

Chapter 1

The Data-Driven ANN-MoC Method to Neutral Particle Transport Problems in 1D

P.H.A. Konzen, N.G. Roman and A. Tchantchalam

1.1 Introduction

Neutral particle transport mainly appears in the radiative heat [HoEtAl21, MoMa23] and the neutron transport [Ku23, Le84] phenomena. The modelling of the first is important in engineering at high temperatures (e.g. glass and ceramic manufactures) [LaEtAl02], combustion chambers [FrEtAl04], solar energy production [FuEtAl17], and many others. The neutron transport modelling is of main importance in nuclear energy production [St07]. Both are related to important applications in optical medicine [Ba09, WaWu07].

We consider the fundamental model of neutral particle transport problems given by the linear Boltzmann equation in 1D geometry

$$\forall \mu : \mu \cdot \frac{\partial}{\partial x} I(x, \mu) + \sigma_t I = \sigma_s \Psi(x) + q(x, \mu), \quad (1.1)$$

where, from the radiative transport terminology, $I = I(x, \mu)$ denotes the radiant intensity at point $x \in D = [a, b]$ and in the direction $\mu \in (-1, 1) \setminus \{0\}$, $q(x, \mu)$ the source term, $\sigma_t, \sigma_s > 0$ are the extinction ($\kappa = \sigma_t - \sigma_s$ the absorption) and the scattering coefficients, respectively. The radiative density is

P.H.A. Konzen
Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, Brazil,
e-mail: pedro.konzen@ufrgs.br

N.G. Roman
Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, Brazil,
e-mail: ngroman1992@gmail.com

A. Tchantchalam
Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, Brazil,
e-mail: augustotchantchalam11@gmail.com

$$\Psi(x) = \frac{1}{2} \int_{-1}^1 I(x, \mu') d\mu'. \quad (1.2)$$

The integro-differential equation (1.1) is assumed to be closed with the following boundary conditions

$$\forall \mu > 0 : I(a, \mu) = I_a, \quad (1.3)$$

$$\forall \mu < 0 : I(b, \mu) = I_b. \quad (1.4)$$

Computational methods for solving direct and inverse transport problems with (1.1)-(1.4) can be found in the textbooks (see, for instance, [HoEtAl21, Ku23, Le84, MoMa23, OzOr21]). More recently, it has been grown attention to the application of deep learning techniques [GoEtAl16] to solve partial differential equations (PDEs). The physics-informed neural networks [RaEtAl19] are one of the notable options, which trains an artificial neural network (ANN) to estimate the given PDE solution by minimizing the equation residual. An application of PINNs to solve transport problems is found in the paper [MiMo21].

The use of deep learning for solving inverse transport problems dates back decades. For parameter estimation problems, the first idea is to train an ANN as a regression model for a given data set. The data set must be big enough to train the network, and it can be generated by solving the direct transport problem for several choices of the goal parameters. We have recently explored this approach in the work [RoEtAl24]. The main disadvantage of this approach is the need to solve the direct transport problem as many times as the size of the required data to train the network. Alternatively, the PINN approach requires less data by training the ANN based on the transport model and given data. However, this involves the training of an ANN to fit the solution of the transport problem for the radiant intensities at all points and directions. Commonly found in the transport solutions, discontinuities in the parameters and in the phase space are difficult to learn by ANNs.

The main goal here is to introduce a viable alternative to the application of ANN to solve direct and inverse transport problems modeled by equations (1.1)-(1.4). The following described ANN-MoC method is designed as a data-driven integration method to the transport equation. Firstly without data, the method was previously presented in the work [Ko23]. Here, it is adapted to solve direct and inverse data-driven neutral particle transport problems.

1.2 Data-Driven ANN-MoC Method

1.2.1 ANN-MoC Model Aware

The ANN-MoC method estimates the radiative density Ψ by a multilayer perceptron (MLP) neural network

$$\tilde{\Psi}(x) = \mathcal{N}(x; \mathbf{p}),$$

where $\mathbf{p} \in \mathbb{R}^{n_p}$ are the n_p parameters $\mathbf{p}_w$ (weights and biases) of the network. An MLP with given activation functions and architecture $1 - n_n \times n_h - 1$ (1 input, n_n neuron per each n_h hidden layers, 1 output) has $n_p = 4n_n + n_n(1 + n_n)(n_h - 1)$ parameters. See Fig. 1.1.

