

Communication

Solution of Electromagnetic Scattering and Inverse Scattering by Integral Equations Through Neural Networks

Xin-Yue Lou¹, Jun-Bo Zhang¹, Da-Miao Yu¹, Deng-Feng Wang, and Xiao-Min Pan¹

Abstract—This work develops a solver based on the neural networks, where the discrete form of integral equations (IEs) is embedded to address both forward and inverse dynamic electromagnetic scattering problems. Since the input is the real-valued coordinates while the output is the complex-valued fields, our proposed solver distinguishes itself from existing neural network solvers. Different network structures to handle the phase information contained in the complex-valued numbers are investigated. Improvements are developed to enhance the performance of the proposed solver. Numerical simulations for 2-D electromagnetic scattering in both forward and inverse problems were conducted to validate the proposed solver.

Index Terms—Electromagnetic scattering, integral equation (IE), physics-informed neural network.

I. INTRODUCTION

With the rapid development of artificial intelligence (AI), there is a growing interest in utilizing the AI technology to tackle electromagnetic problems. Several well-established algorithms in machine learning and deep learning, such as support vector machines (SVMs) [1] and deep neural networks (DNNs) [2], [3], [4], [5], [6], [7], [8], [9], [10], have been successfully employed to solve electromagnetic problems with promising results. A comprehensive review of traditional machine-learning-based computational electromagnetic (CEM) methods can be found in [11] and [12].

Unlike other machine-learning algorithms [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], PINNs can eliminate the need for a large amount of training data because it integrates the laws of physics, in the form of PDEs or their variations, as constraints into neural networks. As a result, PINNs are particularly effective for solving problems with a limited or even zero amount of training samples. Thanks to their ability and great potential as a new scientific computing tool, PINNs [13], including their variations and improvements [14], [15], [16], [17], [18], [19], have gained great attentions as a novel machine learning framework. As far as CEM problems are concerned, a PINN was developed to solve time domain electromagnetic problems by encoding initial conditions, boundary conditions, and Maxwell's equations as constraints during the network training process [20]. Based on the U-Net framework, the so-called MaxwellNet for solving the scattered light field in free space was proposed [21] by employing the frequency-domain vector Helmholtz equation with respect to the electric field as a physics-driven loss function. To design electromagnetism devices and

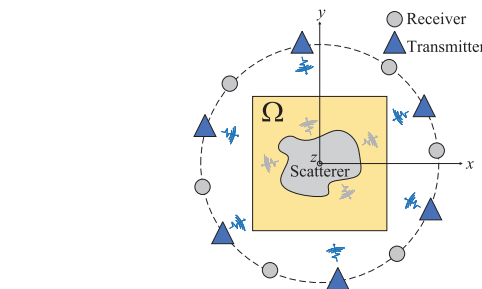


Fig. 1. Geometry configuration of a typical electromagnetic scattering problem with the 2-D transverse magnetic (TM) incident plane wave.

systems, a PINN was proposed for solving 2-D magnetostatic fields [22]. In the low-frequency electromagnetism region, PINNs were utilized to address specific problems [23], where transfer learning (TL) was utilized to generalize the PINN estimator as much as possible. In [24], a hypernetwork was proposed as a parametrized real-time field solver that allows the fast solution of inverse problems. In fact, many researches were conducted on PINNs from different aspects in solving electromagnetic problems [25], [26], [27], [28], [29], [30]. As an alternative to the PDE form, the integral equation (IE) form of the Maxwell's equations is a popular technical line to solve electromagnetic problems. In [31], neural networks were designed to learn and approximate Green's functions, which are subsequently incorporated into the boundary IEs to obtain the desired numerical solutions.

To improve the performance of a neural network, the choice of a suitable activation function is vital. Numerous activation functions have been developed and studied, such as rectified linear units (ReLU) [32] and its variations, exponential linear unit (ELUs) [33] and Gaussian error linear units (GELUs) [34], Swish [35], [36], and Mish [37]. In general, it is hard to find a good activation function working well for all kinds of networks and all types of problems. Most recently, the concept of the adaptive activation function was proposed to combine a set of activation functions together. Studies in [38], [39], [40], and [41] validated their effectiveness in improving the accuracy of the neural networks.

This work develops a solver based on neural networks that incorporates the discrete form of IEs to solve forward and inverse dynamic electromagnetic scattering problems in the same framework. Three network structures are implemented and studied to show the performance of the solver where the phase information is tackled distinctly. In addition, improvements of the proposed solver are developed and studied. The performance of the proposed solver is numerically investigated on a set of 2-D scatterers.

II. VOLUME IE AND METHOD OF MOMENTS

The configuration of a 2-D scattering problem is depicted in Fig. 1 where Ω denotes the computational domain. The background is the

Received 1 April 2024; revised 8 June 2025; accepted 20 July 2025. Date of publication 31 July 2025; date of current version 30 October 2025. This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 62171033. (Corresponding author: Xiao-Min Pan.)

Xin-Yue Lou and Xiao-Min Pan are with the School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing 100081, China (e-mail: xmpan@bit.edu.cn).

Jun-Bo Zhang and Da-Miao Yu are with the School of Integrated Circuits and Electronics, Beijing Institute of Technology, Beijing 100081, China.

Deng-Feng Wang is with Yangzhou Marine Electronic Instruments Institute, Yangzhou 225001, China.

Digital Object Identifier 10.1109/TAP.2025.3592819

0018-926X © 2025 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission.

See <https://www.ieee.org/publications/rights/index.html> for more information.

Authorized licensed use limited to: UNIVERSITAET FREIBURG. Downloaded on February 15, 2026 at 10:52:50 UTC from IEEE Xplore. Restrictions apply.

