PlasmaNet: a framework to study and solve elliptic differential equations using neural networks in plasma fluid simulations

Lionel Cheng
CERFACS
cheng@cerfacs.fr

Ekhi Ajuria Illarramendi
ISAE-SUPAERO /CERFACS
ekhi.ajuria@cerfacs.fr

Michaël Bauerheim
ISAE-SUPAERO
bauerheim@isae-supaero.fr

Bénédicte Cuenot
CERFACS
cuenot@cerfacs.fr

1 Introduction

Elliptic partial differential equations (PDEs) are common in many areas of physics, from the Poisson equation in plasmas and incompressible flows to the Helmholtz equation in electromagnetism. Their numerical solution requires to solve a linear system and many libraries have been developed for this task [1]. Solving a linear system efficiently requires preconditioning the system which is a difficult task. It can become a bottleneck for performance when the number of nodes increases.

The rise of computational power and inherent speed of GPUs offers exciting opportunities to solve PDEs by recasting them in terms of optimization problems. Since the major introduction of *Physics informed neural networks* (PINN) [2], other architectures [3] and frameworks [4] have been created.

In plasma fluid simulations, the Poisson equation is solved, coupled to the charged species transport equations [5]. A pioneer work [6] has shown significant speedup using neural networks to solve the Poisson equation compared to classical linear system solvers on this problem. Coupling the neural network Poisson solver to plasma transport equations has shown promising results and the neural network can be considered as a viable option in terms of accuracy.

This work extends [6] and introduces *PlasmaNet* (`https://gitlab.com/cerfacs/plasmanet`), an open-source library written to study neural networks in plasma simulations. The optimal network parameters are first recalled and metrics to help design appropriate network architectures for solving elliptic differential equations are presented. We then attempt to solve a new class of elliptic differential equations, the screened Poisson equations using neural networks. These equations are used to infer the photoionization source term from the ionization rate in streamer discharges [7]. Finally a simulation running with three neural networks, coupled to plasma transport equations, to solve both the Poisson and the photoionization equations, is performed to assess the accuracy of neural networks predictions.

2 Network architectures for elliptic differential equations

Laplace and Poisson equations form the basis of elliptic PDEs. Studying them can give insights on how to solve all elliptic PDEs. From a given charge density ρ_q the Poisson equation yields the electromagnetic potential ϕ from which we can compute the electric field $\mathbf{E} = -\nabla\phi$ so that in the

Fourth Workshop on Machine Learning and the Physical Sciences (NeurIPS 2021).

end $\mathbf{E} = f(\rho_q)$. In numerical simulations the physical domain is finite so that the Poisson equation is supplemented by Dirichlet and Neumann boundary conditions:

$$\nabla^2 \phi = -\frac{\rho_q}{\varepsilon_0} \quad \text{in} \quad \mathring{\Omega} \quad \text{with} \quad \begin{cases} \phi = 0 & \text{on} \quad \partial\Omega_D \\ \mathbf{E} \cdot \mathbf{n} = E_N & \text{on} \quad \partial\Omega_N \end{cases} \tag{1}$$

The analytical solution of the Poisson equation with boundary conditions depends on the Green function G of the chosen configuration [8, Chap. 1.10]:

$$\phi(\mathbf{x}) = \frac{1}{4\pi\epsilon_0} \int \rho(\mathbf{x}')G(\mathbf{x}, \mathbf{x}') \,\mathrm{d}V' + \frac{1}{4\pi} \int \left(G \frac{\partial \phi}{\partial n'} - \phi \frac{\partial G}{\partial n'} \right) \mathrm{d}S' \tag{2}$$

Expressions of the Green functions [8, Chap. 3] in series for cartesian, cylindrical and spherical coordinates highlight the importance of multiple scales in the Poisson equation which must be incorporated in the neural network architecture. The UNet architecture [9] has proved to work best to infer $\mathbf{E} = \mathrm{NN}(\rho_q)$ in [6] compared to the Multi-Scale architecture [10]. In UNets, the local branch b is defined as the power of 2 by which the initial resolution is divided by and the number of branches n_b is the number of scales involved. Local d_b and global D depths are defined as the number of successive convolutional layers in branch b and across the whole network, respectively.