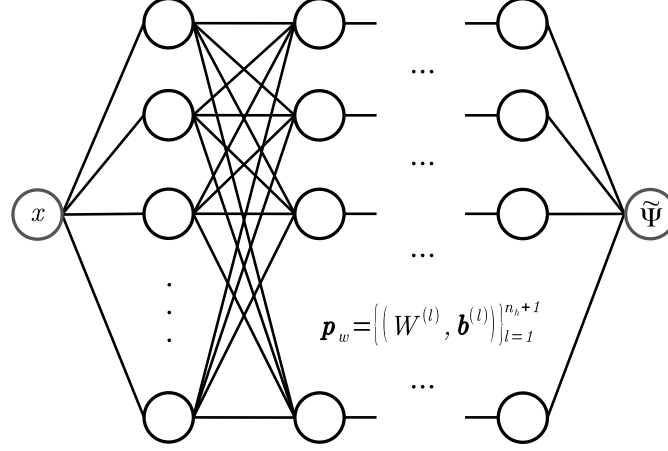


Fig. 1.1 The MLP network $\mathcal{N}$ with architecture $1 - n_n \times n_h - 1$, input x and output the estimated radiative density $\tilde{\Psi}(x)$.

The training of the network $\mathcal{N}$ has the goal to fit the parameters $\mathbf{p}_w$ that minimize the error of the estimated $\tilde{\Psi}$ in relation to the expected $\hat{\Psi}$ solution of the transport problem (1.1)-(1.4). Since the latter is unknown, we calibrate the network by the following gradient-like iteration

$$\mathbf{p}^{(l+1)} = \mathbf{p}^{(l)} + l_r \cdot \mathbf{d} \left(\nabla_{\mathbf{p}} \varepsilon_{\text{M}}^{(l)} \left(\mathbf{x}^{(l)}; \mathbf{p}^{(l)} \right) \right), \quad (1.5)$$

where $\mathbf{p}^{(l)}$ denotes the network parameters at the l -th iterate, $l = 0, 1, \dots, n_{\text{si}}$, n_{si} a given maximum number of iterations. Given a learning rate $l_r > 0$, the direction update $\mathbf{d} \in \mathbb{R}^{n_p}$ is a function of the gradient of the loss function

$$\varepsilon_{\text{M}}^{(l)}(\mathbf{x}; \mathbf{p}) = \frac{1}{n_s} \sum_{s=1}^{n_s} \left| \tilde{\Psi}^{(l)}(x_s) - \tilde{\Psi}(x_s) \right|^2, \quad (1.6)$$

where $\tilde{\Psi}^{(l)}(x_s)$ is an approximation of $\hat{\Psi}(x_s)$ at the domain point $x_s \in D$. To prevent over-fitting, $\mathbf{x}^{(l)} \in \mathbb{R}^{n_s}$ is randomly generated by a uniform distribution.

Given the Gauss-Legendre quadrature $\{(\omega_j, \mu_j)\}_{j=1}^{n_q}$ with n_q pairs, the $\tilde{\Psi}^{(l)}$ approximation of the density (1.2) is computed as

$$\tilde{\Psi}^{(l)}(x) = \frac{1}{2} \sum_{j=1}^{n_q} I_j^{(l)}(x) \omega_j,$$

where $I_j^{(l)}$ is the radiant intensity in the direction μ_j that satisfies

$$\mu_j \cdot \frac{\partial}{\partial x} I_j^{(l)}(x) + \sigma_t I_j^{(l)} = \sigma_s \mathcal{N}(x; \mathbf{p}^{(l-1)}) + q(x, \mu_j), \quad (1.7)$$

and the associated boundary condition. We note that (1.7) is directly derived from the transport equation (1.1) by substituting $\Psi(x)$ by $\mathcal{N}(x)$.