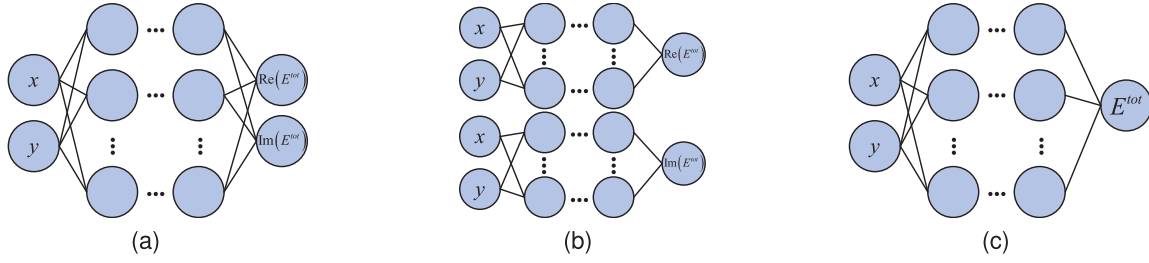


Fig. 2. Three network structures for the forward solution of the IE. The inputs of the networks are the coordinate (x, y) of all sample points within the Ω while the outputs are the total electric field $\bar{E}_{NN}^{\text{tot}}$. All the FCNs has four hidden layers, each with 50 neurons. (a) Pure real-valued. (b) Separated. (c) Complex-valued.

free space with the permittivity ϵ_0 and permeability μ_0 . In Ω , there are N dielectric scatterers with $\epsilon_{r,1}(\mathbf{r}), \epsilon_{r,2}(\mathbf{r}), \dots, \epsilon_{r,N}(\mathbf{r})$, respectively, the relative permittivity of each scatterer. Suppose the scatterers are illuminated by a TM wave and z -axis denotes the invariant direction, the fields can then be uniquely represented by their z -components. The associated electric field IE can be written as

$$\mathbf{E}^{\text{tot}}(\mathbf{r}) = \mathbf{E}^{\text{inc}}(\mathbf{r}) + k_0^2 \int_{\Omega} G(\mathbf{r}, \mathbf{r}') (\epsilon_r(\mathbf{r}') - 1) \mathbf{E}^{\text{tot}}(\mathbf{r}') d\mathbf{r}', \mathbf{r}' \in \Omega \quad (1)$$

where ω is the angular frequency, $k_0 = \omega \sqrt{\mu_0 \epsilon_0}$ is the wavenumber, $\mathbf{r}$ and $\mathbf{r}'$ denote the field and source points, $\mathbf{E}^{\text{inc}}(\mathbf{r})$ is incident electric field, and $G(\mathbf{r}, \mathbf{r}')$ is the Green's function. Often, we define the contrast $\chi(\mathbf{r}')$ and the induced equivalent current $\mathbf{J}(\mathbf{r}')$ within the scatterers as

$$\chi(\mathbf{r}') = \epsilon_r(\mathbf{r}') - 1 \quad \text{and} \quad \mathbf{J}(\mathbf{r}') = \chi(\mathbf{r}') \mathbf{E}^{\text{tot}}(\mathbf{r}'). \quad (2)$$

With (2), the scattered field $\mathbf{E}^{\text{sca}}(\mathbf{r})$ can be obtained by

$$\mathbf{E}^{\text{sca}}(\mathbf{r}) = k_0^2 \int_{\Omega} G(\mathbf{r}, \mathbf{r}') \mathbf{J}(\mathbf{r}') d\mathbf{r}'. \quad (3)$$

To find $\mathbf{J}(\mathbf{r}')$, the method of moments (MoMs) can be employed to solve (1). In the MoM solution, (1) is discretized with the pulse basis function and the delta test function after each scatter in Ω is divided into sufficiently small subdomains. The associated linear system can be written as

$$\bar{\mathbf{E}}^{\text{tot}} = \bar{\mathbf{E}}^{\text{inc}} + \bar{\mathbf{G}}_D \cdot \bar{\chi} \cdot \bar{\mathbf{E}}^{\text{tot}}. \quad (4)$$

Suppose the number of subdomains is M , $\bar{\mathbf{E}}^{\text{tot}} = [E_1^{\text{tot}}, E_2^{\text{tot}}, \dots, E_M^{\text{tot}}]^T$ and $\bar{\mathbf{E}}^{\text{inc}} = [E_1^{\text{inc}}, E_2^{\text{inc}}, \dots, E_M^{\text{inc}}]^T$ contain the total and incident electric field samples of all subdomains; $\bar{\mathbf{G}}_D$ is the impedance matrix and $\bar{\chi}$ is the diagonal matrix storing the contrast of each subdomain. Both $\bar{\mathbf{G}}_D$ and $\bar{\chi}$ are of size $M \times M$. Once (4) is solved, $\bar{\mathbf{J}}$ can be obtained by

$$\bar{\mathbf{J}} = \bar{\chi} \cdot \bar{\mathbf{E}}^{\text{tot}}. \quad (5)$$

With $\bar{\mathbf{G}}_S$ as the discretized Green's function, following equation can be written as

$$\bar{\mathbf{E}}^{\text{sca}} = \bar{\mathbf{G}}_S \cdot \bar{\mathbf{J}}. \quad (6)$$