Concerning losses, the use of a physical loss in the form of a `LaplacianLoss` inspired from PINNs [2] is critical to yield stable trainings and stable simulations when coupled to transport equations [6].

Elliptic PDEs, having no real characteristic curves, need the information of the whole domain at every point. This is highlighted by the analytical solution Eq. (2) which incorporates a domain integral. To quantify the information propagation across the neural network the global receptive field RF is defined as the size of the domain of influence of the input center point in number of points in the original scale n_p. The receptive field can be splitted into local receptive fields per branch RF_b so that

$$\mathrm{RF} = \sum_{b=0}^{n_b - 1} \mathrm{RF}_b \quad \text{with} \quad \mathrm{RF}_b = \begin{cases} 1 + d_b(k_s - 1)2^b & \text{if } b = 0 \\ d_b(k_s - 1)2^b & \text{otherwise} \end{cases} \tag{3}$$

where k_s is the kernel size assumed constant in the network.

In [11], a *theoretical receptive field* is defined as the size of the input domain of influence on the output center point. Tests carried in *PlasmaNet* on the studied UNets show that this definition matches Eq. (3) so that both formulations are equivalent. However, the importance of the domain influence is not uniform, as points closer to the studied pixel will have more paths to influence the output, resulting in a *gaussian-like* distribution [12]. A parametric study in [6, Sec. 5] across multiple UNets showed that the optimal global parameters of the network for a given number of pixels n_p should be:

$$\mathrm{RF} = 2n_p \quad \text{and} \quad n_b = \max\{b \in \mathbb{N} | \lfloor n_p/2^b \rfloor > k_s\} + 1 \tag{4}$$

With these parameters the receptive field fills the entire computational domain for all the input points and convolution is relevant in the downscaled branches [6, Sec. 5].

3 Photoionization in plasma discharges

We model plasma discharges in air using the chemistry from [13]. It consists of electrons (n_e), positive ions (n_p) and negative ions (n_n). Those three species are modeled in a drift-diffusion approximation where the ions are considered not moving [7]:

$$\frac{\partial n_e}{\partial t} + \nabla \cdot (n_e \mathbf{W_e} - D_e \nabla n_e) = n_e \alpha |W_e| - n_e \eta |W_e| - n_e n_p \beta + S_{ph} \tag{5}$$

$$\frac{\partial n_p}{\partial t} = n_e \alpha |W_e| - n_e n_p \beta - n_n n_p \beta + S_{ph} \quad \frac{\partial n_n}{\partial t} = n_e \eta |W_e| - n_n n_p \beta \tag{6}$$

where S_{ph} is the photoionization source term, $\alpha = \alpha(E/N)$ is the ionization coefficient, $\eta = \eta(E/N)$ the attachment coefficient, E the electric field magnitude, N the neutral gas density, β the recombination rate, $\mathbf{W}_e = -\mu_e \mathbf{E}$ the drift-velocity of the electrons and $\mu_e = \mu_e(E/N)$ the electron mobility. These transport equations are coupled to the Poisson equation as $\rho_q = e(n_p - n_e - n_n)$. Such a coupling with the Poisson neural network solver has already been studied in [6]. However a rather high background density of charged species was used so that the photoionization source term S_{ph} could be neglected in previous work. Here, similar cases where photoionization cannot be neglected are considered, highlighting the benefits of the proposed open-source framework *PlasmaNet* to incorporate new physics for CNN-based fluid plasma simulations.

The plasma discharge is a double headed streamer [7], initialized with a neutral Gaussian profile at $x = 2$ mm and $r = 0$ mm with a background density in a rectangular domain of $L_x \times L_r = 4 \times 1$ mm^2, corresponding to an azimuthal cut of the cylindrical geometry.

