2022 International Applied Computational Electromagnetics Society Symposium (ACES-China) | 978-1-6654-5236-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/ACES-China56081.2022.10064917

A Poisson's Equation Solver Based on Neural Network Precondtioned CG Method

Tianchen Shao, Tao Shan, Maokun Li, Fan Yang and Shenheng Xu
Beijing National Research Center for Information Science and Technology,
Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China.
Email: maokunli@tsinghua.edu.cn

***Abstract*—In this study, we investigate the feasibility of utilizing deep learning technique to construct preconditioners for iterative matrix solvers. A neural network (NN) is proposed to simulate the optimum-preconditioner's mapping properties, and participates in NN-precondtioned conjugate gradient (NN-PCG) method. Training and testing sets are generated by finite difference method (FDM). Numerical examples demonstrate that compared to conjugate gradient (CG) method, NN-PCG significantly improves convergence performance on solving 2-D Possion's equation.**

***Index Terms*—deep learning, neural network, conjugate gradient method, preconditioner**

I. Introduction

Computational electromagnetics (CEM) plays an essential role in areas of electromagnetic research and engineering. The finite difference method (FDM) [1] is among the widely used numerical algorithms in CEM. It solves Maxwell's equations by converting them into linear matrix equations. The rapid development of modern electromagnetic systems puts forward higher requirements for CEM numerical algorithms, and it is challenging to perform large-scale electromagnetic simulations while balancing numerical precision and computing speed.

To solve linear matrix equations, non-stationary iterative solvers are usually mentioned, such as CG method [2]. These solvers have advantages of favourable convergence performance and high-degree specialization, as well as small computation memory usage [3]. For iterative solvers, computing speed depends on the scale of the equation and convergence performance depends on spectrum features of the coefficient matrix. Multiple methods are applied to accelerate the computation, such as offline pre-computing, data-driven end-to-end prediction model [4], physics embedded model for integral equation solver [5] and preconditioner. Preconditioner is an algebraic technique to make spectral properties of the coefficient matrix more favourable, thus the iterative convergence process will be accelerated [3]. Typical preconditioner is constructed by matrix's structural properties, and the analysis of it is usually complex. Problem-specific preconditioner is also proposed by integrating physical priors and demonstrates better performance towards specific situations.

In this article, we utilize deep learning technique for preconditioners to accelerate convergence. Therefore, the NN-PCG method is proposed to solve 2-D Possion's equation. This NN-preconditioner is based on the U-net architecture in [6], to approximate the mapping feature of well-designed preconditioner. The training and testing sets are generated by FDM. Numerical examples show that NN-preconditioner can significantly accelerate convergence.

II. Formulation

A. Finite Difference Method on 2-D Possion's Equation

In a 2-D computational domain D, the electric potential follows the Poisson's equation as:

$$\nabla \cdot (\epsilon \nabla \phi) = -\rho \tag{1}$$

and the Dirichlet boundary condition is expressed as

$$\phi|_{\partial D} = 0 \tag{2}$$

where ϕ denotes electric potential, ρ represents excitation source, and ϵ is permittivity.

Finite difference method is used to convert the Poisson's equation to the linear equation as:

$$\frac{\epsilon_{x+\frac{1}{2},y}(\phi_{x+1,y}-\phi_{x,y})-\epsilon_{x-\frac{1}{2},y}(\phi_{x,y}-\phi_{x-1,y})}{(\Delta a)^2}+ \frac{\epsilon_{x,y+\frac{1}{2}}(\phi_{x,y+1}-\phi_{x,y})-\epsilon_{x,y-\frac{1}{2}}(\phi_{x,y}-\phi_{x,y-1})}{(\Delta b)^2} = -\frac{\rho_{x,y}}{\epsilon_0} \tag{3}$$

a, b is the length of grid edge. Relative permittivity ϵ_r is discretized as:

$$\epsilon_{x+\frac{p}{2},y+\frac{q}{2}} = \frac{\epsilon_{x,y}+\epsilon_{x+p,y+q}}{2} \quad p,q \in \{-1,0,1\} \tag{4}$$

where (x,y) is the index of the discrete subdomain.

Above equations can be organized as a linear equation:

$$A \cdot \Phi = -\bar{\rho} \tag{5}$$

Φ denotes potential matrix and is nonlinear to ϵ and ρ. Coefficient matrix A is Hermitian, so CG method is used to solve the equation.

Residual is proposed to evaluate the accuracy of calculated potential in step i, which is defined as:

$$r^{(i)} = -\bar{\rho} - A \cdot \Phi^{(i)} \tag{6}$$

Algorithm 1: NN-Preconditioned Conjugate Gradient Method

Compute $r^{(0)} = b - Ax^{(0)}$ for some initial guess $x^{(0)}$;
for $i = 1, 2, \ldots$ **do**
  if $i = 1$ **then**
    $z^{(0)} = \mathbf{NN}(r^{(0)}, Sig)$;
    $\rho_0 = {r^{(0)}}^T z^{(0)}$;
    $p^{(1)} = z^{(0)}$;
  else
    $\rho_{i-1} = {r^{(i-1)}}^T r^{(i-1)}$;
    $\beta_{i-1} = \rho_{i-1} / \rho_{i-2}$;
    $p^{(i)} = r^{(i-1)} + \beta_{i-1} p^{(i-1)}$;
  end
  $\alpha_i = \rho_{i-1} / {p^{(i)}}^T A p^{(i)}$;
  $x^{(i)} = x^{(i-1)} + \alpha_i p^{(i)}$;
  $r^{(i)} = r^{(i-1)} - \alpha_i A p^{(i)}$;
  check convergence; continue if necessary;
end

Fig. 1. Workflow of NN-PCG method.

