Journal of Quantitative Spectroscopy & Radiative Transfer 331 (2025) 109253

Contents lists available at ScienceDirect

Journal of Quantitative Spectroscopy and Radiative Transfer

journal homepage: www.elsevier.com/locate/jqsrt

Physics-informed neural networks for modeling atmospheric radiative transfer

Shai Zucker [a,*], Dmitry Batenkov [a,b], Michal Segal Rozenhaimer [c]

[a] Department of Applied Mathematics, School of Mathematical Sciences, Tel Aviv University, Tel Aviv, Israel
[b] Basis Research Institute, New York, NY, USA
[c] Bay Area Environmental Research Institute, NASA Ames Research Center, Moffett Field, CA, 94035, USA

ARTICLE INFO

Keywords:
Radiative transfer equation
Physics-informed neural networks
Atmospheric modeling
Remote sensing
Aerosols
Clouds

ABSTRACT

Understanding the radiative transfer processes in the Earth's atmosphere is crucial for accurate climate modeling and climate change predictions. These processes are governed by complex physical phenomena, which can be generally modeled by the radiative transfer equation (RTE). Solutions to the RTE are obtained by various methods including numerical (standard RTE solvers), stochastic (Monte-Carlo), and data-driven (machine-learning) approaches. This paper introduces a novel numerical approach utilizing a Physics-Informed Neural Network (PINN) to solve the RTE in atmospheric scenarios, applying physics constraints in a machine-learning framework. We show that our PINN model offers a flexible and efficient solution, enabling the simulation of radiance values using plane-parallel atmosphere, and under diverse conditions, including clouds and aerosols.

1. Introduction

The Earth's radiative budget plays a significant role in driving weather and climate processes. Therefore, its accurate modeling is critical for understanding current and future climate. Both solar and thermal radiation undergo modifications as they traverse the atmosphere, primarily through scattering and absorption by molecules, aerosol particles, and cloud droplets. The interaction of radiation with clouds and aerosols is highly complex, leading to uncertainties in current radiative forcing models [1]. Many of the algorithms developed for obtaining the optical parameters, which define the aerosol and cloud's radiative forcing properties, rely on forward radiative transfer (RT) simulations generated by different solvers. Among these, retrievals of the optical parameters are made by comparing the measured radiance using methods such as Look-Up Tables (LUTs) [2], optimization methods [3], or neural networks [4]. Classical approaches to numerical solutions of the radiative transfer equation (RTE) include the Discrete Ordinates Method (DISORT) [5], the Monte Carlo Method [6], and Successive Orders of Scattering and Markov Chain methods [7,8]. These methods are widely used in the atmospheric RT community and are considered the standard for radiative transfer simulations. However, these methods have several disadvantages, including the need for a grid to solve the equation, which can limit the ability to simulate radiance over complex geometries, and the vast modifications needed to generate the Jacobian matrices (derivatives of the solution with respect to the parameters) for optical parameter retrieval. Additionally, the run-time for repeatedly running the RTE solvers forward can be long when attempting to incorporate them in an end-to-end retrieval scheme.

An emerging field in the Scientific Machine Learning (SCIML) community is the development of Physics-Informed Neural Network (PINN) models, which provide a unified approach to solving forward and inverse problems for various differential equation models, including ordinary (ODEs), partial (PDEs), and integro-differential equations such as the RTE. The utilization of PINNs for radiative transfer has only emerged recently [9,10], and the existing work focused solely on theoretical derivation.

In this work, we adopt and implement the PINN approach within the context of the atmosphere. We demonstrate that this method can precisely replicate radiance values across a range of atmospheric scenarios, including the presence of molecular gases, aerosols, and clouds. Additionally, we provide a comparative analysis of our PINN-based model against community-standard radiative transfer solvers and offer accuracy assessments. Accompanying this paper is a Python code leveraging the Pytorch framework (RTPINN), designed for simulating radiative transfer under varied atmospheric conditions.

The paper is organized as follows. Section 2 formulates the RTE according to an atmospheric profile. Subsequently, the PINN framework for the forward RTE solution is introduced in Section 3. Section 4 provides a numerical analysis of the PINN model and its convergence

* Corresponding author.
E-mail address: shaizucker@mail.tau.ac.il (S. Zucker).

https://doi.org/10.1016/j.jqsrt.2024.109253
Received 21 May 2024; Received in revised form 23 August 2024; Accepted 28 October 2024
Available online 12 November 2024

properties. A description of the implementation and numerical experiments and comparisons with other solvers is provided in Section 5. Finally, Section 6 offers key conclusions and future work.

2. Atmospheric radiative transfer

In this section we follow the formulation described in [11]. We will consider a plane-parallel atmosphere, representing a 1D version of the radiative transfer equation (RTE) that assumes spatial dependence only on altitude. The term $I(\tau,\mu,\phi)$ describes the (scalar) radiance in a specific wavelength at a given optical depth $\tau \in [0,\tau^*]$, zenith direction $\mu = \cos(\theta) \in [-1,1]$ and azimuth angle $\phi \in [0,2\pi)$. The radiative transfer equation is given by:

$$\mu\frac{\partial I}{\partial \tau} = I(\tau,\mu,\phi) - S(\tau,\mu,\phi), \tag{1}$$

where the source function S is a sum of two terms:

$$S(\tau,\mu,\phi) = Q(\tau,\mu,\phi) + \frac{\omega(\tau)}{4\pi}\int_0^{2\pi}\int_{-1}^{1} P(\tau,\mu,\phi,\mu',\phi')I(\tau,\mu',\phi')d\mu' d\phi'. \tag{2}$$

The first term is the internal source, denoted as Q, corresponding to direct solar radiation, black body emission, or both. The second term is the scattered radiation in the direction specified by μ,ϕ. This term is determined based on the scattering phase function P, which describes the angular scattering pattern of an infinitesimal volume. The single scattering albedo (SSA) is denoted by $\omega(\tau)$ and gives the spatially dependent ratio between total scattering and extinction.

We define the internal source term Q according to the source as follows:

$$Q^{(\text{thermal})}(\tau) = \left\{1-\omega(\tau)\right\} B[\lambda, T(\tau)], \tag{3}$$

where $Q^{(\text{thermal})}(\tau)$ represents the thermal emission source term for the temperature $T(\tau)$ according to the Planck function B. Additionally, the beam source term is defined as:

$$Q^{(\text{beam})}(\tau,\mu,\phi) = \frac{\omega(\tau)I_0}{4\pi}P(\tau,\mu,\phi;-\mu_0,\phi_0)e^{-\tau/\mu_0}, \tag{4}$$

where $Q^{(\text{beam})}(\tau,\mu,\phi)$ accounts for the direct solar radiation.

The RTE (1) is accompanied by the following boundary conditions defined according to the cosine solar zenith angle (SZA) and solar azimuth angle, μ_0 and ϕ_0 respectively, and the surface temperature T_g:

$$I(\tau,\mu,\phi) = \begin{cases} I_{top} & \tau = 0 \wedge \mu < 0, \quad (5a)\\ \varepsilon(\mu)B(T_g) + \frac{1}{\pi}\mu_0 I_0 e^{-\tau/\mu_0}\rho(\mu,-\mu_0,\phi-\phi_0) & \\ \quad + \frac{1}{\pi}\int_0^{2\pi} d\phi' \int_0^1 \rho(\mu,-\mu',\phi-\phi')I(\tau^*,-\mu',\phi')\mu' d\mu' & \tau = \tau^* \wedge \mu > 0. \quad (5b)\end{cases}$$

The constant value of I_{top} is intended to simulate a thermally emitting upper boundary or a highly scattering boundary, like a cloud. In most cases, including this work, I_{top} is set to zero. $B(T_g)$ is the Planck Function given by surface temperature, ε describes the directional emissivity, and ρ is the bidirectional reflectivity distribution function (BRDF).

