Ollscoil Chathair
Bhaile Átha Cliath
Dublin City University

School of Electronic Engineering

CB54: Machine Learning Algorithms for EM Wave Scattering Problems

Portfolio

Anthony James McElwee
ID Number: 20211330

August 2023

MEng in Electronic and Computer Engineering

Supervised by Dr Conor Brennan

Declaration

I hereby declare that, except where otherwise indicated, this document is entirely my own work and has not been submitted in whole or in part to any other university.

Signed: Anthony James Mc Elwee Date: 2023/08/20

Acknowledgements

The student would like to acknowledge the time that Dr Conor Brennan spent on video calls during the duration of the project and the advice offered at various junctions. In return, it is hoped that the student has been able to meet some of the expections envisioned at the initial stages of the process.

School of Electronic Engineering

CB54: Machine Learning Algorithms for EM Wave Scattering Problems

Conference Paper

Anthony James McElwee
ID Number: 20211330

August 2023

MEng in Electronic and Computer Engineering

Supervised by Dr Conor Brennan

Declaration

I hereby declare that, except where otherwise indicated, this document is entirely my own work and has not been submitted in whole or in part to any other university.

Signed: Anthony James Mc Elwee   Date: 2023/08/20

Machine Learning Algorithms for EM Wave Scattering Problems

Anthony James McElwee, *MEng Student, DCU*[1]

***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 electromagnetic scattering problems at 10 MHz. The acceleration of conventional solvers at this frequency is of interest to the medical community where existing methods face computational challenges due to the high-contrast nature of the scenes. Recent works report successes in the area of applying machine learning to electromagnetic scattering problems, however, there lacks documentation as to the potency of combining such models with conventional methods. This paper outlines a statistical experiment to assess the impact of Prescient2DL on SolverEMF2. Experimental evidence indicates a considerably lower initial error than that associated with purely conventional solvers. However, negligible impact on metrics associated with conventional solvers is also reported. Finally, the paper records two simple tests of generalizability where results indicate a degradation in model performance.**

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

I. Introduction

Development of medical segmentation models to accurately differentiate between benign and malignant biological tissue require the generation of large volumes of frequency-domain electromagnetic scattering simulations. The development of such simulations necessitates considerable learning investment and is considered uneconomical [1]. As extolled in [2], benefits to medical practitioners could arise through the deployment of Magnetic Induction Tomography, however, this requires the acceleration of high-contrast simulations for 10 MHz carrier frequencies. Generally, these simulations operate with a constrained set of input parameters, such as fixed incident source wave configurations. Although input parameters are comparable across simulation incidences, conventional methods still require full wave simulation.

The motivation of this paper is to report on the construction of SolverEMF2 for a toy problem with low-contrast values at 10 MHz. SolverEMF2 uses the Biconjugate Gradient Stabilized Method (BICGSTAB) to calculate solutions of contrast-source integral equations. This high performance code uses circular convolutions to accelerate multiplication steps via Fast Fourier Transforms (FFT). SolverEMF2 is then used to create a data set for developing 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 vector 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. This VEFIE formulation uses the Laplace convention derived in [1] and is solved using a conventional MoM methodology[2]. It is assumed that all wave quantities depend sinusoidally on time with a common angular frequency ω. Since the permeability of biological tissues can be considered roughly equal to that of the background vacuum embedding, no permeability contrast is assumed [2]. The embedding medium has an electromagnetic impedance of $Z_0 = \mu_0 c_0$ and propagation coefficient $\hat{\gamma}_0 = s/c_0$, where μ_0 is the permbeability and c_0 is the wave speed within the embedding. It is assumed that no sources exist within the scatterers. The incident waves are generated by a vertical electric-dipole line source and their components are:

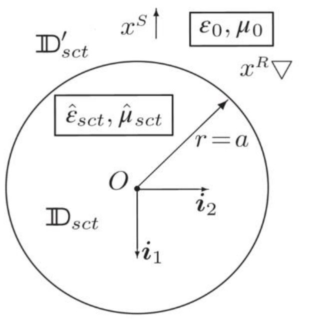


Figure 1. Canonical Problem Diagram

$$\hat{E}_1^{inc}(x_T|x_T^S) = -\frac{Z_0\hat{M}}{\hat{\gamma}_0}\left(-\hat{\gamma}_0^{\,2} + \partial_1\partial_1\right)\hat{G}(x_T - x_T^S), \qquad (1)$$

$$\hat{E}_2^{inc}(x_T|x_T^S) = -\frac{Z_0\hat{M}}{\hat{\gamma}_0}\partial_2\partial_1\hat{G}(x_T - x_T^S), \qquad (2)$$

$$Z_0\hat{H}_3^{inc}(x_T|x_T^S) = \frac{Z_0\hat{M}}{\hat{\gamma}_0}\hat{\gamma}_0\partial_2\hat{G}(x_T - x_T^S), \qquad (3)$$

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 moment is denoted by $\hat{M}$. A simplifying assumption is made such that $Z_0\hat{M} = \hat{\gamma}_0$. All other incident components are zero. The contrast is $\hat{\chi}^{\varepsilon}(x_T) = 1 - \frac{\hat{\varepsilon}_{sct}(x_T)}{\varepsilon_0}$ and the electric contrast source vector is $\hat{w}_k^{\,E}(x) = \hat{\chi}^{\varepsilon}(x)\hat{E}_k(x)$ [1]. As the scene assumes that there is invariance in permittivity contrast in the z-direction, the corresponding equation used to provide the basis to solve for the total electric field is as follows:

$$\hat{\chi}^{\varepsilon}\hat{E}_j^{\,inc}(x_T) = \hat{w}_j^{\,E}(x_T) - \hat{\chi}^{\varepsilon}\left(\hat{\gamma}_0^{\,2}\delta_{j,k} - \partial_j\partial_k\right)\int_{x_T' \in D_{sct}} \hat{G}(x_T - x_T')\hat{w}_k^{\,E}(x_T')\,dA. \quad (4)$$

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 scattered electric field components $\hat{E}_j^{\,sct}(x_T^{\,R}) = \int_{x_T' \in D_{sct}}\left(\hat{\gamma}_0^{\,2}\delta_{j,k} - \partial_j^{\,R}\partial_k^{\,R}\right)\hat{G}(x_T^{\,R} - x_T')\hat{w}_k^{\,E}(x_T')\,dA.$ The permittivity contrasts are frequency independent and assume only a real component with zero conductivity. The diagram, adapted from [1], illustrates the canonical version of the problem. Receivers used to validate the solver against a Bessel-Approach form a ring around the main scatterer lying between the source and the scatterer boundary [4][3].

---

[1] Date of submission: 2023/08/21. e-mail: anthony.mcelwee2@mail.dcu.ie

[2] This is a more advanced problem than the E-polarization problem described in [3]. The lengthy derivation and explanation of the H-polarization problem are given in Section 3.2.1 of [1].

[3] Validation is illustrated in [4].

II. Existing work in deep learning and electromagnetic scattering problems

This is a survey of the state of the art. It should be more than a list of citations of prior work. 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.
- DENSE is useful not just for the meta stuff but because it anchors the use of emulators as sub-steps within larger simulations, they are used as approximating subcomponents [CHECK: is this stated?] whereas this project is trying to go beyond that usage of deep learning.
- Based on lit review, the U-net architecture was choosen as the intial starting point fro experimentation.

III. Technical Description

B. Conventional Solver Creation

As noted in the introduction, generating solutions to forward electromagnetic scattering problems is a potentially complicated, time-intensive task. MATLAB code, provided in conjunction 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 emanates from an extremely experienced researcher that is cited recurrently in other references consulted during the investigation of this paper[4].

Equation 4 is defined continuously over the domain, thus giving rise to an infinite number of linear equations with an infinite number of unknown variables for $\hat{w}_j^E(x_T)$. The MoM scheme is used to discretize this continuous operator problem. A finite set of basis functions is used to generate a weak form of the continuous operator equation. There exists analytical solutions to the weak formulations of Green Functions but in Equation 4 singularities arise when the position vector and the source vector are equal. Details of the averaging and approximation strategies used to fully express Equation 4 are given in Chapter 1.3 of [1]. From these formulations arises a discrete operator that can be represented using circular convolutions. Through the use of a circulant matrix, the multiplication steps required to solve the MoM linear system of N equations is simplified to a vector of 2*N components. The convolution is computed using FFT which reduces the complexity of the matrix-vector multiplication from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log(N))$.

This MoM discretization, and the exploitation of the circulant properties of the operator functions, leads to the use of Iterative Krylov solvers to find a solution that minimizes a residual error criteria set by the engineer. As noted in [2], and eventually [1], the BICGSTAB solver is favored to solve these MoM electromagnetic scattering problems, with both texts reporting a significant reduction in iterations required to achieve threshold error criteria compared to other Iterative Krylov solvers. Through the use of these mathematical techniques, this high-performance Python code provides a defensible comparison, in terms of inference time, to any deep learning emulator model. The general final deep learning model prediction time is 0.3 seconds with the conventional times for the datasets given in the next section.

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 generalizability of negative sample cases (DS2); minor single higher-contrast scatterer dataset for testing model generalizability to increased higher-contrast population (DS3). The input parameters for all simulations were the same except for the scatterer contrast values.

The final generation of the scatterer geometry was kept minimal 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 figures below. It should be noted that the problem was formulated, via the MoM wrapping, as a discrete-to-discrete approximation of a continuous valued problem, rather than an image-to-image problem. The aim of the model development is to test the impact of using Prescient2DL to assist SolverEMF2, rather than provide emulator generated approximations or visualizations.

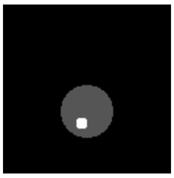


Figure. 2. DS1 geometry sample.

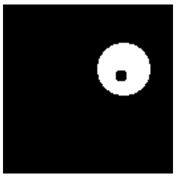


Figure 3. DS2 geometry sample.

Figure 4. DS3 geometry sample.

DS1 geometry contained one higher-contrast scatterer, $\varepsilon_r = 1.75$ inside a geometrically larger but lower contrast-scatterer $\varepsilon_r = 1.25$. The center point of the lower-contrast scatterer was allowed to be within 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

[4] The final adapted code can be found in [5]. 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 development in the electromagnetic scattering domain. The code is fully documented over the breadth of more than one hundred pages in [1] and is the culmination of over 50 articles.

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 higher-contrast scatterer would exist within the boundary of the main scatterer to mimic a 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 negative cases where no secondary tissue exists in the simulation domain.

DS3 has all contrasts set to the higher $\varepsilon_r = 1.75$ value to simulate a total shift in permitivity ε values. This is also a negative sample scenario which tests the model's ability to generalise to larger higher-contrast populations.

With carrier frequency at 10 MHz and the highest permittivity contrast, $\varepsilon_r = 1.75$, the smallest wavelength was 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 increases in the deep learning architecture. While references consulted in [3], such as [6], use very small batch sizes to overcome this issue, the dimensions were deemed sufficient to pose as a challenge for the deep learning model.

The material contrast parameters in the medical domain are much more extreme for incident frequencies at 10 MHz [2]. In order to initiate research in the general area, a much lower contrast value was chosen to allow for a large volume of samples to be generated in a shorter time frame.

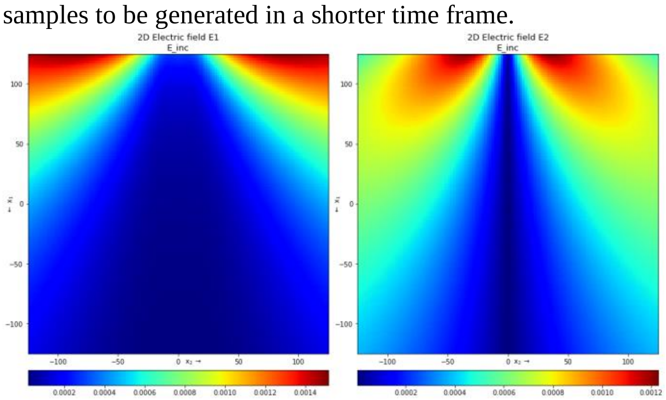


Figure 5. Electric Field incident waves in x and y direction.

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 second per sample, while DS2 samples took 0.75 seconds per sample and DS3 samples took 2.2 seconds per sample.

The generated data was saved in NumPy format. All outputs were saved by splitting the real, imaginary and absolute components of the complex fields by channel. The fields were as follows: the real component of the scatterer geometry; the incident E field in the x and y direction; the incident ZH field in the z direction; the two solved scattered E fields in the x and y directions.

In addition, two extra files were saved separately to document the properties of the generated sample: a PNG file illustrating the contained scatter geometry; a separate NumPy file documenting the iteration count, residual error and iteration duration from the iterative solver call-back information. As Equation 4 is solving for the contrast-source, the predicted scattered fields need to be transformed back into the contrast source formulation though the use of the FFT accelerated operator equation at time of inference.

D. Train/Test/Split Approach

The approach taken was to first establish if the model could emulate the conventional solutions. 5000 NumPy arrays were 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.

E. Model Development

Deep learning was carried out on a laptop GeForce RTX 3070 GPU using TensorFlow and Keras Python libraries.

The training inputs were the scatter geometry as a single channel, followed by the real, imaginary and absolute values of the complex incident wave of relevant axis bringing the total input channel count to four. The outputs were the real and imaginary components of the electric field in the axis of interest bringing the output count to two channels. This resulted in a requirement to train two models, one for each axis, in order to provide a full-dimensional, initial guess to the SolverEMF2 workflow. A key step in the pre-processing of the data was that the target arrays were standardized to a range of [0, 1]. This allows the model to train far 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 standardization can be estimated from a small group of sample solutions making it a robust and simple way to accelerate the training process[5].

1) Model Architecture

The models were fitted using a 54 layer U-net as this type of architecture was core to multiple sources consulted during background research [6]–[10]. 'Elu' activation functions were selected to add non-linearity, as recommended in [6] where experimental evidence was provided to demonstrate that their use gave rise to faster training times in comparison to 'Relu' activation functions. All layers, where possible, had trainable bias terms included in their configurations. Initially a batch normalization layer is applied at the input and each down-sampling block afterwards contains a convolution layer that has stride two. This doubles the channel count and halves 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 the encoder to the decoder side. On the upscaling decoding side of the U-net, a transposed convolution layer with stride two halves the channel count and doubles the dimensions. Upsampling layers were tried but they led to stronger grid-like lines on the output predictions. Transposed convolution layers also have more trainable parameters which increases the model capacity for handling complexity. This is followed by a convolution layer to increase the model complexity. A seeded Dropout layer with a small parameter value was included after the first stage of the decoder to increase regularisation to the model.

The encoder side is also 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 was abandoned due to degradation in initial residual error metrics.

[5] The description of the process is contained in [4].

At the penultimate stage of the decoder, two linear convolution layers with kernal size (3, 3) were included in the model. As there is a certain smoothness quality to the predicted fields, these layers are included with the aim of blurring the output, reducing speckling that appeared in earlier predicted plots.

The Adam optimizer was used when fitting the model and the Keras Mean_Squared_Logarithmic_Error function was used as the loss function. The model training times were both roughly 20 minutes which is to be expected as the same model architecture was used in both instances.

The use of augmentations in the training data was avoided since medical applications require pre-designated incident wave directions. Re-orientating the incident fields would not increase model generalisability in the path of interest. With the exception of horizontal mirroring, data augmentation would not reduce the required number of generated samples. While it would shrink the possible permutations in the scene configuration space, this would increase the probability of duplicates arising from the symmetry in scenes between the training/test/validation sets over the different folder sessions.

IV. Results

F. Model Training Loss Curves

As depicted in Figure 6, the training loss curve features an extremely abrupt initial decrease followed by a slowly declining plateau. These curves are comparable to loss curves reported in [11], [12]. The loss values are so small that it was required to plot the curves on a logarithmic scale. Both the training and validation curves track each other tightly until the final training session when the training curve starts fluctuating rapidly, albeit with small magnitude. This is an indication that the model has started to overfit the training data.

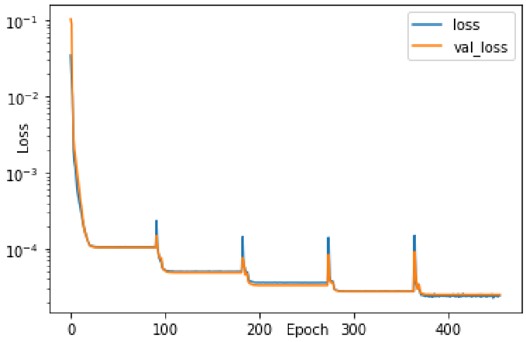


Figure 6. Training history of the E1 model component.

The spikes that occur at the loading of fresh training data indicate there may be possible issues in the model design. The difficulty of predicting the y-dimension, E2, is more apparent since the incident dipole wave has greater variation compared to the x-dimension co-ordinate. The E2 component sufferers from even greater spikes at each training data loading stagethan the E1 component.

While it may appear that training could be terminated after the first session, the model is only achieving visually comparable predictions between epoch 300 and 400, as depicted in Figure 6. The final learning rate frequently reduced to the order of magnitude reflected in Table 2.

G. Model Performance

There is a strong visual similarity between the predicted fields and the fully solved fields, as displayed in Figure 7.

Mean Squared Error	E1 Model	E2 Model
Training	2.3042e-05	3.7006e-05
Validation	2.5000e-05	3.7967e-05
Testing	2.6289e-05	4.1606e-05
Final Learning Rate	1.0000e-22	1.0000e-22

Table 1. Final folder model scores and terminal learning rate.

The Mean Squared Error column reported in Table 2 shows that, if treated as an emulator, Prescient2DL achieves a strong degree of similarity to the fully solved Iterative Krylov solution. The divergence and rotational components of the fields are both being captured by the model and this is evidenced in the plot of the absolute difference in Figure 7.

Table 3 provides descriptive statistics for each test set used to evaluate the impact of Prescient2DL on SolverEMF2. Each set consisted of 100 original samples solved using the naïve initial guess of the incident wave as the scattered field. After training the models for predicting the two scattered fields, a second run of SolverEMF2 was used on the same original samples, allowing for direct comparison across duration of calculation, iteration count and initial error.

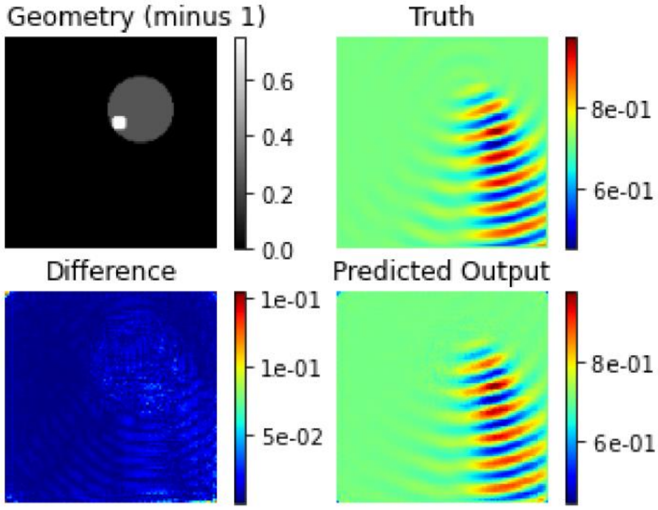


Figure 7. E2 model final prediction on DS1.

Metric	N	Mean	SD
DS1			
Duration_o	100	1.106213	0.055213
Duration_m	100	1.010293	0.093377
Iteration_Count_o	100	22.57	0.655282
Iteration_Count_m	100	22.05	0.479373
Error_Initial_o	100	0.004857	0.002712
Error_Initial_m	100	0.001102	0.000492
DS2			
Duration_o	100	0.772	0.132
Duration_m	100	0.72	0.071
Iteration_Count_o	100	19.57	0.573
Iteration_Count_m	100	19.35	0.52
Error_Initial_o	100	0.003	0.002
Error_Initial_m	100	8.014×10^{-4}	3.702×10^{-4}
DS3			
Duration_o	100	2.218	0.198
Duration_m	100	2.308	0.246
Iteration_Count_o	100	56.65	1.048
Iteration_Count_m	100	56.58	0.955
Error_Initial_o	100	0.03	0.019
Error_Initial_m	100	0.02	0.011

Table 2. Descriptive Statistics of Testing Datasets. "o" corresponds to the naïve guess while "m" gives model-assisted information.

Results indicate that the model generalizes well on DS2. For the generalizability test on DS3, it is evident that the visual resemblance between the predicted fields and the actual fields has broken down, as depicted in Figure 8.

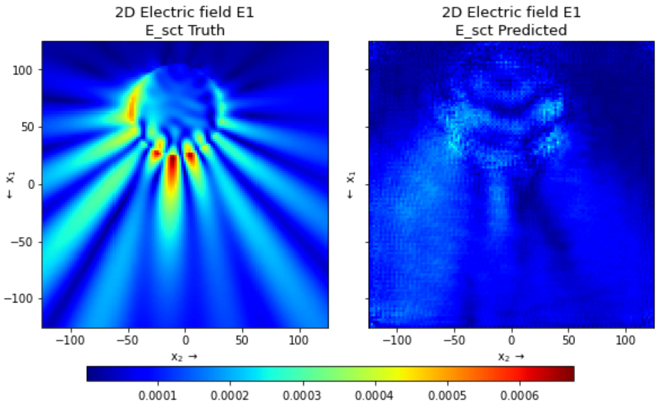


Figure 8. E1 model final prediction on DS3.

Commentary on training issues and a more exhaustive description of the predictive results are available in [13].

V. ANALYSIS[6]

As established in Section IV, Prescient2DL emulates the scenarios of DS1 from both a visual and mean squared error perspective. However, the paper sought to provide experimental data to enable the testing of its impact upon SolverEMF2. Table 3 outlines the statistical results.

	Metric	t	p	Mean Difference
DS1	Duration	8.9132305	< .001	0.0959198
DS2		3.394	< .001	0.052
DS3		-3.3	0.001	-0.09
DS1	Iteration Count	7.2478005	< .001	0.52
DS2		4.2	< .001	0.22
DS3		0.572	0.569	0.07
DS1	Error Initial	16.5942404	< .001	0.0037544
DS2		14.493	< .001	0.002
DS3		12.829	< .001	0.01

Table 3. Paired Samples T-Tests comparing naive to model-assisted performance.

H. Research Test 01 – Initial Solution Conveyance t-Test

"Null Hypothesis H_0: The initial error (Residual Norm) in the Krylov Iterative Metrics in SolverEMF2 is the same as for the non-DL assisted conventional solver. Alternative Hypothesis H_A: The initial error (Residual Norm) in the Krylov Iterative Metrics in SolverEMF2 is lower than the non-DL assisted conventional solver" [14].

For all three test sets, Prescient2DL is able to lower the initial error of SolverEMF2 in a statistically significant manner. The null hypothesis is rejected. In DS1, the initial error is lowered to 18% of that provided by the naïve guess. For DS2, Prescient2DL was able to lower the intial error to 27% of the naïve guess showing that there is some generalisability. Although the initial error achieved on DS3 was 66% of the naïve guess, the visual degradation of the guess is apparent in Figure 8.

[6] A more exhaustive analysis is available in [13].

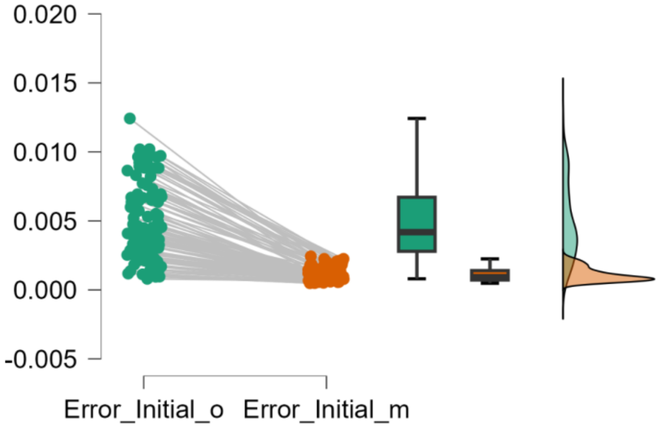


Figure 9. Raincloud plot of initial error comparison in DS1.