B. NN-Preconditioned Conjugate Gradient Method

In this section, we propose a CG method to solve the Poisson's equation in which the preconditioner is structured by a neural network. The pseudocode of NN-PCG method is given in Fig. 1.

For linear equation (5), the optimum-preconditioner is $M = A^{-1}$, by which the iteration can achieve convergence in one step. Therefore, the NN-preconditioner should approximate the algebraic feature of A^{-1}. Mapping relation between Φ and $\bar{\rho}$ demonstrates the function of A^{-1}. We use a neural network as the preconditioner, and it is trained to predict the potential with permittivity distribution and source location.

In CG method, good convergence performance is maintained by the orthogonality of the residual sequence $\{r^{(i)}\}$. For the preconditioner M, the residual sequence should also be M^{-1} orthogonal. Due to the nonlinearity of the neural network, introducing NN-preconditioner into each iterative step will break orthogonality. Hence, NN-preconditioner only participates in the first iterative step. It also results in that NN-PCG method will not increase excessive calculation.

Deep neural network is based on the U-net architecture, and it is composed of seven encoder-decoder layers with skip connections. In each layer, up-sampling and down-sampling operation is implemented by convolution layers to avoid checkerboard artifacts, and improve the prediction accuracy [7]. We take permittivity distribution and source location as the input of the neural network, and potential distribution calculated by FDM as the label. These tensors can be expressed as 2-dimensional arrays. Adam optimizer is utilized to minimize the cost function expressed as relative error [4].

III. Numerical Examples

In this section, we respectively deploy NN-PCG method and CG method in numerical examples to test the convergence performance. 1000 randomly generated samples are utilized to benchmark these two methods. A sample consists of a permittivity array, a source array and a potential array. The computational region is $6.4m \times 6.4m$ and discretized into 64×64 grids. For the permittivity array, four ellipses' semi-axes vary from 1 to 8 and their permittivity values are randomly selected in $[1.25, 2.5, 5, 10, 20]$, while background relative permittivity is 10. For the source array, electric charge is set as $\rho(\mathbf{r})/\epsilon_0 = 100V/m^2$, and the source location is randomly distributed in a limited subdomain. It is noted that the source location cannot be inside the ellipses' area. We use 8000 examples as the training set and 2000 examples as the testing set to update and evaluate the neural network.

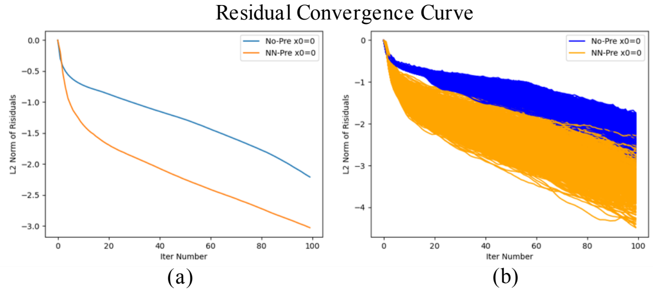

Fig. 2. Residual convergence curve of NN-PCG and CG for 1000 samples. (a) Average convergence curve. (b) Individual convergence curve.

The convergence performance of NN-PCG and CG is shown in Fig. 2. The result indicates that the NN-preconditioner can significantly improve the convergence performance compared to CG method. At the same iterative step, the residual of NN-PCG method is up to 1dB smaller than that of CG method.

IV. Conclusion

In this study, we propose the NN-PCG method, in which a deep neural network is utilized as a NN-preconditioner and embedded in CG mathod. Numerical examples demonstrate that the NN-PCG method can improve the convergence speed.

References

[1] J.-M. Jin, *Theory and computation of electromagnetic fields*. John Wiley & Sons, 2011.
[2] J. R. Shewchuk *et al.*, "An introduction to the conjugate gradient method without the agonizing pain," 1994.
[3] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst, *Templates for the solution of linear systems: building blocks for iterative methods*. SIAM, 1994.
[4] T. Shan, W. Tang, X. Dang, M. Li, F. Yang, S. Xu, and J. Wu, "Study on a fast solver for poisson's equation based on deep learning technique," *IEEE Transactions on Antennas and Propagation*, vol. 68, no. 9, pp. 6725–6733, 2020.
[5] R. Guo, T. Shan, X. Song, M. Li, F. Yang, S. Xu, and A. Abubakar, "Physics embedded deep neural network for solving volume integral equation: 2d case," *IEEE Transactions on Antennas and Propagation*, 2021.
[6] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.
[7] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," *Distill*, vol. 1, no. 10, p. e3, 2016.