As is common in numerical solvers [5], the solution is assumed to be decomposed into Fourier components

$$I(\tau,\mu,\phi) = \sum_{m=0}^{M-1} I^m(\tau,\mu)\cos(m(\phi-\phi_0)). \tag{6}$$

The phase function is assumed to depend solely on the angle Θ between the incident and scattered beam directions and can be expressed as a finite Legendre decomposition, where P_l are the Legendre polynomials and g_l are the phase function coefficients:

$$P(\tau,\Theta) = \sum_{l=0}^{M-1}(2l+1)g_l(\tau)P_l(\cos(\Theta)). \tag{7}$$

The phase function P is also decomposed into its spherical Fourier modes, by utilizing the addition theorem for spherical harmonics

$$P(\tau,\mu,\mu',\phi-\phi') = \sum_{m=0}^{M-1}(2-\delta_{0m})\left[\sum_{l=m}^{M-1}(2l+1)g_l(\tau)\Lambda_l^m(\mu)\Lambda_l^m(\mu')\right]\cos(m(\phi-\phi')), \tag{8}$$

where Λ_l^m are defined as follows

$$\Lambda_l^m(\mu) = \sqrt{\frac{(l-m)!}{(l+m)!}}P_l^m(\mu), \tag{9}$$

and P_l^m are the associated Legendre polynomials. By substituting the Fourier decompositions (6), (8) back into the RTE (1), (2) we get M independent integro-differential equations:

$$\mu\frac{\partial I^m(\tau,\mu)}{\partial\tau} = I^m(\tau,\mu) - S^m(\tau,\mu)\ \ (m=0,1,\dots,M-1), \tag{10}$$

Correspondingly, the source function is given by the following terms:

$$S^m(\tau,\mu) = \int_{-1}^{1} D^m(\tau,\mu,\mu')I^m(\tau,\mu')d\mu' + Q^m(\tau,\mu),$$

$$D^m(\tau,\mu,\mu') = \frac{\omega(\tau)}{2}\sum_{l=m}^{M-1}(2l+1)g_l(\tau)\Lambda_l^m(\mu)\Lambda_l^m(\mu'),$$

$$Q^m(\tau,\mu) = X_0^m(\tau,\mu)e^{-\tau/\mu_0} + \delta_{m0}Q^{thermal}(\tau),$$

$$X_0^m(\tau,\mu) = \frac{\omega(\tau)I_0}{4\pi}(2-\delta_{m0})\sum_{l=m}^{M-1}(-1)^{l+m}(2l+1)g_l(\tau)\Lambda_l^m(\mu)\Lambda_l^m(\mu_0).$$

Here δ_{ij} denotes the Kronecker's delta function.

A similar decomposition is performed over the boundary conditions (5) yielding M independent boundary conditions: for each $m = 0,1,\dots,M-1$ we have

$$I^m(\tau=0,\mu) = 0, \quad \mu < 0, \tag{11}$$

$$I^m(\tau=\tau^*,\mu) = I_g^m(\mu), \quad \mu > 0. \tag{12}$$

Here, the surface reflected radiance $I_g^m(\mu)$ is given by the following equation:

$$I_g^m(\mu) = \delta_{m0}\varepsilon(\mu)B(T_g) + \frac{1}{\pi}\mu_0 I_0 e^{-\tau^*/\mu_0}\rho^m(\mu,-\mu_0) + (1+\delta_{m0})\int_0^1 \rho^m(\mu,-\mu')I^m(\tau^*,-\mu')\mu' d\mu', \tag{13}$$

where $\rho^m(\mu,-\mu') = \frac{1}{\pi}\frac{2-\delta_{m0}}{2}\int_0^{2\pi}\rho(\mu,-\mu',\phi-\phi')\cos(m(\phi-\phi'))d(\phi-\phi')$ are the Fourier components of the reflection kernel.

3. The RTPINN framework

The PINN approach is an innovative framework that combines deep learning techniques with physical laws to solve complex scientific and engineering problems. PINNs aim to leverage the expressive power of neural networks while enforcing the governing physics equations as constraints during the learning process [12]. This is achieved by integrating the knowledge of the underlying equations and the boundary conditions into the loss function, allowing the solution of partial, ordinary, and integro-differential equations by minimizing the loss over many points sampled throughout the domain.

The approach was initially introduced by [12], where the first implementation of a Physics-Informed Neural Network (PINN) was demonstrated, showcasing its capability in solving different differential equations such as the Navier–Stokes and non-linear Schrödinger equations. This idea was further extended to operator solutions in [13]. To date, there have been only a few publications applying the PINN to address the Radiative Transfer Equation (RTE). In [9], they showed how a PINN architecture could solve the scalar RTE. They derived an error bound for the generalized error of a trained network which depends

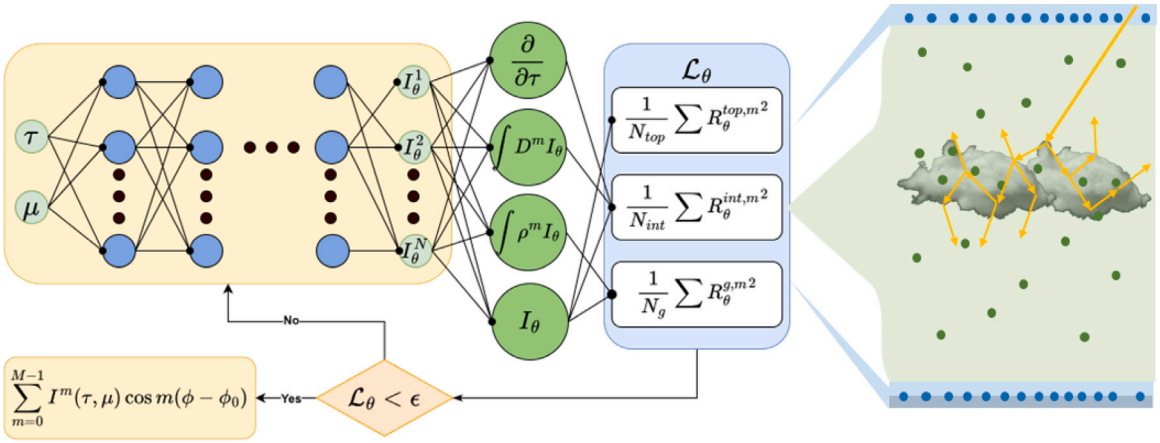


Fig. 1. Evaluation of the loss function over the atmosphere. Collocation points (green dots, S^{int}) are sampled over the domain to validate the RTE term, and boundary points (blue dots, S^g, S^{top}) are sampled to measure the correspondence with the boundary conditions.

solely on the number of training point. In [14], the authors further researched the utilization of PINN for radiative transfer and examined the solution stability. In [10] they considered the 1D equation with azimuthally averaged radiance and demonstrated how the integration kernel can be evaluated using a second network without quadrature. As far as we know, PINNs were never utilized in physical boundary conditions and were never compared to state-of-the-art atmospheric RTE solvers. For all cases mentioned, the boundary condition was generated under the assumption that the radiance values at the surface are known, which is not valid for most realistic applications. Our research did not assume prior knowledge of the radiance solution other than the BRDF function as in (5), thus creating the first PINN model comparable with the standard radiative transfer solvers.

The neural network inputs are the pair $x = (\tau, \mu)$, while the output $u_\theta(x)$ is the vector $\left(I_\theta^0(\tau,\mu),\dots,I_\theta^{M-1}(\tau,\mu)\right)$ which is an approximation to $\left(I^0(\tau,\mu),\dots,I^{M-1}(\tau,\mu)\right)$ as in (6), This approach of considering the network output as the Fourier coefficients of a differential equation solution was first demonstrated in [15]. The optimization of the network parameter vector θ is performed to minimize the discrepancy between the two vectors, by ensuring that each $I_\theta^m(\tau,\mu)$, $m = 0,\dots,M-1$ adheres to the RT Eq. (10) and its boundary conditions (11), (12) and (13) at various randomly selected points. If a sufficient number of points is sampled, the network will accurately represent the solution to the differential equation across the entire domain. This approach therefore provides a "grid-free" solution.