I. Research Test 02 – Initial Solution Convergence t-Test

"Null Hypothesis H_0: A linear approximation of the slope of the curve for plot Residual Norm versus Iteration Count, labelled as convergence rate, in the Krylov Iterative Metrics for SolverEMF2 is the same as for the non-DL assisted conventional solver. Alternative Hypothesis H_A: A linear approximation of the slope of the curve for plot Residual Norm versus Iteration Count, labelled as convergence rate, in the Krylov Iterative Metrics for SolverEMF2 is the not equal to the non-DL assisted conventional solver" [14].

Again, there is a difference in the mean between the iteration count of naïve and Prescient2DL informed solutions in DS1 and DS2. For DS3, there is no evidence to show a reduction and so the null fails to be rejected. All differences were less than a single iteration on average.

J. Wider Analysis & Impact Requirements

Prescient2DL is able to generalise to DS2 without too much of a degradation in terms of lowering the initial error but struggles on DS3. Overall, although the initial error in the model-assisted approach is lower and more consistant then the naive approach, it is not low enough to impact SolverEMF2. Indeed, the difference in iteration count for DS1is roughly only half of one iteration. The descriptive statistics in Section IV and the statistical analysis indicate that Prescient2DL does not accelerate SolverEMF2 in generating solutions. A subset of the information gathered during a sample run is captured in Table 4. If the current tests and results show low mean squared error but an immaterial impact upon SolverEMF2, how would results that actually achieve MoM solution acceleration manifest in this problem formulation?

Sample 5101	**Naïve Original Run**	**Model-Assisted Run**
Iteration	**Residual Error**	**Residual Error**
0	0.00207481	0.000512025
1	0.000234899	7.02384E-05
2	3.68364E-05	1.58675E-05
3	1.02315E-05	4.2492E-06
4	1.0167E-06	5.663E-07
5	2.997E-07	6.35E-08
6	4.01E-08	1.71E-08

Table 4. Partial Iterative Solver information for Sample 5101

Due to dependance of convergence rates for Krylov solvers on the conditioning of the matrices and the eigenvalue properties of the matrices [15], Prescient2DL would need to be achieving initial residual errors of 10^{-8} or lower in this toy scenario to diminish the final iteration count by even 25% of the naïve solution iteration count.

VI. Conclusions

Although a deep learning model has been shown to achieving statistical differences in metric performance compared to naïve approaches, there appears to be nothing materially gained from using the model with regard to accelerating the MoM solver.

Before further experimentation with generalizability and extrapolation is to be carried out, the evidence presented in this paper highlights the need for models to achieve much lower mean squared error results than achieved during this research. In the forward problem, further deep learning model development work should focus on lowering the initial error so that the steps required to achieve iterative solution convergence can be reduced. This paper has also presented evidence that research remains in model development.

If the creation of models that bring the initial residual error to orders of magnitude lower than currently reported is deemed non-viable, after more extensive experimentation, then the attempted application of deep learning in electromagnetic scattering, frequency domain, forward problems could be regarded as frivolous.

In terms of immediate directions for future work arising from this paper, the author has the following suggestions. The use of dense layers [6], residual learning blocks [16] and general adversarial networks [8] may improve training performance for this problem. The inclusion of PINNs or loss functions that make stronger assumptions about the physics underlying the data generating process could help to improve the training behaviour [9]. Combining the information available in each axis to increase the information available to the model may lower the final loss. Co-ordinate systems were siloed for the purposes of this paper and the author expects that inclusion of secondary field information at model inception or at a secondary model stage may increase the final prediction performance. Exploring the domain of deep learning explainability may provide insight into boundary region errors that can be seen in the predictive plots. Deliberately increasing the depth and attempting to overparameterize the model may led to tractability. Theoretical work attempting to increase the mathematical understanding of deep learning models has conjectured that overfitting very complex models does not inevitably lead to poor predictive accuracy [17]. Finally, implementing some of these ideas in the context of predicting only the absolute component of each field may accelerate the experimental process.

While positive results are lacking with regard to the goal of the paper, the author points out that the research has been fruitful in providing the electromagnetics community with reproducible data, as well as raising concerns about the model training results. The paper is written with the view that the engineering problem formulation, design approach, data generation process, experimental design and model development are substantial and meaningful in a pre-results context.

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] A. J. McElwee, ‘Machine Learning Algorithms for EM Wave Scattering Problems, Appendix A: Literature Survey’, DCU, 2023.

[4] A. J. McElwee, ‘Machine Learning Algorithms for EM Wave Scattering Problems, Appendix D: Project Design & Implementation’, DCU, 2023.

[5] A. J. McElwee, ‘Machine Learning Algorithms for EM Wave Scattering Problems, Appendix F: Source Code Listing’, DCU, 2023.

[6] 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.

[7] C. Brennan and K. McGuinness, ‘Site-specific Deep Learning Path Loss Models based on the Method of Moments’. arXiv, Feb. 02, 2023. doi: 10.48550/arXiv.2302.01052.

[8] Z. Ma, K. Xu, R. Song, C.-F. Wang, and X. Chen, ‘Learning-Based Fast Electromagnetic Scattering Solver Through Generative Adversarial Network’, *IEEE Transactions on Antennas and Propagation*, vol. 69, no. 4, pp. 2194–2208, Apr. 2021, doi: 10.1109/TAP.2020.3026447.

[9] J. Lim and D. Psaltis, ‘MaxwellNet: Physics-driven deep neural network training based on Maxwell’s equations’, *APL Photonics*, vol. 7, no. 1, p. 011301, Jan. 2022, doi: 10.1063/5.0071616.

[10] C. Gigli, A. Saba, A. B. Ayoub, and D. Psaltis, ‘Predicting nonlinear optical scattering with physics-driven neural networks’, *APL Photonics*, vol. 8, no. 2, p. 026105, Feb. 2023, doi: 10.1063/5.0119186.

[11] T.-T.-H. Le, H. Kang, and H. Kim, ‘Towards Incompressible Laminar Flow Estimation Based on Interpolated Feature Generation and Deep Learning’, *Sustainability*, vol. 14, no. 19, Art. no. 19, Jan. 2022, doi: 10.3390/su141911996.

[12] Y. Guan, T. Fang, D. Zhang, and C. Jin, ‘Solving Fredholm Integral Equations Using Deep Learning’, *Int. J. Appl. Comput. Math*, vol. 8, no. 2, p. 87, Mar. 2022, doi: 10.1007/s40819-022-01288-3.

[13] A. J. McElwee, ‘Machine Learning Algorithms for EM Wave Scattering Problems, Appendix E: Testing & Results’, DCU, 2023.

[14] A. J. McElwee, ‘Machine Learning Algorithms for EM Wave Scattering Problems, Appendix B: Project Design Plan’, DCU, 2023.

[15] H. A. van der Vorst, *Iterative Krylov Methods for Large Linear Systems*, 1st ed. Cambridge University Press, 2003. doi: 10.1017/CBO9780511615115.

[16] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems*, Second edition. Beijing [China] ; Sebastopol, CA: O’Reilly Media, Inc, 2019.

[17] P. L. Bartlett, A. Montanari, and A. Rakhlin, ‘Deep learning: a statistical viewpoint’, *Acta Numerica*, vol. 30, pp. 87–201, May 2021, doi: 10.1017/S0962492921000027.

School of Electronic Engineering

CB54: Machine Learning Algorithms for EM Wave Scattering Problems

Appendix A: Literature Survey

Anthony James McElwee
ID Number: 20211330

August 2023

MEng in Electronic and Computer Engineering

Supervised by Dr Conor Brennan

Declaration

I hereby declare that, except where otherwise indicated, this document is entirely my own work and has not been submitted in whole or in part to any other university.

Signed: Anthony James Mc Elwee   Date: 2023/08/20

Preface

This is the summary of changes to the Literature Review as submitted in January 2023.

- Minor corrections transforming the word "scatter" to "scatterer" were required.
- The problem specification of the project was changed from the Transverse Magnetic Scalar 2D problem to the Transverse Electric Vector problem. This is not reflected in the Literature Review due to time and space limitiations. Instead please refer to the final IEEE conference style paper for the final problem specification.
- Minor changes to two neighboring sentences were required on page two where the words "matrix inversions, often" was changed to "matrix inversions or" and the words "accelerate the matrix inversions by Fast" was changed to "accelerate the matrix multiplications by Fast".
- No other changes were required. Although many more references and sources were recovered during the duration of the project, most of these have been referred to in the Project Research Log, the main IEEE conference style paper and the various other appendices. There are comments regarding the literature review in the Project Research Log that complement the comments below.

Comments on Literature Review

At the time of submission of the final portfolio, the original Literature Review remains extremely relevant in terms of the direction that advanced model development could take in future work. Many of the advanced ideas presented in January remain unexplored due to time limitations. In light of the progress made around experimentation with data generation and U-net architecture stages, the student include more advanced generative adversarial models and denoising architectures if a further review was required. Although this was mentioned, in terms of existing work directly dealing with electromagnetic scattering forward problems, the student would focus on papers dealing with the pure domain of denoising architectures so that recent developments could be transferred to the toy problem in order to achieve lower initial errors. The student would also focus more on the low-frequency high-contrast problem, as noted in the final IEEE-style conference paper submission. It is the student's view, as reflected in the analysis of the testing and results section, that model development to a specific application of scattering should be prioritiesed before trying to test the ability of the model to generalise to unseen parameter domains. These comments would inform the direction of a second stage literature review if the project was to be continued in the future.

Machine Learning Algorithms for EM Wave Scattering Problems: Literature Review

Anthony James McElwee, *MEng Student, DCU*

***Abstract* – This review examines the possibility of using machine learning (ML) algorithms in the search for solutions to Electric Field Volume Integral Equations (VEFIE) formulated, forward scattering problems. A short overview of existing, conventional approaches to approximating solutions to such problems is included, along with a reflection on some recent attempts to augment these methods and create ML emulators by using deep learning (DL) approaches. Based on the review, a brief proposal for the direction of the project activity is offered for deliberation. The aspiration of the review is to communicate recent developments in nascent ML approaches and to provide groundwork for the development of a solver, SolverEMF2, that resolves to reduce the computational cost of providing a solution to the scattering problem at time of inference via a DL model called Prescient2DL.**

Index Terms - computational electromagnetics, deep learning, knowledge integration, neural networks, physics-guided, physics-informed, VEFIE, Volume Electric Field Integral Equation

I. Introduction

A. Task Motivation

The construction of object classifiers using electromagnetic scattering characteristics and the competent planning of wireless network design are undertakings that can require large numbers of frequency-domain simulations and the ability to iteratively adjust input configurations through intervention by a design engineer [1], [2]. Typically, these tasks operate with a constrained set of input parameters, such as incident source and material/geometry attributes of scatterers. Although input parameters are comparable across simulation incidences, conventional methods typically require full uninterrupted simulations, below the wavelength, to provide solutions. As a consequence, the generation of large volumes of such simulations takes an uneconomical amount of time and computer memory. Design methodologies appreciate the incorporation of rapidly adjustable, human mediated input configurations but conventional approaches lead to inflexible workflows. In addition, early-stage designs are usually afforded significantly higher error thresholds than full simulations deliver, resulting in over-simulation and a waste of computational resources. With restrictions on the volume of simulations afforded to designers, it is postulated that final classifier metrics and planning layouts are typically sub-optimal.

Just as the requirement to build expensive, physical prototypes in design development workflows has been minimized through the use of computational electromagnetics (CEM), research is now underway to reduce the computationally intense attributes of CEM through the use of data-driven ML. The aim of this project is to accelerate VEFIE-formulated, two-dimensional, scattering simulations at time of inference using ML algorithms in a bid to alleviate the described design workflow issues. The CEM aspect of the problem is acknowledged to have a steep learning curve [3].

Date of submission: 2023/01/22. e-mail: anthony.mcelwee2@mail.dcu.ie

B. Problem Specification

The forward problem constitutes the realization of scattered wave fields based on information regarding the material contrast and incident field [3]. Typically, Maxwell's equations are formulated in a manner which gives rise to the Helmholtz Wave Equation, which degenerates into Fredholm Integral Equations, through boundary and continuity conditions, in particular, the second kind for VEFIE. The design properties of interest are assumed to depend sinusoidally on time with a shared angular frequency ω.

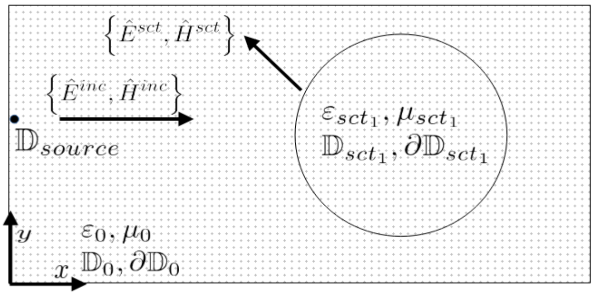


Fig. 1 Problem Illustration. A single source emitting incident waves is located at a fixed x-axis location on the left-hand side of the scatter. Material values are complex valued, frequency-dependent permittivity (ε) and permeability (μ). Background points indicate discretization.

In this project, the aim is to solve for the total electric field $E^{tot}(\boldsymbol{r})$ so that the scattered field, $E^{sct}(\boldsymbol{r})$, can be approximated in the simulation domain, as shown in Fig. 1. Positions in the 2D domain are denoted $\boldsymbol{r} = (x, y)$. A scatterer is located in free-space with surface boundary geometry that will vary in deformation. Material constituents of the scatterer give rise to permittivity contrast only, so permeability is assumed to be the same as free-space ($\mu = \mu_0$).

Incident waves, $E^{inc}(\boldsymbol{r})$, are emitted in Transverse Magnetic Mode by a sole transmitter at the left-hand side of the domain. As a result, the incident electric field has no x or y component, only a z one, although Transverse Electric Mode can also be considered with similar consequences.

As given in [1], setting $k_b = \omega\sqrt{\varepsilon_0\mu_0}$ as the wavenumber, the described configuration gives rise to the electric field integral equation:

$$E^{tot}(\boldsymbol{r}) = E^{inc}(\boldsymbol{r}) + k_b^2 \int_D G(\boldsymbol{r} - \boldsymbol{r}')\, \chi(\boldsymbol{r}') E^{tot}(\boldsymbol{r}') d\boldsymbol{r}', r \in D, \quad \textbf{(1)}$$

where $\chi(\boldsymbol{r})$ is the contrast function and $G(\boldsymbol{x})$ is the 2D free space Greens function

$$G(\boldsymbol{r} - \boldsymbol{r}') = -\frac{j}{4} H_0^{(2)}(k_b|\boldsymbol{r} - \boldsymbol{r}'|). \quad (2)$$

It is assumed that no sources exist within scatterers. The scattered field, $E^{sca}(r^R)$, can be computed by

$$E^{sca}(r^R) = k_b^2 \int_D G(r^R - \boldsymbol{r}')\, \chi(\boldsymbol{r}') E^{tot}(\boldsymbol{r}') d\boldsymbol{r}', r^R \notin \boldsymbol{D}. \quad \textbf{(3)}$$

A similar formulation in Chapter 3 of [3] is given with more exhaustive derivations for various material assumptions, as well as MATLAB code.

II. Review & Analysis of Prior Work

A. Existing approaches and their related use with ML

Awareness of existing approaches is important when developing SolverEMF2. Concepts underpinning such methods may be assimilated into the DL architecture [4]. Appreciation of computational bottlenecks may also allow Prescient2DL to be specifically targeted.

1) Monte Carlo (MC)

MC methods estimate the value of an integral via repeated random sampling and can evaluate arbitrary points in a domain, including integrals with singularities and discontinuities. The rate of convergence for naïve MC is $\mathcal{O}(\mathrm{n}^{-\frac{1}{2}})$, making it computationally expensive.

2) Analytical

Integrals may admit approximate solution methods, such as infinite series solutions, due to the simple nature of the geometry in the formulation. For VEFIE, these methods are dominated by Bessel-function approaches [3]. Infinite summations can be truncated to suit the required accuracy of the solution, provided the infinite series actually converges analytically. Such solutions are used to benchmark CEM solvers for canonical problems, assess accuracy requirements and debug development code. Analytical methods are also useful for generating initial training data for developing Prescient2DL. When problems contain non-trivial geometries, analytical Bessel-Function approaches breakdown.

3) Conventional Computational Electromagnetics (CEM)

More usually, numerical approximation methods are used for solving VEFIE formulated integrals. They typically use discretized grid systems generating large linear systems of equations [4]. They offer high fidelity solutions for a wide variety of problem formulations, are in widespread use and have been analytically validated for canonical problems [3].

Boundary Element Methods, known idiomatically as Method of Moments (MoM), require the computation of matrix inversions, or using iterative Krylov Methods [3]. It is possible to formulate the integral operator as a discrete convolution and accelerate the matrix multiplications by Fast Fourier Transforms [3]. The exploitation of circulant properties of Toeplitz matrices or eigenvalue deflation can also reduce computational requirements. With such formulation adjustments, the rate of convergence for BICGSTAB solver can be reduced to $\mathcal{O}(\mathrm{n}\log n)$ [1].

CEM also covers the Finite Difference Frequency Domain Method (FDFD), Finite Difference Finite Time Method (FDFT) and Finite Element Method (FEM). All CEM require an accuracy threshold or bound on resources as an input so that they can be realized on a computer. As a problem becomes larger, CEM eventually becomes uneconomical in both computational time and memory management [1].

4) High-Frequency and Empirical Approaches

Ray tracing approaches can be used for indoor propagation problems [1]. While the contrary has been reported in [1], ray tracing formulations are typically faster than CEM approaches as they are high-frequency approximations that exploit assumptions from geometrical optics. An example of how developments in ray tracing may stimulate the development of SolverEMF2 is briefly mentioned in the final section of this review. Also considered in [1] are empirical path loss models that may give insight into how DL architectures can be simplified to reduce training burdens.

In summary, existing approaches can be used to generate development data for Prescient2DL, help validate results and offer insights into how SolverEMF2 can be constructed.

B. Possible ML approaches to the problem

In a naïve sense, this a supervised regression problem and deployed ML models can offer an inference in a smaller number of computations than the preferred CEM [5]. A variety of ML algorithms exist and can be appropriated to almost any research domain where data is plentiful. The survey [6] gives a wide overview of application-centric objectives for using ML in engineering and physics domains. With regard to this project, and its resource limitations, exploring downscaling, reduced order modelling, forward PDE solving, inverse modelling, data generation and uncertainty quantification may contribute to development.

One ML development in particular, DL, has led to exceptional advancements in computer vision over the last decade. Indeed, [6] classifies physics-guided methods to integrate scientific knowledge into ML and all are applicable to DL: loss functions; training weight initialization; architecture design; hybrid modelling. Efforts to develop understanding of statistical properties of DL have led to conjectures about the benign nature of its overfitting and how over parameterization leads to tractability when dealing with very complex models. Consequently, DL is now of interest to researchers, more than any other aspect of ML, in trying to combat expensive computational physics problems.

While DL approaches have been more extensively applied to inverse problems, EM scattering forward problems have only recently been reported. Applications of ML to forward problems in other domains can be found more easily. There are research papers reaching back to the 1990s that strive to use neural networks to solve fluid dynamics, process modelling problems and differential equations [7].

C. Surrogate Replacements

Surrogate models, or emulators, are built with the intention of assimilating an entire method, typically CEM, within an approximation model. The surrogate requires minimal human interaction and can be used as a sub-model in a hierarchical framework. The cost of data generation and training is realized in an offline stage prior to deployment time which results in an exchange of computationally intensive algorithms with data-driven inferences. The emulator avoids solving large systems of equations generated by the approximation over basis functions of non-linear integrals, thus removing a computational bottleneck.

As profiled in Chapter 8 of [4], DL architectures have already been proposed as ML duals of CEM methods in a bid to emulate their abstract properties. A variety of Long Short-Term Memory, Convolutional Neural Networks (CNN), Encoder-Decoder structures and Physics-Informed Neural Networks (PINN) are combined with other DL techniques, depending on the approach the CEM captures in its solution.

Surrogates are usually trained for specific problem parameter ranges and, as a result, are assumed to have limited generalization ability [7]. Even with immense advances in ML, these models introduce uncertainties and compromise interpretability and explain-ability of results [7].

Chapter 4.3 of [7] gives a short description of peer-reviewed, non-electromagnetic case studies that used DL surrogates. Typically, training data was constructed from a small number of FEM simulations and used to develop emulators for human tissue stress determination. These surrogates allowed real-time interventions with patients. In further examples, [7] mentions CNN architectures used in field estimation for fluid dynamics. Although accuracy was reduced compared to conventional methods, ML was deemed sufficient for early stage design workflows. In summary of [7], surrogates developed using DL have been deployed to act as decision support mechanisms to humans in medical settings and, in resource restricted design scenarios, emulators enhanced composition methodologies for engineers.

1) Direct solvers using input-output pairs

The paper [8] precedes and forms the groundwork for the new book [9] that reports on the implementation of a U-Net structured emulator. The architecture takes two input images that establish the source as well as the material/geometry of the scatterer as depicted in Fig. 2 taken from [8]. In essence, it is an autoencoder styled structure with CNN and residual blocks. The residual blocks help to overcome common problems with training DL networks, via skip layers, and the technique features across the surveyed literature.

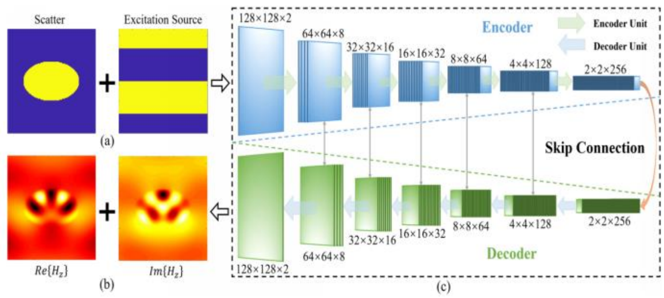


Fig. 2 (a) Input images for scatter & source. (b) Two output images representing Re() and Im() parts of solved field. (c) U-Net Architecture [8].

The problem of estimating complex spatial relationships, where global information influences local values, typically requires deep networks that give rise to vanishing gradients unless the architecture is augmented with said remediation structures.

The paper outlines FDFD discretization for an ellipse and modeling of a TE plane wave is validated against commercial solver COMSOL Multiphysics. The implemented FDFD solver is applied to solve for scattering caused by a 2D training set whose geometry and material properties are coherently bounded by parameter ranges.

One difference between [8] and [9] is that the paper [8] used CReLU activation functions while the book [9] discusses various considered options, finally opting for ELU based on an experiment. Neither of these activation functions are considered immediate choices for DL development. The varied documentation regarding this aspect of the architecture points to the intricacies involved in ML emulator development and that superior DL configurations may yet be found. While [9] uses a mean-squared error loss function to compare the output with the FDFD solution, PINN approaches in the next section offer a different approach to this aspect of the DL training approach.

The test results are presented as having low error when compared to the same FDFD code used to generate training data. In addition, shapes not present in training are evaluated using the emulator and reported error remains low. The paper notes that the emulator does not generalize well for permittivity contrasts beyond the range provided at training. No code was available for either [8], [9]. Explicit experimental documentation is a desirable reproducibility feature when reporting such results.

A more complex surrogate DL architecture, called a General Adversarial Network (GAN), has also been applied to the problem [5]. It uses input-output pairs and reformulates the problem as one of ML image translation. GAN development is currently enjoying success, driven by media attention from beyond the ML community. In [5], the generator is constructed using U-Net architecture, similar to that already described in [8], [9]. Through the addition of a discriminator stage, the approach is redirected to find a solution to a Nash Equilibrium problem. By adding such complexity to the architecture, the discriminator also allows negative examples to be generated and tested. [5] describes in atypical detail the computational complexity of the implementation, as opposed to most literature where such considerations are simplified or ignored totally.