In plasma air discharges, the electron-impact reactions produce excited states of N_2. Radiative relaxations of these states are absorbed by O_2 causing ionization [14]. Integral models have been developed to model this photoionization source term but are very costly in numerical simulations [14]. Simplifiying assumptions allow to recast this photoionization source term in terms of screened Poisson equations [7]:

$$S_{ph}(\mathbf{r}) = \sum_{j=1}^{j_m} S_{ph}^j(\mathbf{r}) \quad \forall j \in [\![1, j_m]\!] \; \nabla^2 S_{ph}^j(\mathbf{r}) - (\lambda_j p_{\mathrm{O2}})^2 S_{ph}^j(\mathbf{r}) = -A_j p_{\mathrm{O2}}^2 I(\mathbf{r}) \tag{7}$$

where $I(\mathbf{r}) = f(E/p) n_e \alpha |W_e|$ is an effective ionization term, p_{O2} the oxygen pressure, λ_j, A_j are fitting parameters and $j_m = 2$ or 3 depending on the level of precision wanted. Note that j_m resolutions of linear systems must be performed inside a numerical time iteration when solving S_{ph} so that the computational cost of photoionization is high. Here only $j_m = 2$ is considered for simplicity. Each component of the photoionoization source term obeys a screened Poisson equation:

$$\nabla^2 \phi - \lambda^2 \phi = -R \tag{8}$$

where λ controls the amount of diffusion of the solution, the higher the value of λ, the lower the diffusion ($\lambda = 0$ is the maximum diffusion and reduces to the Poisson equation). We apply a neural network to solve each component of the photoionization source term using the optimal parameters in Eq. (4). The same network architecure as the Poisson neural network solver (UNet5-RF_x800-RF_y200) is thus applied for each component of the photoionization source term. We introduce a physical loss associated to the screened Poisson equation called `PhotoLoss`:

$$\mathcal{L}_P(\phi_{\mathrm{out}}; \lambda) = \frac{L_x^2 L_y^2}{b_s(n_x - 1)(n_y - 1)} \sum_{b,j,i} \left[\nabla^2 \phi_{\mathrm{out}}^{b,j,i} - \lambda^2 \phi_{\mathrm{out}}^{b,j,i} + R_{\mathrm{in}}^{b,j,i} \right]^2 \tag{9}$$

Training is done using random datasets of 10 000 snapshots introduced in [15] and already used in [6, Sec. 4]. A snapshot of such a dataset is shown in Fig. 1. It can be seen that S_{ph}^2 is less diffusive than S_{ph}^1 which behaves closer to the Poisson equation due to $\lambda_1 \ll \lambda_2$.

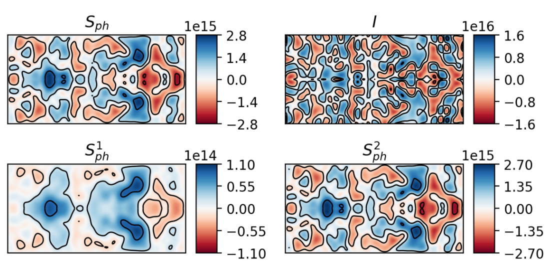

Figure 1: Example of `random_12` source term input in a 4×1 mm^2 cylindrical domain.

4 Results

First, snapshots of the propagation of the double headed streamer without photoionization using a linear system solver and a neural network solver for the Poisson equation are shown in Fig 2. Electron density and electric field profiles tend to be more diffusive when the Poisson equation is solved by the neural network. However the speed of propagation is well-captured as well as the discharge energy [6, Sec. 7].

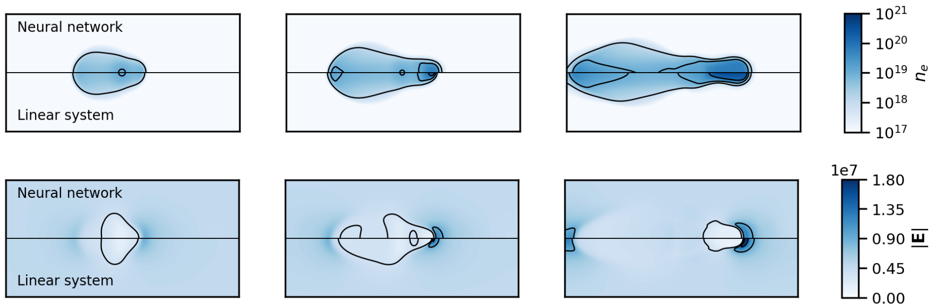

Figure 2: Electron density n_e and norm of electric field $|\mathbf{E}|$ of a double headed streamer using UNet5-RFx800-RFy200 (top-half) and a linear system solver (bottom-half) at 1.2, 2.0 and 2.8 ns.