The PINN architecture we consider here is in the form of a feed-forward fully connected neural network, which forms a function $u_\theta(x)$ that calculates its output through n layers of units (neurons) composed of affine-linear maps of weights (W_k) and biases b_k fed through non-linear activation functions. Formally, we have:

$$
\begin{aligned}
z_1 &= x,\\
z_{k+1} &= \sigma(W_k z_k + b_k), \quad k = 1,\dots,n,\\
u_\theta(x) &= z_{n+1},
\end{aligned}
$$

with

$$W_k \in \mathbb{R}^{d_{k+1}\times d_k}, b_k \in \mathbb{R}^{d_{k+1}}, \quad k = 1,\dots,n,$$

and

$$\theta = \{W_k, b_k\}_{k=1}^n.$$

Here, σ is a scalar (non-linear) activation function applied to vectors component-wise. As recommended in [16], we use the hyperbolic tangent (tanh) function as the activation function. While other choices can also be considered, the commonly used ReLU function is not effective for PINNs due to its zero second derivative, which can cause the gradient of the differential term in the equation residual to vanish, significantly affecting training.

Neural networks are trained to optimize the weights $\{W_k\}$ and biases $\{b_k\}$ via minimizing a loss function with a gradient-based method. Our PINN approach relies on training over points spread in the spatial domain (i.e., collocation points) over which residual terms (describing adherence to the physical constraint) in the loss function are minimized. For interior collocation points, we define a residual term $R_\theta^{int,m}$ that measures how well the solution satisfies the equation. For points on the boundary, we define residual terms $R_\theta^{top,m}$ and $R_\theta^{g,m}$ measuring how well the model adheres to the top and the bottom boundary conditions, respectively. Fig. 1 demonstrates how the sampled radiance function is optimized over different points in the atmospheric domain.

The interior residual term is formulated based on the differential equation specific to each mode, approximating the integral of the source term through Gaussian quadrature using N_S quadrature nodes $\{\mu_j', w_j'\}_{j=1}^{N_S}$.

$$R_\theta^{int,m}(\tau,\mu) = \mu\frac{\partial I_\theta^m}{\partial\tau} - I_\theta^m + \sum_{j=1}^{N_S} w_j' D(\tau,\mu,\mu_j') I_\theta^m(\tau,\mu_j') + Q^m(\tau,\mu). \tag{14}$$

Here the derivative term is evaluated using the automatic differentiation of neural networks.

The residual term for the top boundary points is defined according to the top boundary condition for $\mu < 0$ as follows:

$$R_\theta^{top,m}(\mu) = I_\theta^m(0,\mu) - I_{top} = I_\theta^m(0,\mu). \tag{15}$$

For the surface reflection condition described in (12), (13), we get the following residual term where the reflection integral is approximated using Gaussian quadrature of N_R nodes $\{(\tilde{\mu}_i, \tilde{w}_i)\}_{i=1}^{N_R}$:

$$
\begin{aligned}
R_\theta^{g,m}(\mu) =& I_\theta^m - \delta_{m0}\epsilon(\mu)B(T_g)\\
&- \frac{1}{\pi}\mu_0 I_0 e^{-\tau^*/\mu_0}\rho^m(\mu,-\mu_0)\\
&- (1+\delta_{m0})\frac{1}{\pi}\sum_{j=1}^{N_R}\tilde{w}_j\rho^m(\mu,-\tilde{\mu}_j)I_\theta^m(\tau^*,-\tilde{\mu}_j)\tilde{\mu}_j.
\end{aligned}
\tag{16}
$$

Having defined the residual terms, we now move to the loss function. Let the interior and the boundary points be denoted as follows:

$$
\begin{aligned}
S^{int} &:= \{(\tau_i,\mu_i), \quad i = 1,\dots,N_{int}\},\\
S^{top} &:= \{(0,\mu_j^{top}), \quad j = 1,\dots,N_{top}\},\\
S^{g} &:= \{(\tau^*,\mu_l^{g}), \quad l = 1,\dots,N_g\}.
\end{aligned}
$$

We take the loss $\mathcal{L}_\theta$ to be the average of the residual terms over the sampled nodes in the domain

$$\mathcal{L}_\theta = \frac{1}{M}\sum_{m=0}^{M-1}\left[\mathcal{E}^{int,m} + \mathcal{E}^{top,m} + \mathcal{E}^{g,m}\right], \tag{17}$$

where the error terms $\mathcal{E}^{int,m}, \mathcal{E}^{top,m}, \mathcal{E}^{g,m}$ approximate the integrated residuals over the domain:

$$\mathcal{E}^{int,m} = \frac{1}{N_{int}}\sum_{i=1}^{N_{int}} R_\theta^{int}(\tau_i,\mu_i)^2,\ \mathcal{E}^{top,m} = \frac{1}{N_{top}}\sum_{j=1}^{N_{top}} R_\theta^{top,m}(\mu_j^{top})^2,\ \mathcal{E}^{g,m} = \frac{1}{N_g}\sum_{l=1}^{N_g} R_\theta^{g,m}(\mu_l^g)^2. \tag{18}$$

Although it is not typically treated as a boundary condition in most radiative transfer literature, another constraint is imposed on the radiance solution: at the boundaries of $\mu = -1$ and $\mu = 1$, the solution must remain constant over $\phi \in [0, 2\pi)$. When $M = 1$, this constraint is naturally satisfied. However, for higher modes, it can be used to regularize the optimization process. Since the variation is constant, all modes with $m > 1$ are eliminated, leading to the following constraint for the PINN solution:

$$\mathcal{E}^{reg} = \frac{1}{2N_{int}(M-1)}\sum_{m=1}^{M}\sum_{i=1}^{N_{int}} I_\theta^{m2}(\tau_i, 1) + I_\theta^{m2}(\tau_i, -1), \tag{19}$$

we add this term to the loss function to ensure that the solution remains constant.

The network is trained by optimizing the parameters θ to minimize $\mathcal{L}_\theta$. According to (6), the predicted output is computed for a new input τ, μ, ϕ as the Fourier sum of the outputs of the M networks:

$$I_\theta(\tau,\mu,\phi) := \sum_{m=0}^{M-1} I_\theta^m(\tau,\mu)\cos m(\phi-\phi_0). \tag{20}$$

4. Numerical analysis

In this section, we provide a theoretical analysis of the convergence of the PINN model for the radiative transfer equation. We start by considering the error estimation of the PINN model and then provide a theorem for the convergence of each Fourier mode over the domain $S := [0,\tau^*]\times[-1,1]$ and then generalized it to the solution function $I(\tau,\mu,\phi)$. The analysis is based on the work of [9] generalized to the atmospheric domain.

Theorem 1. *Let $I \in L^2([0,\tau^*]\times[-1,1]\times[0,2\pi])$ be the unique weak solution of the radiative transfer Eq. (1), and let I^m be its Fourier mode of order m. Let D^m, ρ^m satisfy the following conditions for every $\tau \in [0,\tau^*]$:*

- $D^m(\tau,\mu,\mu') \in C^{2N_S}([-1,1]\times[-1,1])$;
- $\rho^m(\mu,\mu') \in C^{2N_R}([-1,1]\times[-1,1])$.