The solution of (1.7) is computed by an application of the method of characteristics (MoC). Assuming the change of variables $x(u) = x_0 + u \cdot \mu_j$, $u \in \mathbb{R}$, (1.7) is equivalent of

$$\frac{d}{du} I_j^{(l)} + \sigma_t I_j^{(l)}(u) = \sigma_s \mathcal{N}(u; \mathbf{p}^{(l)}) + q(u),$$

where we assumed the notation $I_j^{(l)}(u) = I_j^{(l)}(x(u))$ and analogous for the other terms. This ordinary differential equation has the solution

$$I_j^{(l)}(u) = I_j^{(l)}(0) e^{-\sigma_t u} + \int_0^u [\sigma_s \mathcal{N}(u') + q(u')] e^{-\sigma_t(u-u')} du',$$

where x_0 must be chosen so $I_j^{(l)}(0)$ is known (for instance, at the entrance boundary point).

We note that the above procedure is an adapted source iteration, which is expected to converge to $\hat{\Psi}$ for $\sigma_s/\sigma_t \leq 1$ (see, for instance, [AdLa02]). One advantage of the ANN-MoC method is that it provides a neural network $\mathcal{N}(x)$ that estimates the radiative density $\Psi(x)$ at any point x in the computational domain. It does not require any assumptions on interpolating discrete values of Ψ . The ANN-MoC method presented this way is model-only aware, i.e. it trains $\mathcal{N}$ based just on the transport model (1.1)-(1.4). However, $\mathcal{N}(x)$ can be forced to fulfill extra constraints by adding them to the loss function (1.6). For instance, for direct and inverse data-driven problems, the solution can be forced to fit the data. These are the applications that we are exploring in the following.

1.2.2 ANN-MoC Data and Model Aware

The ANN-MoC can be easily adapted to data-driven problems. Let's assume the given n_d data $\left\{ \left(x_d^{(s)}, \Psi_d^{(s)} \right) \right\}_{s=1}^{n_d}$. The data-driven problem is to estimate Ψ such that it satisfies the transport model (1.1)-(1.4) and the given data. This can be computed by the ANN-MoC method with a network $\tilde{\Psi}(x) = \mathcal{N}(x; \mathbf{p} = \mathbf{p}_w)$ trained to minimize the loss function

$$\mathcal{E}_{\text{DM}}^{(l)}(\mathbf{x}; \mathbf{p}_w) = \frac{w_1}{n_s} \sum_{s=1}^{n_s} \left| \bar{\Psi}^{(l)}(x_s) - \tilde{\Psi}(x_s) \right|^2 + \frac{w_2}{n_d} \sum_{s=1}^{n_d} \left| \Psi_d^{(s)} - \tilde{\Psi}(x_d) \right|^2,$$

where $w_1, w_2 > 0$ are penalty scalars that control the weight of each term. With this loss function, the network training is based on the transport model, and the given data, i.e. $\mathcal{N}$ is model and data-aware.

1.2.3 Data-Driven ANN-MoC for Inverse Problems

The adaptation of the ANN-MoC method to inverse transport problems is also feasible. For instance, let us consider the problem of estimating the extinction coefficient σ_t , given the n_d data $\left\{ \left(x_d^{(s)}, \Psi_d^{(s)} \right) \right\}_{s=1}^{n_d}$. Here, the idea is to add radiant intensity information on selected directions to the loss function, to add an explicit relation between the data and the unknown parameter σ_t . Therefore, let us assume the n_μ directions $\{\mu_{j_k}\}_{k=1}^{n_\mu}$ have been selected, $(j_k)_{k=1}^{n_\mu} \subset (1, \dots, n_q)$. The neural network is then chosen as

$$\tilde{\mathbf{y}}(x) = \mathcal{N}_{\text{IP}}(x; \mathbf{p} = (\mathbf{p}_w, \sigma_t)),$$

where the output is $\tilde{\mathbf{y}}(x) = (\tilde{\Psi}(x), \tilde{I}_{j_1}(x), \dots, \tilde{I}_{j_{n_\mu}}(x))$, i.e. the estimations of the radiative density and the radiant intensities in the selected directions. See Fig. 1.2.

