LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-ABS-814240

Heterogeneity, Hyperparameters, and GPUs: Towards Useful Transport Calculations Using Neural Networks

M. M. Pozulp, P. S. Brantley, T. S. Palmer, J. L. Vujic

September 3, 2020

The International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2021)
Raleigh, NC, United States
April 11, 2021 through April 15, 2021

ANS M&C 2021 - The International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering · Raleigh, North Carolina · April 11–15, 2021

HETEROGENEITY, HYPERPARAMETERS, AND GPUS: TOWARDS USEFUL TRANSPORT CALCULATIONS USING NEURAL NETWORKS

Michael M. Pozulp[1,2], Patrick S. Brantley[1], Todd S. Palmer[3], and Jasmina L. Vujic[2]
[1]Lawrence Livermore National Laboratory,
[2]University of California, Berkeley, [3]Oregon State University

pozulp1@llnl.gov, brantley1@llnl.gov,
todd.palmer@oregonstate.edu, vujic@nuc.berkeley.edu

ABSTRACT

We use a neural network to solve the discrete ordinates neutron transport equation in slab geometry. We extend previous work which solved homogeneous medium problems by considering problems with spatial heterogeneity. We also include a list of hyperparameters which can be tuned to improve convergence, and we include a description of our GPU implementation along with the runtime we observed using Sierra GPUs at LLNL.

KEYWORDS: transport, neural networks, heterogeneity, hyperparameters, GPUs

1. INTRODUCTION

There has been widespread interest in neural networks (NNs) since at least 2012 when a NN known today as AlexNet made substantial progress in an image classification benchmark called ImageNet. Since then, NNs: surpassed human performance at Go, matched human performance at detecting some eye diseases and cancers, and became the leading technology for anomaly detection, recommender systems, and speech recognition. In the physical sciences, NNs have solved: Burgers' equation, Schrödinger's equation, Laplace's equation, the slab geometry neutron diffusion equation [1], and the slab geometry discrete ordinates (S_N) neutron transport equation [2,3]. NNs have also been used to accelerate components of traditional S_N solvers [4].

In [2,3] NNs solved homogeneous medium problems. We extend [2,3] by considering problems with spatial heterogeneity. We also include a list of hyperparameters which can be tuned to improve convergence, and we include a description of our GPU implementation along with the runtime we observed using a GPU from the Sierra supercomputer at LLNL. Heterogeneity, a hyperparameter study, and GPU results are the new aspects of this work.

2. METHODS

We solve Equation (1), the slab geometry neutron transport equation, in a non-multiplying medium using a single group and linearly-anisotropic scattering. We employ the loss function from [3] reproduced here in Equation (2). See [3] for variable descriptions and derivations.

$$\mu\frac{\partial\psi(z,\mu)}{\partial z}+\Sigma_t(z)\psi(z,\mu)=2\pi\int_{-1}^{1}d\mu'\Sigma_s(z,\mu_0)\psi(z,\mu')+\frac{1}{2}\left[\nu\Sigma_f(z)\phi(z)+Q_{ext}(z)\right] \quad (1)$$

$$\begin{aligned}\mathcal{L}=&\left\|\nabla\hat{\mathbf{\Psi}}\text{diag}(\boldsymbol{\mu})+\Sigma_t\hat{\mathbf{\Psi}}-\frac{1}{2}\left(\Sigma_{s0}\hat{\mathbf{\Phi}}_0\mathbb{1}_N^T-3\Sigma_{s1}\hat{\mathbf{\Phi}}_1\boldsymbol{\mu}^T-\mathbf{Q}\right)\right\|_F^2\\&+\gamma_L\|\hat{\mathbf{\Psi}}^{\mu>0}(z=0)-\mathbf{\Psi}_L\|_F^2\\&+\gamma_R\|\hat{\mathbf{\Psi}}^{\mu<0}(z=z_{max})-\mathbf{\Psi}_R\|_F^2\end{aligned} \quad (2)$$

Our implementation is available on LLNL's Github [5].

3. RESULTS

3.1. SPATIAL HETEROGENEITY

Figure 1 presents flux plots for six problems:

Problem 1. Full Slab Homogeneous purely-absorbing medium with a uniform source and vacuum boundaries.

Problem 2. Half Source Homogeneous absorbing and linearly-anisotropic scattering medium with a source on $[0, 0.5]$ and vacuum boundaries.

Problem 3. Reed Heterogeneous absorbing and scattering medium with multiple sources and vacuum boundaries. Four regions. Identical to Problem 1 from [6] except with a vacuum boundary at 0 instead of a reflecting boundary.

Problem 4. Sharp Slab 0 Heterogeneous strongly-absorbing medium with a source on $[0, 1]$ adjacent to a void on $[1, 5]$ and vacuum boundaries. Two regions.