III. PROPOSED NEURAL NETWORK SOLVER

A. Incorporation of the MoM Into Neural Networks

The essence of PINNs to solve a PDE is to embed the given PDE or its variations into the loss function of the neural network, which can be defined as the weighted summation of the L_2 norm of residuals for the PDE equation and boundary conditions. To feed into a neural network, the underlying physics of the electromagnetic scattering in the form of (1), for the forward problem, the loss function is defined as the mean squared error (mse)

$$\mathcal{L} = \frac{1}{M} \left\| \bar{\mathbf{E}}^{\text{inc}} - (\bar{\mathbf{I}} - \bar{\mathbf{G}}_D \cdot \bar{\chi}) \cdot \bar{\mathbf{E}}_{NN}^{\text{tot}} \right\|_F^2 \quad (7)$$

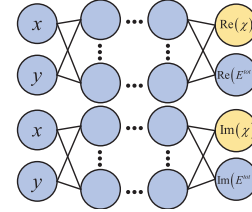


Fig. 3. Network structure for the inverse solver. Here, the NN shown in Fig. 2(b) is taken as the backbone.

where $\bar{\mathbf{E}}_{NN}^{\text{tot}}$ represents the predicted total field within the scatterer by the propose neural network solver, M is the number of observation points, and $\|\cdot\|_F$ is the Frobenius norm operator. Different from [31], our proposed solver incorporates the IE into the loss function without explicitly handling the singularity in the network. Due to the incorporation of (4) into the loss function, no training dataset is required to solve the corresponding IE for the forward scattering problems.

B. Network Structures

1) *Complex-Valued Numbers*: The output of the neural network for forward problems is complex-valued fields while the input are real-valued coordinates. As it is well-known, the phase information contained in complex numbers plays an important role in the underlying physics. It is therefore interesting to study the performance of different neural network structures where the phase information is handle distinctively. In this work, three network structures are investigated, as shown in Fig. 2. The backbone FCN consists of four hidden layers, each with 50 neurons. The three networks share the same combination of the input and output. In particular, for scattering problems, the input consists of x - and y -coordinates of the observation points (or sampling points) where the output is the corresponding total electric fields $\mathbf{E}^{\text{tot}}$'s, i.e., $\bar{\mathbf{E}}_{NN}^{\text{tot}}$'s. The network shown in Fig. 2(a) has a single real-valued FCN. Fig. 2(b) employs two real-valued FCNs to separately predict the real and imaginary parts of $\mathbf{E}^{\text{tot}}$ (i.e., $\bar{\mathbf{E}}_{NN}^{\text{tot}}$). The network in Fig. 2(c) uses a complex-valued FCN [42], [43], [44] to produce the required complex-valued total field.

2) *Inverse Problems*: The three networks shown in Fig. 2 can be also employed for inverse problems if $\bar{\chi}$, $\bar{\mathbf{E}}^{\text{tot}}$ are taken as outputs, as shown in Fig. 3. For a typical inverse problem, a set of observation scattering data corresponding to N_i incident waves, namely, $\bar{\mathbf{E}}_1^{\text{inc}}, \bar{\mathbf{E}}_2^{\text{inc}}, \dots, \bar{\mathbf{E}}_{N_i}^{\text{inc}}$, should be provided to retrieve $\bar{\chi}$. Here, the number of receivers for each incident is denoted by N_o . The observation data are employed to compute the loss function, which can be written as

$$\mathcal{L} = \frac{\sum_{p=1}^{N_i} \left\| \bar{\chi}_{NN} \cdot \bar{\mathbf{E}}_p^{\text{inc}} - (\bar{\mathbf{I}} - \bar{\chi}_{NN} \cdot \bar{\mathbf{G}}_D) \cdot \bar{\chi}_{NN} \cdot \bar{\mathbf{E}}_p^{\text{tot}} \right\|_F^2}{\sum_{p=1}^{N_i} \left\| \bar{\mathbf{E}}_p^{\text{inc}} \right\|_F^2} + \frac{\sum_{p=1}^{N_i} \left\| \bar{\mathbf{G}}_S \cdot \bar{\chi}_{NN} \cdot \bar{\mathbf{E}}_p^{\text{tot}} - \bar{\mathbf{E}}_p^{\text{sca}} \right\|_F^2}{\sum_{p=1}^{N_i} \left\| \bar{\mathbf{E}}_p^{\text{sca}} \right\|_F^2} \quad (8)$$

where $\bar{E}_p^{\text{sca}} = [E_1^{\text{sca}}, E_2^{\text{sca}}, \dots, E_{N_o}^{\text{sca}}]^T$ is the scattered field away from the scatterer under the p th incident. It is clear that the first term in the right hand-side of (8) is similar to the term in (7), which evaluates the loss corresponding to the operator in (1). The difference lies in that, in (8), a scaling operation is employed to balance the two loss terms.

It should be noted that $\bar{E}_p^{\text{sca}}$'s are not training data because they are indispensable for an inverse problem.

C. Improvements

1) *Accuracy*: To improve the accuracy of our solver, an adaptive activation function [38], [41] is employed. It [i.e., $f(x)$] is a linear combination of F candidate activation functions with learnable coefficients, which can be written as

$$f(x) = \sum_{i=1}^F G(\beta_i) \sigma_i(x) \quad (9)$$

where $\sigma_i(\cdot)$ and β_i denote the i th candidate activation function and the corresponding learnable coefficient, $G(\cdot)$ denotes a gate function, in the form of a Softmax function here. In this work, $F = 2$ and the candidate activation functions are $\{\sin, \cos\}$. The learnable coefficients (β_1 and β_2) are initialized as zeros. Identical to [39] and [40], the adaptive activation function is layer-wise locally, aiming to enhance the learnability of the activation function.