$$\begin{array}{l} \mathbf{E} = \mathrm{NN}_1(\rho_q) \\ S_{ph} = \mathrm{NN}_2(I) + \mathrm{NN}_3(I) \end{array} \longrightarrow \mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t[\mathbf{R}_t(\mathbf{E}) + \mathbf{R}_c(\mathbf{E}, S_{ph})]$$

Figure 3: Interaction of the neural network with plasma transport equations at each iteration. $\mathbf{U} = (n_e, n_p, n_n)$, $\mathbf{R}_t$ and $\mathbf{R}_c$ are transport and chemistry residuals.

Second, simulations with photoionization have been performed with two networks inferring S^1_{ph} and S^2_{ph}. Another network is used to infer the electric field $\mathbf{E}$ (as in [6] and results from Fig. 2) so that three networks are coupled to the plasma transport equations to replace linear system solvers. Results are shown in Fig. 4 with a sketch of coupling in Fig. 3. The interaction of multiple neural networks solutions seem to yield promising results as the photoionization source term and electric field are correctly predicted. The right-propagating positive streamer ia slightly more diffuse when looking at the electron density than the reference solution whereas the left-propagating negative streamer is better captured by the neural networks. We note however negative values in the prediction of S_{ph} by the neural network (white regions in the snapshots due to clipping), which strictly speaking should not be allowed as the ionization source term is positive. These values have a relative amplitude of 10^{-3} and have been clipped to get good streamer propagation.

To prevent the rise of these negative values, a positive-valued dataset can be used so that the network learns only to infer positive-values. A similar dataset as the one shown in Fig. 1 but without negative values for the ionization rate I is used to train the networks and results are shown in Fig. 5. The negative values are indeed removed but the overall shape of the electron density is more diffused especially for the right-propagating positive streamer: the network struggles to predict very low values of S_{ph}.

5 Conclusion

We have introduced *PlasmaNet* and shown its ability to couple neural network solvers to plasma transport equations. The range of applicability of the method developed in [6] has been extended to the more general screened Poisson equations, which highlights the flexibility of the present framework *PlasmaNet* to incorporate new complex physics simulated by neural networks. Future work will try to integrate the screening length λ directly inside the network so that one network and not three will be necessary to solve both the Poisson equation and the photoionization source term. Dedicated regularized terms, such as penalty on negative S_{ph} could also be tested. Interaction of the neural network with other plasma test cases such as Hall effect thrusters are also planned.

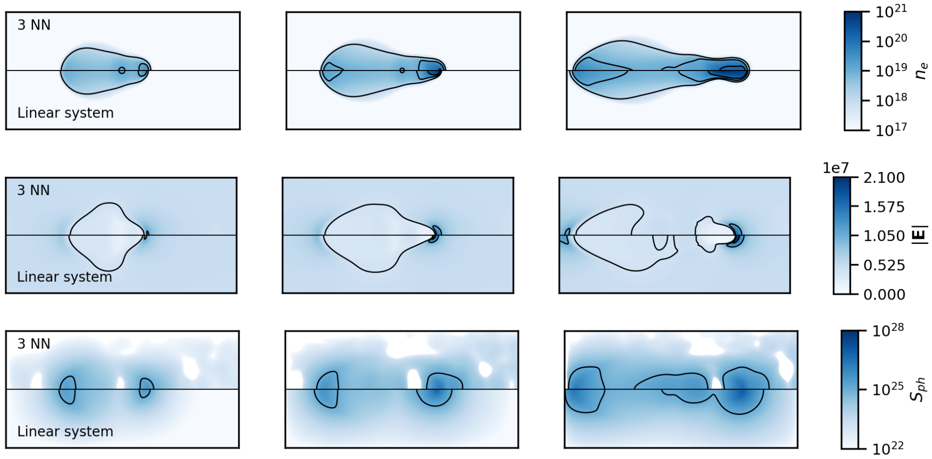

Figure 4: Electron density n_e, norm of electric field $|\mathbf{E}|$ an photoionization source term S_{ph} of a double headed streamer using three neural networks (top-half) and a linear system solver (bottom-half) at 1.6, 2.2 and 2.8 ns.

