Machine Learning Algorithms for EM Wave Scattering Problems

Anthony James McElwee, *MEng Student, DCU*

***Abstract* – This paper details the construction and evaluation of a deep learning emulator, Prescient2DL, to assist a Method of Moments (MoM) iterative solver, SolverEMF2, in generating solutions to two-dimensional, H-polarization, electromagnetic scattering problems at the 10 MHz range. The acceleration of conventional solvers at this frequency is of particular interest to the medical community where existing methods face computational disadvantage due to the high contrast nature of the scenes. Recent referenced works report successes in the general area of applying machine learning to electromagnetic scattering problems, however, there are conflicting testimonies to the potency of the efforts. This paper outlines a statistical experiment to assess the impact of the hybrid methodology, where Prescient2DL contributes to SolverEMF2. Experimental evidence indicates a considerably lower initial error than that associated with purely conventional solvers. However, negligible impact on more important metrics associated with conventional solvers is also reported. The paper also records a simple test of generalizability for Prescient2DL where results indicate a degradation in model performance. Finally, the coupling of the two predicted electric fields into a second stage model in an effort to reduce initial error fails to yield positive results.**

Index Terms* – *computational electromagnetics, deep learning, VEFIE, Volume Electric Field Integral Equation, Transverse Electric, Contrast-Source Integral Equations, U-net, scientific emulation, forward problem, frequency domain

I. Introduction

Medical diagnostic tools, such as biological segmentation and classification models, constitute a methodology that can increase the capacity for healthcare practitioners to rapidly and accurately differentiate between benign and malignant biological tissue. Developing such aides requires the generation of large quantities of synthetic data using frequency-domain electromagnetic scattering simulations. The development of such simulations necessitates considerable learning investment [1].

As extolled in [2], significant benefits to patients and medical practitioners could arise through the deployment of Magnetic Induction Tomography (MIT). This requires the acceleration of high-contrast simulation scenarios in the 10 MHz carrier frequency range.

Generally, these simulations operate with a constrained set of input parameters, such as incident source wave configurations and dielectric material attributes of scatterers. Although input parameters are comparable across simulation incidences, conventional methods require full wave simulation and cannot be estimated with very low or high-frequency approximations. Generating large volumes of such simulations is currently uneconomical.

The motivation of this paper is to report on the construction of a electromagnetic scattering solver workflow, SolverEMF2, for a toy problem with low contrast values for an incident wave frequency of 10 MHz SolverEMF2 adapts code from [1] to use the Krylov iterative solver Biconjugate Gradient Stabilized Method (BICGSTAB), to calculate the solution to contrast-source integral equations. This high performance code uses circular convolutions to accelerate multiplication steps via Fast Fourier Transforms. SolverEMF2 is used to create a training data set for developing a deep learning model called Prescient2DL. Prescient2DL can feed back into SolverEMF2 to assist in the provision of solutions to the scattering simulations. Experiments to establish the impact of infusing Prescient2DL into SolverEMF2 are provided with commentary.

A. Problem Specification

The paper reports on the forward H-polarization problem, otherwise known as the Transverse Electric (TE) problem, solving for the electric field strength in a domain with two contrast scatterers, one inside the other, receiving a dipole incident wave with E_x, E_y and ZH_z components[1]. The formulation uses the Laplace convention derived in [1][2]. The problem requires finding two electric fields and as a result is a full vector problem. As the task motivation is based around future medical developments, and the permeability of biological tissues can be considered roughly equal that of the background vacuum embedding of the domain, no permeability contrast is assumed [2]. The embedding medium, in this paper a vacuum, has an electromagnetic impedance of $Z_0 = \mu_0 c_0$ and popagation coefficient $\hat{\gamma}_0 = s/c_0$, where μ_0 is the permbeability and c_0 is the wave speed within the embedding. The incident waves are generated by a vertical electric-dipole line source and are given by the following formulae:

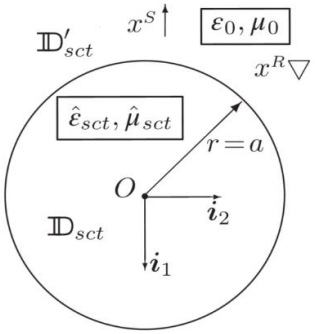

Fig. 1. Canonical Problem Diagram

$$\hat{E}_1^{inc}(x_T|x_T{}^S) = -\frac{Z_0\hat{M}}{\hat{\gamma}_0}(-\hat{\gamma}_0{}^2 + \partial_1\partial_1)\hat{G}(x_T - x_T{}^S)$$

$$\hat{E}_2^{inc}(x_T|x_T{}^S) = -\frac{Z_0\hat{M}}{\hat{\gamma}_0}\partial_2\partial_1\hat{G}(x_T - x_T{}^S)$$

$$Z_0\hat{H}_3^{inc}(x_T|x_T{}^S) = \frac{Z_0\hat{M}}{\hat{\gamma}_0}\hat{\gamma}_0\partial_2\hat{G}(x_T - x_T{}^S)$$

where the 2D Green's function is given by $\hat{G}(x_T) = \frac{1}{2\pi}K_0(\hat{\gamma}_0|x_T|)$. The modified Bessel function of the second kind with second order is denoted by K_0. The electric-dipole momen is denoted by $\hat{M}$. A simplifying assumption is made that the $Z_0\hat{M} = \hat{\gamma}_0$. All other incident components are zero.

As the model assumes that there is an invariance in the permittivity contrast in the z-direction, the corresponding

Date of submission: 2023/08/21. e-mail: anthony.mcelwee2@mail.dcu.ie
Note: The author of this paper will be referred to as "the student" to avoid confusion.

[1] This is a more advanced problem than the E-polarisation problem described in the Literature Review appendix.

[2] The lengthy, full derivation and explanation of the problem are given in section 3.2.1 of [1].

equation for the total electric field using contrast source notation is as follows:

$$\hat{\chi}^{\varepsilon}\hat{E}_j^{\,inc}(x_T) = \hat{w}_j^{\,E}(x_T) - \hat{\chi}^{\varepsilon}\left(\hat{\gamma}_0^{\,2}\delta_{j,k} - \partial_j\partial_k\right)\int_{x_T'\in D_{sct}} \hat{G}(x_T - x_T')\hat{w}_k^{\,E}(x_T')\,dA$$

where $\hat{\chi}^{\varepsilon}(x_T) = 1 - \frac{\hat{\varepsilon}_{Sct}(x_T)}{\varepsilon_0}$ and the electric contrast source vector is $\hat{w}_k^{\,E}(x) = \hat{\chi}^{\varepsilon}(x)\hat{E}_k(x)$ [1]. An indicator function $\delta_{j,k}$ assumes the property that $\delta_{j,k} = 1$ if j and k are equal, otherwise it is zero. Thus two solutions are required to solve for the electric contrast sources which can then be used to solve for the electric field components at the recievers given by $\hat{E}_j^{\,sct}(x_T^{\,R}) = \int_{x_T'\in D_{sct}}\left(\hat{\gamma}_0^{\,2}\delta_{j,k} - \partial_j^{\,R}\partial_k^{\,R}\right)\hat{G}(x_T^{\,R} - x_T')\hat{w}_k^{\,E}(x_T')\,dA.$
The permittivity contrasts also assume only a real component with zero conductivity and are frequency independent. This is a permittivity contrast only problem. The diagram, adapted from [1], illustrates the canonical version of the problem. A radially orientated electric dipole emitting incident waves is located at a fixed x-axis location in the negative direction above the larger cylindrical scatterer. Receivers used to validate the solver against a Bessel Function Approach form a ring around the main scatterer lying between the source and the scatterer boundary. Within the main scatterer lies a second scatter, not pictured, with contrast different to both the background embedding and main scatterer.

II. Existing work in the domain of electromagnetic scattering and machine learning

This is a survey of the state of the art. It should be more than a list of citations of prior work. Give this section a title relevant to your project ("Existing techniques for chronological displacement"). Organize prior work in groups and evaluate them. What are their common features, strengths and weaknesses? This section should be persuading the reader that there is a gaping hole in the research literature, and hint that the technique you are about to describe will fill that hole. The prior art on which you base this section will have already been discussed by you in your Literature Survey. However, you should have greater insight into prior research now, having completed your own project. Do not simply cut and paste text from your literature survey into this section rewrite it so that it is concise enough to meet the length requirements of a research paper and to reflect your improved understanding of your research topic.