[5] claims improved accuracy over the sole U-Net but also indicates some weaknesses associated with this particular form. GANs typically require multiple adjustments to architectural elements, relative to U-Net, and [5] also adjusts the loss functions in addition to these changes. Much larger training sets are required to compensate for the complex form. The range of contrast permittivity tested is narrow and small in [5] compared to the other literature. It is an open question whether specific EM scattering GANs are the architectures that will yield SOTA results.

2) Physics-Informed Neural Network (PINN)

In [2], a DL model is trained using a Maxwell informed, physics-integrated loss function to find the electric field given scatterer geometry and material information, replacing FDFD. The residual is based on the time-harmonic Helmholtz EM Wave Equation. This would be considered a PINN, an area of research that has expanded significantly since 2019. In contrast to [5], [8], [9], where the surrogate is developed using a database of input-output pairs, [2] relies on indirect learning dependent on penalizing the physics-informed loss function. A significant advantage to this approach is that the training process does not require intensive computations to generate the model. In [2], the DL model is coupled with a second stage DL model that helps to solve an inverse optimization design problem.

Where full surrogates are implemented, a solution difference gap relative to CEM is typically not clarified. This uncertainty opens surrogates to questions of robustness. Stating input parameter ranges used in training is frequently the unsatisfactory rebuttal.

D. Combined/Hybrid Methods

As already stated, ML can be used to achieve diverse objectives and knowledge of underlying physics can be infused into DL models in a variety of ways. In the feasibility study [10] regarding DL and the Poisson Equation, the authors give a thoroughly documented demonstration of a CNN based architecture, orientated around Algebraic Multigrid approaches, that can act as a surrogate to solving the PDE or as the provider of an initial guess for a CFD solver to achieve the same aim. The stated aspiration is that the informed guess allows the iterative solver to reach convergence in a smaller *wall-clock* runtime compared to those not given an initial guess. This paper makes an attempt to integrate a variety of approaches mentioned in [6]. [10] gives insights into development, provides narrative around creating special loss functions to enhance training rates and achieve lower error metrics than more typical PINN and MSE loss functions, as well as provide results that include impacts on BICGSTAB initial error rates. An ablation study focuses on changes to model architecture. Although this paper does not examine electromagnetics, it offers fertile ground for development proposals.

While other uses for combined approaches are mentioned in [7], the underlying theme of this hybrid form is that ML acts as a support mechanism for deterministic methods. Many examples exist where ML controlled systems are actively discouraged, such as medical applications. The pervasive attitude is that ML should never be used in a stand-alone fashion but instead aid or accelerate a guided method. Aside from risk aversion, this approach may reduce robustness testing requirements. In the case of supplying initial guesses, this aspect is drastically reduced since deterministic iterative algorithms should converge to a unique solution.

E. Culs-De-Sac

During the review, some pre-print and peer reviewed material presented possible research routes that transpired to be inapplicable or worse. Fundamentally, such material was underpinned by an inappropriate use of DL for directly solving linear problems or through sub-algorithmic augmentation. Their inappropriate nature can be identified from plots of loss and error functions with extreme convergence rates. In these cases, DL adds more computational expense and creates needless uncertainty.

III. Relation of Prior Work to Project Problem

Even though there is a relative poverty of research into the application of ML to forward EM scattering problems, there are already multiple approaches to infusing ML in the construction of new engineering solvers. Acceleration might be achieved by considering new objectives in the engineering workflow, such as increasing design process flexibility. ML may aid in producing early-stage design solvers with small inference times whose estimations are satisfactory for error requirements less stringent than final design criteria.

F. Proposal of the direction of the project activity

The central hypothesis is that ML can be used in a combined-hybrid manner to robustly lower the computational burden of CEM. Based on the cited literature, the project proposes the creation of SolverEMF2 that will encapsulate the entire solution workflow. The computational cost of providing CEM convergent solutions will be reduced via an initial guess, via a DL model called Prescient2DL, to a MoM iterative solver, such as BICGSTAB. SolverEMF2 will then complete the MoM approach with this guess, reducing the iteration count required to achieve convergence.

G. Potential routes of experimentation

Prescient2DL will initially be developed using the existing architectures already cited. Attempting to resolve mathematical features that various CEM methods utilize [4] and synthesizing physics-informed loss functions to reduce required training data and increase robustness [2] are both routes that can be expanded upon. By amending existing architectures, via meta-architectures or assimilating developments in GANs, an EM scattering focused DL architecture may finally diverge from the U-Net architecture originally intended for biomedical segmentation. MATLAB and Tensorflow in Python will be used with Git to facilitate reproducibility.

Finally, light rendering typically also involves solving Fredholm Integral equations of the Second Kind, generally dependent on MC and ray tracing approaches. Significant developments in this domain have occurred recently. Through a multi-staged solver, the challenge of solving VEFIE could be recast as an inverse problem. By iteratively populating MC samples in the forward manner, a DL model in the second stage could denoise the inferred field as SolverEMF2 converges to the MoM validated solution. This approach may be less resource intensive than developing GAN structures.

IV. Conclusion

The sources considered in the process of completing this literature review agree on the positive potential of ML to shift the computational effort of current conventional approaches from time of inference to the training stage, as well as reduce the required duration to provide a solution to the problem of electromagnetic scattering. The review finds DL as the best route that presents experimental opportunities.

References

[1] I. Kavanagh and C. Brennan, 'Validation of a volume integral equation method for indoor propagation modelling', *IET Microwaves, Antennas & Propagation*, vol. 13, no. 6, pp. 705–713, 2019, doi: 10.1049/iet-map.2018.5849.

[2] J. Lim and D. Psaltis, 'MaxwellNet: Physics-driven deep neural network training based on Maxwell's equations', *APL Photonics*, vol. 7, no. 1, p. 011301, Jan. 2022, doi: 10.1063/5.0071616.

[3] 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

[4] M. Martínez-Ramón, A. Gupta, J. L. Rojo-Álvarez, and C. G. Christodoulou, *Machine learning applications in electromagnetics and antenna array processing*. Boston London: Artech House, 2021.

[5] Z. Ma, K. Xu, R. Song, C.-F. Wang, and X. Chen, 'Learning-Based Fast Electromagnetic Scattering Solver Through Generative Adversarial Network', *IEEE Transactions on Antennas and Propagation*, vol. 69, no. 4, pp. 2194–2208, Apr. 2021, doi: 10.1109/TAP.2020.3026447.

[6] J. Willard, X. Jia, S. Xu, M. Steinbach, and V. Kumar, 'Integrating Scientific Knowledge with Machine Learning for Engineering and Environmental Systems', *ACM Comput. Surv.*, vol. 55, no. 4, pp. 1–37, May 2023, doi: 10.1145/3514228.

[7] S. Kollmannsberger, D. D'Angella, M. Jokeit, and L. Herrmann, *Deep Learning in Computational Mechanics: an Introductory Course*. Cham: Springer, 2021.

[8] 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.

[9] 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.

[10] A. G. Özbay, A. Hamzehloo, S. Laizet, P. Tzirakis, G. Rizos, and B. Schuller, 'Poisson CNN: Convolutional neural networks for the solution of the Poisson equation on a Cartesian mesh', *DCE*, vol. 2, p. e6, 2021, doi: 10.1017/dce.2021.7.

Ollscoil Chathair Bhaile Átha Cliath
Dublin City University

School of Electronic Engineering

CB54: Machine Learning Algorithms for EM Wave Scattering Problems

Appendix B: Project Design Plan

Anthony James McElwee

ID Number: 20211330

August 2023

MEng in Electronic and Computer Engineering

Supervised by Dr Conor Brennan

Updates since original plan (May until August 2023)

The original plan below has not been changed in order to save the reader the effort of reviewing all contents twice. Instead the student lists the main deviations from the original submitted plan.

- The original plan promoted the idea that graphs could be used to illustrate the improvement that Prescient2DL would provide to SolverEMF2. The results of the experimentation mean that any such graphs would be identical for all of the hypothesis tests regarding solution convergence and solution conveyance. While the initial solution conveyance was statistically significant for both models, it was not impactful for the assistance of the Iterative Krylov Solver. There may be an illustrative graph present in the final IEEE paper, space permitting, but all information can be presented to the reader in table form.
- Due to time issues, the student has not been able to investigate the secondary research question surrounding more sophisticated deep learning implementations in order to solve the electromagnetic scattering problem. All of the energy was focused on generating a validated data generator and trying to lower the initial error of SolverEMF2 to make substantive impact on the required solution time. As a result this remains an open problem for future research.
- In “Project Scope”, the student sought to equate the electromagnetic formulation of equation (6.5) in [2] and equation (3.86) in [3]. These are not equivalent problems. However, equation (6.5) in [2] is equivalent under certain conditions to equation (1.43) in [3] and it can be shown that equation (1.43) in [3] is a reduced form of equation (3.86) in [3]. While these are all closely related formulations, great difficulty arose when trying to reconcile code and parameters. The student decided to drive the project with equation (3.87) in [3] to ensure that the underlying problem was physically sound and that an opportunity arose for combining field predictions in the vector case in more advanced model development. Unfortunately time meant that this opportunity was not grasped before the submission deadline.
- Although the student was granted access to the DCU GPU rig, the student did not avail of the hardware as the personal laptop was sufficient to generate samples and train the models. In future work, with higher-permittivity contrasts, this opportunity would have to have been utilised. To keep experimental conditions uniform across the three experiments, the student found it easier to use their own local laptop.
- The student decided against using data augmentation in the context of the physical nature of the problem and concerns about contaminating test/validation sets with symmetrical samples.
- In the original plan, the student outlined advanced architectures that could be investigated under the Secondary Research Questions header. These were not investigated due to time limitations but the student would like to highlight that this eventuality was included at the time of the original submission.
- The “Success Criteria” section listed 7 key points at the time of the original submission. All points on the list were either fully completed, negating the experimental findings that Prescient2DL does not accelerate the SolverEMF2 workflow, except for the final point regarding the drafting and submission of a paper covering the findings of the project on ArXiv. The time limitations, and multiple setbacks faced during the development process, has resulted in the student not writing a paper for submission. The student managed to complete tests on all of the Primary Research questions in a limited form as only one architecture was developed.
- Finally, the project timeline that was outlined by the student underestimated the amount of time available in August that could be taken due to work commitments but underestimated the time it would take to generate useable data in the training of the deep learning model. A result of this imbalance is that the deep learning experimentation was limited to a single model architecture and the write-up of the portfolio was condensed into three days. The student realised that the training data was unstuitable to use in experiments on Thursday 17th August and hence all results and documentation had to be finalised very close to the submission time. Upon reflection, the student finds that the main aspects of the original project plan were achieved but regrets not getting to experiment with the Secondary Research Questions due to time limitations.

Contents

Project Design Plan 4
Research Question 4
Primary Research Question 4
Secondary Research Questions 4
Project Scope 5
Design Approach 5
Timeline 9
Success Criteria 10
Bibliography 10
Approval 10

Project Design Plan

Research Question

The student proposes to sub-divide the research question into two stages to allow for iterative assessment of plausibility of research opportunities and increased agility around the resource and time constraints available.

- The primary stage is aimed at establishing elements of the work of [1]–[5] in a reproducible workflow called SolverEMF2 and creating a supervised regression model called Prescient2DL to test the various hypothesis associated with the Primary Research Question, as explained in the Design Approach section.
- The secondary stage is based around investigating the rudiments of Prescient2DL. Possible aspects for analysis are covered in the various hypothesis associated with the Secondary Research Questions, as explained in the Design Approach section.

Primary Research Question

"Can the manner in which deep learning has been shown to solve two-dimensional, forward electromagnetic scattering problems applied to the problem of predicting EM wave propagation over rural terrain be improved upon?"

As already reported in the Literature Review, various sources have described using deep learning to tackle forward electromagnetic scattering problems, however, to the knowledge of the student, none have provided a public, reproducible, open-source workflow or a model to the research community. The student proposes to approach the integration of the developed deep learning model, Prescient2DL, into SolverEMF2 through the use of Prescient2DL to generate initial guesses for the Krylov Iterative Solver. By establishing the SolverEMF2 workflow, this primary research question will be approached through the investigation of simulations with several segmented statistical hypothesis tests in lieu of qualitive mathematical proofs. Implicitly, the primary aim of the project is to implement a solver with a deep learning model that optimally shifts calculation metrics to towards the lower left corner of the Residual Error versus Iterations/Time graph when solving permittivity contrast source only Volume Electric Field Integral Equations.

Secondary Research Questions

The secondary research questions below are based around trying to expand knowledge around whether the application of deep learning to this domain is fundamentally underpinned by the attributes of the training data set or whether a hierarchical approach to model generation exists. The questions offer a rich range of potentially publishable findings and opportunities to contribute to the field of scientific machine learning.

"Are all models equal in the framework of the performance analysis conducted in the primary research question? Can improvements be made to model development approaches? Can model deployment be improved so that there is an increased opportunity/impact by the model on SolverEMF2 performance?"

Based on the literature review, and echoed in the more recent [6], the models reported in the available literature are either developed in a U-net based architecture or through a Physics Informed Neural Network (PINN). The student proposes to conduct studies into the performance between siloed models to illustrate advantages and changes in performance given a consistent testing environment.

From the insights arising from the model comparison stage, the student believes that these approaches can initially be expanded upon and ensembled. Stemming from this investigation will be the final Prescient2DL model that should demonstrate a more mature application of deep learning to the problem domain. This facilitates the exploration of deep learning attributes and features specifically refined for the problem domain with the possibility of reporting on refined architectures or properties of the model development purpose previously not recorded in journals.

As reflected in the literature review, there is almost no diversity in the deployment strategies for deep learning in this problem domain. The process of emulation is the baseline approach to harnessing deep learning models. The model is expected to resolve problems whose inputs are confined to the parameter ranges associated with the input data used to train the model in the first place. The online-stage of the process thus mirrors the use of Look-Up Tables (LUTs) in conventional problem approaches. The primary stage aims to expand this baseline approach by wrapping Prescient2DL solutions in the Method of Moments framework. In addition to this expansion, the literature review has highlighted sub-algorithmic adjustments to Krylov Iterative Solvers that may yield improved performance metrics. The area of probabilistic numerics may also offer alternative ways that deep learning can be integrated into SolverEMF2.

The benchmarking and characterisation of existing model formulations, the expansion of the model development approaches and the deployment techniques used in SolverEMF2 should facilitate investigation into the generalisability of the deep learning in the domain. The use of Prescient2DL as a basis for transfer learning may also be investigated. These secondary research questions will also be approached through the investigation of several segmented statistical hypothesis tests in lieu of qualitive mathematical proofs.

Project Scope

- The electromagnetic formulation that underpins the area will remain as the two-dimensional, permittivity contrast source only Volume Electric Field Integral Equation as derived at equation (6.5) in [2] or equation (3.86) in [3]. These are scalar integral equations. The derivations in [2] and [3] differ in their uses of time harmonic convention, however, the results from the equations should be equivalent. The Time Harmonic Dependence Convention (j=-i) can be used to translate between them. In [2], the Real-Value Transform with complex notation “i” is used while in [3] the Laplace Transform with complex notation “j” is favoured with the Laplace variable $s = -i\omega$.
- Establish a fresh Github version control repository for code maintenance and sharing.
- Develop the initial SolverEMF2 by migrating VEFIE code from MATLAB to Python and validating the code on a canonical problem. The student has purchased a new laptop with i7-11800H @ 2.3GHz CPU and NVIDIA GeForce RTX 3070 GPU. Access has also been granted to remote DCU GPU rigs and these will be utilised as the project matures.
- Establish and follow the dataset generation pathway as outlined in the Design Approach section below.
- Develop the initial iteration of Prescient2DL using Python deep learning packages.
- There are a number of advanced research areas outlined in the Literature Review, such as the expansion of models via Monte Carlo integration. Due to resource/time constraints, the student proposes to only approach these areas if the potential of the research questions, as outlined in the previous sections, has been exhausted.
- The visualisation and flexibility of the solution information conveyed to the user of SolverEMF2 was initially of much interest to the student, however, since the literature review was conducted the student has discovered a project hosted at (https://github.com/chiuhans111/fdtd-html). That project tackles a time-domain, rather than frequency-domain, problem but the student proposes to focus on the more technical performance metrics and deep learning model development due to the increased probability of generating publishable content.

Design Approach

Code Development Environment

Code will be developed in a manner where all outputs are reproducible through input documentation and seeding. Metrics around the non-DL simulations will be recorded. Time of training and time of inference with initial error (for hybrid models incorporated into conventional methods) will also be recorded.

Code Development Validation – Initial MATLAB migration

Verify code through the solution of canonical toy problems in MATLAB and Python, comparing results to a satisfactory degree of accuracy.

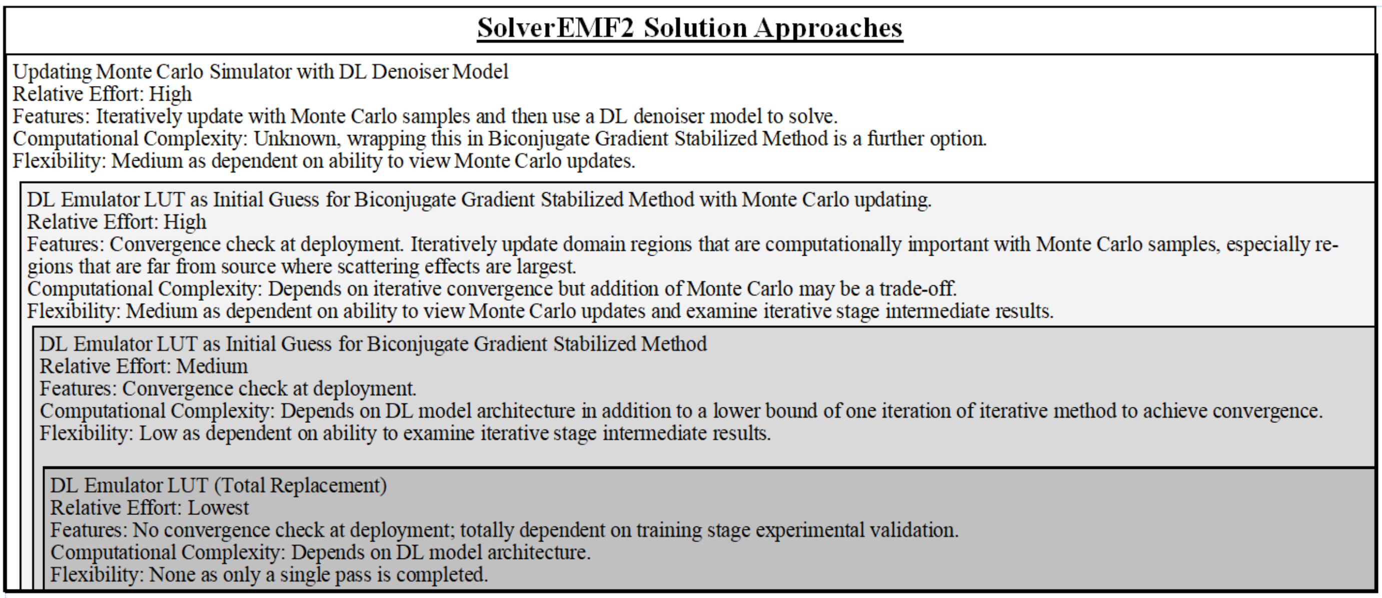
Deep Learning Dataset Development– Pathway

The creation of a benchmark dataset would be a publishable accomplishment in itself. The student proposes to start with a simple, narrow parameter settings database and as the feasibility of a model develops, the student plans on adding network architecture complexity.

- In the first instance, only train the model on data generated from canonical problem formulations with Bessel-function type solutions. Then benchmark an independent conventional model (MoM) against the Bessel-Function solutions and the ML model. If all three agree then the ML model can be trained next on the conventional model.
- Generate non-canonical dataset using MoM python code.
- Generating the training dataset will be computationally intensive. The student proposes to squeeze value out of the simulations by using data augmentation such as rotation and reflection. A large number of simulations should be possible by having a range over the input parameters.

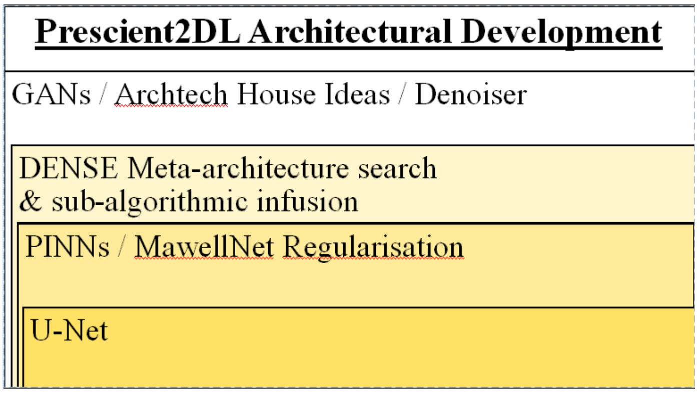
SolverEMF2 – Architecture

The architecture of SolverEMF2 initially sets all guesses to the Krylov Iterative Solver as an array of zeros. This step is later replaced with Prescient2DL informed guesses. The student proposes following the development of the solver methodology using the pathway illustrated in the image below. The student does not anticipate that the later Monte Carlo stages will be achieved in the time limitations of the project.



Prescient2DL – Architecture

The student proposes to follow the pathway outlined in the image below when expanding the model architecture and physics infusion.



Primary Research Test 01 – Initial Solution Conveyance t-Test

Null Hypothesis H_0: The initial error (Residual Norm) in the Krylov Iterative Metrics in SolverEMF2 is the same as for the non-DL assisted conventional solver.

Alternative Hypothesis H_A: The initial error (Residual Norm) in the Krylov Iterative Metrics in SolverEMF2 is lower than the non-DL assisted conventional solver.

Note: Conveyance is used to mean an indication of the level of information from the informative guess, be it Prescient2DL or a more vanilla approach, conveyed to SolverEMF2. The closer the Kyrlov Iterative Solver is to the be within the acceptable solution threshold, with respect to the naïve guess, then the more useful information has been conveyed via the guess. In the Test 01 situation, this manifests as simply the lower initial error.

Primary Research Test 02 –Solution Convergence t-Test

Null Hypothesis H_0: A linear approximation of the slope of the curve for plot Residual Norm versus Iteration Count, labelled as convergence rate, in the Krylov Iterative Metrics for SolverEMF2 is the same as for the non-DL assisted conventional solver.

Alternative Hypothesis H_A: A linear approximation of the slope of the curve for plot Residual Norm versus Iteration Count, labelled as convergence rate, in the Krylov Iterative Metrics for SolverEMF2 is the not equal to the non-DL assisted conventional solver.

Note: Further tests to establish if the absolute value of the slope is greater for Solver EMF2 may be required, however, it is expected that the convergence rate will remain unless sub-algorithmic integration of the model is successful in the secondary stage.

Primary Research Test 03 – Solution Conveyance t-Test

Null Hypothesis H_0: The area under the curve for plot Residual Norm versus Iteration Count, labelled as AbsementKIM, in the Krylov Iterative Metrics for SolverEMF2 is the same as for the non-DL assisted conventional solver.

Alternative Hypothesis H_A: The area under the curve for plot Residual Norm versus Iteration Count, labelled as AbsementKIM, in the Krylov Iterative Metrics for SolverEMF2 is smaller than for the non-DL assisted conventional solver.

Explanation: If both the Residual Norm and the Iteration/time vary, it may be difficult to judge the difference in performance based on a single parameter. This naïve approach to combining the parameters by finding the area under the curve intersecting the two axis may be a way to compare the computational expense expended by the various SolverEMF2 variants. AbsementKIM stands for Absement Krylov Iterative Method and is analogous to Absement in the domain of kinematics.