Problem 5. Sharp Slab 1 Same as above but with a weak absorber ($\Sigma_a = 1$) instead of a void. The ratio of the absorption cross sections of the two regions is 50.

Problem 6. Sharp Slab 5 Same as above but with $\Sigma_a = 5$ and absorption cross section ratio 10.

In Figure 1, the flux calculated using the NN very closely matches MC for **Problems 1-2** which are homogeneous medium problems. This is not the case for **Problems 3-6**. **Problems 4-6** demonstrate that the NN has trouble with sharp gradients. The spatial gradient, which is a component of the neutron streaming term, is the first term in Equation (1). In the NN we calculate this term using automatic differentiation (AD) in `PyTorch` [7]. We were surprised by AD's sensitivity to sharp gradients and are working to understand it. We also want to try more traditional numerical methods for calculating the gradient, like finite differences.

3.2. HYPERPARAMETER TUNING

In deep learning parlance, the weights and biases are the parameters of the NN. Any quantity that is not a weight or bias is a hyperparameter. We believe that the following hyperparameters could effect convergence and iteration runtime, where convergence is defined as the number of iterations required to achieve the convergence criterion. We are actively varying these and will present

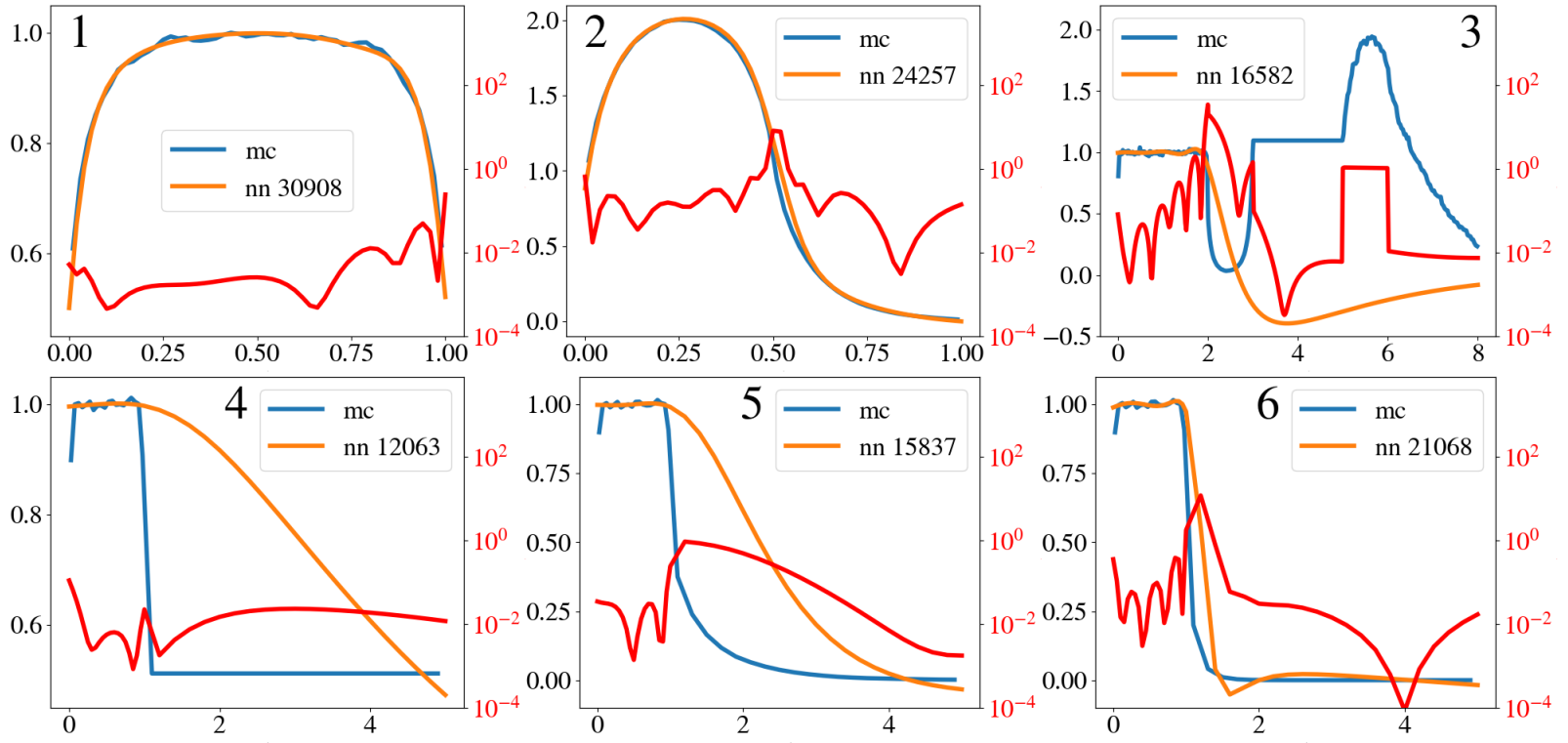

Figure 1: Scalar flux plots for six different problems. The horizontal axis is the spatial coordinate z and the left vertical axis is the scalar flux $\phi(z)$. The curves are labeled with the algorithm used to find the solution, where mc is Monte Carlo and nn is neural network. The integer in the nn label is the number of iterations required to achieve convergence. The red curve, which corresponds to the right vertical axis, is the spatial distribution of the loss, which has the same units as $\phi(z)$. The loss is calculated using Equation (2). Problems 1-6 are Full Slab, Half Source, Reed, Sharp Slab 0, Sharp Slab 1, and Sharp Slab 5.

results in the full paper: number of nodes in the hidden layer, number of hidden layers, learning rate, activation function, optimizer settings, choice of optimizer, regularization, initialization, and scaling inputs.

Notable omissions include mini-batch size, early stopping, and dropout. These hyperparameters may not be relevant to our NN. Consider a NN for image recognition, which might use one billion images that are partitioned into training and test sets. Mini-batch size could affect iteration runtime. Early stopping and dropout could improve performance on the test set by preventing overfitting. Our data is tiny; instead of a billion images we have one input vector. Thus, our mini-batch size is fixed at one and overfitting is not a problem because the training and test sets are identical.

3.3. GPU EXECUTION