End of page 1 (0 pages)

- Look at [2] section A.
- Look at the project log!
- DO THIS WHEN OTHER SECTIONS ARE COMPLETE TO MANAGE SPACE. IF YOU NEED TO BULK OUT THEN DO SO, OTHERWISE NOTE THAT THE PROJECT LOG IN THE APPENDCICES CONTAINS MORE COMMENTRAY AND THE LITERATURE REVIEW REMAINS RELEVANT.

III. Technical Description

This section details the key aspects in the overall workflow required to test experimental hypothesis surround the hybrid-methodology as outlined in the abstract and introduction.

B. Conventional Solver Creation

As noted in the introduction, generating solutions to forward electromagnetic scattering problems is a vast, time-intensive task. MATLAB code, provided in conjuction with [1], was translated by the student to Python and then adapted to generate solutions in a bid to accelerate the experimental development. The source of the original code eminates from an extremely experienced researcher that is cited recurrently in other references consulted during the investigation of the paper[3]. Some key elements of the code are now described below[4].

1) Problem Formula Description

The problem is formulated in the s-domain (Laplace convention).

2) Green's Function

There are points where the Green's Function is singular.

3) Core Solver

BICGSTAB and why that algorithm was choosen

4) FFT Acceleration Technique

Power of 2 requirement, circulant matrices forming the convolutions can be computed using FFT more efficiently.

5) Bessel-Approach Validation

- *images of the validation in the appendices*

End of page 2 (1 pages)

[3] The final adapted code is attached as an appendix. Considerable effort has been made to maintain the original structure of the code as a source of truth so that it can be used more widely in future research efforts, as well as be tied back to the reference text for documentation. Significant gains have been made in the last decade in machine learning due to the open and transparent nature of shared code. The aspiration is that this adaptation can add to this development in the electromagetic scattering domain.

[4] The code is fully documented over the breadth of more than one hundred pages in [1] and is the culmination of over 50 articles.

C. Dataset Generation Description

Three types of dataset were generated to conduct experimental analysis: major base dataset with two contrast scatterers (DS1); minor single lower-contrast scatterer dataset for testing model generalisability of negative sample cases (DS2); minor single higher-contrast scatterer dataset for testing model generalisability to different contrast population (DS3). The input parameters for all simulations were the same except for the scatterer geometry.

The final generation of the scatterer geometry was kept minimial and close to the canonical formulation in order to reduce generation time. A variation on the generator used to validate against the Bessel-Approach was adapted to create DS1, DS2 and DS3. In all cases, cells outside the major scatterer area were replaced with the zero contrast value, as illustrated in the [CHECK: Figure used to show the differences between the 3 datasets].

DS1 geometery contained one higher-contrast scatterer, $\varepsilon_r = 1.75$ inside a geometrically larger lower contrast-scatterer $\varepsilon_r = 1.25$. The centre point of the lower-contrast scatterer was allowed to be within the a distance from the domain origin of its own radius ensuring that it was contained entirely within the domain simulation grid. Both scatterers were of constant fixed size with the smaller scatter populating 5% of the area of the larger scatterer. A seeded random number generator was used to shift the smaller scatterer within a range where at least one cell of scatterer would exist within the boundary of the main scatterer in order to mimic a true-positive sample in a biomedical screening scenario.

DS2 has the same geometric rules as DS1 except that the higher-contrast value was set to $\varepsilon_r = 1.0$, thus forming a vacuum void within or piercing the larger scatter. This is equivalent to generating cases where no secondary tissue exists in the simulation domain, the case of true negatives.

DS3 has all contrast values set to the higher $\varepsilon_r = 1.75$ value to simulate a total shift in permitivity values. This is also a true negative scenario which tests the models ability to generalise to large grid populations not seen at training time. The computational time for generating these data samples also increased in contrast to DS1 and DS2.

[CHECK: Figure, sample of DS1 DS2 DS3 showing different geometric properties.]