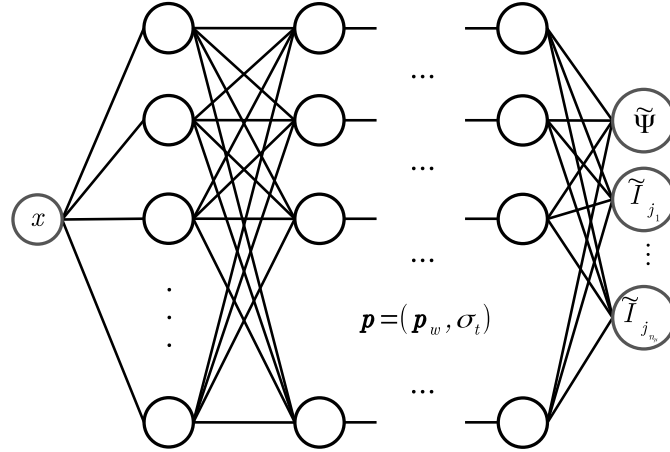


Fig. 1.2 The MLP network $\mathcal{N}_{\text{IP}}$ with architecture $1 - n_n \times n_h - 1 + n_\mu$, input x and output the estimated radiative density and radiant intensities in selected directions.

The network $\mathcal{N}_{\text{IP}}$ is then trained to minimize the following loss function

$$\begin{aligned}
\varepsilon_{\text{IP}}^{(l)}(\mathbf{x}; \mathbf{p}) = & \frac{w_1}{n_s} \sum_{s=1}^{n_s} \left| \tilde{\Psi}^{(l)}(x_s) - \tilde{\Psi}(x_s) \right|^2 + \frac{w_2}{n_d} \sum_{d=1}^{n_d} \left| \Psi_d^{(s)} - \tilde{\Psi}(x_d) \right|^2 \\
& + \frac{w_3}{n_s n_\mu} \sum_{k=1}^{n_\mu} \sum_{s=1}^{n_s} \left| \mu_{jk} \frac{\partial \tilde{I}_{jk}}{\partial x}(x_s) + \sigma_t \tilde{I}_{jk}(x_s) - \sigma_s \tilde{\Psi}(x_s) - q(x_s, \mu_{jk}) \right|^2 \\
& + \frac{w_4}{n_{\mu>0}} \sum_{k, \mu_{jk}>0} \left| \tilde{I}_{jk}(a) - I_a \right|^2 + \frac{w_5}{n_{\mu<0}} \sum_{k, \mu_{jk}>0} \left| \tilde{I}_{jk}(b) - I_b \right|^2,
\end{aligned}$$

where w_j , $j = 1, 2, 3, 4, 5$, are the weights of each term, $n_{\mu>0}$ is the number of selected positive directions μ_{jk} , and analogous for $n_{\mu<0}$. We note that the third term is the mean squared residual of the transport equation (1.1) on the sample points and the selected directions (usually a small subset of the ordinates directions). The last two terms are the boundary conditions (1.3)-(1.4). The training of the network $\mathcal{N}_{\text{IP}}$ is again performed by the gradient-like method (1.5), but it now has $(1 + n_n)n_\mu + 1$ more parameters to be fitted. The derivative of $\tilde{I}_{jk}$ required in the residual can be efficiently computed by automatic differentiation [BaEtAl18].

1.3 Numerical Experiments

We now present numerical experiments to show the availability of the ANN-MoC method to solve data-driven problems of neutral particle transport. In all the following test cases, we are considering the transport problem (1.1)-(1.4) with the manufactured solution

$$\hat{I}(x, \mu) = e^{-\alpha \sigma_t x}, \quad (1.8)$$

where $\alpha = 5$ and $\sigma_s = 0.5$. Therefore, radiative density is

$$\hat{\Psi}(x) = e^{-\alpha \sigma_t x},$$

and the source is

$$q(x, \mu) = [(1 - \mu \alpha) \sigma_t - \sigma_s] e^{-\alpha \sigma_t x}.$$

In the previous work [Tc24], it has been found that an adequate network $\mathcal{N}$ to this problem is an MLP of architecture $1 - 200 \times 3 - 1$ with the hyperbolic tangent and the softplus as the activation functions in the hidden and the output layers, respectively. From that, we also choose the Adam gradient-like method [KiBa17] to perform the parameter updates (1.5), with learning rate $l_r = 10^{-3}$. At each iteration $n_s = 100$ collocation samples $x_s \in D$ are randomly selected. The Gaussian quadrature set is fixed with $n_q = 4$. The iterations stop when the relative absolute difference of successive $\tilde{\Psi}^{(l-1)}$ and $\tilde{\Psi}^{(l)}$ are less than 0.49×10^{-4} .