Assuming the scattering phase function is energy-conserving, and ρ^m and D^m not fully scattering, i.e.

$$\int_{-1}^{1}|D^m(\tau,\mu,\mu')|d\mu' < M_S < 1, \tag{21}$$

$$(1+\delta_{m0})\int_0^1 |\rho^m(\mu,-\mu')|\mu' d\mu' = M_R < 1. \tag{22}$$

Assume the residual terms (14), (15) and (16) are evaluated at N_{int}, N_g, N_{top} Sobol points respectively. Further, the scattering and reflection kernels are computed using Gauss Legendre quadrature points $\{\mu_i', w_i'\}_{i=1}^{N_S}$, and $\{\tilde{\mu}_i, \tilde{w}_i\}_{i=1}^{N_R}$ for the scattering and reflection kernels. Then the following expression bounds the generalization error of each mode of the PINN solution $\widehat{I^m}(\tau,\mu) := I^m(\tau,\mu) - I_\theta^m(\tau,\mu)$:

$$\mathcal{E}_G^m = \int_S \left(\widehat{I^m}(\tau,\mu)\right)^2 d\tau d\mu \le C\left[\mathcal{E}^{top,m} + \mathcal{E}^{g,m} + \mathcal{E}^{int,m} + \frac{\log N_g}{N_g} + \frac{\log N_{top}}{N_{top}} + \frac{\log^2 N_{int}}{N_{int}} + \left(\frac{e}{N_S}\right)^{4N_S} + \left(\frac{e}{N_R}\right)^{4N_R}\right]. \tag{23}$$

The constant C is defined in the proof (see Eq. (A.23)) and depends on the variation of the residual terms (cf. Definition 2), as well as the norms of the network prediction function and the initial data.

The proof of Theorem 1 is provided in Appendix A. Here let us remark on the conditions (21), (22) of Theorem 1, which may not seem fully intuitive. As we argue below, they should be satisfied in any reasonable scenario.

First, since the scattering phase function can be considered as a probability density function (PDF), it is non-negative and integrates to 1, i.e.

$$\frac{1}{4\pi}\int_0^{2\pi}\int_{-1}^{1} P(\tau,\mu,\mu',\phi-\phi')d\mu' d(\phi-\phi') = 1. \tag{24}$$

Since $|\cos(m(\phi-\phi'))| \le 1$ and $P(\tau,\mu,\mu',\phi-\phi') \ge 0$ we get the following inequality:

$$\frac{1}{2}\int_{-1}^{1}\left|\frac{1}{2\pi}\int_0^{2\pi} P(\tau,\mu,\mu',\phi-\phi')\cos(m(\phi-\phi'))d(\phi-\phi')\right| d\mu' \le 1, \tag{25}$$

which implies that

$$\int_{-1}^{1}\left|\frac{2-\delta_{0m}}{2}\left[\sum_{l=m}^{M-1}(2l+1)g_l(\tau)\Lambda_l^m(\mu)\Lambda_l^m(\mu')\right]\right| d\mu' \le 1. \tag{26}$$

One can see that the last inequality is equivalent to (21) up to $\omega(\tau)$ which means that as long as $\omega(\tau) < 1$ the condition is satisfied. While the case of $\omega(\tau) = 1$ is theoretically possible, it cannot be realized in practice, since all materials exhibit some level of absorption. A similar argument can be made for the reflection kernel, where the condition (22) is equivalent to a minimal level of absorption on the surface.

By applying Theorem 1 to each Fourier mode and assuming that the solution is decomposed as in (6), we obtain the following corollary for the accuracy of the network output (20):

Corollary 1. *Assuming all the conditions of Theorem 1 hold, and that the solution can be decomposed into a finite Fourier decomposition, the generalization error of the PINN satisfies the following bound:*

$$\begin{aligned}\mathcal{E}_G^T &= \int_S\int_0^{2\pi}\hat{I}^2 d\tau d\mu d\phi = \int_S \left(I_\theta(\tau,\mu,\phi) - I(\tau,\mu,\phi)\right)^2 \\ &\le \tilde{C}\left[\frac{\log N_g}{N_g} + \frac{\log N_{top}}{N_{top}} + \frac{\log^2 N_{int}}{N_{int}} + \left(\frac{e}{N_S}\right)^{4N_S} + \left(\frac{e}{N_R}\right)^{4N_R} + \sum_{m=0}^{M-1}\left\{\mathcal{E}^{top,m} + \mathcal{E}^{g,m} + \mathcal{E}^{int,m}\right\}\right].\end{aligned} \tag{27}$$

Proof. By (6), (20) and the Parseval's theorem, we have

$$\int_S\int_0^{2\pi}\hat{I}^2 d\tau d\mu d\phi = \pi\sum_{m=0}^{M-1}\int_S \widehat{I^m}^2 d\tau d\mu = \pi\sum_{m=0}^{M-1}\mathcal{E}_G^m.$$

Applying Theorem 1 to each mode we get the desired result. □

The bounds provided in Theorem 1 and Corollary 1 limit the generalization error based on the training loss terms, N_S, N_R, and the number of training points. When considering these bounds, it is important to note that the training error cannot be estimated prior to the solution process but is available at every epoch of the optimization. From the user's perspective, as the loss values approach zero, the error order will be determined by N_S, N_R and the number of training points according to the convergence order specified in Theorem 1 and Corollary 1.

Fig. 2. Flow chart describing simulation process using RTPINN, the optical properties can either be generated by the standalone software or by the Libradtran code.

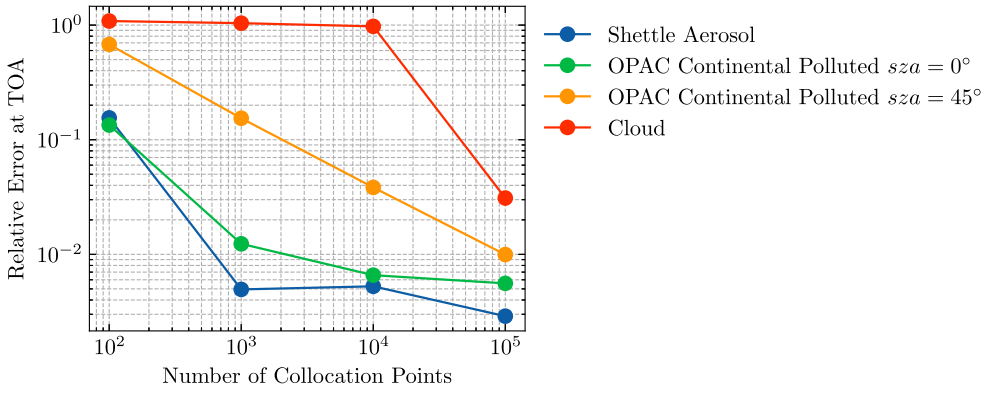


Fig. 3. PINN solution relative error from DISORT as a function of the number of training points. The x-axis indicates the number of collocation points N_{int}, with the number of boundary points N_g, N_{top} fixed at one-tenth of the collocation points, and the y axis represents the relative error in absolute values. The error is evaluated over 90 equidistant μ_i sampled in $(0, 1]$.