We used the `device` argument to `torch.tensor()` along with `torch.tensor.to()` to move tensors into GPU-accessible memory and `torch.nn.Sequential.cuda()` to move the model parameters and buffers into GPU-accessible memory. We ran on one node of LLNL's

RZAnsel cluster [8], a small system with the same hardware as LLNL's Sierra supercomputer [9]. The node has two IBM POWER9™ sockets each with 22 cores, and four NVIDIA Tesla™ V100-SXM2-16GB Volta™ GPUs. We ran **Problem 1** to convergence at 29599 iterations and 5 minutes and 44 seconds of walltime. The flux calculated using the V100 closely matches the flux calculated using one P9 core, but the latter only took 2 minutes and 3 seconds to run. We were surprised that the V100 is slower than one P9 core and we are trying to understand why.

4. CONCLUSIONS

We presented the solution to six different slab geometry neutron transport problems using NNs. We also listed hyperparameters worth examining and we presented GPU results using a GPU from the Sierra supercomputer at LLNL. We were surprised by the challenge that sharp spatial gradients present to AD in `PyTorch`. We were also surprised to see a 2x slowdown going from one P9 core to the V100. We expect more surprises when we do the hyperparameter study. Resolving the gradient issue and GPU performance issue and undertaking the hyperparameter study are good directions for future work. If we add multigroup we could solve a two-material, two-group reactor problem in slab geometry. Would we then finally have a useful transport calculation using NNs?

ACKNOWLEDGEMENTS

The authors thank the LLNL WSC Research Coordination Council for funding this work through the LEARN program. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

REFERENCES

[1] P. S. Brantley. "Artificial Neural Network Solutions of Slab-Geometry Neutron Diffusion Problems." *Trans Am Nuc Soc*, (83), p. 251 (2000).

[2] P. S. Brantley. "Spatial Treatment of the Slab-Geometry Discrete Ordinates Equations Using Artificial Neural Networks." In *Proceedings of M&C01*. Salt Lake City, UT, USA (2001).

[3] M. M. Pozulp. "1D Transport Using Neural Nets, SN, and MC." In *Proceedings of M&C19*, pp. 876–885. Portland, OR, USA (2019).

[4] M. Tano and J. Ragusa. "Acceleration of Radiation Transport Solves Using Artificial Neural Networks." (2019).

[5] "Narrows." (2020). URL https://github.com/llnl/narrows. LLNL-CODE-806068.

[6] W. H. Reed. "New Difference Schemes for the Neutron Transport Equation." *Nuclear Science and Engineering*, **volume 46**(2), pp. 309–314 (1971).

[7] A. Paszke et al. "Automatic Differentiation in PyTorch." In *Proceedings of 31st NeurIPS*. Curran Associates, Inc., Long Beach, CA, USA (2017).

[8] "LLNL RZAnsel Cluster." (2020). URL https://hpc.llnl.gov/hardware/platforms/rzansel.

[9] S. S. Vazhkudai et al. "The Design, Deployment, and Evaluation of the CORAL Pre-exascale Systems." In *Proceedings of SC18*, volume 52, pp. 1–12. Dallas, TX, USA (2018).