CSE 6730 - Spring 2022
Project - Final Report

Team Members: Guangyu Cui Chi-Nuo Lee Hao Wu
Project Title: Simulating Physical Systems with Modern Techniques

1 Abstract

Electrical Impedance Tomography (EIT) is a non-invasive medical imaging technique that uses the difference in electric conductivity and permittivity of tissues in the body. By injecting current and measuring with surface electrodes, one can use reconstructing algorithms and transform measurements into a tomographic image. This process is an example of solving differential equations in a class of inverse problems (IP). In this report, we explore how three different modern deep learning methods can be used to solve this inverse problem, and compare the effectiveness of these methods. These methods are physics-informed neural networks (PINN) , deep Ritz method (DRM), and weak adversarial network (WAN). We followed the procedure of [1], and compared the result in detail with PINN and DRM which is the unique contribution of our report.

2 Project Description

Inverse problems (IP) are a common and important topic for a wide variety of scientific disciplines. They can be generally formulated in the following way [1]: Let Ω be an open and bounded set in $\mathbb{R}^d$, and

$$\mathcal{A}[u,\gamma] = 0, \quad \text{in } \Omega \tag{1a}$$
$$\mathcal{B}[u,\gamma] = 0, \quad \text{on } \partial\Omega \tag{1b}$$

$\mathcal{A}$ is a differential equation that can be a ordinary differential equation (ODE), partial differential equation (PDE), or an integro-differential equation (IDE). $\mathcal{B}$ denotes the boundary valus on $\partial\Omega$ u denotes the solution and γ the coefficient in the inverse medium problem or the source function in the inverse source problem.

In this project, we consider the special IP of the electrical impedance tomography(EIT) [2, 3] with the following formulation[1]:

$$-\nabla \cdot (\gamma \nabla u) - f = 0, \quad \text{in } \Omega \tag{2a}$$
$$u - u_b = 0,\ \gamma - \gamma_b = 0,\ \partial_{\vec{n}} u - u_n = 0, \quad \text{on } \partial\Omega \tag{2b}$$

Here, the solution we attempt to solve γ signifies the electrical conductivity distribution and u the potential. u_b is the measured voltage, γ_b is the conductivity near the surface of the object and $u_n := \nabla u \cdot \vec{n}$, where $\vec{n}$ the outer normal of $\partial\Omega$.

We will apply three different types of deep learning methods to this problem: physics-informed neural networks (PINN) [4], deep Ritz method (DRM) [5], and weak adversarial network (WAN) [6]. We will compare the accuracy of the three methods.

- PINN aims to minimize the l2 or some other norm of the error

$$MSE = \frac{1}{N_u} \sum_{i=1}^{N_f} \left| u(x_u^i, t_u^i) - u^i \right|^2 + \frac{1}{N_f} \sum_{i=1}^{N_f} \left| f(x_f^i, t_f^i) \right|^2, \tag{3}$$

- DRM uses the variational formula of elliptic PDE and aims to minimize

$$\min_{u \in H} \int_\Omega \left(\frac{1}{2} |\nabla u|^2 - f(x)u(x) \right) dx, \tag{4}$$

- WAN uses weak formulation of PDE and introduces the test function ϕ, it aims to solve a minmax problem

$$min_{\gamma,u} max_{\phi}\{|\langle \nabla \cdot (\gamma \nabla u), \phi \rangle| + \lambda ||u - a(x)||_{\partial\Omega}\} \tag{5}$$

3 Literature Review

3.1 Problem Statement

Many physical problems can be formulated as differential equations (ODE or PDE). They are described by a set of equations which may evolve the (time/space) derivatives of functions. We aim to solve the functions from the given information which is typically the boundary condition or initial condition.
For example, the transfer of heat is described by the heat equation:

$$\frac{\partial u}{\partial t} = \alpha \Delta u \tag{6}$$

Here $u(x, t)$ is the function that gives temperature at time t and position x, α is the thermal diffusivity of the medium.
And the propagation of wave is described by another type of equations with the following form:

$$\frac{\partial^2 u}{\partial t^2} = c^2 \Delta u \tag{7}$$

We can get some qualitative results by analyzing the equations. For example, it's well known that heat propagation doesn't take time while the propagation of wave has speed c.
However, we need to do numerical simulations to get more detailed information. Nowadays there are many numerical methods available, including traditional PDE solvers like finite difference method or finite element method, and modern machine learning based approaches. In this project, we plan to study and simulate some physical systems with these techniques and compare the results. Hopefully, we can gain insights about the real physical system from the numerical solutions.