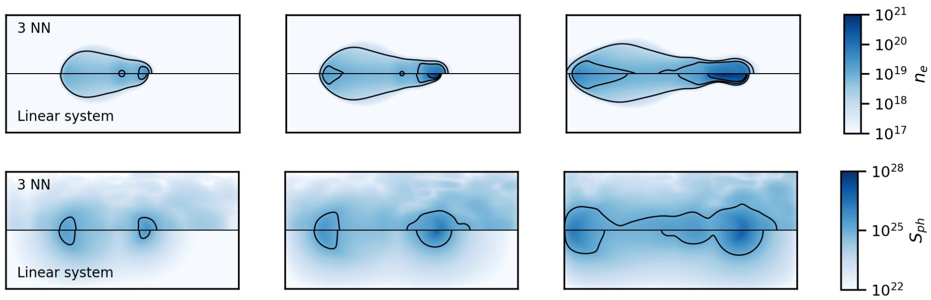

Figure 5: Electron density n_e and photoionization source term S_{ph} of a double headed streamer using three neural networks (top-half) and a linear system solver (bottom-half) at 1.6, 2.2 and 2.8 ns using a positive-valued training dataset.

References

[1] Satish Balay, Shrirang Abhyankar, Mark F. Adams, Jed Brown, Peter Brune, Kris Buschelman, Lisandro Dalcin, Alp Dener, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Dave A. May, Lois Curfman McInnes, Richard Tran Mills, Todd Munson, Karl Rupp, Patrick Sanan, Barry F. Smith, Stefano Zampini, Hong Zhang, and Hong Zhang. PETSc Web page, 2021. https://petsc.org/.

[2] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. *arXiv preprint arXiv:1711.10561*, 2017.

[3] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

[4] Philipp Holl, Vladlen Koltun, Kiwon Um, and Nils Thuerey. phiflow: A differentiable pde solving framework for deep learning via physical simulations. *DiffCVGP workshop at NeurIPS 2020*, 2020.

[5] J.A. Bittencourt. *Fundamentals of Plasma Physics*. Springer, third edition, 2004.

[6] Lionel Cheng, Ekhi Ajuria Illarramendi, Guillaume Bogopolsky, Michael Bauerheim, and Benedicte Cuenot. Using neural networks to solve the 2d poisson equation for electric field computation in plasma fluid simulations. *arXiv preprint arXiv:2109.13076*, 2021.

[7] Sébastien Célestin. *Study of the dynamics of streamers in air at atmospheric pressure*. PhD thesis, 2008. Thèse de doctorat dirigée par Bourdon, Anne et Rousseau, Antoine Physique Châtenay-Malabry, Ecole centrale de Paris 2008.

[8] John David Jackson. *Classical Electrodynamics*. John Wiley & Sons, 1999.

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[10] Michael Mathieu, Camille Couprie, and Yann Lecun. Deep multi-scale video prediction beyond mean square error. *Proceedings of the ICLR*, 2016.

[11] Leon O Chua and Tamas Roska. The cnn paradigm. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 40(3):147–156, 1993.

[12] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 4905–4913, 2016.

[13] R Morrow and J J Lowke. Streamer propagation in air. *Journal of Physics D: Applied Physics*, 30(4):614–627, February 1997.

[14] N. Liu. *Dynamics of positive and negative streamers in sprites*. PhD thesis, The Pennsylvania State University, 2006.

[15] Ali Girayhan Özbay, Arash Hamzehloo, Sylvain Laizet, Panagiotis Tzirakis, Georgios Rizos, and Björn Schuller. Poisson cnn: Convolutional neural networks for the solution of the poisson equation on a cartesian mesh. *Data-Centric Engineering*, 2, 2021.