org/10.6038/pg2019CC0228
- Zhdanov M, Varentsov IM, Weaver J, Golubev N, Krylov V (1997) Methods for modelling electromagnetic fields results from COMMEMI-the international project on the comparison of modelling methods for electromagnetic induction. Journal of applied geophysics 37(3–4):133– 271. https://doi.org/10.1016/S0926-9851(97)00013-X
- Zyserman FI, Guarracino L, Santos JE (1999) A hybridized mixed finite element domain decomposed method for two dimensional magnetotelluric modelling. Earth, Planets and Space 51(4):297–306. https://doi.org/10. 1186/BF03352233

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.