Secondary Research Tests – General

All tests conducted in the Primary Research Test stage will be applied to the secondary stage. To illustrate the idea of the primary metric from the Primary Research Test, the diagram below shows the baseline permutations that could arising when comparing the output metrics of the SolverEMF2 activities. The diagram on the left indicates an impact on initial errors while the right diagram indicates an impact on the rate of convergence of the solver. Both of these changes of parameter could change simultaneously so a third graph showing the exhaustive list of such graphs could be produced but is deemed too busy to be informative. The areas in green indicate the AbsementKIM that may arise. A smaller AbsementKIM in the final metrics would indicate that more information in general was passed through the initial guess relative to whatever configuration that was used to establish the null.

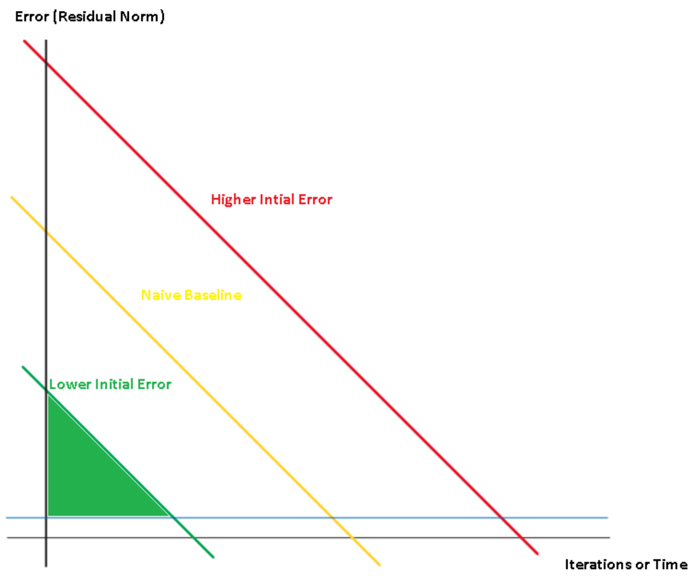


Figure 1: Impacts on Initial Error. Slopes are constant.

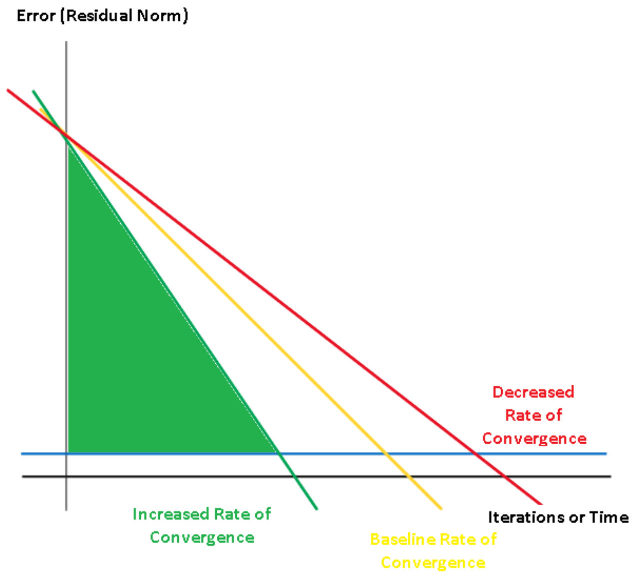


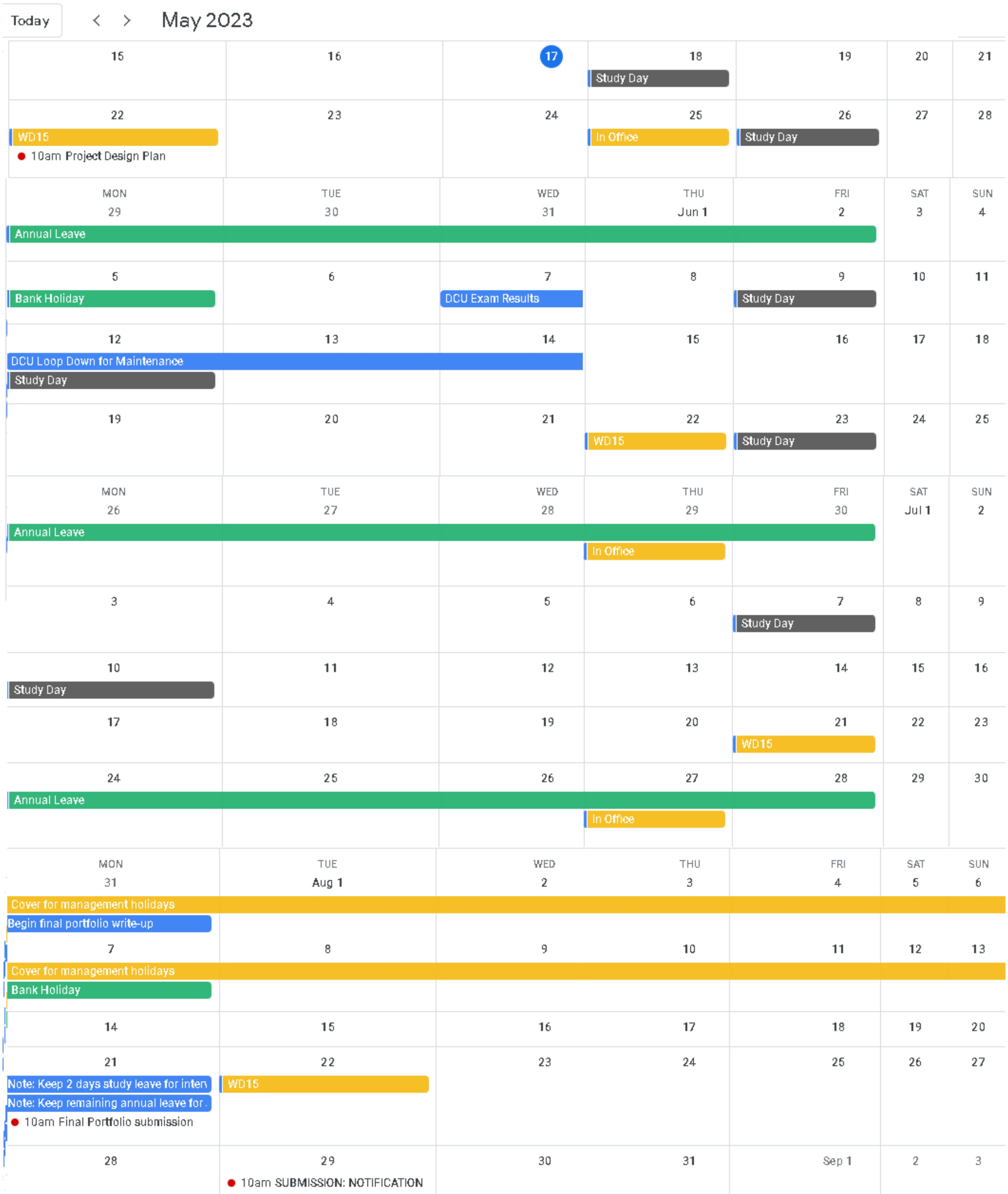
Figure 2: Impacts on Rate of Convergence. Initial error is held constant.

Secondary Research Tests – Model Development & Integration

The metrics used in the development process for Prescient2DL will be the residual sum of squares normally used in such regression problems. The developments will follow the baseline established in [1], [4] and expand to include aspects highlighted in the Literature Review. The generalisability of the model will be tested by generating a random shapes dataset, as opposed to the terrain dataset used in the rest of the project. Machine learning models are largely data-driven but the data in this experiment will be totally synthetic. As a result, the developed ML model will be derived from governing equations that are an idealization and representative of that centric world-view. This is the same idea as the Helio-centric versus Geo-centric interpretation in inter-planetary motion. The objective is to build an ML model of an existing conventional method. Testing generalizability could be done by seeing how far Prescient2DL diverges from canonical solutions when initially only trained on canonical datasets, then trained on general datasets, then compare the model on canonical validation set and general test set to see if performance degrades on the canonical test in order to generalize to the generalized inputs. A decision around metrics may also be required, for example, tolerance of some extra resources for time of inference may be tolerated for lower error and vice-versa.

Timeline

As of the date of the submission of this project proposal document (2023/05/21) there are only 92 days to the final submission deadline for the completed project (2023/08/21). In this period, there are 64 weekdays and 28 weekend days. The student has 7 study leave days and up to 23 annual leave days from work. Due to the constraints of work, it is highly likely that not all of the annual leave can be taken in this period. In August, the student will be required to act as cover for management. There is also a strong bias towards taking leave towards the end of the month to avoid impact on the bulk of regulatory reporting between working days 8 and 15. The student is populating the Google Calendar facility, available with their DCU email account, with key milestones and constraints. This is illustrated below. Due to the iterative and unknown time requirements for the tasks outlined in the Project Scope and Design Plan, the student has avoided pinning dates to specific actions with the exception of “Final Portfolio Write-Up”. Due to severe constraints in August, the preparation for the final portfolio will need to commence at the start of August and while experimentation can run in parallel in the background, it is highly unlikely that new developments or additions to the model development architecture will be completed beyond this date.



Success Criteria

The student proposes that the criteria for success remain open-ended with the aspiration of completing all items on the list below while acknowledging that the time and resource constraints may curtail its completion. Chapter 13 of [6] offers some open problems in the domain of applying DL to electromagnetic problems. The relevant ones for this specific subdomain can be summarised as trying to generate more generalisable models using less data in a more efficient manner without overfitting. The suggestion is to enhance the model architecture with physically-based loss functions and generate foundation-type models that can be adapted and fine-tuned via transfer learning approaches.

- Validated literature review. See log entry relevant to [6].
- 10 research log submissions updating work progress and developments in the field.
- Python implemented SolverEMF2 workflow that can be used and expanded in future by students of the field.
- Mature Prescient2DL model architecture that can be deployed to accelerate the generation of solutions to these types of problems.
- A dataset of solved simulations for future use by researchers in this field.
- A completed final report portfolio for submission to DCU for masters accreditation that provides answers to the outlined primary and secondary research questions.
- A paper on (https://arxiv.org/) or in a peer-reviewed journal reporting some finding from the second stage of research questions.

Bibliography

[1] C. Brennan and K. McGuinness, “Site-specific Deep Learning Path Loss Models based on the Method of Moments.” arXiv, Feb. 02, 2023. doi: 10.48550/arXiv.2302.01052.

[2] V. Pham-Xuan, “Accelerated iterative solvers for the solution of electromagnetic scattering and wave propagation propagation problems,” doctoral, Dublin City University. School of Electronic Engineering, 2016. Accessed: Mar. 20, 2023. [Online]. Available: https://doras.dcu.ie/20951/

[3] 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

[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] J. Lim and D. Psaltis, “MaxwellNet: Physics-driven deep neural network training based on Maxwell’s equations,” *APL Photonics*, vol. 7, no. 1, p. 011301, Jan. 2022, doi: 10.1063/5.0071616.

[6] A. P. M. Li, M. Li, and M. Salucci, *Applications of Deep Learning in Electromagnetics: Teaching Maxwell’s Equations to Machines*. Institution of Engineering & Technology, 2023.

Approval

Signature of Project Worker: Anthony James McElwee Date: 2023/05/20

Print name of Project Worker: **ANTHONY JAMES MC ELWEE**

Signature of Project Supervisor: Conor Brennan Date: 2023/05/21

Print name of Project Supervisor: **DR CONOR BRENNAN**

School of Electronic Engineering

CB54: Machine Learning Algorithms for EM Wave Scattering Problems

Appendix C: Project Research Log

Anthony James McElwee

ID Number: 20211330

August 2023

MEng in Electronic and Computer Engineering

Supervised by Dr Conor Brennan

Please read before making entries in this log

The purpose of this Project Research Log is to capture concise, focused summaries of research materials you read, as you progress through your project. The emphasis is to record (i) how the material you have read will determine or influence your project solution approach and (ii) your assessment of the key strengths and weaknesses of the solutions, methods, technologies, etc. proposed in the material you have read.

In the first stage of your project, the literature review, use the Log to capture this information for the key papers you have read (for example, the three most important papers of your 10 literature review references). As your project progresses into the design and implementation phases, you will need to continue to search the literature so you can review, revise and refine your initial thinking and the details of your approach to a project solution. Use this Research Log to capture your continued research reading and its influence on your project design and implementation.

Be selective about what you record in this log. Do not use it as an informal notebook while you are reading a new paper. Only make an entry after you have read a paper that you consider important to the development of your project solution. It is expected that, by the end of the project, you will have made between 10 and 20 entries (20 maximum).

Share your log with your supervisor for viewing throughout the project. You will submit the final version of the log for grading, at the end of the project implementation period. It will be assessed on the basis of how well you have used your analysis of the literature to inform your project design, implementation and the evaluation of your project results.

The Research Log contributes 5% to the overall project mark.

Note: All entries you make in this log must use the prescribed format shown on the next page. You will maintain other notes as you progress through your project but they should not be recorded here. Fill in the details where the *** signs are.

Contents

Please read before making entries in this log 2

Log Entry 00: 2022/11/28 3

Log Entry 01: 2022/12/31 4

Log Entry 02: 2023/01/11 5

Log Entry 03: 2023/05/16 6

Log Entry 04: 2023/05/17 8

Log Entry 05: 2023/06/06 9

Log Entry 06: 2023/07/13 10

Log Entry 07: 2023/07/15 11

Log Entry 08: 2023/07/28 12

Log Entry 09: 2023/08/05 13

Final Information as per guidelines 14

Log Entry 00: 2022/11/28

Statement of project problem / research question (maximum 200 words)
This statement should be periodically reviewed and updated, as necessary, as your project progresses and you gain further insight into the detailed project challenges, requirements and objectives as your project work moves from background reading, literature review, initial project design planning and detailed design and implementation. Initially, start by stating your current understanding of the project objectives. After each meeting with your supervisor, review and refine your project problem statement, as required.
THIS IS JUST TO RECORD THE INITIAL PROJECT STARTING POINT *"When an electromagnetic wave encounters an object it scatters, with some energy being transmitted into the object and the rest propagating in a variety of directions depending on the material composition and local geometry. A precise knowledge of the scattering phenomenon is desirable for a variety of applications, such as medical imaging, radar and wireless communications. Numerical techniques such as the method of moments give highly accurate results, but are computationally expensive. An emerging alternative is the use of machine learning tools that can be trained using a training set of data covering a sufficiently wide feature set (i.e. problem geometry, material, frequency etc). This project will use an in-house, Matlab-based, implementation of the method of moments to train an artificial neural network to solve the problem of EM scattering from convex dielectric bodies."*

A complete reference for the paper
***
Summary of paper (maximum 100 words)
***
How is this paper relevant to solving your project problem or addressing your research question? (maximum 100 words)
***
What are the strengths and weaknesses of the solutions/methods/technologies proposed in this paper? (maximum 100 words)
***

Log Entry 01: 2022/12/31

Statement of project problem / research question (maximum 200 words)

"Can the manner in which deep learning has been shown to solve two-dimensional, forward electromagnetic scatting problems applied to the problem of predicting EM wave propagation over rural terrain, namely emulation, be expanded or improved upon?"

A complete reference for the paper

[1] , please refer to the final bibliography.

Summary of paper (maximum 100 words)

This paper claims that the sub-algorithmic infusion of a deep learning model into an iterative solver, essentially replacing steps in the iterative solver at every iteration, can accelerate the realization of a solution to the VEFIE formulation of the electromagnetic scattering problem.

How is this paper relevant to solving your project problem or addressing your research question? (maximum 100 words)

It is extremely relevant to the project as it deals directly with the problem domain and takes the outlook of a hybrid methodology where a conventional approach is enhanced with a new machine learning technique. The paper describes how the solver algorithms are augmented and includes diagrams of deep learning architecture designs.

What are the strengths and weaknesses of the solutions/methods/technologies proposed in this paper? (maximum 100 words)

The paper gives a lot of detail to each element of the problem and even gives information about the experimental apparatus. The student remains sceptical that the approach can be implemented to yield consistent results as the deep learning model should give a single guess upon which every successive completion of the iterative solver would beat in terms of minimising the residual vector. The student feels that the deep learning model would hinder, rather than assist, the iterative solver after the initial guess. The paper is also advanced in terms of implementation requirements to achieve a duplication of the paper so the idea may not be suitable to approach in the time frame of the project. Also the hardware used in the paper is far beyond the budget deemed reasonable by the student to achieve experimental results as the price for the GPU at time of writing is over $10, 000.

Log Entry 02: 2023/01/11

Statement of project problem / research question (maximum 200 words)

"Can the manner in which deep learning has been shown to solve two-dimensional, forward electromagnetic scatting problems applied to the problem of predicting EM wave propagation over rural terrain, namely emulation, be expanded or improved upon?"

A complete reference for the paper

[2], please refer to the final bibliography.

Summary of paper (maximum 100 words)

The paper describes a deep learning architecture that can search for suitable problem-specific architectures as it is training on the domain data. The general problem areas that the paper discusses covers ten scientific simulation topics that exist at a variety of contrasting physical scales. While electromagnetic scattering is not specifically mentioned, the approach of the paper is highly-likely to be transferrable to the domain.

How is this paper relevant to solving your project problem or addressing your research question? (maximum 100 words)

All material referenced in the literature review relies on pre-existing deep learning architectures, such as U-net, that were not initially developed with the project topic problem at the core of their inception. The student believes that a new architecture should be developed or sought-out that deals with the intricacies of simulating electromagnetic scattering. For example, max-pooling may cause unacceptable domain border errors when considering a hybrid conventional/deep learning solver design. The requirements in terms of layer count and parameters for electromagnetic scattering problems is also extremely vague in the literature reviewed so far by the student. Using a neural network meta-architecture may assist in developing a model that can be trained faster that existing architectures and may require less training data simulations, thus reducing the major bottleneck in deploying machine learning algorithms in the electromagnetic domain.

What are the strengths and weaknesses of the solutions/methods/technologies proposed in this paper? (maximum 100 words)

A weakness of the paper is that it claims the benefits of emulators while it is actually dealing with a search meta-architecture that only implicitly acts as an emulator. The student found this slightly misleading upon first review. From the perspective of the student's project aim, the paper is also limited in that the main signals it handles are 1-dimensional. However, the general idea of the paper and the description of the "super-architecture" as visualised in Figure 1. make this paper a stimulating read that may lead to a direct contribution to the student's final emulator design. The inclusion of zero-layers in the architecture is something that the student had never considered or even heard about in previous reading or course materials.

Log Entry 03: 2023/05/16

Statement of project problem / research question (maximum 200 words)

"Can the manner in which deep learning has been shown to solve two-dimensional, forward electromagnetic scatting problems applied to the problem of predicting EM wave propagation over rural terrain, namely emulation, be expanded or improved upon?"

As already reported in the Literature Review, various sources have described using deep learning to tackle forward electromagnetic scattering problems, however, to the knowledge of the student, none have provided a public, reproducible, open-source workflow or a model to the research community. The student proposes to approach the integration of the developed deep learning model, Prescient2DL, into SolverEMF2 through the use of Prescient2DL to generate initial guesses for the Krylov Iterative Solver. By establishing the SolverEMF2 workflow, this primary research question will be approached through the investigation of simulations with several segmented statistical hypothesis tests in lieu of qualitive mathematical proofs. Implicitly, the primary aim of the project is to implement a solver with a deep learning model that optimally shifts calculation metrics to towards the lower left corner of the Residual Error versus Iterations/Time graph when solving permittivity contrast source only Volume Electric Field Integral Equations.

A complete reference for the paper

[3], please refer to the final bibliography.

Summary of paper (maximum 100 words)

This is a new book (2022) dealing with the application of deep learning to electromagnetic problems that the student did not know existed until 2023/05/08 well after the literature review was submitted. The student has read the relevant chapters 1, 2 and 13 of this book and it conforms with the student's literature review with strong overlaps in the references covered. The student views this an as independent confirmation that their research to date and literature review reflects much of the current research energy in the project domain.

How is this paper relevant to solving your project problem or addressing your research question? (maximum 100 words)

In Chapter 13 there is a section dealing with the pros and cons of using DL in the domain whose synthesis would be helpful in the final project portfolio.

Chapter 13 also raises some problems that may be faced in the project. There is a lack of transparency and understanding of the inner workings of the DL architectures. The student believes there are developments in ML space that are working on reducing this lack of transparency, for example, Professor Paul Whelan's visualization methodology for the various layers in the Computer Vision module assignment and the student's understanding that a recent new research domain of explainability in DL may have yielded recent breakthroughs.

What are the strengths and weaknesses of the solutions/methods/technologies proposed in this paper? (maximum 100 words)

One difference to the literature review was that there seems to be a greater consideration given to the sub-algorithmic approaches (references 126-130) that the student had partially avoided. As a result the student may reconsider these approaches and consider their inclusion. The sources not previously considered in the literature review have been recorded in the student's Zotero database for future consideration. In terms of downsides of the book, there was nothing that the student hadn't previously considered or covered in the literature review already.

Log Entry 04: 2023/05/17

Statement of project problem / research question (maximum 200 words)

"Can the manner in which deep learning has been shown to solve two-dimensional, forward electromagnetic scatting problems applied to the problem of predicting EM wave propagation over rural terrain, namely emulation, be expanded or improved upon?"

A complete reference for the paper

[4], please refer to the final bibliography.

Summary of paper (maximum 100 words)

This is a book so new recent that no copy is available at time of writing, however, the student was able to consult the table of contents. The book is concerned with recent advancements in deep learning with application to electromagnetics and is part of the IEEE Press Series on Electromagnetic Wave Theory. This series was also includes [5], as recommended by the supervisor, and has a wide range of titles that concern the general project domain.

How is this paper relevant to solving your project problem or addressing your research question? (maximum 100 words)

The student reviewed the table of contents to check for any forward problem developments that might be useful or prompt queries at the final stage of the project implementation. The vast bulk of the book seems to be concerned with solving inverse problems, such as design optimisation. The only chapter that looks pertinent is the short chapter "Machine Learning Advances in Computational Electromagnetics". Most of the sub-headers have already been covered in the literature review and those that are not obvious to the student, such as "Deep Surrogate Solvers Trained with Physical Regularization" seem to have less than a page of material.

What are the strengths and weaknesses of the solutions/methods/technologies proposed in this paper? (maximum 100 words)

The student does not feel that a review of the table of contents is enough to pass comment on the strengths/weaknesses etc. of the text except for the fact that the overwhelming majority of the book is focused on inverse, rather than forward, problems. If the text becomes available within the time limit of the project, the student will review Chapter 7 in case there are ideas that can be easily incorporated into the project.

Log Entry 05: 2023/06/06

Statement of project problem / research question (maximum 200 words)

"Can the manner in which deep learning has been shown to solve two-dimensional, forward electromagnetic scatting problems applied to the problem of predicting EM wave propagation over rural terrain, namely emulation, be expanded or improved upon?"

A complete reference for the paper

[6] , please refer to the final bibliography.

Summary of paper (maximum 100 words)

The paper discusses a hybrid approach that incorporates neural networks into a finite element method (FEM) solver. The approach is to calculate a residual from the finite element method and a custom loss function from the deep learning model to form a new solver algorithm. The idea is to create surrogate models that can be more generalisable and wrapped in a conventional solver. The applications are not in the domain of electromagnetics.

How is this paper relevant to solving your project problem or addressing your research question? (maximum 100 words)

The paper reaffirms the student's project objective of testing the ability of deep learning models to enhance conventional forward problem solvers. The paper does not totally abandon the conventional solver but instead finds a way to integrate the new machine learning approach with the more established FEM methodology. The benefits of this framework have already been expounded in the student's literature review.

What are the strengths and weaknesses of the solutions/methods/technologies proposed in this paper? (maximum 100 words)

The paper lays out the algorithm for integrating the neural network into the FEM solver in a clear fashion. Such visual description would be a useful addition to the student's own write-up towards the end of the project. Although the student's domain knowledge of the applications, and indeed FEM, tackled in the paper are limited, the two case studies are nicely detailed and give toy examples that may be useful in future work if problems beyond electromagnetics were to be developed. The paper also highlights some of the issues in generating the deep learning surrogates which is useful towards planning the student's project implementation. The paper is lacking detail on contrasting the computational improvement of the new hybrid solver and focuses on the accuracy of the predictions.