Fig. 1.3 shows the ANN-MoC *versus* the exact solutions for $\sigma_t = 0.5, 1.0, 1.5$. These results have been found without any extra data, i.e. the network $\mathcal{N}$ has been trained to minimize the loss function (1.6). It is notable the good precision of the ANN-MoC estimations.

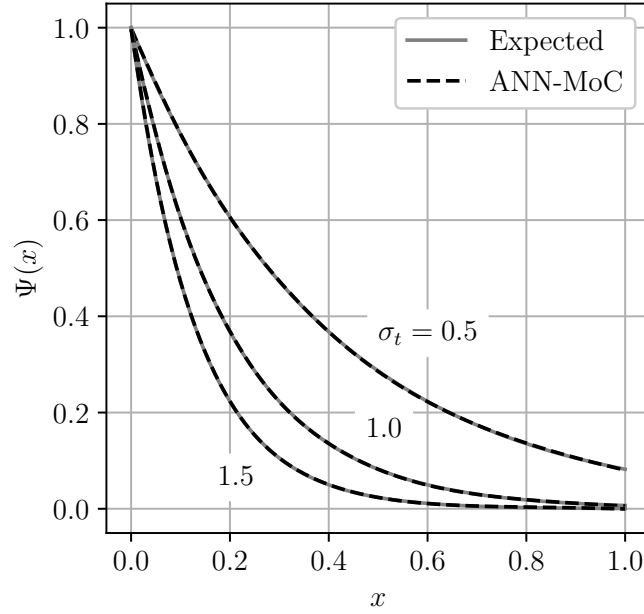


Fig. 1.3 The ANN-MoC *versus* the exact solutions for $\sigma_t = 0.5, 1.0, 1.5$ without extra data.

1.3.1 Model and Data-Driven Application

The data-driven ANN-MoC method allows extra data to be considered in the training of the network $\mathcal{N}$. An application of it is the acceleration of the convergence of the source iteration procedure. For instance, let us assume the given $n_d = 5$ data $\{x_d^{(s)}, \Psi_d^{(s)}\}_{s=1}^5$, $x_d^{(s)} = 0.25(s-1)$, $\Psi_d^{(s)} = \hat{\Psi}(x_d^{(s)})$. We also warm start the parameters of the network $\mathcal{N}(x; \mathbf{p}_w^{(0)})$ to fit the linear interpolator of the given data. The loss function penalty parameters have been fixed to $w_1 = w_2 = 1$.

Fig. 1.4 shows the convergence of the absolute error $\|\tilde{\Psi}^{(l)} - \hat{\Psi}\|$ for both the model-only and the data-driven ANN-MoC iterations. Taking into consideration the stochastic behavior of the iterates (samples points $\mathbf{x}^{(l)} \in D$ are randomly chosen at each iterate l), we have performed 5 runs for each version of the method. We have discarded the fastest and the slower runs. We note that the data-driven method requires about 50% less iterates to reach the stop criteria.

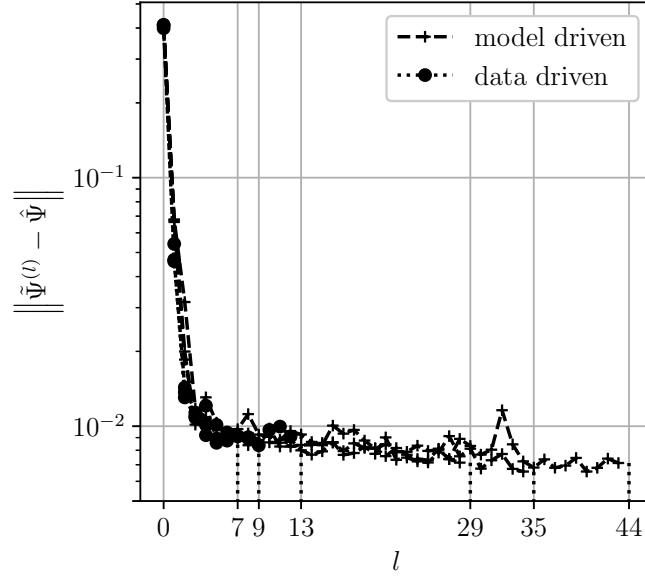


Fig. 1.4 Error converge comparison between model-only *versus* data-driven ANN-MoC iterations.