4.1. Intensity correction

For cases with a strong forward scattering component in the phase function, many Legendre coefficients and Fourier modes are required to accurately solve the radiance solution. In order to address this, we implemented the $\delta - M$ transformation [17]. The δ–M method is used to decompose the phase function P into a combination of a delta function in the forward direction and a truncated phase function P', which is then expanded into a series of Legendre polynomials and over which the RTPINN is trained. In more detail, we have

$$P(\tau, \cos\Theta) \approx 2f\delta(\cos\Theta) + P'(\tau, \cos\Theta),$$

where f is the separated fraction chosen to be $f = g_N(\tau)$ where N is the truncation index, usually set to the number of streams N_S. According to f we define the $\delta - M$ scaled optical parameters relevant to the RTE over P'

$$g_l'(\tau) = \frac{g_l(\tau) - f}{1 - f} \tag{28}$$

$$\omega'(\tau) = \frac{\omega(\tau)(1 - f)}{1 - \omega(\tau)f} \tag{29}$$

$$\tau' = (1 - \omega(\tau)f)\tau \tag{30}$$

Nakajima and Tanaka [18] demonstrated that although the $\delta - M$ transformation is an effective truncation method, it introduces oscillations in the radiance solution, particularly pronounced in the aureole area. In their study, they illustrated that modifying the $\delta - M$ solution to incorporate direct computations of single and double scattering can significantly mitigate these errors in the radiance solution, those intensity corrections are reffered as the IMS corrections. The DISORT method applies IMS corrections when the delta-M transformation is used. In contrast, we found that RTPINN should not always include these corrections, as the PINN solution naturally produces smooth functionals, and adding counteracting oscillations from the correction could distort the results, this phenomena is demonstrated in Fig. 6.

Table 1
Parameters used for the creation of Fig. 3, featuring OPAC and Shettle aerosol models [20,21] and cloud models. The cloud simulation shown was generated with an $r_{\text{eff}} = 10\,[\mu\text{m}]$ and stretching between 2–5 [km]. All networks used $n = 8$ hidden layers with $d_k = 30$ neurons each.

Scenario	τ^*	Phase function	N_S	M	μ_0
Shettle aerosol model	0.29	Henyey-Greenstien	16	1	1
OPAC continental polluted	0.39	Mie simulated	16	1	1
OPAC continental polluted	0.39	Mie simulated	16	16	$\cos(\pi/4)$
Cloud	15.13	Mie simulated	16	1	1

5. Implementation and validation

A radiative transfer forward solver must be coupled with additional software for defining the atmospheric composition (aerosol, gases, clouds) and the scene geometry (solar angles, surface properties etc.) For this work, we developed the RTPINN in Python, while implementing the necessary Mie simulations for the aerosol optical parameters and supporting various options for BRDF and phase functions. In addition to the standalone option, the software was integrated with the community standard library - Libradtran [19], and operated on the same inputs passed into the DISORT solver. As such, the RTPINN can be used as a solver within the Libradtran package, thus allowing to include the generic scene description of Libradtran. Fig. 2 presents the flow of the software structure.

We investigated the convergence behavior of our RTPINN model across various configurations, focusing on the number of collocation points and boundary points. Fig. 3 depicts the RTPINN error when applied to solve the radiative transfer equation in a plane-parallel atmosphere for non-polarized radiance intensity $I(\tau, \mu, \phi)$ under various scenarios described in Table 1 where the number of collocation and boundary points varies. The error is quantified as the deviation between the PINN model's predictions and those obtained from the DISORT code [5] at the top of the atmosphere (TOA). In all simulations, the $\delta - M$ transformation is applied, truncating the phase function's Legendre decomposition.

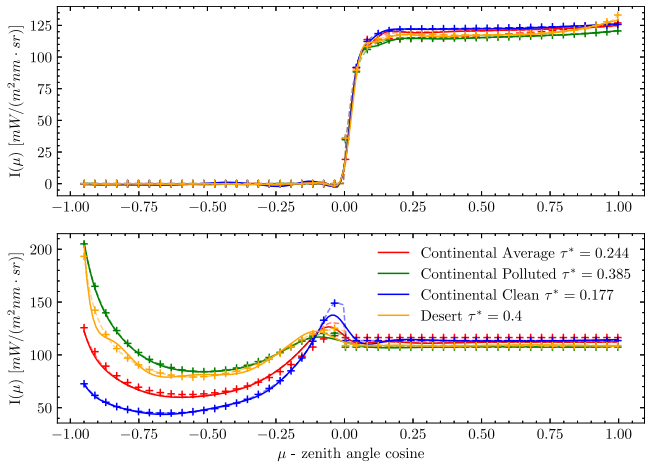


Fig. 4. Simulation of the mean diffused radiance for continental aerosol models from OPAC over above a Lambertian BRDF with albedo set to 0.2. The continental aerosols are set at the base altitude of 2 [km] and the desert aerosol model at the base altitude of 6 [km]. The upper subplot shows the TOA mean radiance, and the bottom shows the bottom of atmosphere (BOA) mean radiance. The boundary condition of zero isotropic radiance at the top of the atmosphere is demonstrated. Solid lines represent the simulation using the RTPINN model, dashed lines represent the utilization of the DISORT method and the + signs represent results from the Mystic Monte Carlo code.

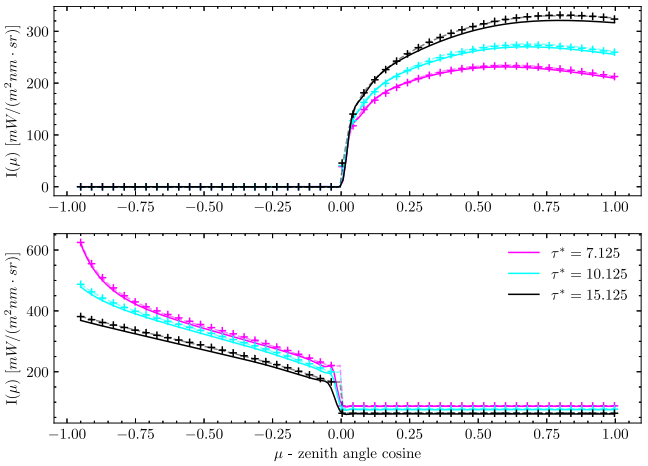


Fig. 5. Simulation of the mean diffused radiance for cloud models of varying optical depth. Simulation was performed with $M = 1$ and $N_S = N_R = 16$. Cloud wes generated with an effective droplet radius of 10 [μm].

Table 2
The parameters values of the different BRDF models considered in the simulations presented in Fig. 7.

Model			
Lambertian	$\rho_0 = 0.5$		
RPV	$\rho_0 = 0.076$	$k = 0.648$	$g = -0.290$
Ross-Li	$k_{\text{iso}} = 0.3$	$k_{\text{vol}} = 0.023$	$k_{\text{geo}} = 0.023$

To assess the ability to model aerosol measurements, we conducted simulations of aerosol diffuse radiance for various aerosols using the Optical Properties of Aerosols and Clouds (OPAC) database supported by Libradtran [20]. Fig. 4 illustrates sevral simulations performed over different aerosol species data from the OPAC parameterization in LibRadtran. The sun was positioned directly at the zenith, resulting in azimuthal symmetry ($I^m = 0; \forall m > 0$), as such the simulation was preformed with $M = 1$. For all simulations presented in this paper, the wavelength was fixed at $\lambda = 555,\text{nm}$, and the atmospheric model was set to the US-Standard atmosphere from [19]. The neural network employed in the simulations featured a *tanh* activation function and was fine-tuned using the L-BFGS [22] optimizer. It included 8 fully connected (FC) layers, each with 30 neurons, and utilized 10,000 collocation points along with 1000 boundary points. We configured $N_S = N_R = 16$ for the integral quadrature, aligning with a 16-stream simulation in the DISORT solver. Additionally, we compared results with the Mystic - Monte Carlo code [6] within LibRadtran, setting the photon count to 10^5. As depicted in the figure, comparable outcomes across all solvers confirm that the PINN model performs on par with industry-standard solvers.