2) *Efficiency*: To improve the efficiency for solving forward problems, the TL technique [45] is employed when the shape of the scatterer varies but the computational domain and contrast are fixed. In particular, the learnable parameters of the NN used for a scatterer are taken as the initial parameters for the subsequent scatterer. For the solution of the latter, the NN is fine-tuned according to the corresponding profile of the scatterer. Since the computational domain is invariant, the operator associated with G_D is fixed. The parameters in a well-trained NN for a given scatterer can be a nice starting point of a next scatterer.

In addition, we make the NN solver able to conduct angular sweeping. Suppose the number of sweeping angles is N_{swp} , the corresponding loss function (7) is revised as

$$\mathcal{L} = \frac{1}{MN_{\text{swp}}} \sum_{p=1}^{N_{\text{swp}}} \left\| \bar{E}_p^{\text{inc}} - (\bar{I} - \bar{G}_D \cdot \bar{\chi}) \cdot \bar{E}_{p, \text{NN}}^{\text{tot}} \right\|_F^2 \quad (10)$$

In our implementation, the number of output neurons is simply increased by a factor of N_{swp} to store fields related with all incidents.

IV. NUMERICAL RESULTS

The performance of the proposed solver is investigated on a server configured by a NVIDIA RTX 4090 GPU and an AMD EPYC 9654 CPU. Four types of 2-D dielectric scatterers with different shapes are selected. They are the rectangular-shaped objects with dimensions of $1.6 \times 1.6 \text{ m}^2$, a floral-shaped objects, the digit-shaped, and letter-shaped objects that are generated from the extended MNIST (EMNIST) database. Scaling all scatterers to a comparable size, we set the computational domain Ω to be $2 \times 2 \text{ m}^2$ with the center of $(0, 0) \text{ m}$. The uniformly sampled collocation points are employed in this work for simplicity even though nonuniformed ones are applicable to the proposed method.

The adaptive moment estimation with decoupled weight decay (AdamW) method is used to minimize the loss function. The adaptive moment estimation with decoupled weight decay (AdamW) method is used to minimize the loss function. The learning rate is 1.0e-^4 and the training process stops when the number of epochs reaches 50 000.

To evaluate the accuracy of the proposed solver for forward problems, the relative error (RE) is employed, which has the form

$$\text{RE} = \frac{\left\| \bar{E}_{\text{NN}}^{\text{tot}} - \bar{E}_{\text{MoM}}^{\text{tot}} \right\|_F}{\left\| \bar{E}_{\text{MoM}}^{\text{tot}} \right\|_F} \quad (11)$$

TABLE I

ACCURACY FOR DIFFERENT NETWORK STRUCTURES

Network Structure	Fig. 2(a)	Fig. 2(b)	Fig. 2(c)
$M = 1024, \varepsilon_r = (1.5, 0.0)$			
RE of Current	0.0051	0.0045	0.0051
RE of Total Field	0.0047	0.0039	0.0042
RE of Scattering Field	0.0018	0.0019	0.0023
$M = 4096, \varepsilon_r = (3.0, 0.5)$			
RE of Current	0.0357	0.0231	0.0231
RE of Total Field	0.0224	0.0146	0.0154
RE of Scattering Field	0.0131	0.0098	0.0098

where $\bar{E}_{\text{MoM}}^{\text{tot}}$ is obtained by the traditional MoM. Except for $\bar{E}_{\text{MoM}}^{\text{tot}}$, the equivalent current (J) obtained by the MoM, and the scattering fields from the MoM, denoted by $\bar{E}_{\text{MoM}}^{\text{sca}}$, are also employed to reach comprehensive comparisons.

A. Network Structures

The implemented three different network structures are studied in terms of their performances for forward problems. Table I lists the detailed information on the computational configurations and REs associated with the three different network structures for the digit-shaped scatterers. In the case with the relative permittivity of $(1.5, 0)$, M is 1024, whereas it is increased to 4096 in the case with the relative permittivity of $(3.0, 0.5)$. From the table, it can be seen that three networks with different structures can all provide acceptable solutions. The three networks exhibit comparable accuracy when the relative permittivity of the scatterer is $(1.5, 0)$. However, the network described by Fig. 2(b) and (c) give a more accurate result than that in Fig. 2(a) when the relative permittivity becomes $(3.0, 0.5)$.

Different from [43], [44], [46], [47], and [48], where the complex-valued network can always perform better than its real-valued counterparts, here, the treatment of phase information contained in the complex-valued numbers exhibits compound impacts on the performance. One reason may lie in that both the input and output of networks in [43], [44], [46], [47], and [48] are complex-valued while the input in this work is real-valued. More research on this topic are required to figure out a suitable network structure to cope with the phase information [49].

The width and depth of the backbone FCN can also impact the performance of the three networks. We will continue the associated investigation after finding a more suitable network structure to handle the phase information.

B. Performance of Forward Solver

1) *Accuracy*: The network structure shown in Fig. 2(b) is used to study the performance of different activation functions against the tanh function.

Fig. 4 shows the distribution of equivalent currents, total electric fields, and scattered electric fields for the computations on floral-shaped and letter-shaped lossless scatterers with the relative permittivity of $(1.5, 0.0)$ and $M = 1024$. To quantitatively evaluate the performance of our solver, the associated computational configurations and REs of four scatterers are detailed in Table II.

Fig. 5 presents the current and field results of floral-shaped and letter-shaped lossy scatterers with the relative permittivity of $(3.0, 0.5)$ and $M = 4096$. Table III provides the computational configurations and the corresponding REs in detail.

The two set of computations demonstrate that our employed adaptive activation function can obviously improve the performance of the solver.