3.2 Previous Work

Traditional numerical methods often divide the space with many grids. The function value at these grid points are unkowns, and the equations are discretized to get a linear system. Solving the linear system can give us the values at each grid point. However, as dimensional goes higher, the number of grid points required grows exponentially, and the computation cost will become extremely expensive. That's the so called 'curse of dimensionality'. On the hand, most machine learning appoaches use sample points instead of grids, and the functions are more flexible. This can reduce he computational cost and make the problems solvable. That's one of the advantages of machine learning approaches against traditional PDE solvers.
Each type of differential equations has its own properties, so people developed different neural network based approaches for each type.

3.2.1 PINN (Physics-Informed Neural Networks)

In [4], the authors trained the network u as the solution of the PDE, where they consider parametrized and nonlinear differential equations of general form

$$u_t + \mathcal{N}[u;\lambda] = 0. \tag{8}$$

Denote $f(x,t) := u_t(x,t) + \mathcal{N}[u(x,t);\lambda]$, one can obtain the solution u by minimizing the mean squared error

$$MSE = \frac{1}{N_u}\sum_{i=1}^{N_f}\left|u(x_u^i, t_u^i) - u^i\right|^2 + \frac{1}{N_f}\sum_{i=1}^{N_f}\left|f(x_f^i, t_f^i)\right|^2, \tag{9}$$

where $\{x_u^i, t_u^i, u^i\}_1^{N_u}$ denotes the initial and boundary training data for $u(x,t)$, and $\{x_f^i, t_f^i\}_1^{N_f}$ specifies the collocation points for $f(x,t)$.

3.2.2 DRM (Deep Ritz Method)

In [5], the authors use the variational form of elliptic PDEs to derive the formulation and design the algorithm.

Consider the variational form of PDE problem

$$\min_{u\in H}\int_\Omega\left(\frac{1}{2}|\nabla u|^2 - f(x)u(x)\right)dx, \tag{10}$$

and H is the set of admissible functions (also called trial function, here represented by u), f is a given function, representing external forcing to the system under consideration. u is approximated by a neural network, and the authors use SGD to minimize the loss. The minimizer gives solution to the original PDE. Problems of this type are fairly common in physical sciences.

3.2.3 WAN (Weak Adversarial Networks)

In [6], the authors use weak form of PDEs and introduces the test function ϕ, it aims to solve a minmax problem

$$min_{\gamma,u} max_\phi\{|\langle\nabla\cdot(\gamma\nabla u), \phi\rangle| + \lambda||u - a(x)||_{\partial\Omega}\}, \tag{11}$$

where the authors proved that with the energy minimized, u converge to the solution of the following PDE:

$$\nabla\cdot(\gamma\nabla u) = f \tag{12}$$

$$u|\partial\Omega = a(x). \tag{13}$$

With the energy(11), the order of the derivative of u is reduced such that it can handle solution with discontinuity.

4 Conceptual Model

In this project, we are interested in the classical inverse conductivity problem in electrical impedance tomography (EIT) [2]. The goal of EIT is to determine the electrical conductivity distribution $\gamma(x)$ on Ω based on the potential u, the current $-\gamma\partial_{\overrightarrow{n}}u$ measurements, and the knowledge of γ on the boundary $\partial\Omega$:

$$-\nabla\cdot(\gamma\nabla u) - f = 0, \quad \text{in } \Omega \tag{14}$$

$$u - u_b = 0, \gamma - \gamma_b = 0, \partial_{\overrightarrow{n}}u - u_n = 0, \quad \text{on } \partial\Omega. \tag{15}$$

where u_b and γ_b are measured voltage and conductivity near the surface of the object. With such boundary condition provided, one needs to solve the BVP and find correct γ at the same time.