1.3.2 Inverse Problem Application

The inverse problem to be considered here is to estimate the extinction coefficient σ_t from given data sensors $\left\{ \left(a, \Psi_d^{(a)} \right), \left(b, \Psi_d^{(b)} \right) \right\}$ at the boundaries of the domain $D = [a, b]$ and assuming the transport problem holds with the manufactured solution (1.8). To this goal, the network $\mathcal{N}$ has been chosen to output the estimated $\tilde{\Psi}(x)$ and only the two radiant intensities $\tilde{I}_{\mu_0}$ and $\tilde{I}_{-\mu_0}$ for $\mu_0 = \sqrt{(3/7) + (2/7)\sqrt{6/5}} \approx 0.861136$. The loss function penalty parameters have been fixed to $w_1 = w_2 = w_3 = w_4 = w_5 = 1$. Here we have changed the stop criteria to $|\tilde{\sigma}_t^{(l)} - \tilde{\sigma}_t^{(l-1)}| < 0.49 \times 10^{-5}$.

Fig. 1.5 shows the results of three ANN-MoC runs with the initial approximations $\tilde{\sigma}_t^{(0)} = 0.25, 0.75, 1$. The expected coefficient is $\hat{\sigma}_t = 0.5$. We note that as usual, the far is the initial approximation from the expected value, the more ANN-MoC iterates were required to reach the stop criteria. In the worse case $\tilde{\sigma}_t^{(0)} = 1$, the number of iterations was 20, which is very reasonable.

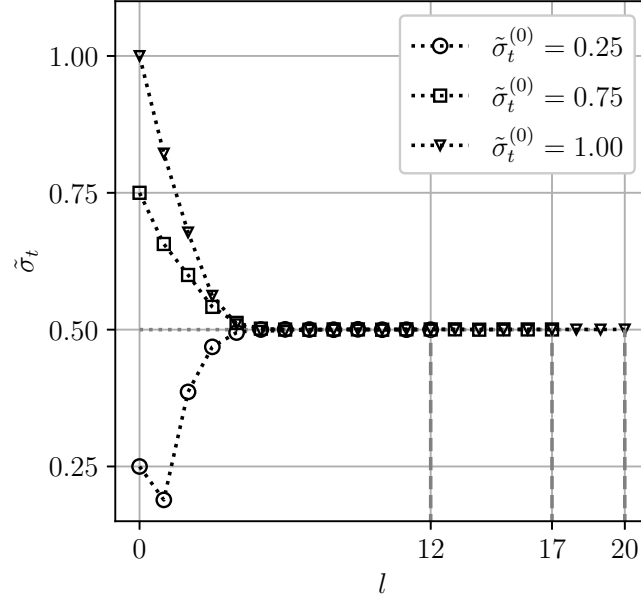


Fig. 1.5 The estimations $\tilde{\sigma}_t^{(l)}$ over the iterates l for three different initial guesses. $\hat{\sigma}_t = 0.5$.

1.4 Final Considerations

We presented the ANN-MoC method to solve data-driven neutral particle transport problems. The method can be adapted to direct and inverse (parameter estimation) problems. It estimates the radiative density by an artificial neural network (ANN), which is iteratively trained from the method of characteristic (MoC) solution in each source iteration. At each iteration, the ANN loss function is the mean squared error between expected and estimated densities from the MoC solution.

For direct data-driven problems, the method fits the data and accelerates the source iterations. This is possible by simply adding a penalty term to the loss function related to the data. In the numerical experiment shown above, with a data set of just 5 points, the data-driven ANN-MoC took at least 50% less iterations to reach the stop criteria than its model-only version.

The method can also be adapted to solve indirect parameter estimation problems. It requires an additional change to the architecture of the ANN to estimate the density together with selected radiant intensities. With this, the loss function is built with extra penalty terms regarding the residual of the transport equation, which has a direct relation with the parameter to be estimated. Automatic differentiation can be applied to compute the directional derivatives of the selected radiant intensities. For the ANN, the parameter to be estimated is just one more in many other of its

weights and biases. The numerical test shown above indicates the method converges to the expected parameter value in a reasonable number of iterations.