Log Entry 06: 2023/07/13

Statement of project problem / research question (maximum 200 words)

"Can the manner in which deep learning has been shown to solve two-dimensional, forward electromagnetic scatting problems applied to the problem of predicting EM wave propagation over rural terrain, namely emulation, be expanded or improved upon?"

A complete reference for the paper

[7]–[9], please refer to the final bibliography.

Summary of paper (maximum 100 words)

This log entry deals with references concerning the domain of application for solver, namely biomedical. The references consulted were:

- Section 5.4 of "Case Study: Scattering from Red Blood Cells" of [7];
- Section 2.5.1 & Section 6 of [8];
- Table 1 of [9] titled "Microwave parameters of three breast tissue types at low (0.5 GHz), middle (2 GHz, 4 GHz, 6 GHz), and high (8 GHz) frequencies".

How is this paper relevant to solving your project problem or addressing your research question? (maximum 100 words)

These references were consulted when considering topic and parameter selection. The setting of carrier wave incident frequency, geometric scale and discretization outputs matter in sizing the data inputs for the deep learning model.

What are the strengths and weaknesses of the solutions/methods/technologies proposed in this paper? (maximum 100 words)

- [7] indicated that a 474 THz incident wave would be required, with a red blood cell having a length of roughly 7.7 micrometers. According to the text, such scales lead to matrix equations with dimensions of over 200,000. Using such large arrays for building ML models is not suitable with current resources and even generating a dataset with solved fields is far beyond what the remaining project time would allow.
- In [8], a model of the relative complex permittivity of human muscle tissue is described in Section 2.5.1. This is the basis for an illustration of Deep Regional Hyperthermia Treatment Planning in Section 6. The example is in the time domain and is too computationally intensive as it depends on three dimensions with multiple incident waves in the 90 MHz range. The main reason for setting aside this source is that, due to time constraints, the incident wave is fixed for all simulations. Creating a sophisticated look-up table for a set of relative permittivities and conductivities based on the carrier incident wave frequency would be wasteful.
- Leading on from [8], the student found Table 1 in [9]. It gives a description of the effective dielectric permittivity and conductivity for normal, benign tumor and cancer cell tissues in the GHz range. This allows for discretization in the scale of interest of 128 and 256 which are more easily accommodated in deep learning architectures.

Log Entry 07: 2023/07/15

Statement of project problem / research question (maximum 200 words)

"Can the manner in which deep learning has been shown to solve two-dimensional, forward electromagnetic scatting problems applied to the problem of predicting EM wave propagation over rural terrain, namely emulation, be expanded or improved upon?"

A complete reference for the paper

[10], please refer to the final bibliography.

Summary of paper (maximum 100 words)

Chapter 6 of this thesis describes an accelerated implementation the Volume Electric Field Integral Equations. Accompanying MATLAB code was sent by the supervisor.

How is this paper relevant to solving your project problem or addressing your research question? (maximum 100 words)

The student was able to match the equations from the derivations in this thesis, culminating with Equation 6.5, to the derivation in [11] for the scalar scenario in Chapter 1, specifically Equation 1.43. The texts use different conventions and approaches to deriving the VEFIE. Initially the student hoped to adapt this MATLAB code to python in order to generate a dataset. The dataset would then be used to train a deep learning model. After enhancing the code parameters to accept a Debeye material model for common building materials, unfortunately, the student realised that the code did not produce scattering simulations due to a possible error in the way the contrast was assigned to the domain of interest.

What are the strengths and weaknesses of the solutions/methods/technologies proposed in this paper? (maximum 100 words)

The focus of the thesis is on conventional approaches to solving the forward problem of electromagnetic scattering. The convention is different to [11], which is the main reference for the student's project. The common complaint, at least to the electromagnetics community, of a lack of code in the body or appendices of the text that reflects the experimental findings of the thesis arises here.

Log Entry 08: 2023/07/28

Statement of project problem / research question (maximum 200 words)

"Can the manner in which deep learning has been shown to solve two-dimensional, forward electromagnetic scatting problems applied to the problem of predicting EM wave propagation over rural terrain, namely emulation, be expanded or improved upon?"

A complete reference for the paper

[12], please refer to the final bibliography.

Summary of paper (maximum 100 words)

This paper was published soon after the completion of the literature review stage of the project. The paper discusses the use of U-net architecture and physics-informed loss functions to predict nonlinear optical scattering problems and the solution to an inverse design problem. Both TE and TM problems are referenced. The paper is accompanied by a supplementary materials appendix. The simulations are generated using finite difference schemes.

How is this paper relevant to solving your project problem or addressing your research question? (maximum 100 words)

The paper provides supplementary material that describes elements of the deep learning design that may be relevant to the student's project. Aside from tackling both Transverse Electric and Transverse Magnetic problems, elements of the U-net architecture used in the deep learning model is described as well as the loss function that tries to embed physics properties arising from Maxwell's equations . The paper highlights the increased difficulty of attempting to train a model in the TE scenario.

What are the strengths and weaknesses of the solutions/methods/technologies proposed in this paper? (maximum 100 words)

The paper presents material useful to researchers trying to prioritise and implement approaches to using deep learning to solve scattering problems. Evidence of training equipment and time requirements are welcome as such information is lacking in the general literature. The paper has made the student reconsider the urgency of using physics-based loss functions due to the large amounts of time required to achieve modest results. A major weakness of the paper is the lack of computer code or dataset available to the reader at the time of writing. It is difficult to assess the diagrams and the layers in the U-net architecture are not clear to the reader.

Log Entry 09: 2023/08/05

Statement of project problem / research question (maximum 200 words)

"Can the manner in which deep learning has been shown to solve two-dimensional, forward electromagnetic scatting problems applied to the problem of predicting EM wave propagation over rural terrain, namely emulation, be expanded or improved upon?"

A complete reference for the paper

[13], please refer to the final bibliography.

Summary of paper (maximum 100 words)

The paper describes a developed conventional approach to reducing the complexity of solving low frequency, high-contrast problems in the domain of non-destructive biomedical evaluation, in particular magnetic induction tomography (MIT). Although the paper is following a conventional, forward-problem methodology, it outlines the difficulties in electromagnetic scattering simulation that are relevant to the masters project. The paper also makes multiple references to the work of Peter van den Berg, a key reference in the project.

How is this paper relevant to solving your project problem or addressing your research question? (maximum 100 words)

This paper grounds the project domain in a viable, applied research area concerning magnetic induction tomography (MIT). The preferred carrier frequencies used in MIT are in the 10 MHz region, the same as used to develop the deep learning dataset. There are major difficulties with modelling biological tissue in this frequency range since permittivity values present extreme contrast values. The paper points to future applications and project developments that could lead to medical applications. Unfortunately, the paper deals with a three-dimensional scenario which is beyond the scope of the student's current interest.

What are the strengths and weaknesses of the solutions/methods/technologies proposed in this paper? (maximum 100 words)

In relation to the project that the student is undertaking, the paper provides the existential reason to simulate 10 MHz carrier frequency and allows the student to establish a raison d'être for attempting to use innovative deep learning techniques in this frequency range. Lack of code is a major weakness with regard to this paper.

Final Information as per guidelines

The reasons why you selected the papers that you have entered in your research log.

I have tried to select papers that touch on all aspects of the project workflow to gain insights into as many subparts of the process as possible. The area is relatively new no source consulted contained enough information in a standalone manner to offer a complete solution to the project problem. I also wanted to ground the project in a real-world application, such as the biomedical domain, to increase the usefulness of any insights gained during the project development process.

How you have used the literature that you have read to guide your project plan and implementation.

I used the papers and books to inform choices in the deep learning architecture, establish expected behaviours of the solvers and to avoid following routes that were beyond the time scope of the project in terms of complexity. The books were largely used as a springboard to find related papers in the domain. One major regret was running out of time and not being able to investigate the zero layer idea in [2]. I really wanted to test this idea as a route to creating bespoke architectures specific to the electromagnetic scattering domain. If I was to extend the research time I would prioritise this step immediately before trying to do anything else. The literature, and associated videos on the internet, also directed me away from trying to implement physics embedded loss functions.

How you compared your implementation and results to previous outputs described in the selected papers.

In order to avoid directly copying existing work, I used the literature to rule out some design implementations that have been already completed. For example, the image-to-image approach is the main implementation of the emulator design so I routed towards keeping the problem as an array instead of an image array. In many cases though, due to the scant literature availble on the project topic and lack of transparency in many cases with regard to results and design choices, I tried to use the literature review to set default values in the design choices.

Describe the value of your continued reading of literature relating to your project.

There were a number of texts I found after the literature review that reiterated the lack of development in this area compared to the solutions developed for the inverse problem. This reiterates the difficulty surrounding this topic and that trying to minimialise every single aspect of the toy problem was the correct thing to do in terms of achieving any results at all. I continued to seek out new material all the way until the final week of the project, including trying to infuse second stage DnCNN denoising models to improve the performance without success.

Briefly describe any other impacts that literature had on your project.

In hindsight, reviewing literature constantly actually was a hinderance in terms of achieving results and experimentation. Instead of reading about existing literature, after the exams in May, I should have just tried to build a basic implementation of the U-net architecture. I also feel that reading so much about the electromagnetic components of the data generation was wasted. Early on in my literature review I found a number of key texts, such as [11] and [14]. I should have drawn a line at these texts and proceeded with an attempt to replicate their findings instead of aiming to be comprehensive in literature review, project presentation and project proposal. I would have achieved much more by learning as I built than trying to plan ahead so much. I also took a lot of books out of the library and this biased my reading towards the electromagnetic end of the project since most of the relevant books available were to do with scattering simulation rather than deep learning. Another major negative impact was the late realisation that, although the Unet architecture originated out of biomedical segmentation problems, the problem at hand was closer to a denoising/generative formulation. I had referenced and read [15] in the literature review and although an investigation into GANs would be well beyond the time limits for this project, I feel I might have avoided spending so much time looking at PINNs now that I appreciate the intended function of the Unet in these problems.

Complete Bibliography

[1] R. Guo *et al.*, “Physics Embedded Deep Neural Network for Solving Volume Integral Equation: 2-D Case,” *IEEE Trans. Antennas Propag.*, vol. 70, no. 8, pp. 6135–6147, Aug. 2022, doi: 10.1109/TAP.2021.3070152.

[2] M. F. Kasim *et al.*, “Building high accuracy emulators for scientific simulations with deep neural architecture search,” *Mach. Learn. Sci. Technol.*, vol. 3, no. 1, p. 015013, Dec. 2021, doi: 10.1088/2632-2153/ac3ffa.

[3] A. P. M. Li, M. Li, and M. Salucci, *Applications of Deep Learning in Electromagnetics: Teaching Maxwell’s Equations to Machines*. Institution of Engineering & Technology, 2023.

[4] S. D. Campbell and D. H. Werner, Eds., *Advances in electromagnetics empowered by artificial intelligence and deep learning*. Hoboken, New Jersey: Wiley-IEEE Press, 2023.

[5] A. F. Peterson, S. L. Ray, and R. Mittra, *Computational methods for electromagnetics*. New York, Oxford: IEEE Press ; Oxford University Press, 1998.

[6] R. E. Meethal *et al.*, “Finite element method-enhanced neural network for forward and inverse problems,” *Adv. Model. Simul. Eng. Sci.*, vol. 10, no. 1, p. 6, May 2023, doi: 10.1186/s40323-023-00243-1.

[7] Ö. Ergül and L. Gurel, *The multilevel fast multipole algorithm (MLFMA) for solving large-scale computational electromagnetics problems*, 1st edition. in Ieee press series on electromagnetic wave theory. Chichester, West Sussex: Wiley-IEEE Press, 2014.

[8] J. E. Houle and D. M. Sullivan, *Electromagnetic simulation using the FDTD method with Python*, Third edition. Hoboken, NJ: Wiley, 2020.

[9] Y. Cheng and M. Fu, “Dielectric properties for non-invasive detection of normal, benign, and malignant breast tissues using microwave theories,” *Thorac. Cancer*, vol. 9, no. 4, pp. 459–465, Apr. 2018, doi: 10.1111/1759-7714.12605.

[10] V. Pham-Xuan, “Accelerated iterative solvers for the solution of electromagnetic scattering and wave propagation propagation problems,” doctoral, Dublin City University. School of Electronic Engineering, 2016. Accessed: Mar. 20, 2023. [Online]. Available: https://doras.dcu.ie/20951/

[11] 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

[12] C. Gigli, A. Saba, A. B. Ayoub, and D. Psaltis, “Predicting nonlinear optical scattering with physics-driven neural networks,” *APL Photonics*, vol. 8, no. 2, p. 026105, Feb. 2023, doi: 10.1063/5.0119186.

[13] 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 Propag.*, vol. 69, no. 1, pp. 366–378, Jan. 2021, doi: 10.1109/TAP.2020.3008618.

[14] 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.

[15] Z. Ma, K. Xu, R. Song, C.-F. Wang, and X. Chen, “Learning-Based Fast Electromagnetic Scattering Solver Through Generative Adversarial Network,” *IEEE Trans. Antennas Propag.*, vol. 69, no. 4, pp. 2194–2208, Apr. 2021, doi: 10.1109/TAP.2020.3026447.

School of Electronic Engineering

CB54: Machine Learning Algorithms for EM Wave Scattering Problems

Appendix D: Project Design & Implementation

Anthony James McElwee

ID Number: 20211330

August 2023

MEng in Electronic and Computer Engineering

Supervised by Dr Conor Brennan

Contents

Project Design & Implementation 3
Code Validation 3
Model Architecture Description 5
SovlerEMF2 Infusion 7
Bibliography 8

Project Design & Implementation

Code Validation

The Python code was validated in the same manner as the MATLAB code was validated in [1]. This code is called using the function EmsctCircle in the file custom_functions_EM.py found in the folder lib found in the project GitHub repository "https://github.com/spookworm/CB54". Details of the validation code and lengthy derivation are covered in Section 3.A.1.3 of [1] and the main points with plots are briefly described below. In the code ForwardBICGSTABFFTwE.py code, by changing the variable "validation" to equal "'True'" the validation code will be reproduced. Ensure that the other two variables "guess_validation_answer" and "guess_model" are both set to 'False'.

- The parameters of the simulations are loaded using the initEM function with the custom "bessel" argument enabled to indicate that the programme returns the base scatterer geometry.

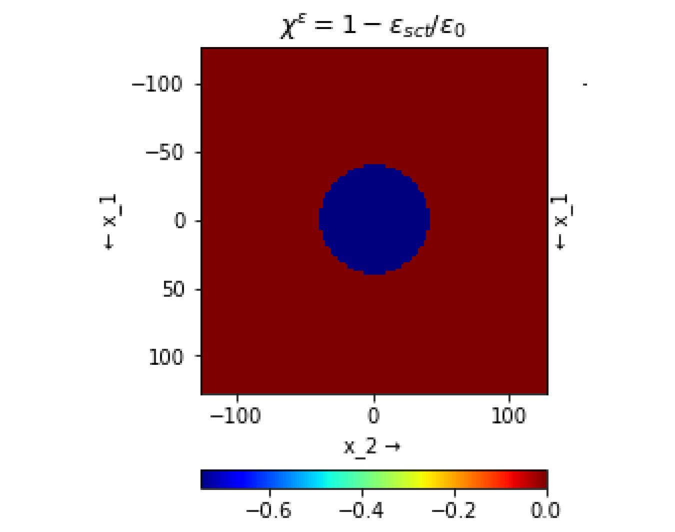


Figure 1. Validation scatterer geometry for validation case. As there is no permeability contrast.

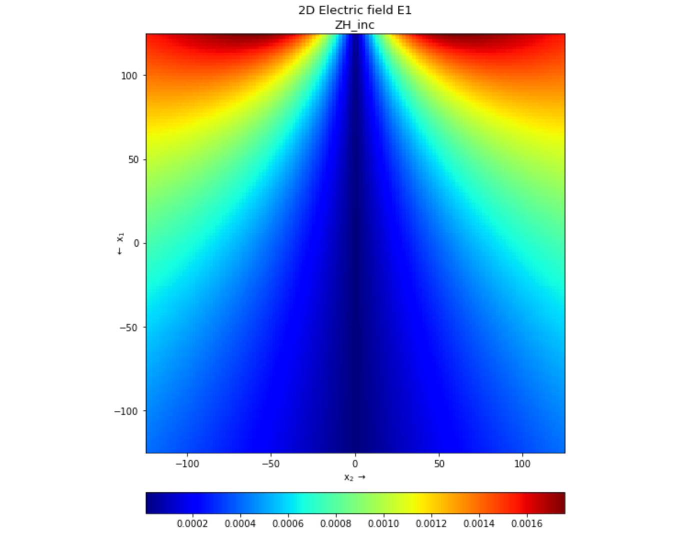


Figure 2. Incident ZH_z field.

- The receiver and source emitter have their co-ordinates transformed from cartesian to polar form.

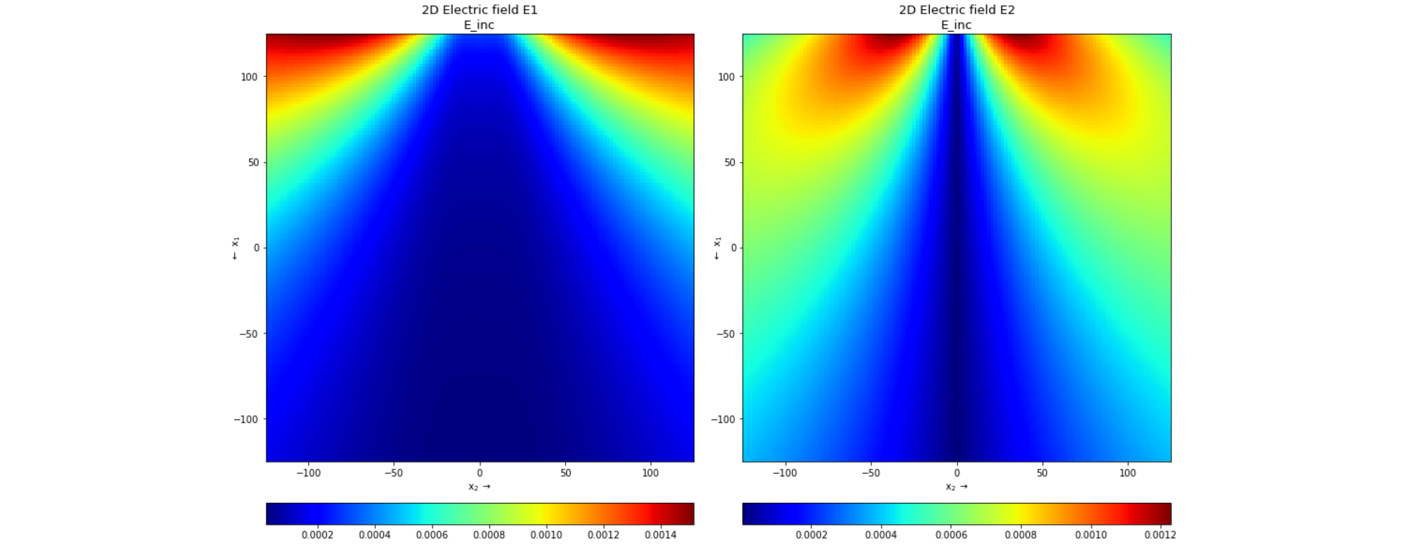


Figure 3. Incident x and y components of the incident Electric Fields.

- The Bessel-Function series are calculated in a manner to reduce the required algebraic steps using a for loop with a higher number of terms originally found in [1] due to experimentation with initEM parameters during the development process. Coefficients A and B are used in determining the reflected electric field at the receivers as denoted in equations (3.A.15-16) in [1].

- The reflected electric field and incident fields are transformed from polar co-ordinates back into cartesian co-ordinates.
- The absolute difference between the magnitudes of the analytical Bessel-Function Approach and the fields produced by the Contrast-Integral Approach is calculated and printed to the plotted diagrams below.

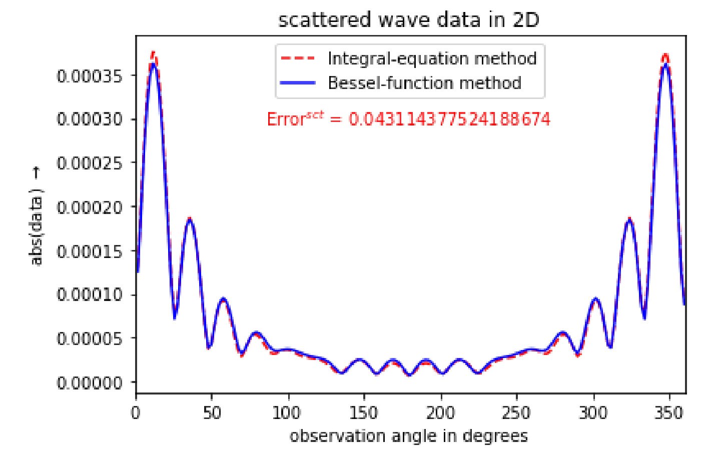


Figure 4. Scattered E-Field at receivers.

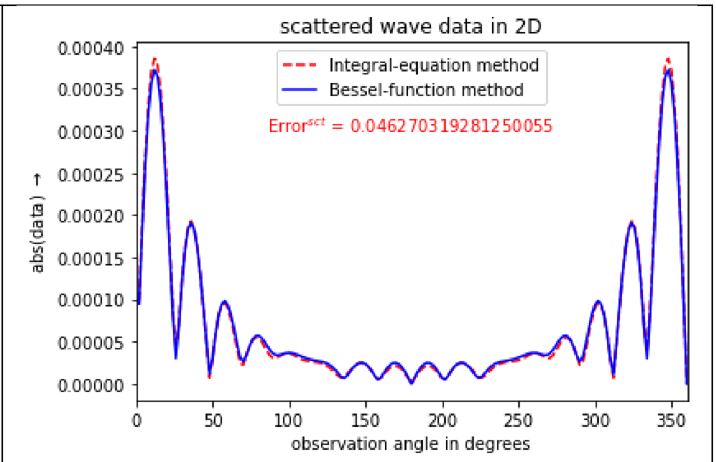


Figure 5. Scattered ZH-Field at receivers.

- The validation used in [1] is that of a 2D electric dipole line source so that comparison can be made more easily with the 3D case. Aside from validating the solver against an analytical solution, by reproducing this MATLAB code in Python, the code itself is validated for the input parameters.

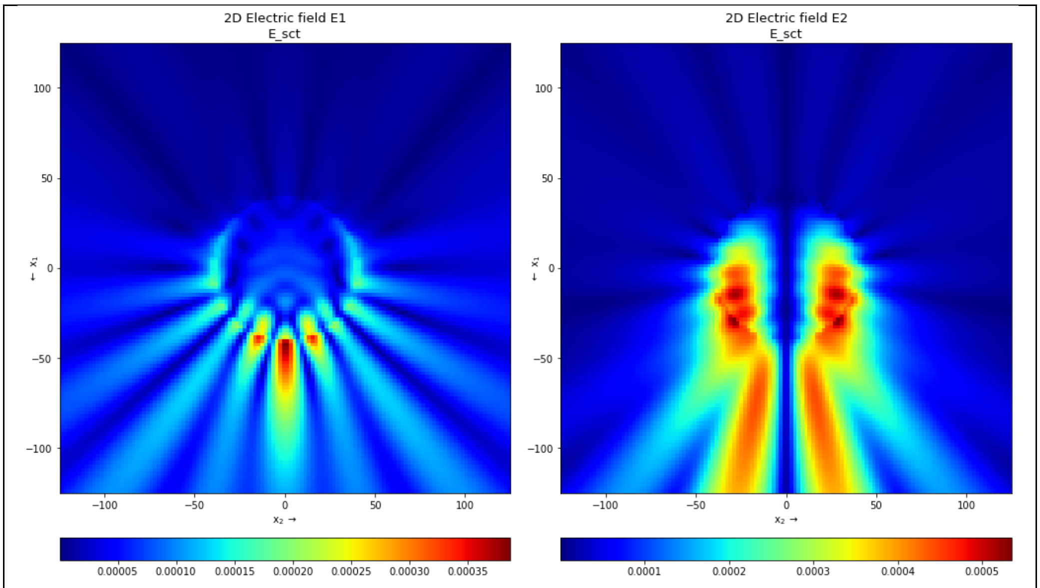


Figure 6. Final scattered electric fields for the validation case.