The carrier frequency is set at 10 MHz and the highest permitivity contrast, $\varepsilon_r = 1.75$, resulted in the smallest wavelength of 22.7m. The source emmiter is located 170m in the negative x direction. A grid dimension of 128x128 was choosen in order to comply with the FFT requirement that the grid be an integer power of 2, and the typical computer vision approach of using grids divisible by 32. The grid delta was 2m giving rise to a sample per cell of 11. Training a model where the grid dimension is greater than 128 becomes computationally difficult as memory issues arise when the number of layers in the deep learning architecture increases. Although the material contrast parameters in the medical domain are much more extreme for incident frequencies at 10 MHz [2] [CHECK: Change this source to something more direct around contrast values], in order to initiate research in the general area a much lower contrast value was choosen to allow for a large volume of samples to be generated in a shorter time frame. All iterative solver computations were carried out on a local laptop CPU i7-11800H @ 2.3 GHz using Python, in particular NumPy and SciPy libraries. DS1 samples took 1 seconds per sample, while DS2 samples took 0.75 seconds per sample and DS3 samples took 2.2 seconds per sample.

- Put in geometry generation algorithms: base scatter; internal scatterers with base centered at origin; validation scatterers shifted around domain.

The generated data was saved in NumPy format on an external hard drive due to the excessive collective size of the samples. The outputs saved for each base sample, aside form the scatter geometry array, with each field splitting the real, imaginary and absolute components of the complex field by channel were as follows: the incident E field in the x and y direction; the ZH field in the z direction; the two solved scattered electric fields in the x and y directions.

In addition, two extra files were saved seperately to document the properties of the generated sample: a PNG file illustrating the contained scatter geometry; a separate NumPy file documenting the iterative solver run information documented in the experimental results section later in the paper.

- Illustrate with diagram similar to figure 2 in [3] that covers the incident & geometry inputs and the model architecture and the output predicted fields.
- Illustrate the training model dataset similar to figure 5 in [3] to show static nature of main scatterer.

D. Train/Test/Split Approach

Usually, independence is required between the training data, the validation data and the testing data to ensure that the model is actually learning. In this paper, the approach taken was to first establish if the model could emulate the generated examples. Due to the nature of the geometric set-up, only 1264 cells exist within the main scatterer. Although every scene is guarenteed to contain two scatterers, there are only so many samples that can be generated before samples start repeating. The data was saved in folders of 1000 samples. At the train/validate/test splitting stage, 20% of the 1000 samples were retained for testing (200) and 20% of the training data was retained for validation (160) leaving 640 samples for training. The base model was trained on 46 folders of samples and tested in terms of impact on the SolverEMF2 on the final folder of 1000 samples. Within the final testing folder there were

Dataset	Sample Count #			

E. Architecture Description

- for the iterative wrapping, the predicted scattered field needs to be transformed back into the contrast source formulation though the use of the FFT accelerated operator equation.

All deep learning computations were carried out on a local laptop GeForce RTX 3070 GPU using Tensorflow and Keras Python libraries.

The training inputs were the scatter geometery as a single channel, followed by the real, imaginary and absolute values of the complex incident wave bringing the total input channel count to four. The outputs were the real and imaginary components of the electric field in the dimension of interest bringing the output count to two channels. This resulted in a

requirement to train two models, one for each field, in order to provide an initial guess to the SolverEMF2 workflow. The target arrays was standardised to a range of [0, 1]. This allows the model to train faster. In order to provide the predictions as inputs in SolverEMF2, the process is reversed at the provision of initial guess time. The information required to perform this standardisation can be estimated from a small group of sample solutions making it a robust and simple way to accelerate the training process[5].

1) Base Model Architecture

The base architecture was a 37 layer U-net type architecture using 'Elu' activation functions as recommended in [4]. All layers, where possible, had bias terms included in their configurations. Initially a batch normalisation layer is applied at the input and each downsampling block afterwards contains a convolution layer that has stride two that double the channel count and half the width and height of the input. This is followed by a convolution layer with a single stride to add enhanced complexity to the model. The bottom layer is a bottleneck convolution layer that brings the dimension to 2x2x256. On the upscaling decoding side of the U-net, a transposed convolution layer with stride two halves the channel count and doubles the dimensions. This is followed by a convolution layer to increase the model complexity. The encoder side is connected to the decoder side via skip concatenation layers. In order to add non-linearity after the concatenation step, a convolution layer is added on the decoding side for each skip connection. Max pooling and Up-Sampling were not implemented due to higher error rates near the domain boundaries that caused a degradation in the intial residual error metrics.