The data-driven ANN-MoC is a new conceptual method to solve direct and inverse neutral particle transport problems. Much work is still necessary to study the limitations of the method, such as sensibility tests on noisy data. Its most notable advantage is its versatility and capacity to solve problems with few extra data available.

1.5 Acknowledgment

This work has been partially funded by the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* (CAPES).

References

- [AdLa02] Adams, M.L. and Larsen, E.W.: Fast iterative methods for discrete-ordinates particle transport calculations. *Progress in Nuclear Energy*, **40**, 3–159 (2002)
- [Ba09] Bal, G.: Inverse transport theory and applications. *Inverse Problems*, **25**, 053001 (2009).
- [BaEtAl18] Baydin, A.G., Pearlmutter, B.A., Radul, A.A. and Sisking, J.M.: Automatic Differentiation in Machine Learning: A Survey. *Journal of Machine Learning Research*, **18**, 1–43.
- [HoEtAl21] Howel, J.R., Mengüç, M.P., Daun, K. and Siegel, R.: *Thermal Radiation Heat Transfer*, 7th ed., CRC Press, Boca Raton, FL (2021).
- [FrEtAl04] Frank, M., Seiad, M., Klar, R., Pinnau, R. and Thömmes, G.: A comparison of approximate models for radiation in gas turbines. *Computational Fluid Dynamics, an International Journal*, **4**, 191–197 (2004).
- [FuEtAl17] Fuqiang, W., Lanxin, M., Ziming, C., Jianyu, T., Xing, H. and Linhua, L.: Radiative heat transfer in solar thermochemical particle reactor - a comprehensive review. *Renewable and Sustainable Energy Reviews*, **73**, 935–949 (2017).
- [GoEtAl16] Goodfellow, I., Bengio, Y. and Courville, A.: *Deep Learning*, MIT Press, Cambridge, MA (2016).
- [KiBa17] Kingma, D.P. and Ba, J.L.: Adam: a method for stochastic optimization. *arXiv*, **1412.6980** (2017).
- [Ku23] Kuridan, R.M.: *Neutron Transport - Theory, Modelling, and Computations*, Springer, Cham (2023).
- [Ko23] Konzen, P.H.A.: ANN-MoC method for solving unidimensional neutral particle transport problems. In *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics*, Buske, D. (ed.), vol. 10 (CNMAC 2023), Brazil (2023), pp. 010019-1–7.
- [LaEtAl02] Larsen, E.W., Thömmes, G., Klar, A., Seaid, M. and Götz, T.: Simplified P_N approximations to the equations of radiative heat transfer and applications. *Computational Physics*, **183**, 652–675 (2002)
- [Le84] Lewis, E.E.: *Computational Methods of Neutron Transport*, John Wiley & Sons, New York (1984).
- [MiMo21] Mishra, S. and Molinaro, R.: Physics informed neural networks for simulating radiative transfer. *Journal of Quantitative Spectroscopy & Radiative Transfer*, **270**, 107705 (2021).
- [MoMa23] Modest, M.F. and Mazumder, S.: *Radiative Heat Transfer*, 4th ed., Academic Press, London (2023).

- [OzOr21] Özisik, M.N. and Orlande, H.R.B.: *Inverse Heat Transfer - Fundamentals and Applications*, CRC Press, Boca Raton, FL (2021).
- [RaEtAl19] Raissi, M., Perdikaris, P. and Karniadakis, G.E.: 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).
- [RoEtAl24] Roman, N.G., Santos, P.C. and Konzen, P.H.A.: ANN-MoC method for inverse transient transport problems in one-dimensional geometry. *Latin-American Journal of Computing*, **11**, 41–50 (2024).
- [St07] Stacey, W.M.: *Nuclear Reactor Physics*, Wiley-VCH, Darmstadt (2007).
- [Tc24] Tchanchalam, A.: Método ANN-MoC para problemas de transporte de partículas neutras em geometria 1D. *Master dissertation*, Konzen, P.H.A. (supervisor), PPGMAp, UFRGS, Brazil (2024).
- [WaWu07] Wang, L.V. and Wu, H.: *Biomedical Optics - Principles and Imaging*, John Wiley & Sons, Hoboken, NJ (2007).