5 Simulation Model

In our simulation model, we design a suitable loss function, apply stochastic gradient descent (SGD) for our parameterized u and γ simultaneously, since the loss is designed such that the energy can be minimized with the ground truth conductivity, we consider γ as simulated result when the algorithm converges.

As for the validation, since ground truth is known from the design of experiment, we can simply compare our predicted result with the ground truth.

6 Experiment

6.1 Experiment setup

In our experiment, we considered an artificial EIT problem on $\Omega = (0,1)^2$ but replace the classical boundary condition with $\gamma\nabla u\cdot\overrightarrow{n}$. We set the ground truth as:

$$\gamma^*(x) = \pi^{-1}\exp\big\{(d-1)\pi^2(x_1 - x_1^2)/2\big\}$$
$$u^*(x) = \exp\big\{\pi^2(x_1^2 - x_1)/2\big\}\cdot\sin(x_2).$$

During the training, boundary information is provided only on $\Gamma = \{x \in \partial\Omega : x_1 = 0, 1\}$. Figure 1 shows the ground truth γ^* and u^*.

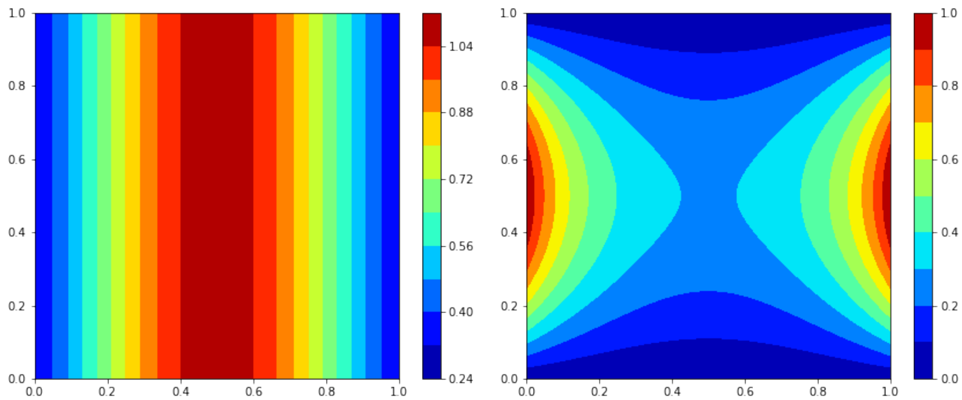


Figure 1: Ground truth of γ and u.

6.2 Network architecture

For all these three methods, we use multilayer perceptron (MLP) as the network. A multilayer perceptron is a fully connected class of feedforward artificial neural network. It consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. In most cases, the number of layers and number of neurons can be really large, which enables MLP to have much larger parameter set than traditional functions in approximation theory. Hence it has stronger representation capacity and learns the target better. The drawback of MLP is that it doesn't have any close form solution for the problem, while tradition class such as the triangle family have exact expression for specific optimization tasks. We apply gradient-based optimization techniques to find out the optimal parameter that can represent or approximate the true solution.

In our experiments, we utilize an MLP with 4 or 6 hidden linear layers and 20 neurons in each layer. We try all possible combinations of number of layers and type of activation functions to get the best result from each method. However, we still want to stress that the results may be improved by changing the number of neurons, learning rate of optimizer or some other hyperparameters.

We use 6 hidden linear layers in WAN. For DRM and PINN we only use 4 hidden linear layers, since 6 hidden layers MLP doesn't produce a better result. It's a common phenomenon in machine learning problems, called the over fitting, which claims that larger parameter set sometime makes the training procedure harder and less stable. As for the activation functions, we consider sin, tanh (hyper tangent), softplus and elu. It turns out that sin and softplus are most suitbale in our problems. More details can be found in our github: https://github.gatech.edu/clee685/CSE-6730-Project

6.3 Experimental Results and Validations

In the following figure, we show the experiment result of different network models. Figure 2 shows the result of WAN. Comparing to the other models, it provides the most accurate prediction of γ^*.