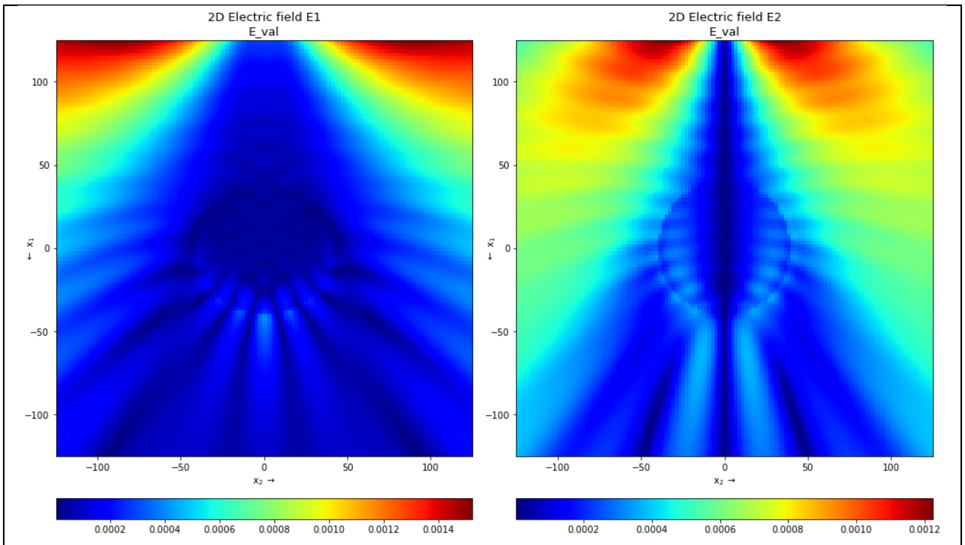


Figure 7. Final total electric fields for the validation case.

In coding the scalar code problem in Chapter 1 of [1], the student used the Python library "fn_graph" to create an illustration of the complexity of the code. While the scalar problem described is simpler the problem tackled in this project, the code flow is included below for the Bessel-Approach and the full integral solver approach.

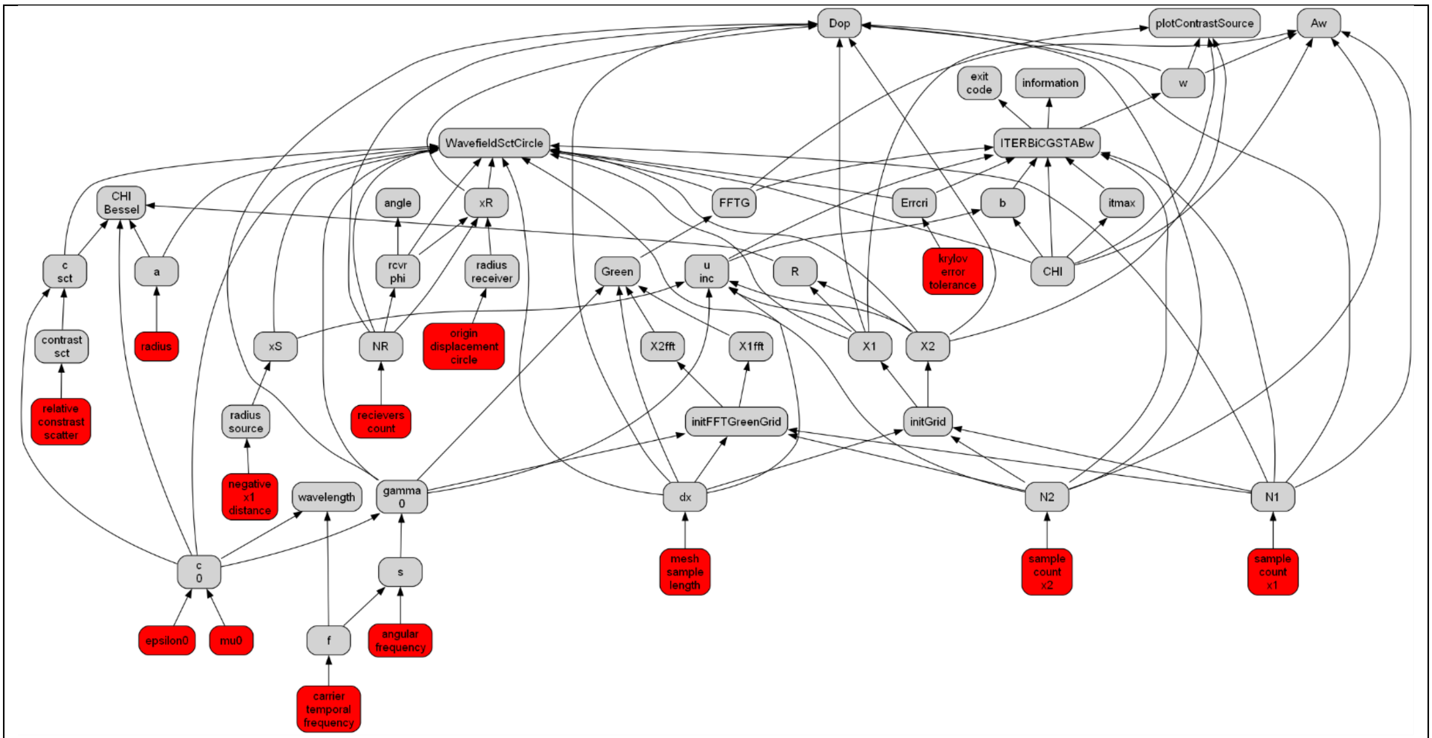


Figure 8. Bessel-Approach function flow for the scalar problem. Red indicates user inputs that determine the simulation scenario.

Model Architecture Description

The deep learning model was built using the TensorFlow and Keras libraries in Python. Details of the installation versions are found in the conda environment files on the project GitHub repository at https://github.com/spookworm/CB54 .

A simple visualisation of the layer dimensions is captured using the visualkeras Python library below. This diagram does not feature the skip connections between the encoder and decoder sides of the model.

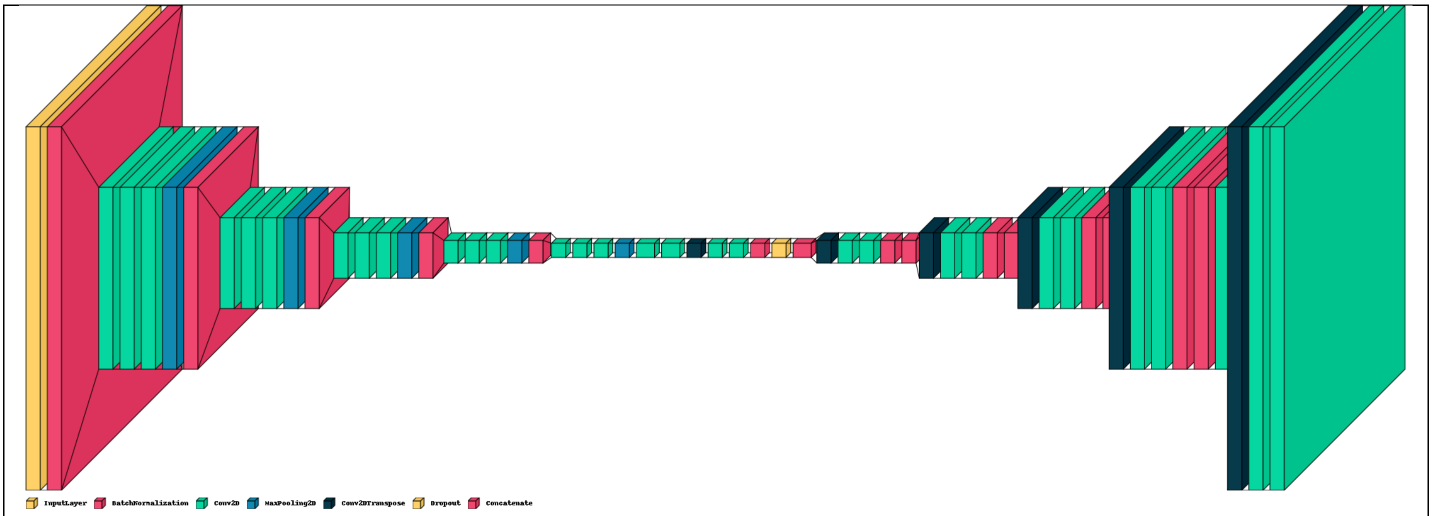


Figure 9. The model architecture used to predict both E1 and E2 scattered fields.

The full detailed model architecture is visualised using the TensorFlow plot_model library below. This can be used, aside from the code itself, to reconstruct the model architecture in future experimentation. The student recognises that the image resolution suffers within the pdf format and recommends that the reader refer directly to the source “model_plot_EM.png” file located in the subfolder “doc\Project_Design_Implementation” in the GitHub repository.

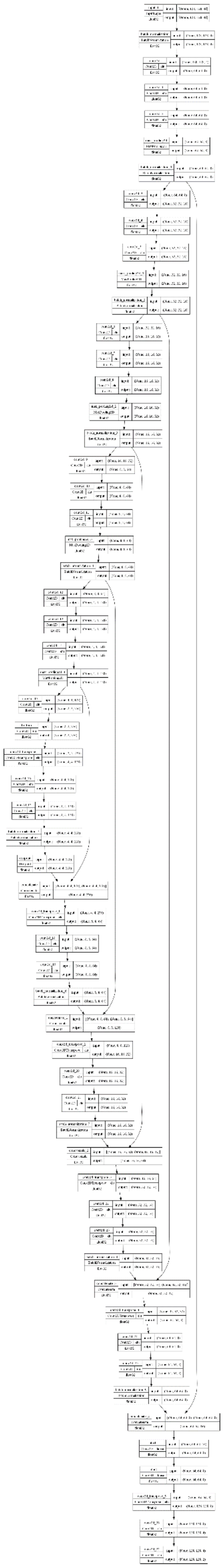


Figure 10. Model achitecture, please refer to source image on GitHub.

The initial input shape for the model was the batch size of the data followed by the height and width of the array with the channels given in the last position. This required transformation of the NumPy arrays the the time of training preparation using the function prescient2DL_data. The first channel consisted of the real component of the scatter geometry. If complex dielectric materials were to be used this would have to be adjusted and a second channel to hold the imaginary component of the scatterer would need to be included in the input. The rest of the channels consisted of the real, imaginary and absolute components of the relevant incident electric fields in the required direction. No information regarding the ZH field was used as this was considered constant for both fields where the carrier frequency was a constant. If the carrier frequency was to vary, the inclusion of the ZH incident wave would add information that may help in the training of the model.

The output of the model is a 128x128x2 tensor where the channels are the real and imaginary component of the scattered electric fields in the pertenaint directions. The student had coded, with code remaining in the files commented out, and considered using the custom loss functions for the Helmholtz-Hodge decomposition as well as training on a target field where just the absolute values of the fields were considered, however, time limitations and the usefulness of predicting this information to the SolverEMF2 workflow led to their deprioritisation.

The architecture used for both sub-models of Prescient2DL is called “DL_model” in the code for ease of adjustment. Other model architectures and variations remain as relics in the custom_architectures_EM.py for future reference and inspiration. The model is a U-net style architecture that uses “Elu” activation functions at every convolution and transposed convolution layer, as suggested in [2]. All layers also include a bias term that can be learned, rather than explicitly included, as in [3].

Each stage of the encoder/decoder configuration has a batch normalisation layer. Combined with the properties of the ‘Elu’ activation function, these layers reduce the probability of vanishing gradients during training [4]. Although this layer should also help the model with standardisation of inputs, the student found that it was nessecary to perform pre-processing standardisation of the inputs to ensure the model begins learning straigt-away during training. This was achieved by taking the minimum value found per channel from the each cell value and then dividing by the range of the minimum value to the maximum value found across all training samples. Since the standardisation proceedure can be estimated from a small batch of sample solutions, the student preferred to employ this method rather than rely purely on the model to adapt over time. The use of batch normalisation layers also makes the model more rebust to weight initialisation issues [4]. As a result, the student did not employ weight initialisation parameters in the layers as they may be prone to bad/good seeding starting points, thus moving the model development into the “seed hacking” zone. Each batch normalisation layer will also add some noise to the model, thus providing some regularization.

On the encoder side, convolution layers with stride equal to two step the height/width dimension down and increase the channel count. This is followed by a max pooling layer with pool size of 1. This essentially has no impact on the model performance as a minimum pool size is required to share information in local regions. The student left the layer in as a reminder of what the original U-net architecture employs when developing segmentation models. After the convolutions, the batch normalisation layer is again employed to skip connect layers from the encoder to the decoder side.

This encoding is carried out until a bottleneck layer is reached where the tensor dimensions are 2x2x256. A seeded dropout layer with a small value is included after the first stage of upscaling on the decoder side to add regularisation to the model.

The upscaling operations are achieved by using transposed convolution layers with stride 2. Upsampling layers were also tried but this led to stronger grid-like lines on the output predictions. Transposed convolution layers also have more trainable parameters which increases the model capacity for complexity.

At the penultimate stage of the decoder, two linear convolution layers of kernal size 3 are included in the model. Since there is a certain smoothness to the predicted fields, they are included with the aim of adding some blurring effect to the output. The student expects that this reduces some of the speckled noise that appears in the predicted versus final solution comparison graphs that could cause a sharp deteriation in the performance of the prediction as an initial guess in the SolverEMF2 workflow.

SovlerEMF2 Infusion

The manner in which Prescient2DL can be used as a stand-alone emulator is obvious: the model is provided with similar information available to the Krylov Iterative Solver and produces a prediction for the target fields with a comparable or lower error that the Krylov solver provides after one iteration. Infusing the predictions into a Method-of-Moments workflow is a more open question. The student decided to go with the simplest approach of using the predicted outputs from Prescient2DL as the initial guess for the Krylov solver. There may be more complex ways to infuse deep learning into conventional methods and the reader is referred to the recent [5] for inspiration for more entry points. As the Krylov Iterative solver at the heart of SolverEMF2 is actually searching for a solution to the contrast source as opposed to the direct scattered fields, the output from the Prescient2DL models undergoes a transformation in ForwardBICGSTABFFTwE.py using the custom_functions_EM.KopE function so it can be used as an initial guess in the BICGSTAB solver. This is not evident in [1] since the code was not intended to be used to infer guesses. No initial guess was postulated in [1] so for the naïve implementation, the incident wave was taken to be the total wave.

SovlerEMF2 Krylov Solver

In the more explicit implementation of the MATLAB code in [1], the main solver is the Conjugate Gradient Krylov solver. However, in section 3.2.3.1 it is made clear that the BICGSTAB Krylov solver performs much better in terms of reduced iteration count. Translating the MATLAB code to Python for these solvers was not straight forward as custom call-back functions had to be written so that the outputs from the MATLAB code were comparable to the outputs produced by the ITERBiCGSTABwE function. The Python library scipy.sparse.linalg was used to provide the core bicgstab and LinearOperator functions, however, they calculate the iteration steps in a different manner to the MATLAB calculation and are not easily accessible to debug. The student has formulated the code so that the outputs from the Python code match those produced by MATLAB in case development is required due to errata in [1] or other solvers or adaptations are desired in future work.

Bibliography

[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] 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.
[3] C. Brennan and K. McGuinness, “Site-specific Deep Learning Path Loss Models based on the Method of Moments.” arXiv, Feb. 02, 2023. doi: 10.48550/arXiv.2302.01052.
[4] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems*, Second edition. Beijing [China] ; Sebastopol, CA: O’Reilly Media, Inc, 2019.
[5] P. Hennig, M. A. Osborne, and H. Kersting, *Probabilistic numerics: computation as machine learning*. Cambridge New York, NY Melbourne New Delhi Singapore: Cambridge University Press, 2022.

School of Electronic Engineering

CB54: Machine Learning Algorithms for EM Wave Scattering Problems

Appendix E: Testing & Results

Anthony James McElwee
ID Number: 20211330

August 2023

MEng in Electronic and Computer Engineering

Supervised by Dr Conor Brennan

Contents

Introduction 3
Description of Training Dataset 3
E1 Model Training Commentary 3
E1 Field Model Training Loss Curves 3
E1 Field Model Training Test Scores 4
E1 Field Model Training Visual Prediction Results Absolute Component 4
E2 Model Training Commentary 6
E2 Field Model Training Loss Curves 7
E2 Field Model Training Test Scores 7
E2 Field Model Training Visual Prediction Results Absolute Component 8
Collected comments on the E1 & E2 Training Process 10
Descriptive Statistics of Testing Datasets 12
Paired t-Tests of Testing Datasets 13
DS1 Results & Analysis 13
DS2 Results & Analysis 14
DS3 Results & Analysis 14
Impact Demands 16
Conclusions 16
Bibliography 16

Introduction

All statistical tests were performed using the free and open-source statistical analysis programme JASP [1]. The JASP programmes and the associated CSV datasets are available on GitHub. A simple Bayesian alternative to the frequentist approach also resides within the JASP files.

Description of Training Dataset

The generation of seeded samples was conducted in batches of one thousand samples per folder due to the input configuration used in Python causing memory exhaustion. Five thousand samples were generated. All folders were scanned using Auslogics Duplicate File Finder 10, on the PNG files, to search for duplicate geometry samples. These duplicates and their associated NumPy files were moved to a separate duplicate folder and excluded from all further experimental activity. Earlier runs of the experiment had used an even more minimal geometric scenario generator where the main scatter was anchored at the origin. After generating 49000 samples, and extremely late in the project timeline, it was realised by the student that roughly 84% of the samples were duplicates. The training, validation and testing sets were all totally overlapping and the model was immediately overfitting on every run. After correcting for this issue, the student found that between 4000 and 5000 samples were enough to develop the model before the training loss curve started to indicate possible overfitting issues. The training set was split at 80% from the entire folder with the remaining 20% used for testing. A further split of 20% from the training set was used for validation at the end of each epoch leaving 640 samples for training in each session. Due to the removal of duplicates some folders had less than 640 training samples.

E1 Model Training Commentary

The final total model fitting time for the E1 component of the Prescient2DL deep learning model was 1117.3092517852783 seconds, which is roughly twenty minutes. This does not include the initial creation of the training/validation/testing splits which take time to transform from the sample data to correct tensor format. The screen command line printout of the final two training data batches is available on GitHub for reference.

E1 Field Model Training Loss Curves

It should be noted that GitHub hosts all of the loss curves plotted at the end of each epoch. Tensorboard was also used to track the weight updates at each layer, however, analysis and improvement on weight behaviours will remain in the domain of future work due to time limitations.

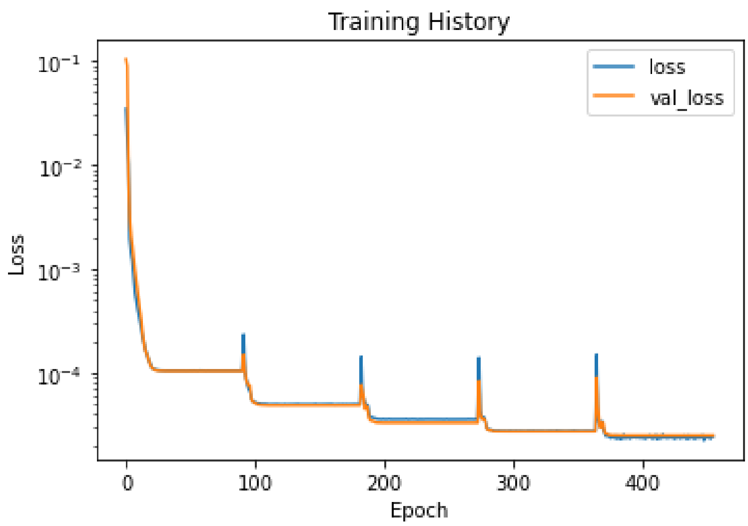


Figure 1Final Loss Curve Plot. Note the axis scale is logarithmic.

The loss curve is extremely severe when plotted on the normal axis and corresponds with loss curve plots found in the paper [2] that deals with physics deep learning applications to a laminar flow problem in the domain of fluid dynamics. Both training and validation curves track each other well In the final training session, the training curve starts to oscillate compared to the validation curve indicating possible overfitting. Although the loss curve generally follows the desired shape of a loss curve graph, albeit on a log scale, the loading spikes indicate a number of possible issues in the model design. These issues are discussed later in the appendix.

E1 Field Model Training Test Scores

For the final training dataset, the following scores were achieved:

Metric	#
Training Mean Squared Error Loss	2.3042e-05
Validation Mean Squared Error Loss	2.5000e-05
Final Learning Rate	1.0000e-22

Key Metric: For the final training dataset, the mean squared error test score after training was 2.628892798384186e-05. Since no cross-contamination of samples exist in each split, the testing error here is the correct indicator of model performance from the perspective of the deep learning model compared to the final solved solution provided as the prediction target. The final learning rate frequently was reduced to this order of magnitude throughout each training dataset run and this is reflected in the loss curves.

E1 Field Model Training Visual Prediction Results Absolute Component

The student collected prediction plots of a sample test case at the end of each training set loading stage to illustrate the improvement in the prediction power of the model as it was being trained. Each dataset contained roughly 640 samples of training data, depending on duplicates, after the testing and validation splits were removed. The plots are shown below in chronological order and can be found on GitHub. They illustrate the input geometry contrast values and the absolute field assembled form the two predicted real and imaginary component fields. The rapid convergence to low mean squared error loss is clearly captured in the difference plot on the bottom left corner, however, the source of truth and predicted fields really only become comparable as the model starts to overfit in the final two plots.

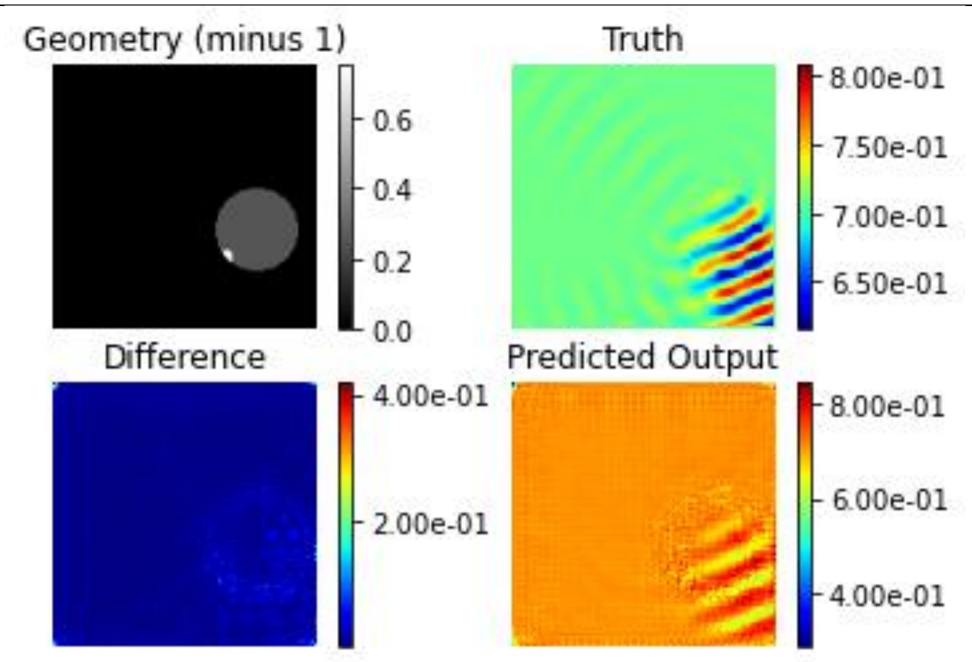


Figure 2End of first training session for E1.