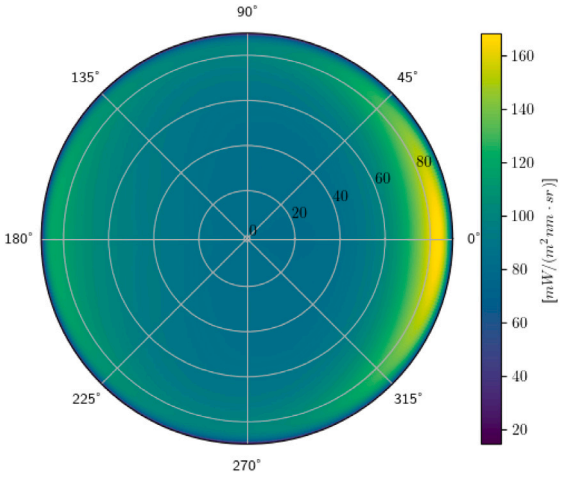
As illustrated in Fig. 3, cloud simulations necessitate a greater number of training points to ensure accurate convergence compared to aerosol simulations. This requirement arises from the increased complexity of the phase function and scattering properties of clouds. Fig. 5 displays cloud simulations generated with $N_{int} = 100,000$ collocation points and $N_{top} = N_g = 10{,}000$. The simulation was produced using an atmospheric profile that includes a cloud extending from 2–5 [km] over US-Standard atmosphere, and the cloud optical depth (COD) varies between each simulation.

Symmetry is not given in most atmospheric scenarios, and the radiance distribution is expected to vary with the viewing azimuth angle. Fig. 6 displays asymmetric scenes, showcasing the variation in upwelling radiance at the top of the atmosphere (TOA) across different azimuth and zenith angles. Here, the depicted neural network was trained by employing $M = 16$ modes, $N_{int} = 10{,}000$ collocation points, and $N_g = N_{top} = 1000$ boundary points for the aerosol and $N_{int} = 100,000$ and $N_g = N_{top} = 10{,}000$ for the cloud, with a configuration of $n = 12$ hidden layers with $d_k = 50$ neurons each. The aerosol model considered was the OPAC continental polluted aerosol from 4 and a cloud set with an optical depth of 7. The SZA is set to 45° for which the radiance distribution should be non-symmetric according to the viewing azimuth angle. We have examined the effectiveness of intensity correction, which resulted in a negative effect for the cloud case, as the PINN training already provides a smoothed solution, where without corrections the error relative to DISORT is 2.74% and with the corrections it is 3.14%.

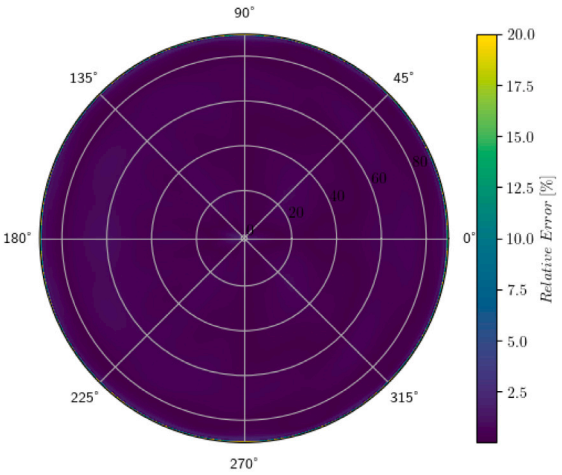
The RTPINN software supports three BRDF models: the Lambertian model, the Ross-Li model, and the Rahman-Pinty-Verstraete (RPV) model [23]. The Lambertian model assumes constant reflectance across all angles, defined by the surface albedo ρ_0. The RPV model is a widely used semi-empirical model for the BRDF of terrestrial surfaces. It is defined by three parameters: ρ_0, controlling the magnitude of reflectance k, characterizing the anisotropy and g representing the asymmetry of the reflectance. The Ross-Li is also used for land surfaces and represents the reflectance as a sum of three kernels corresponding to isotropic scattering, volumetric scattering, and geometric-optics surface scattering, the kernels are summed according to the weights $k_{iso}, k_{geo}, k_{vol}$ which allows the variability and interparetability of the surface models. Fig. 7 shows the radiance distribution at the top of the atmosphere (TOA) for a clear atmosphere using different BRDF models, the parameters used in each BRDF model are described in Table 2. The PINN solution network was trained with $M = 16$, $N_{int} = 10{,}000$, and $N_g = N_{top} = 1000$, using a configuration of $n = 12$ hidden layers with $d_k = 12$ neurons per layer.

For most remote sensing needs, radiance measurement is considered at either at TOA or BOA. However, for some applications, radiance is measured at different altitudes. The functional solution of the PINN is beneficial in these cases, as it provides a functional description of radiance according to both the viewing angles and altitude.

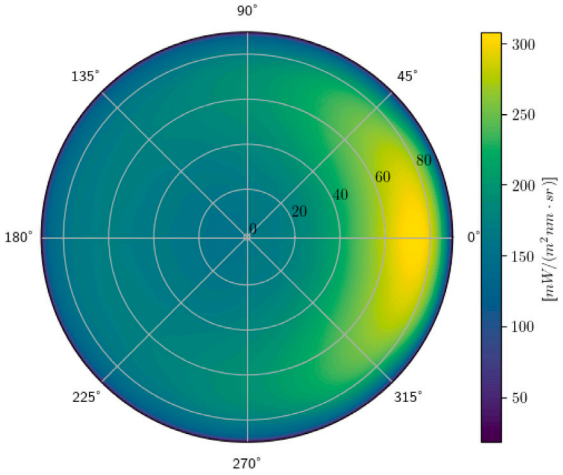
Fig. 8 illustrates how diffuse radiance varies with altitude for continental polluted aerosol in the OPAC parameterization located at an altitude of 2 km. The network was trained by with $M = 1$, $N_{int} =$



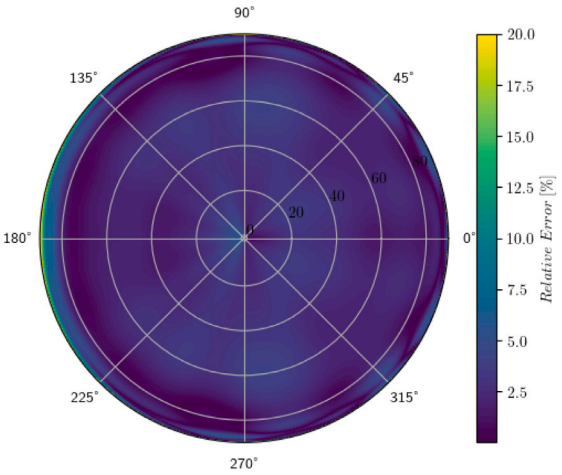
(a) PINN solution for polluted aerosol at the TOA.



(b) Relative error compared to the DISORT solution for aerosol at the TOA without the IMS intensity corrections.



(c) PINN solution for cloud at the TOA.



(d) Relative error compared to the DISORT solution for cloud at the TOA without the IMS intensity corrections

Fig. 6. Comparison of PINN solutions and errors at the top of the atmosphere (TOA) for both aerosol and cloud cases. (a) and (b) show the solution and error for the polluted aerosol scenario, while (c) and (d) present the same for the cloud scenario. The average relative error across viewing angles is 0.74% for the aerosol case. For the cloud scene, the error relative to DISORT is 2.74%.

10,000, and $N_g = N_{top} = 1000$, with a configuration of $n = 8$ hidden layers with $d_k = 30$ neurons each. By examining both the radiance and the error maps, it is evident that the PINN model effectively approximates the radiance transmitted through the aerosol to the bottom of the atmosphere and the radiance reflected at various altitudes, maintaining a bias of less than 1% from the DISORT model output.

6. Summary and discussion

In this work, we demonstrated the capabilities of Physics-Informed Neural Networks (PINNs) for solving atmospheric radiative transfer problems. We compared our RTPINN framework with community-standard atmospheric radiative transfer solvers, generating results within an error of less than 1% for aerosol cases and around 3% for clouds, assuming a sufficient number of collocation and boundary points. We showed that error bounds depend solely on the choice of collocation points, boundary points, number of streams, and the estimated loss.