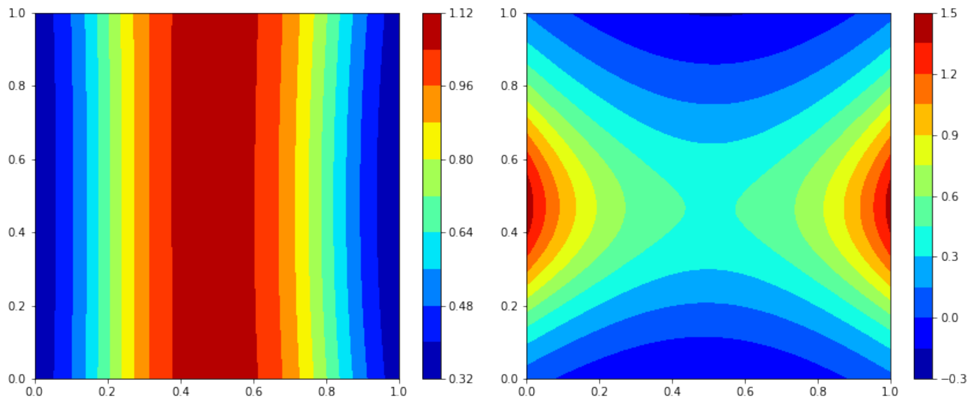


Figure 2: WAN of γ and u.

Figure 3 shows the result of PINN. It provides less accurate model but since its simple network structure, the training time in practice is significantly less than WAN model.

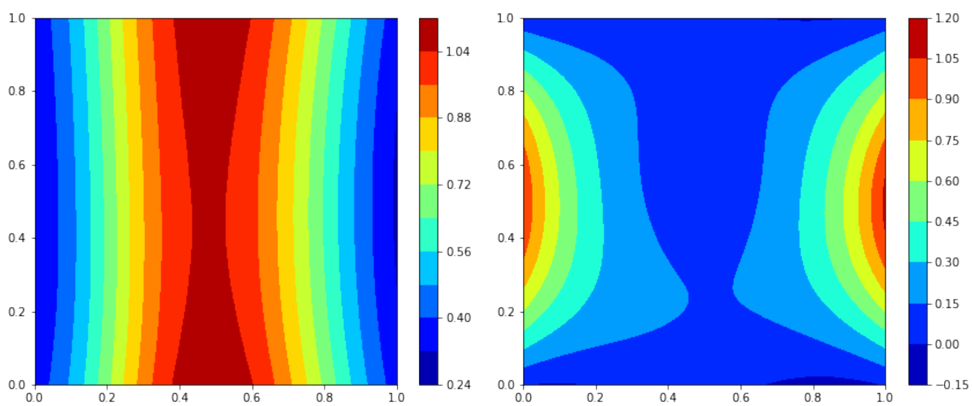


Figure 3: PINN's γ and u.

Figure 4 shows the result of DRM. It has a good prediction of u^*, but γ^* is the worst among three different models, the reason is that the loss function of DRM model is derived from the variational form of the problem, where the energy can be seem as a quadratic minimization problem for u, but on the other hand, it is non-convex with respect to γ.