- How is model saved and loaded
- Detail the reasoning behind activation function choice and architecture (use references)
- Loss function as an equation

F. Hybrid-Methodology Infusion

- Put in an algorithmic flow chart like in [5] figure 1
- Discuss linear convolution layers using appendix piece

End of page 3 (1 pages)

[5] The description of the process is contained in Appendix PDI. [CHECK REF ZOTORO HERE]

IV. Results Obtained

Document your results here. Use tables and scatter plots, histograms, etc. to present numerical results. Make sure that the scenario used to obtain each set of results is described unambiguously. There should be sufficient information in this section and the previous one for the interested reader to replicate your results. Put some thought into how you visualize your results. If you generated lots of data, should it be presented in a 3D plot? On multiple 2D plots? What scales should use use? Log? Linear? If you use colour in your plots, will the traces still be distinguishable if printed in monochrome? Describe how you know your results are valid. What testing strategies were used? Were enough results obtained? Does your algorithm perform correctly? Does your code implement your algorithm accurately? Does your input data set contain features of the kind the algorithm is supposed to extract?

End of page 4 (1 page)

LOOK AT THE HYPOTHESIS IN THE PROJECT PLAN PROPOSAL

H. Dataset Generation Statistics

- Put in descriptive statistics around the time to generate the naïve guess and the model informed guess data samples.

I. Model Development Results

- Put in loss curves

J. Comparing Model Output to Full-Solution on single instance basis

- This is not the batch stats but instead a comparison based on MSE etc between an instance from model and original. How do they differ? Are there structurial issues etc?

V. Analysis

Interpret your results here. You've obtained lots of data. What have you learned from it? Does accepting transit traffic overload the router? What do the peaks in the spectral response indicate? Why are there no fluctuations in the EEG data? This section should be ONE PAGE in length. The division of the body of the report into three sections (here named "Technical Description", "Results Obtained" and "Analysis") may be inappropriate for some projects. If you wish to change this structure, you may do so only in consultation with your supervisor and only with his/her written agreement to the revised structure any such revised paper format must have an aggregate length of four pages for the sections equivalent to the above four.

End of page 5 (1 page)

- Is this project not a waste of energy and time…

K. Experimental Results

- T-tests for the three metrics from the project plan proposal. There needs to be results for the two architectures tested (final architecture and alternative with max pooling) and there needs to be results for the moved geometry validation set too.
- Wobbley training line in loss curve at the end means the model is starting to overfit

There are two out of distribution tests changing the contrast values. One frops cancer to zero contrast. Slight degradation in performance but visually not too bad. The second ups the main scatter to that of cancer. This takes double the amount of iterations to solve…what is the result? Visually poor but statistically still lower error than naïve guess.

VI. Conclusions

This is the conclusion. *Here you summarize what has been achieved and learned, and the implications for future research and suggestions for future work that could follow on from your work. This section resembles the introduction in some ways, but remember that by now the reader has read the body of the paper. The introduction was your attempt to encourage them to do so. You can present insights in the conclusions.*

References

[1] P. M. van den Berg, *Forward and inverse scattering algorithms based on contrast source integral equations*. Hoboken, NJ: Wiley, 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/book/10.1002/9781119741602

[2] P. De Tillieux and Y. Goussard, 'Improving the Computational Cost of Image Reconstruction in Biomedical Magnetic Induction Tomography Using a Volume Integral Equation Approach', *IEEE Trans. Antennas Propagat.*, vol. 69, no. 1, pp. 366–378, Jan. 2021, doi: 10.1109/TAP.2020.3008618.

[3] S. Qi, Y. Wang, Y. Li, X. Wu, Q. Ren, and Y. Ren, 'Two-Dimensional Electromagnetic Solver Based on Deep Learning Technique', *IEEE Journal on Multiscale and Multiphysics Computational Techniques*, vol. 5, pp. 83–88, 2020, doi: 10.1109/JMMCT.2020.2995811.

[4] Q. Ren, Y. Wang, Y. Li, and S. Qi, *Sophisticated Electromagnetic Forward Scattering Solver via Deep Learning*. Singapore: Springer, 2022. doi: 10.1007/978-981-16-6261-4.

[5] R. E. Meethal *et al.*, 'Finite element method-enhanced neural network for forward and inverse problems', *Advanced Modeling and Simulation in Engineering Sciences*, vol. 10, no. 1, p. 6, May 2023, doi: 10.1186/s40323-023-00243-1.