2) *Efficiency*: As the iteration count always varies with the incident angle and the scatterer's shape, it takes GMRES-equipped MoM $4.0\text{--}12.0\text{e-}^3 \text{ s}$ to reach a solution with the 1.0e-^3 accuracy corresponding to a single incident for a letter-shaped scatterer with

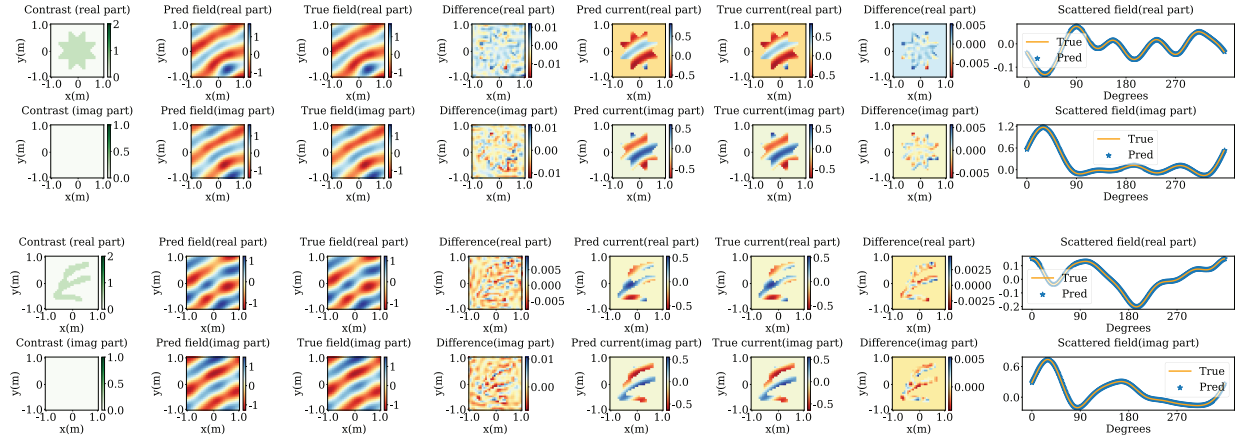


Fig. 4. Results for scatterers with relative permittivity of (1.5, 0.0) illuminated by the incident with direction $\varphi = 30^\circ$. The first column is the real- and imaginary-components of contrast. The second column shows the total electric fields obtained by our solver. The third column shows the real- and imaginary-components of the total electric fields computed by the MoM, which is employed as the reference result. The fourth column shows the absolute error between the predicted total fields and the reference result. The fifth–seventh columns show, respectively, the reference, the predicted results, and the difference between the former two with respect to the equivalent current. The eighth column shows the real- and imaginary-components of the reference and predicted scattered fields within $[0^\circ, 360^\circ]$.

TABLE II
COMPUTATIONAL INFORMATION FOR THE SCATTERERS WITH THE RELATIVE PERMITTIVITY OF (1.5, 0.0)

Activation Function	Adaptive Activation Function				tanh			
Scatterer's Shape	rectangular	floral shape	digit	letter	rectangular	floral	digit	letter
RE of Current	0.0035	0.0037	0.0042	0.0040	0.0078	0.0080	0.0089	0.0084
RE of Total Field	0.0045	0.0035	0.0035	0.0032	0.0101	0.0082	0.0083	0.0071
RE of Scattering Field	0.0013	0.0013	0.0015	0.0018	0.0021	0.0034	0.0044	0.0029

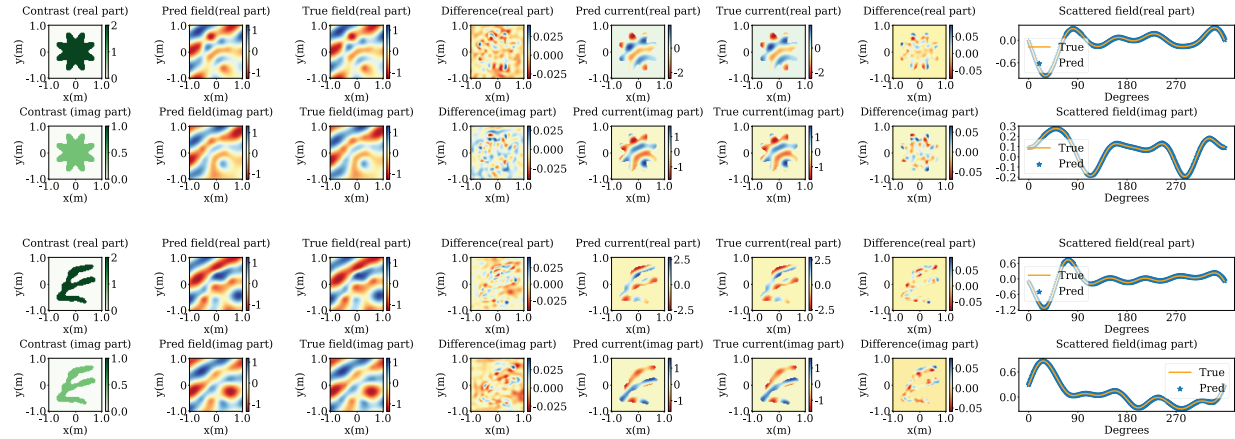


Fig. 5. Results for scatterers with relative permittivity of (3.0, 0.5) illuminated by the incident with $\varphi = 30^\circ$. The first column is the real and imaginary components of contrast. The second column shows the total electric field obtained by our solver. The third column shows the real and imaginary components of the total electric field computed by the MoM, which is employed as the reference result. The fourth column shows the absolute error between the predicted total field and the reference result. The fifth–seventh columns show, respectively, the reference, the predicted results, and the difference between the former two with respect to the equivalent current. The eighth column shows the real and imaginary components of the reference and predicted scattered fields within $[0^\circ, 360^\circ]$.