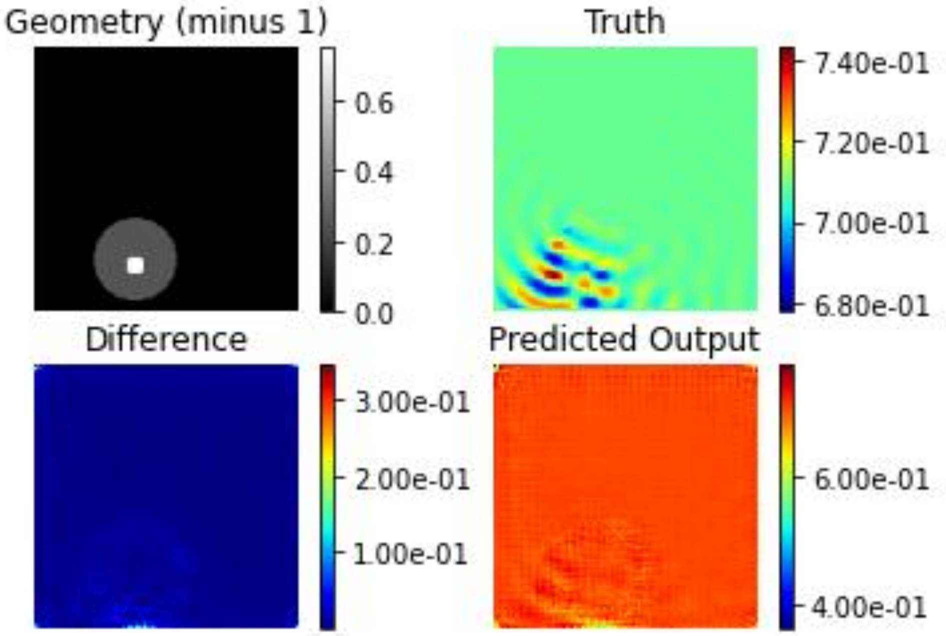


Figure 3. End of second training session for E1.

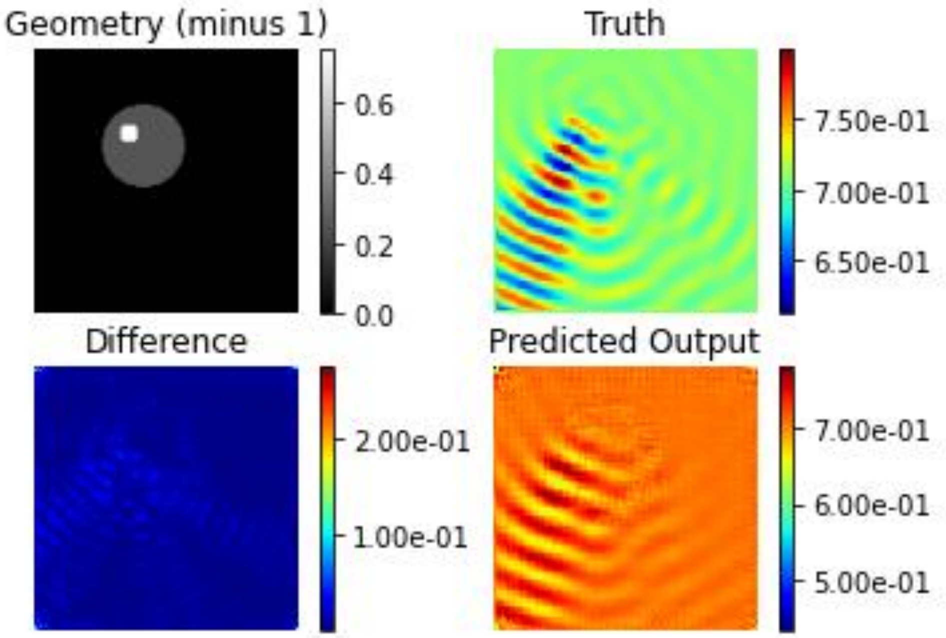


Figure 4. End of third training session. For E1.

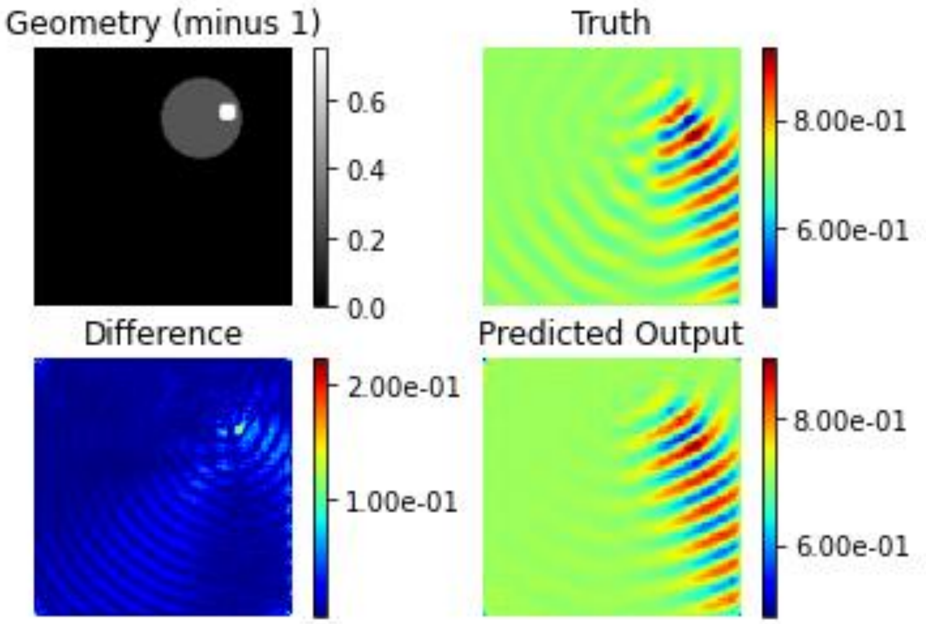


Figure 5. End of fourth training session for E1.

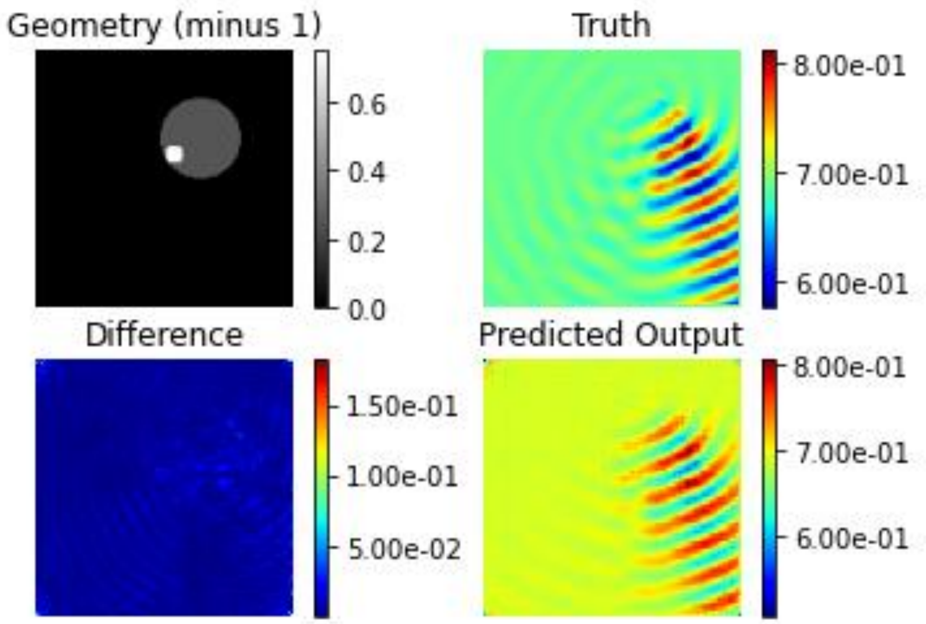


Figure 6. End of fifth training session for E1.

E2 Model Training Commentary

The final total model fitting time for the E2 component of the Prescient2DL deep learning model was 1183.819656610489 seconds, which is roughly twenty minutes. This is almost the same as the E1 component training time and this is to be expected as the same architecture was used in both cases. This does not include the initial creation of the training/validation/testing splits which take time to process from the sample data to correct tensor format. The screen command line printout of the final two training data batches is available on GitHub for reference.

E2 Field Model Training Loss Curves

It should be noted that GitHub hosts all of the loss curves plotted at the end of each epoch. Tensorboard was also used to track the weight updates at each layer, however, analysis and improvement on weight behaviours will remain in the domain of future work due to time limitations.

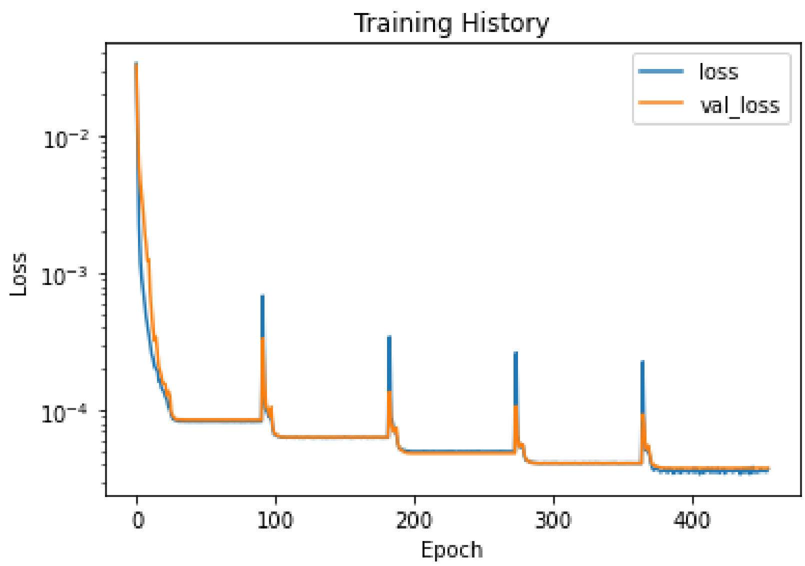


Figure 7. Final Loss Curve Plot E2. Note the axis scale is logarithmic.

E2 Field Model Training Test Scores

For the final training dataset, the following scores were achieved:

Metric	**#**
Training Mean Squared Error Loss	3.7006e-05
Validation Mean Squared Error Loss	3.7967e-05
Final Learning Rate	1.0000e-22

Key Metric: For the final training dataset, the mean squared error test score after training was 4.160570097155869e-05. Since no cross-contamination of samples exist in each split, the testing error here is the correct indicator of model performance from the perspective of the deep learning model compared to the final solved solution provided as the prediction target. This is notably higher than the E1 score, however, the complexity of the y-dimension is more apparent since the incident dipole wave has greater magnitude of change compared to the x-dimension co-ordinate. The final learning rate frequently was reduced to this order of magnitude throughout each training dataset run and this is reflected in the loss curves.

E2 Field Model Training Visual Prediction Results Absolute Component

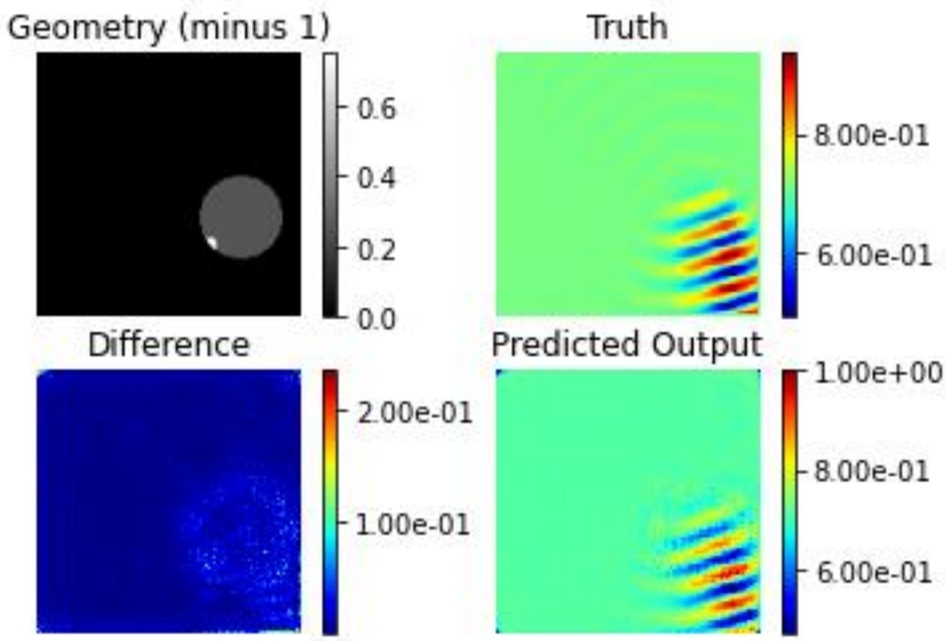


Figure 8. End of first training session for E2.

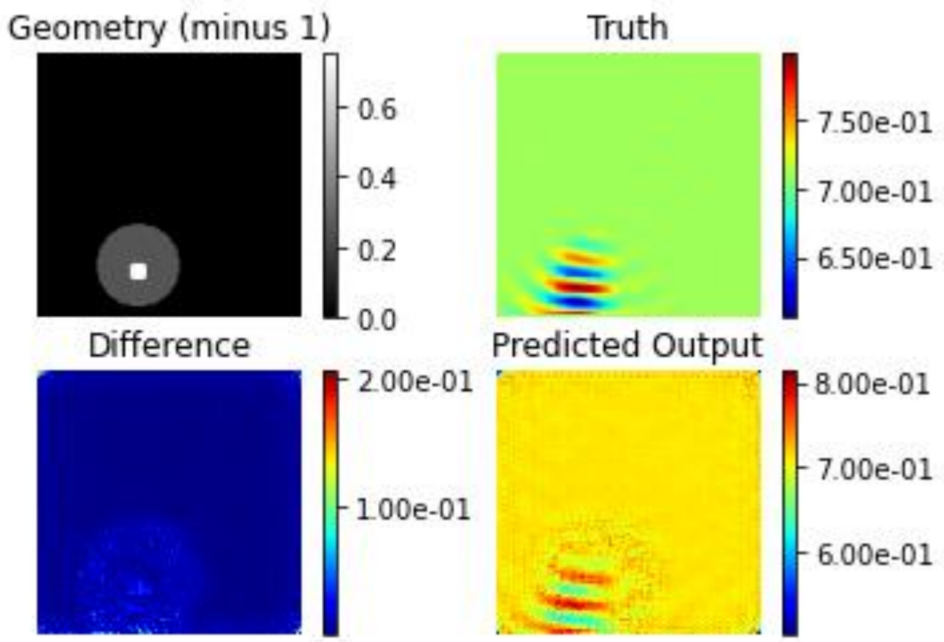


Figure 9. End of second training session for E2.

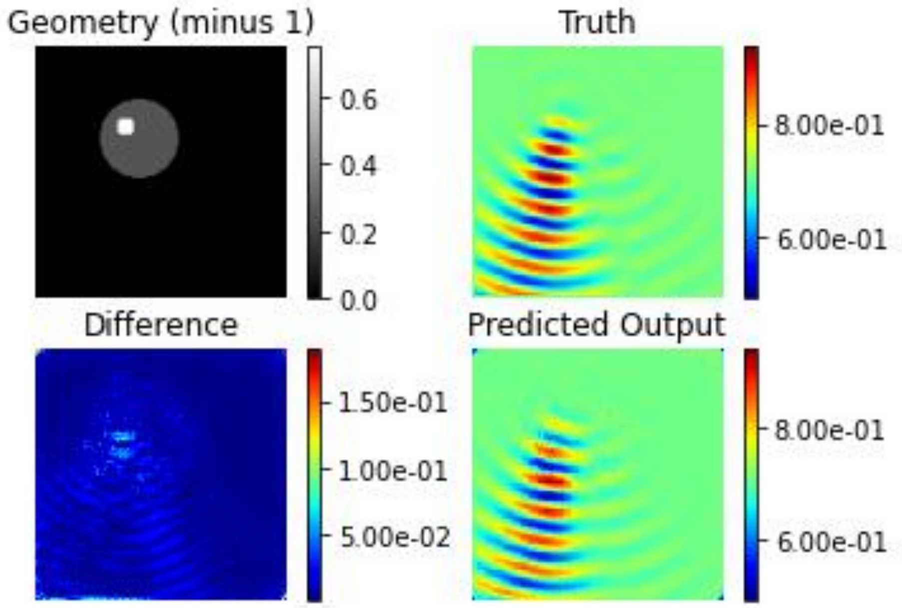


Figure 10. End of third training session for E2.

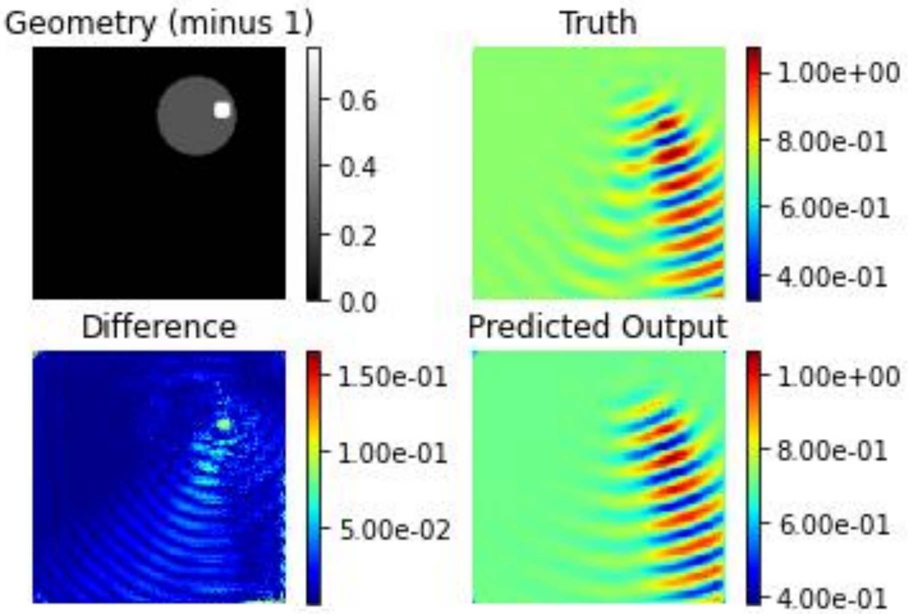


Figure 11. End of fourth training session for E2.

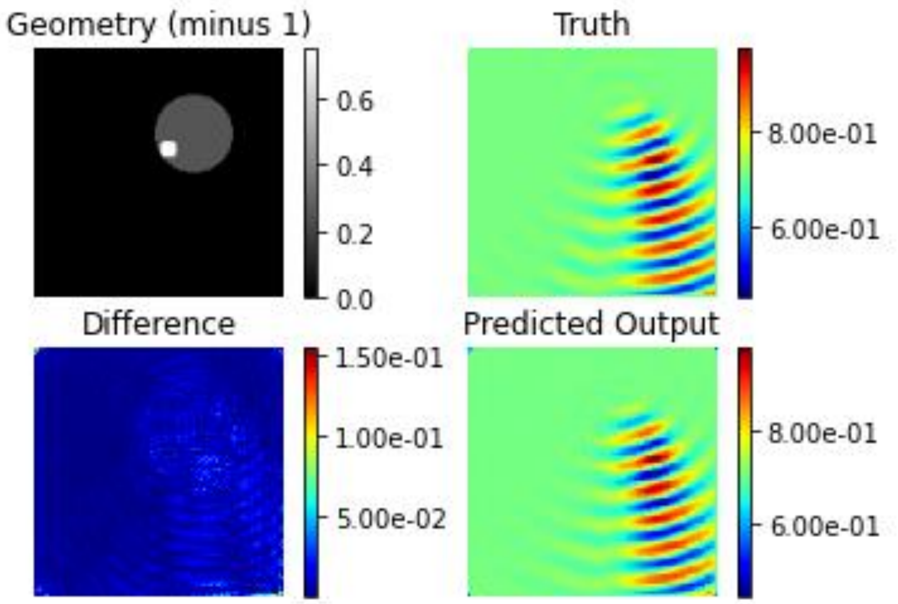


Figure 12. End of fifth training session for E2.

Collected comments on the E1 & E2 Training Process

It appears that both models are not learning in a manner usually desired in deep learning development processes. The student will now provide some commentary on this topic.

The search space of possible geometric configurations for this problem can be approximated with an upper bound on the number of cells in the grid multiplied by the possible starting positions for the smaller scatterer in the scenario. This can be roughly estimate to be (128^2 * 1600) which gives 26 million possibilities. Five-thousand geometric samples is roughly 0.02% coverage of this space. The means squared error results indicate that the deep learning models are successfully capturing a compressed version of the fields in their weights but only up to a certain degree of accuracy. Based on the oscillating training curve in the final run of the training, it is unlikely that there is insufficient training data. Indeed, more training data could be generated and up to 49000 data samples had been generated in earlier conceptions of the project development. The use of training augmentations is also eliminated since medical applications require pre-designated incident directions. Re-orientating the incident fields will not increase model generalisability that is useful to the long term goal of the research. With the exception of horizontal mirroring, data augmentation would shrink the possible permutations in the scene configuration space and increase the probability of duplicates between the training/test/validation sets.

The student finds it much more likely, based on the literature review, that the U-net architecture in its current form is sub-optimal or inappropriate for the problem at hand. The student has reflected on the use of meta-architecture in the project research log and this would be a route that research could follow off the back of the findings made in this project.

At time of writing, the student has begun to interpret the general application of deep learning to the forward problem using U-Net as an implicit recasting of the problem to an inverse one where the inputs are basically noisy inputs that need to be recovered backwards to their scattered fields. This had been the students long term outlook when conducting the literature review, except using different prior information gleamed from Von Neumann integral expansions in the higher frequency approximations to the wave scattering configurations and then applying denoising to complete the scene. There is a conceptual connection in this idea to the GANs approach documented in [3].

Although the student cannot rule out the possibility that the model has received inadequate training time, the rapid convergence to a similar order of magnitude of loss through all training sessions implies that this is not a training time duration issue. The student also believes increasing the training data size to cover more of the possible permutations would not scale well when higher-contrast problems are tackled since this would dramatically increase the amount of training data required to achieve results, in direct contradiction to developing the model in the first instance. It is a key requirement that only a small sample set is required to train and test the model, as this is what would be available to researchers at the low-frequency high-contrast end of the problem spectrum.

While the predicted target data was pre-processed and then post-processed to bring the values within a smaller range suitable for deep learning, this was not done for the input data. For this low-contrast problem where the incident waves are already tightly bounded in value, it is unlikely that this is causing a problem. However, the student notes that in the high-contrast scenarios such processing would also need to be carried out on the input data in order for the model weights to adjust quickly and avoid losing permutations to bias adjustment. As already commented, the model incorporates a bias term in each convolution layer but this bias term still needs to learn so processing the input data further may help this delivery. The student has included many batch normalisation layers in the architecture in a bid to alleviate this issue.

The student has adapted a learning rate that reduces over time as the loss curve saturates. The use of an increased batch size compared to [4] up to where the computer memory would allow should also have aided the improved training performance. The model also had two extra final linear layers that aimed to provide a blurring effect on the arrays, given the smoothness assumption for the scattered fields. The student also included a Dropout layer need the lower end of the decoding side of the model. The student notes that the use of batch layer normalisation may help the model to train more smoothly, however, implicit regularisation may also be arising from these layers and the model may be struggling to reduce the loss values with higher accuracy because of their inclusion at so many levels. The student also did not use advanced hyper-parameter tuning or meta-search libraries such as AutoKeras. In general, the student would suggest experimenting with regularisation on the activity of some of the convolution layers should further research be conducted on this specific model architecture.

Although not formally reported, the student tried to use Xavier weight initialisation as suggested in [4]. This did not positively impact the learning curves. The student used Adam optimiser based on the literature review.