Our PINN model differs in many ways from most neural network-based architectures as the training process is performed at each inference. While slower than other methods, this approach allows for the creation of accurate results as needed by extending the optimization process without the need for any pre-built database.

This paper presents the first example of using PINNs in atmospheric radiative transfer simulations, which can be easily upgraded for different complex scenarios. In practice, 3D and spherical atmospheres can be modeled using PINNs since the solution process remains similar, with only adjusted residual terms.

There are several advantages to the PINN approach, primarily due to the algorithm's flexibility. Since the solution requires no complete input parameter structure, it allows users to apply settings with limited information by excluding unknowns from the loss sampling. This option

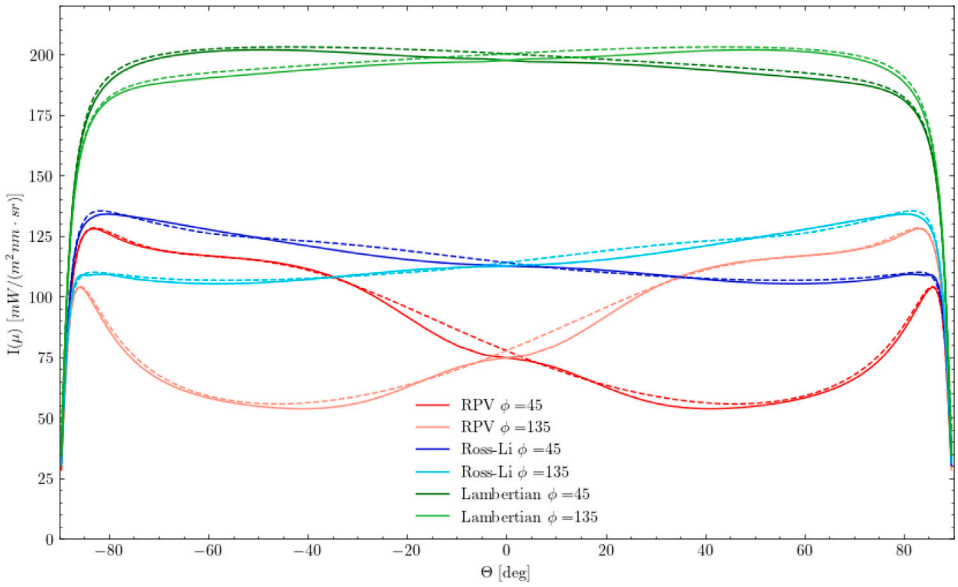


Fig. 7. Radiance Scans at TOA for different BRDF models. The x axis describes the vza centered around the "down" direction. For all simulations the sun was set with a zenith angle of 45° and an azimuth angle of 0°. The PINN solution is described by solid lines and the DISORT solution by dashed lines.

can be further generalized by setting different unknowns as learned parameters during the training process, allowing for the inclusion of optical parameters retrieval and inverse problems. The optimization-based solution process allows for the simultaneous optimization of the atmospheric parameters and the radiance solution.

CRediT authorship contribution statement

Shai Zucker: Writing – original draft, Visualization, Software, Methodology, Formal analysis, Conceptualization. **Dmitry Batenkov:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization. **Michal Segal Rozenhaimer:** Writing – review & editing, Supervision, Methodology, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Shai Zucker reports financial support was provided by Israel Science Foundation. Michal Segal Rozenhaimer reports financial support was provided by NASA Ames Research Center. Michal Segal Rozenhaimer reports financial support was provided by Tel Aviv University. Dmitry Batenkov reports financial support was provided by Israel Science Foundation. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We would like to thank the two anonymous reviewers, whose inputs helped us improve this paper. Dmitry Batenkov and Shai Zucker were supported by the Israel Science Foundation, Grant 1793/20. Michal Segal Rozenhaimer was partially supported by NASA AOS Science Team funding, United States. Part of this work was completed while Michal Segal Rozenhaimer was affiliated with the Department of Geophysics, Porter School of the Environment and Earth Sciences, Tel Aviv University, Israel.

Appendix A. Proof of Theorem 1

Consider the quadrature error of the scattering kernel $D^m(\tau,\mu,\mu')$ and the reflection kernel $\rho^m(\mu,-\mu')$ over the Gauss–Legendre quadrature nodes and weights. The error of the quadrature is defined as follows:

$$E_S[I^m_\theta, D^m](\tau,\mu) = \left(\int_{-1}^{1} D^m(\tau,\mu,\mu') I^m_\theta(\tau,\mu') d\mu' - \sum_{j=1}^{N_S} w'_j D^m(\tau,\mu,\mu'_j) I^m_\theta(\tau,\mu'_j) \right), \tag{A.1}$$

for the scattering kernel, and

$$E_R[I^m_\theta, \rho^m](\tau,\mu) = (1+\delta_{m0}) \left(\int_0^1 \rho^m(\mu,-\mu') I^m_\theta(\tau^*,-\mu') \mu' d\mu' - \sum_{j=1}^{N_R} \tilde{w}_j \rho^m(\mu,-\tilde{\mu}_j) I^m_\theta(\tau^*,-\tilde{\mu}_j) \tilde{\mu}_j \right) \tag{A.2}$$

for the reflection kernel.

Now we consider the RTPINN error. From (10) we obtain

$$\widehat{I^m} = I^m - I^m_\theta = \mu \frac{\partial I^m}{\partial \tau} + \int_{-1}^{1} D^m(\tau,\mu,\mu') I^m(\tau,\mu') d\mu' + Q^m(\tau,\mu) - I^m_\theta.$$

According to our definition of $R^{int,m}_\theta$ (14) we have:

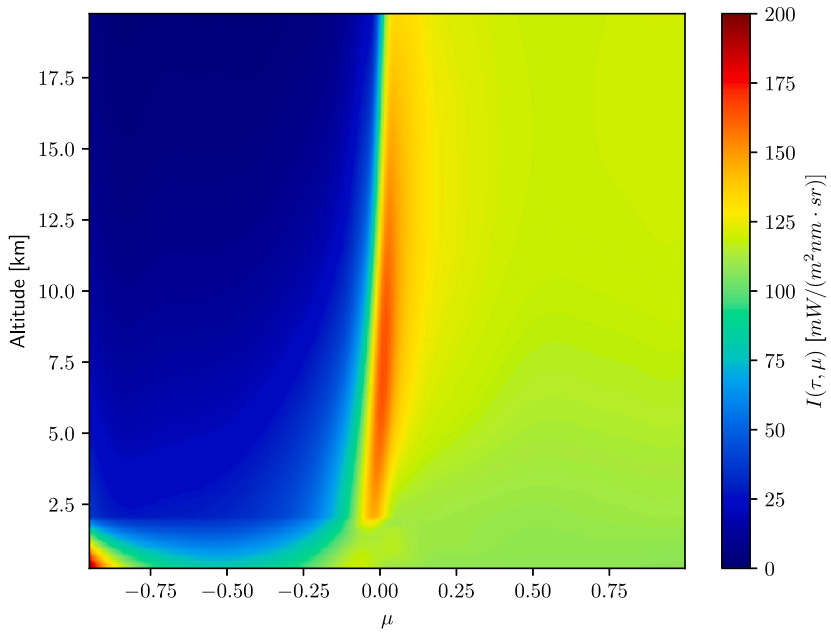
$$I^m_\theta = -R^{int,m}_\theta + \mu \frac{\partial I^m_\theta}{\partial \tau} + \sum_{j=1}^{N_S} w'_j D^m(\tau,\mu,\mu'_j) I^m_\theta(\tau,\mu'_j) + Q^m(\tau,\mu). \tag{A.3}$$