TABLE III
COMPUTATIONAL INFORMATION FOR THE SCATTERERS WITH THE RELATIVE PERMITTIVITY OF (3.0, 0.5)

Activation Function	Adaptive Activation Function				tanh			
Scatterer's Shape	rectangular	floral	digit	letter	rectangular	floral	digit	letter
RE of Current	0.0228	0.0231	0.0190	0.0211	0.0660	0.0483	0.0489	0.0325
RE of Total Field	0.0203	0.0151	0.0122	0.0129	0.0563	0.0293	0.0306	0.0210
RE of Scattering Field	0.0052	0.0097	0.0052	0.0086	0.0197	0.0224	0.0168	0.0130

$M = 1024$ and $\chi = 1.0$ after the impedance matrix is established in the same mesh is employed, the NN solver consumes around 3.0 s to reach a comparable accuracy. MoM requires additional 1.5 s to

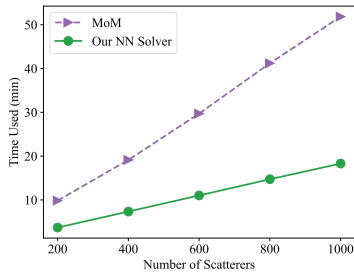


Fig. 6. Time used as a function of the number of tested scatterers.

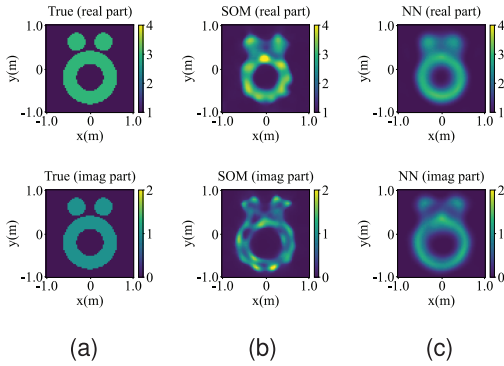


Fig. 7. Inversion results for a scatterer with a permittivity of (3.0, 1.0). The three columns show the scatterer's permittivity, respectively, for its ground truth value, that obtained by the SOM inversion after 800 iterations and that by the NN solver. (a) Ground truth. (b) SOM. (c) NN solver.

complete an angular sweeping with $N_{\text{swp}} = 360$ due to the repeated iteration solutions. Although the NN solver requires only $1.0e^{-3}$ s for each incident after being well-trained, it is still less efficient than MoM in terms of the total solution time.

When the number of scatterers is increased while χ and the computational domain are fixed, the TL-accelerated NN solver can be more efficient than MoM. The time used as a function of the number of scatterers is given in Fig. 6. In the study, the angular sweeping with $N_{\text{swp}} = 360$ is conducted for each scatterer. It can be observed that our NN solver outperforms MoM with FFT acceleration in terms of computational time under a comparable accuracy. The TL-accelerated NN solver can finish the computation involving 1000 scatterers in around 18 min while MoM requires approximately 52 min.

C. Inverse Problem

Using the inverse solver in Fig. 3 whose backbone is the NN shown in Fig. 2(b), the performance of the proposed solver is investigated in terms of an inverse problem where the Australia model with the permittivity of (3.0, 1.0) is employed where $N_i = 16$ and $N_o = 32$. Both the incident angles and observation points are uniformly sampled. For the latter, a circle with a radius of 2 m is employed. To make the observation data reliable, $\vec{E}^{\text{sca}}$'s are generated numerically with a 100×100 grid, in contrast to the 64×64 one employed by the inverse solver. Additive white Gaussian noise with a level of 10%, which defined similar to that in [50], is added to $\vec{E}^{\text{sca}}$'s to reach a rigid test of the proposed NN inverse solver.

In this study, our solver is employed to improve the accuracy of a subspace optimization method (SOM) [50] solution. In particular, a supervised pretraining is conducted on a single sample where the total field and permittivity obtained from the tenth iteration of SOM for the inverse problem under study are taken as the training data and label to avoid our NN solver falling into local minimal points. After 40 epochs of pretraining, the obtained network parameters are then given to our proposed solver for its training. Fig. 7 presents

the inversion results. The first column illustrates the ground truth distribution of the scatterer's permittivity. The second column shows the permittivity distribution obtained through the 800 iterations of the SOM, where 50% of the singular values are truncated. The resultant accuracy can be viewed as the upper bound of the accuracy, in the numerical sense, which can be delivered by the SOM at this configuration. The third column depicts the permittivity distribution reconstructed using our solver with the backbone of Fig. 2(b). The RE and structural similarity index (SSIM) [51] of χ obtained by MoM are 30.33% and 47.69%, respectively, while those by our solver are 28.62% and 53.20%, respectively. As illustrated in Fig. 7, our solver can substantially improve the accuracy.

Complex-valued DL structures may be preferred for inverse scattering problems [46], [47], [48] if the input and output of the networks are complex-valued. Since the input of the our solver is real-valued, complex-valued DL is not an optimized choice.

V. CONCLUSION

A solver which can incorporate physics into neural networks is proposed to solve the IE for 2-D dynamic electromagnetic scattering problems. Numerical experiments demonstrate that the proposed solver can effectively solve the electric IE. It is revealed that the manner to handle the phase information and the choice of the activation functions have compound impact on the performance of the proposed solver. Compared to traditional solvers, the proposed solver can solve both forward and inverse problems within the same framework accurately and efficiently.