Finally, the student wants to raise the possibility that the application of deep learning to this problem may not be suitable beyond the emulation case. This project has failed to find evidence that infusing deep learning models into conventional Krylov based solvers can improve their convergence properties. While the model has been shown to provide a decent estimation of the target fields, substantial benefits to existing methodologies can only be claimed if either the initial error arising from using the deep model lower the iteration count of the Krylov solver or help the Krylov solver to converge at a faster rate to a solution that meets the error criterion of the simulation. Evidence for either of these goals was not found in the experiments conducted in this project. This is evidenced in the next section.

Descriptive Statistics of Testing Datasets

The following table gives the descriptive statistics for each statistical test set used to evaluate the impact of Prescient2DL on SolverEMF2. Each set consisted of 100 original samples solved using the naïve initial guess of the incident wave as the scattered field. None of these samples appeared in the training/validation or test sets used during model development. After training the models for predicting the two scattered fields, a second run of SolverEMF2 was used on the same original samples, allowing for direct comparison across duration of calculation, iteration count and initial error between the original sample information ("_o") and the model-assisted sample information ("_m").

Metric	N	Mean	SD	SE	Coefficient of variation
DS1					
Duration_o	100	1.106213	0.055213	0.005521	0.0499113
Duration_m	100	1.010293	0.093377	0.009338	0.0924252
Iteration_Count_o	100	22.57	0.655282	0.065528	0.0290333
Iteration_Count_m	100	22.05	0.479373	0.047937	0.0217402
Error_Initial_o	100	0.004857	0.002712	0.000271	0.5583008
Error_Initial_m	100	0.001102	0.000492	4.92E-05	0.4466384
DS2					
Duration_o	100	0.772	0.132	0.013	0.171
Duration_m	100	0.72	0.071	0.007	0.099
Iteration_Count_o	100	19.57	0.573	0.057	0.029
Iteration_Count_m	100	19.35	0.52	0.052	0.027
Error_Initial_o	100	0.003	0.002	1.963×10^{-4}	0.619
Error_Initial_m	100	8.014×10^{-4}	3.702×10^{-4}	3.702×10^{-5}	0.462
DS3					
Duration_o	100	2.218	0.198	0.02	0.089
Duration_m	100	2.308	0.246	0.025	0.106
Iteration_Count_o	100	56.65	1.048	0.105	0.019
Iteration_Count_m	100	56.58	0.955	0.096	0.017
Error_Initial_o	100	0.03	0.019	0.002	0.611
Error_Initial_m	100	0.02	0.011	0.001	0.533

Paired t-Tests of Testing Datasets

Between the descriptive statistics of the testing datasets and the t-tests conducted on the variables, the impact of Prescient2DL on Solver EMF2 can be established. In the attached appendix Project Plan Proposal, the Primary research hypothesis were tested. The Primary Research question was orientated around using Prescient2DL to improve the Krylov Iterative solver at the heart of SolverEMF2. DS1, DS2 and DS3 testing datasets had 100 samples each which gives enough degrees of freedom to ensure a large enough sample size to compare the original solution information to the information gathered during the model assisted solution. Please review the Project Plan Proposal for the details of the three hypothesis: "Primary Research Test 01 – Initial Solution Conveyance t-Test"; "Primary Research Test 02 –Solution Convergence t-Test"; "Primary Research Test 03 – Solution Conveyance t-Test".

DS1 Results & Analysis

DS1: Paired Samples T-Test								
Measure 1	**Measure 2**	**t**	**df**	**p**	**Mean Difference**	**SE Difference**	**Cohen's d**	**SE Cohen's d**
Duration_o	Duration_m	8.9132305	99	< .001	0.0959198	0.0107615	0.8913231	0.1656468
Iteration_Count_o	Iteration_Count_m	7.2478005	99	< .001	0.52	0.0717459	0.7247801	0.1394524
Error_Initial_o	Error_Initial_m	16.5942404	99	< .001	0.0037544	0.0002263	1.659424	0.0587952

In the results for DS1, all p-values were deemed statistically significant. However, looking beyond the t-test results and comparing the mean differences for the Duration and Iteration Count variables, the impact the model has on Solution Convergence and Solution Conveyance is inconsequential to the solution progress of the Krylov Iterative Solver. On average, the same number of iterations are required to meet the error criterion of the solver. This is reflected in the Raincloud plots for the Duration and Iteration count variables.

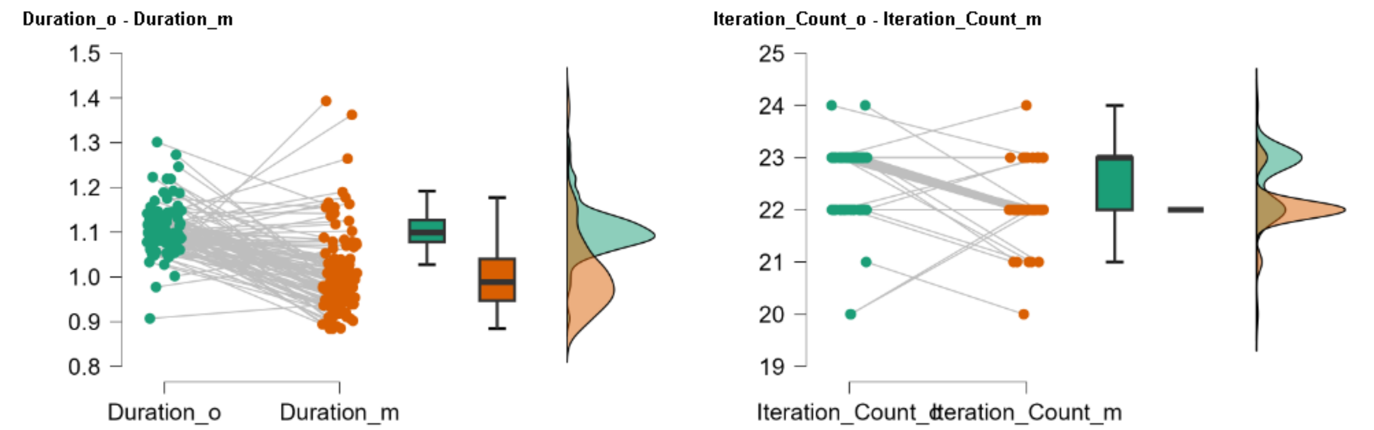


Figure 13. Raincloud plots illustrating the similiarity between the original and model-assisted samples. Although statistically different, the lowering of the variables by the model is immaterial to the solver.

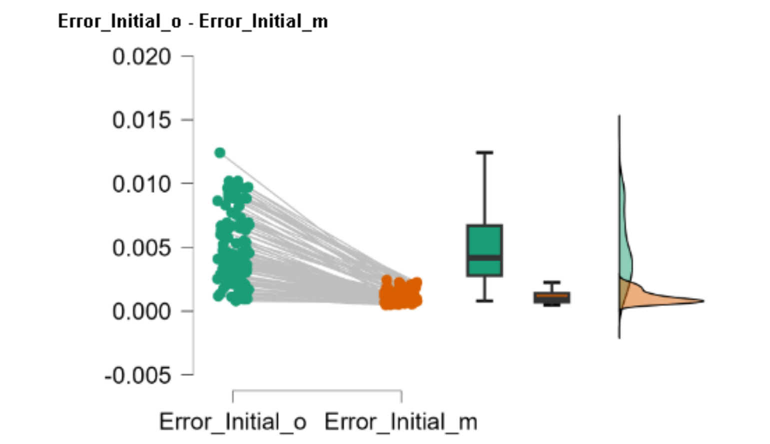


Figure 14. Raincloud plot of initial error comparison. While the initial error in the model-assisted approach is lower and more consistant then the naive approach, it is not low enough to impact SolverEMF2.

The Initial Solution Conveyance based on a comparison of the means of the Error Initial variable is statistically significant with a seemingly impressive mean difference of 0.0038. This is roughly 18% of the original initial error in the naïve approach. As demonstrated, this is enough to give a close visual similarity that would be useful in the early stages of a design methodology where error criteria are low. Although the statistical test rejects the null, indicating a clear performance difference exists between the two approaches, the student concludes that the impact of Prescient2DL is immaterial to achieving the error criterion set out at the initialization stage of the solver.

DS2 Results & Analysis

The aim of the DS2 dataset was to establish how Prescient2DL would handle geometric configurations beyond those it was exposed to in the training stage of development. By setting the smaller scatterer to the same zero contrast value as the background embedding, a deformation or hole was created in the main scatterer geometry.

DS2: Paired Samples T-Test								
Measure 1	**Measure 2**	**t**	**df**	**p**	**Mean Difference**	**SE Difference**	**Cohen's d**	**SE Cohen's d**
Duration_o	Duration_m	3.394	99	< .001	0.052	0.015	0.339	0.15
Iteration_Count_o	Iteration_Count_m	4.2	99	< .001	0.22	0.052	0.42	0.1
Error_Initial_o	Error_Initial_m	14.493	99	< .001	0.002	1.634×10^{-4}	1.449	0.061

Again, all p-values indicate a rejection of the null hypothesis. Prescient2Dl is able to operate with this mild generalization. All three metrics have reported a decrease in the mean difference between naïve and model-informed runs, however, the decrease for the Error Initial metric is mild with a score of roughly 27% compared to the initial error offered by the original, naïve approach.

DS3 Results & Analysis

The aim of the DS3 was to push Prescient2DL into predicting scenarios where the contrast values populating the scene were significantly different to those it was exposed to during training. In this case, both the small and larger scatterers took the same higher contrast value.

DS3: Paired Samples T-Test								
Measure 1	**Measure 2**	**t**	**df**	**p**	**Mean Difference**	**SE Difference**	**Cohen's d**	**SE Cohen's d**
Duration_o	Duration_m	-3.3	99	0.001	-0.09	0.027	-0.33	0.125
Iteration_Count_o	Iteration_Count_m	0.572	99	0.569	0.07	0.122	0.057	0.122
Error_Initial_o	Error_Initial_m	12.829	99	< .001	0.01	8.095×10^{-4}	1.283	0.017

The Duration metric indicates that the model actually has a detrimental impact on SolverEMF2. The evidence supplied to the t-test results in an inability to reject the null hypothesis for Iteration Count. As already commented in DS1 and DS2, neither of these mean differences are material to the Krylov solver in the grand scheme of the workflow. The success measured by the Error Initial variable has also deteriorated but still reports roughly 66% of the original, naïve approach. Even though Prescient2DL was not exposed to single scatterer geometries with such a high population of higher contrast values, it was still able to lower the initial error. A sample prediction outcome is presented in the figure below. The titles indicate the figure content. It is evident that the visual resemblence between the predicted fields and the actual fields has broken down, even if the residual error remains lower than that of a naïve guess.

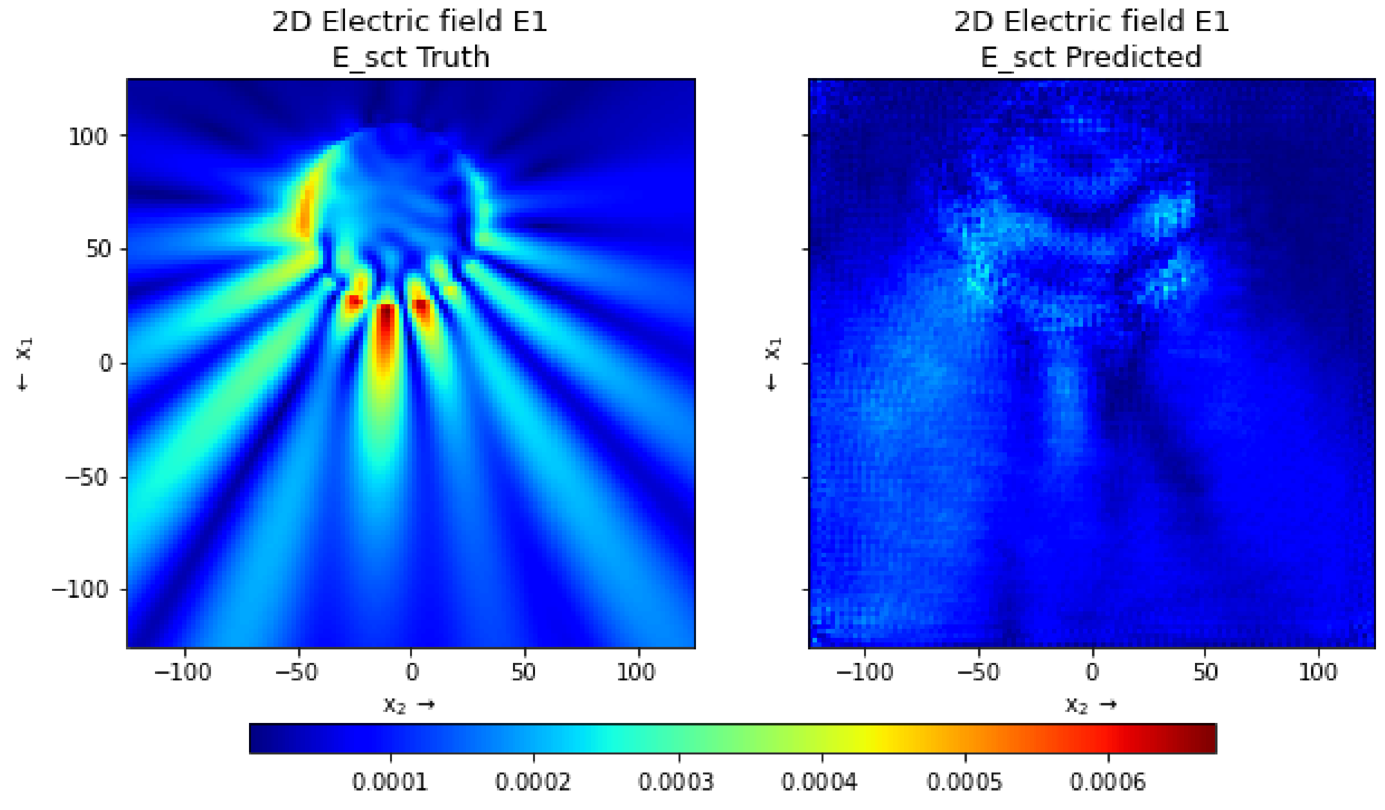


Figure 15. Comparison between the actual scattered field on the left and the predicted field on the right for E1.

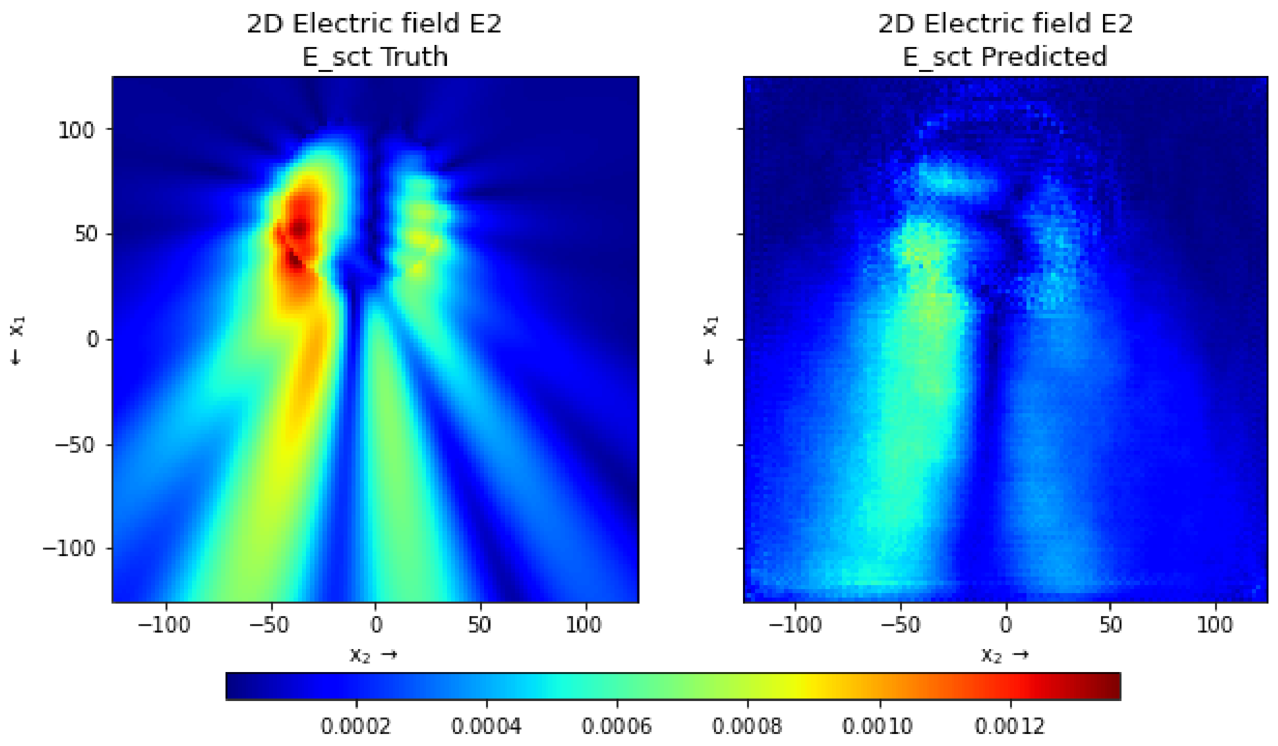


Figure 16. Comparison between the actual scattered field on the left and the predicted field on the right for E2.

Impact Demands

If the current tests and results show low means squared error but immaterial impact upon SolverEMF2, how would results that actually achieve Method of Moments solution acceleration manifest in this scenario? Using a DS1 sample case, the student prints out the first seven entries from the information gathered during a naïve and model-assisted run in the table below.

Sample 5101	**Naïve Original Run**		**Prescient2DL Assisted Run**	
Iteration	**Residual Error**	**Duration**	**Residual Error**	**Duration**
0	0.0020748100	0.0102043000	0.0005120250	0.0085418200
1	0.0002348990	0.0905674000	0.0000702384	0.0611167000
2	0.0000368364	0.1306520000	0.0000158675	0.0931430000
3	0.0000102315	0.1706970000	0.0000042492	0.1249560000
4	0.0000010167	0.2137400000	0.0000005663	0.1580040000
5	0.0000002997	0.2488170000	0.0000000635	0.1890330000
6	0.0000000401	0.2823450000	0.0000000171	0.2185600000

Table 1. Iterative solver information for sample 5101 using DS1 scene parameters. Only first 7 entries are displayed.

In order for Prescient2DL to impact the Method of Moments solver by accelerating the solution process, the initial residual error would need to be lowered below at least these severn error levels reported in the residual error column for the Naïve Original Run. Due to dependance of convergence rates for Krylov solvers on the conditioning of the matrices and the eigenvalue properties of the matrices, deep learning models would need to be achieving initial residual errors of 10^{-8} or lower in this toy scenario to lower the final iteration count by even 25% of the naïve solution iteration count.

Conclusions

Although Prescient2DL is statistically achieving differences in metric performance compared to naïve approaches to the initial guess in SolverEMF2, there appears to be nothing gained from using the model at all with regard to accelerating the Method of Moments solver. Plots of convergence and conveyance, as postulated in the Project Plan Proposal, would all be identical between original and model-assisted runs. Before further experimentation with generalizability and extrapolation is to be carried out, it would be the student's opinion that lowering the initial error so that Prescient2DL impacts the SoverEMF2 in terms of convergence would be made the priority. This project has presented evidence and test results that show research in this area remains at the model development stage. If the creation of models that bring the initial residual error to orders of magnitude lower than currently reported is deemed non-viable, after more extensive experimentation, then the attempted application of deep learning in electromagnetic scattering forward problems could be regarded as frivolous.

Bibliography

[1] JASP Team, "JASP (Version 0.17.3)." 2023. [Online]. Available: https://jasp-stats.org/

[2] T.-T.-H. Le, H. Kang, and H. Kim, "Towards Incompressible Laminar Flow Estimation Based on Interpolated Feature Generation and Deep Learning," *Sustainability*, vol. 14, no. 19, Art. no. 19, Jan. 2022, doi: 10.3390/su141911996.

[3] Z. Ma, K. Xu, R. Song, C.-F. Wang, and X. Chen, "Learning-Based Fast Electromagnetic Scattering Solver Through Generative Adversarial Network," *IEEE Trans. Antennas Propag.*, vol. 69, no. 4, pp. 2194–2208, Apr. 2021, doi: 10.1109/TAP.2020.3026447.

[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.

School of Electronic Engineering

CB54: Machine Learning Algorithms for EM Wave Scattering Problems

Appendix F: Source Code Listing

Anthony James McElwee
ID Number: 20211330

August 2023

MEng in Electronic and Computer Engineering

Supervised by Dr Conor Brennan

Declaration

I hereby declare that, except where otherwise indicated, this document is entirely my own work and has not been submitted in whole or in part to any other university.

Signed: Anthony James Mc Elwee

Date: 2023/08/20

Source Code Location

All code is hosted on a GitHub repository at https://github.com/spookworm/CB54 and will reside there until the final grade for the project is released. Aside from code, the repository also contains the documentats required for the final portfolio, legacy code that was used as raw reference during development and compilation commands that generate the conda environment and wrap the final documents together into the submission template.

Figure 1 The presentation of the portfolio materials on GitHub.

Code Description

The important code files with brief description are as follows:

- Main Folder: ForwardBiCGSTABFFTwE.py: This is actually SolverEMF2, due to changes in the development scheme the code for generating the data reside here rather than the scripts with "solveremf2" in their name.
- Main Folder: prescient2dl.py: This is the deep learning model development part of the code. The training files are assembled here too.

- Lib Folder: custom_architectures_EM.py: This is where the various deep learning model architectures were saved during development. Some plotting functions for the predictions and loss curves also reside here.
- Lib Folder: custom_functions_EM.py: This is where the van den Berg code, adapted from [1] resides along with support scripts for generating reports and plotting diagrams etc. The student stresses that they tried to keep the code in the same structure as the original MATLAB code to enable referral to the main text if required by future developers. No plagerism is intended and credit and referencing of [1] is frequent throughout the documentation.
- Lib Folder: custom_tensorboard.py: Basic code to terminate the Tensorboard instance before refreshing the deep learning model. This caused some trouble on Windows as the instance was blocking anything from running in the background unless the previous session was terminated first.

All other code is either reference/legacy code used in the development process or documentation code that does not apply to the project problem of electromagnetic wave scattering.

Note: Even more code exists that was developed during the course of the project but it has not been posted on GitHub as there was no time to implement it directly in the main code bodies. For example, the student had developed code to tie in the Python library Gradio to demonstrate the models with a user input geometry file.

[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

Ollscoil Chathair
Bhaile Átha Cliath
Dublin City University

School of Electronic Engineering

CB54: Machine Learning Algorithms for EM Wave Scattering Problems

Appendix G: Risk Assessment

Anthony James McElwee
ID Number: 20211330

August 2023

MEng in Electronic and Computer Engineering

Supervised by Dr Conor Brennan

Risk Assessment Experimental Method Form for Undergraduate and Taught PG Projects

All operations/procedures being assessed (give specific details):

All operations/procedures are purely theoretical or computational requiring no written risk assessment.

Risk Category Rating:

E

Known or expected hazards associated with the activity:

None

Precautions to be taken to reduce the level of risk:

None

Training prerequisite:

None

Risk remaining:

None

Emergency procedures:

None

Detail references if any:

For the Project Worker and Project Supervisor:

We have carried out a risk assessment for the above operation/procedure in accordance with those guidelines as detailed in the School Safety Handbook.

Signature of Project Worker: Anthony James McElwee Date: 2023/02/05

Print name of Project Worker: **ANTHONY JAMES MC ELWEE**

Signature of Project Supervisor: … Conor Brennan Date 27/02/2023

Print name of Project Supervisor: **DR CONOR BRENNAN**

Print name of Technical Officer assigned to Project: **CONOR MURPHY**

N.B.

- Copies of completed forms should be submitted to the Project Supervisor and the Technical Officer assigned to the project.
- A signed copy of the completed form should be kept in close proximity to the project bench/space where the project is taking place.