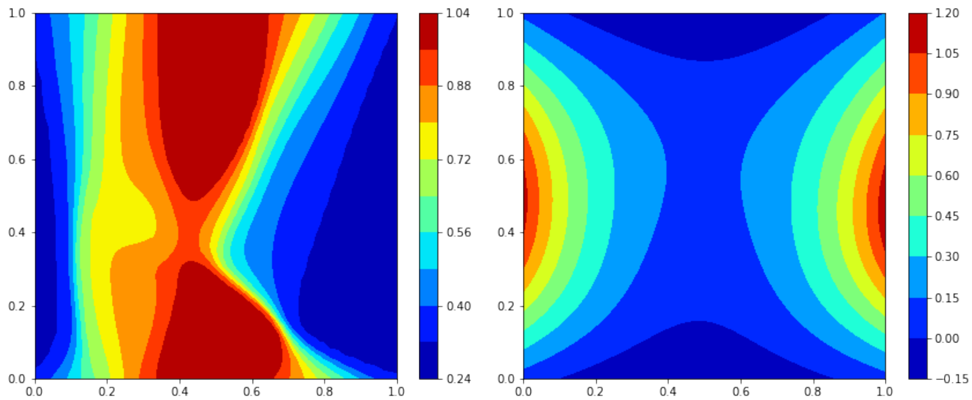
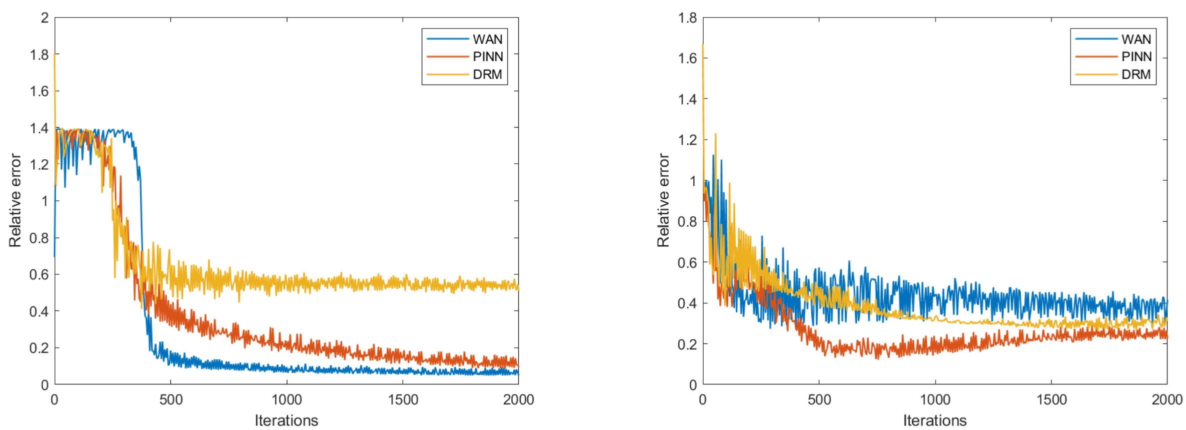


Figure 4: DRM's γ and u.

Since we have the ground truth data u^* and γ^*, the validation can be done by simply checking the L_2 distance of predicted result and the ground truth as relative error. Figure 5 shows the relative error w.r.t. number of iterations.



(a) (b)

Figure 5: (a) Relative error of γ, (b) Relative error of u.

An interesting observation is, with all three models, the relative error of potential u decreases with iteration immediately, but for γ it delayed until the error for u dropped to certain level. The model is unable to find the optimal shape of γ with u being incorrect, but on the other hand, the training of u is independent of the quality of γ.

7 Conclusion

We consider real world problem in this project, and simplify it as an inverse problem. With the modern machine learning tools, we are able to solve the inverse problem much more efficiently. Three machine learning approaches are implemented in our project. For each approach, we analysis its pros and cons, and validate our analysis with experiment results. The results may be improved with stronger networks and fine tuning of hyperparameters, however, we can still conclude that these methods produce better results than traditional ones. And WAN gives the best approximation, its γ, u are very close to the ground truth, despite the intrinsic diffculty of inverse problems.

We may conclude that, machine learning tools are very power in the Numerical PDE and Inverse problem area. In many cases, they are more flexible, powerful and stable compared to traditional methods. With the benefit large networks and efficient training algorithms, machine learning based approaches can be applied to many real world problems and simulations, and help us understand and solve the problems.

References

[1] Gang Bao, Xiaojing Ye, Yaohua Zang, and Haomin Zhou. Numerical solution of inverse problems by weak adversarial networks. *Inverse Problems*, 36(11):115003, 2020.

[2] Margaret Cheney, David Isaacson, and Jonathan C Newell. Electrical impedance tomography. *SIAM review*, 41(1):85–101, 1999.

[3] Talha Ali Khan and Sai Ho Ling. Review on electrical impedance tomography: Artificial intelligence methods and its applications. *Algorithms*, 12(5):88, 2019.

[4] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

[5] Bing Yu et al. The deep ritz method: a deep learning-based numerical algorithm for solving variational problems. *Communications in Mathematics and Statistics*, 6(1):1–12, 2018.

[6] Yaohua Zang, Gang Bao, Xiaojing Ye, and Haomin Zhou. Weak adversarial networks for high-dimensional partial differential equations. *Journal of Computational Physics*, 411:109409, 2020.