REFERENCES

- [1] Y. Yang, S. M. Hu, and R. S. Chen, "A combination of FDTD and least-squares support vector machines for analysis of microwave integrated circuits," *Microw. Opt. Technol. Lett.*, vol. 44, no. 3, pp. 296–299, Dec. 2004.
- [2] P. Ramuhalli, L. Udpa, and S. S. Udpa, "Finite-element neural networks for solving differential equations," *IEEE Trans. Neural Netw.*, vol. 16, no. 6, pp. 1381–1392, Nov. 2005.
- [3] S. Qi, Y. Wang, Y. Li, X. Wu, Q. Ren, and Y. Ren, "Two-dimensional electromagnetic solver based on deep learning technique," *IEEE J. Multiscale Multiphys. Comput. Techn.*, vol. 5, pp. 83–88, 2020.
- [4] T. Shan et al., "Study on a fast solver for Poisson's equation based on deep learning technique," *IEEE Trans. Antennas Propag.*, vol. 68, no. 9, pp. 6725–6733, Sep. 2020.
- [5] L. Li, L. G. Wang, F. L. Teixeira, C. Liu, A. Nehorai, and T. J. Cui, "DeepNIS: Deep neural network for nonlinear electromagnetic inverse scattering," *IEEE Trans. Antennas Propag.*, vol. 67, no. 3, pp. 1819–1825, Mar. 2019.
- [6] T. Yin, C.-F. Wang, K. Xu, Y. Zhou, Y. Zhong, and X. Chen, "Electric flux density learning method for solving 3-D electromagnetic scattering problems," *IEEE Trans. Antennas Propag.*, vol. 70, no. 7, pp. 5144–5155, Jul. 2022.
- [7] M. Salucci, M. Arrebola, T. Shan, and M. Li, "Artificial intelligence: New frontiers in real-time inverse scattering and electromagnetic imaging," *IEEE Trans. Antennas Propag.*, vol. 70, no. 8, pp. 6349–6364, Aug. 2022.
- [8] H. Zhang, Y. Chen, T. J. Cui, F. L. Teixeira, and L. Li, "Probabilistic deep learning solutions to electromagnetic inverse scattering problems using conditional renormalization group flow," *IEEE Trans. Microw. Theory Techn.*, vol. 70, no. 11, pp. 4955–4965, Nov. 2022.
- [9] J.-Y. Wang, Y. Li, B.-W. Xue, and X.-M. Pan, "Physics-informed deep learning to solve electromagnetic scattering problems," in *Proc. IEEE Conf. Antenna Meas. Appl. (CAMA)*, Dec. 2022, pp. 1–3.
- [10] J.-Y. Wang and X.-M. Pan, "Physics-informed deep learning to solve 2D electromagnetic scattering problems," *Appl. Comput. Electromagn. Soc. J. (ACES)*, vol. 38, no. 9, pp. 667–673, Sep. 2023.
- [11] A. Massa, G. Oliveri, M. Salucci, N. Anselmi, and P. Rocca, "Learning-by-examples techniques as applied to electromagnetics," *J. Electromagn. Waves Appl.*, vol. 32, no. 4, pp. 516–541, Mar. 2018.
- [12] A. Massa, D. Marcantonio, X. Chen, M. Li, and M. Salucci, "DNNs as applied to electromagnetics, antennas, and propagation—A review," *IEEE Antennas Wireless Propag. Lett.*, vol. 18, pp. 2225–2229, 2019.

- [13] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.*, vol. 378, pp. 686–707, Feb. 2019.
- [14] W. E and B. Yu, "The deep Ritz method: A deep learning-based numerical algorithm for solving variational problems," *Commun. Math. Statist.*, vol. 6, no. 1, pp. 1–12, Feb. 2018.
- [15] C. Leake and D. Mortari, "Deep theory of functional connections: A new method for estimating the solutions of partial differential equations," *Mach. Learn. Knowl. Extraction*, vol. 2, no. 1, pp. 37–55, Mar. 2020.
- [16] H. Gao, L. Sun, and J.-X. Wang, "PhyGeoNet: Physics-informed geometry-adaptive convolutional neural networks for solving parameterized steady-state PDEs on irregular domain," *J. Comput. Phys.*, vol. 428, Mar. 2021, Art. no. 110079.
- [17] P. Ren, C. Rao, Y. Liu, J.-X. Wang, and H. Sun, "PhyCRNet: Physics-informed convolutional-recurrent network for solving spatiotemporal PDEs," *Comput. Methods Appl. Mech. Eng.*, vol. 389, Feb. 2022, Art. no. 114399.
- [18] G. Lin et al., "BINet: Learning to solve partial differential equations with boundary integral networks," 2021, *arXiv:2110.00352*.
- [19] S. Basir and I. Senocak, "Physics and equality constrained artificial neural networks: Application to forward and inverse problems with multi-fidelity data fusion," *J. Comput. Phys.*, vol. 463, Aug. 2022, Art. no. 111301.
- [20] P. Zhang, Y. Hu, Y. Jin, S. Deng, X. Wu, and J. Chen, "A Maxwell's equations based deep learning method for time domain electromagnetic simulations," in *Proc. IEEE Texas Symp. Wireless Microw. Circuits Syst. (WMCS)*, May 2020, pp. 1–4.
- [21] J. Lim and D. Psaltis, "MaxwellNet: Physics-driven deep neural network training based on Maxwell's equations," *APL Photon.*, vol. 7, no. 1, Jan. 2022, Art. no. 011301.
- [22] Z. Gong, Y. Chu, and S. Yang, "Physics-informed neural networks for solving 2-D magnetostatic fields," *IEEE Trans. Magn.*, vol. 59, no. 11, pp. 1–5, Nov. 2023.
- [23] A. Khan and D. A. Lowther, "Physics informed neural networks for electromagnetic analysis," *IEEE Trans. Magn.*, vol. 58, no. 9, pp. 1–4, Sep. 2022.
- [24] M. Baldan, P. Di Barba, and D. A. Lowther, "Physics-informed neural networks for inverse electromagnetic problems," *IEEE Trans. Magn.*, vol. 59, no. 5, pp. 1–5, May 2023.
- [25] J.-B. Zhang, D.-M. Yu, and X.-M. Pan, "Physics-informed neural networks for the solution of electromagnetic scattering by integral equations," in *Proc. Int. Appl. Comput. Electromagn. Soc. Symp. (ACES-China)*, Dec. 2022, pp. 1–2.
- [26] Y. Ge, L. Guo, and M. Li, "Physics-informed deep learning for time-domain electromagnetic radiation problem," in *IEEE MTT-S Int. Microw. Symp. Dig.*, May 2022, pp. 114–116.
- [27] S. Qi and C. D. Sarris, "Electromagnetic-thermal analysis with FDTD and physics-informed neural networks," *IEEE J. Multiscale Multiphys. Comput. Techn.*, vol. 8, pp. 49–59, 2023.
- [28] T. Shan et al., "Physics-informed supervised residual learning for electromagnetic modeling," *IEEE Trans. Antennas Propag.*, vol. 71, no. 4, pp. 3393–3407, Apr. 2023.
- [29] T. Shan, M. Li, F. Yang, and S. Xu, "Solving combined field integral equations with physics-informed graph residual learning for EM scattering of 3-D PEC targets," *IEEE Trans. Antennas Propag.*, vol. 72, no. 1, pp. 733–744, Jan. 2024.
- [30] Y. Zhang, H. Fu, Y. Qin, K. Wang, and J. Ma, "Physics-informed deep neural network for inhomogeneous magnetized plasma parameter inversion," *IEEE Antennas Wireless Propag. Lett.*, vol. 21, pp. 828–832, 2022.
- [31] G. Lin et al., "BI-GreenNet: Learning Green's functions by boundary integral network," *Commun. Math. Statist.*, vol. 11, no. 1, pp. 103–129, Mar. 2023, doi: [10.1007/s40304-023-00338-6](https://doi.org/10.1007/s40304-023-00338-6).
- [32] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814.
- [33] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," 2015, *arXiv:1511.07289*.
- [34] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," 2016, *arXiv:1606.08415*.
- [35] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," 2017, *arXiv:1710.05941*.
- [36] S. Elfving, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," 2017, *arXiv:1702.03118*.
- [37] D. Misra, "Mish: A self regularized non-monotonic activation function," 2019, *arXiv:1908.08681*.
- [38] H. W. H. Wang, L. L. Lu Lu, S. S. S. Song, and G. H. G. Huang, "Learning specialized activation functions for physics-informed neural networks," *Commun. Comput. Phys.*, vol. 34, no. 4, pp. 869–906, Jan. 2023.
- [39] A. D. Jagtap, K. Kawaguchi, and G. E. Karniadakis, "Adaptive activation functions accelerate convergence in deep and physics-informed neural networks," *J. Comput. Phys.*, vol. 404, Mar. 2020, Art. no. 109136.
- [40] A. D. Jagtap, K. Kawaguchi, and G. E. Karniadakis, "Locally adaptive activation functions with slope recovery for deep and physics-informed neural networks," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 476, no. 2239, Jul. 2020, Art. no. 20200334.
- [41] L. R. Sütfeld, F. Brieger, H. Finger, S. Füllhase, and G. Pipa, "Adaptive blending units: Trainable activation functions for deep neural networks," in *Advances in Intelligent Systems and Computing*, 2020, pp. 37–50.
- [42] C. Trabelsi et al., "Deep complex networks," 2017, *arXiv:1705.09792*.
- [43] X.-M. Pan, B.-Y. Song, D. Wu, G. Wei, and X.-Q. Sheng, "On phase information for deep neural networks to solve full-wave nonlinear inverse scattering problems," *IEEE Antennas Wireless Propag. Lett.*, vol. 20, pp. 1903–1907, 2021.
- [44] B.-W. Xue, R. Guo, M.-K. Li, S. Sun, and X.-M. Pan, "Deep-learning-equipped iterative solution of electromagnetic scattering from dielectric objects," *IEEE Trans. Antennas Propag.*, vol. 71, no. 7, pp. 5954–5966, Jul. 2023.
- [45] K. R. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *J. Big data*, vol. 3, no. 1, pp. 1–40, 2016.
- [46] L. Guo, G. Song, and H. Wu, "Complex-valued Pix2pix—Deep neural network for nonlinear electromagnetic inverse scattering," *Electron.*, vol. 10, no. 6, p. 752, 2021. [Online]. Available: <https://www.mdpi.com/2079-9292/10/6/752>
- [47] H. M. Yao, W. E. I. Sha, and L. Jiang, "Two-step enhanced deep learning approach for electromagnetic inverse scattering problems," *IEEE Antennas Wireless Propag. Lett.*, vol. 18, pp. 2254–2258, 2019.
- [48] H. M. Yao, L. Jiang, and W. E. I. Sha, "Enhanced deep learning approach based on the deep convolutional encoder–decoder architecture for electromagnetic inverse scattering problems," *IEEE Antennas Wireless Propag. Lett.*, vol. 19, pp. 1211–1215, 2020.
- [49] J.-Y. Wang and X.-M. Pan, "Universal approximation theorem and deep learning for the solution of off-frequency domain electromagnetic scattering problems," *IEEE Trans. Antennas Propag.*, vol. 72, no. 12, pp. 9274–9285, Oct. 2024.
- [50] X. Chen, "Subspace-based optimization method for solving inverse-scattering problems," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 1, pp. 42–49, Jan. 2010.
- [51] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.