Combining the above equations, we get

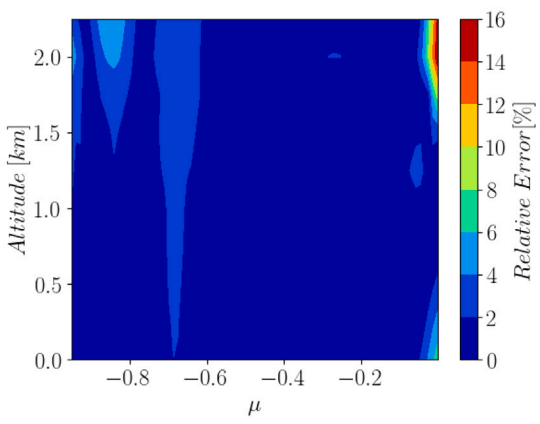
$$\begin{aligned} \widehat{I^m} &= \mu \frac{\partial I^m}{\partial \tau} + \int_{-1}^{1} D^m(\tau,\mu,\mu') I^m(\tau,\mu') d\mu' + Q^m(\tau,\mu) \\ &\quad + R^{int,m}_\theta - \mu \frac{\partial I_\theta}{\partial \tau} - \sum_{j=1}^{N_S} w'_j D^m(\tau,\mu,\mu'_j) I^m_\theta(\tau,\mu'_j) - Q^m(\tau,\mu) \\ &= \mu \frac{\partial \widehat{I^m}}{\partial \tau} + \int_{-1}^{1} D^m(\tau,\mu,\mu') I^m(\tau,\mu') d\mu' + R^{int,m}_\theta - \sum_{j=1}^{N_S} w'_j D^m(\tau,\mu,\mu'_j) I^m_\theta(\tau,\mu'_j). \end{aligned}$$

By adding and subtracting $\int_{-1}^{1} D^m(\tau,\mu,\mu') I^m_\theta(\tau,\mu') d\mu'$, and using (A.1) we further have

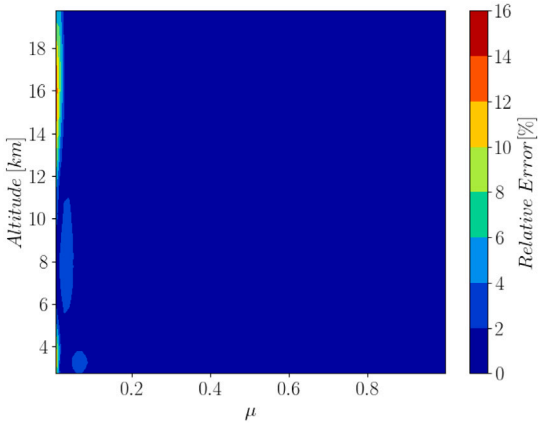
$$\widehat{I^m} = \mu \frac{\partial \widehat{I^m}}{\partial \tau} + \int_{-1}^{1} D^m(\tau,\mu,\mu') \widehat{I^m}(\tau,\mu') d\mu' + R^{int,m}_\theta + E_S[I^m_\theta, D^m].$$



(a) Intensity variation of the PINN solution in altitude



(b) downwelling radiance error map



(c) upwelling radiance error map

Fig. 8. (8(a)) The variation of the radiance intensity with altitude for the PINN model. (8(c),8(b)) The error maps of the PINN model compared to the DISORT model for transmitted and reflected radiance. The error map is focused in the most informative regions of the radiance distribution, downwelling radiance at the altitudes below the aerosol layer and the upwelling radiance above the aerosol layer.

Multiplying both sides by $\widehat{I^m}$ we get

$$\left(\widehat{I^m}(\tau,\mu)\right)^2 = \mu\widehat{I^m}\frac{\partial\widehat{I^m}}{\partial\tau} + \int_{-1}^{1} D^m(\tau,\mu,\mu')\widehat{I^m}(\tau,\mu')\widehat{I^m}(\tau,\mu)d\mu' + R_\theta^{int,m}\widehat{I^m} + E_S[I_\theta^m, D^m]\widehat{I^m}. \tag{A.4}$$

Define the following boundaries:

$$\Gamma^{\tau^*} = \{(\tau^*,\mu) \mid \mu \in [0,1]\}, \tag{A.5}$$

$$\Gamma^{0} = \{(0,\mu) \mid \mu \in [-1,0)\}. \tag{A.6}$$

Next, we integrate (A.4) over S using integration by parts. We obtain:

$$\int_S \widehat{I^m}^2 d\tau d\mu = \frac{1}{2}\left\{\int_{\Gamma_{\tau^*}} \mu\widehat{I^m}^2 d\mu + \int_{\Gamma_0} -\mu\widehat{I^m}^2 d\mu\right\} + J + \int_S \left\{R_\theta^{int}\widehat{I^m} + E_S[I_\theta^m, D^m]\widehat{I^m}\right\} d\tau d\mu, \tag{A.7}$$

where

$$J = \int_S \int_{-1}^{1} D^m(\tau,\mu,\mu')\widehat{I^m}(\tau,\mu')\widehat{I^m}(\tau,\mu)d\mu' d\tau d\mu.$$

Using the Cauchy–Schwarz inequality, we get

$$\begin{aligned} |J| &\le \int_0^{\tau^*}\int_{-1}^{1}\int_{-1}^{1} |D^m(\tau,\mu,\mu')|(\widehat{I^m}(\tau,\mu))^2 d\mu' d\mu d\tau \\ &= \int_0^{\tau^*}\int_{-1}^{1} (\widehat{I^m}(\tau,\mu))^2 \int_{-1}^{1} |D^m(\tau,\mu,\mu')| d\mu' d\mu d\tau \\ &\le M_S \int_S \widehat{I^m}^2 d\tau d\mu. \end{aligned}$$

Since $|\mu| \le 1$, the integration over the boundary sets in (A.7) can be bounded:

$$(1 - M_S)\int_S \widehat{I^m}^2 d\tau d\mu \le \frac{1}{2}\left\{\int_{\Gamma_{\tau^*}} \widehat{I^m}^2 d\mu + \int_{\Gamma_0} \widehat{I^m}^2 d\mu\right\} + \left|\int_S (R_\theta^{int,m} + E_S[I_\theta^m, D^m])\widehat{I^m} d\tau d\mu\right|. \tag{A.8}$$

Consider the following inequality, commonly denoted as the ε - Young's inequality:

Lemma 1. *Let $a, b > 0$, then for any $\varepsilon > 0$*

$$ab \le \frac{a^2}{2\varepsilon} + \frac{\varepsilon b^2}{2}.$$

Proof. See Appendix B. □

Now let there be $0 < \varepsilon_S < 1 - M_S$, then by Lemma 1 we have

$$\left| \int_S R_\theta^{int,m} \widehat{I^m} ds \right| \le \int_S |R_\theta^{int,m}| \, \| \, \widehat{I^m}| ds \le \int_S \left\{ \frac{(R_\theta^{int,m})^2}{2\varepsilon_S} + \frac{\varepsilon_S \widehat{I^m}^2}{2} \right\} ds \quad \text{(A.9)}$$

and

$$\left| \int_S E_S[I_\theta^m, D^m] \widehat{I^m} ds \right| \le \int_S |E_S[I_\theta^m, D^m]| \, \| \, \widehat{I^m}| ds \le \int_S \left\{ \frac{E_S[I_\theta^m, D^m]^2}{2\varepsilon_S} + \frac{\varepsilon_S \widehat{I^m}^2}{2} \right\} ds. \quad \text{(A.10)}$$

Combining (A.8) with the above inequalities, we get

$$(1 - M_S - \varepsilon_S) \int_S \widehat{I^m}^2 d\tau d\mu \le \frac{1}{2} \left\{ \int_{\Gamma_{\tau^*}} \widehat{I^m}^2 d\mu + \int_{\Gamma_0} \widehat{I^m}^2 d\mu \right\} + \frac{1}{2\varepsilon_S} \left(\int_S ((R_\theta^{int,m})^2 + E_S[I_\theta^m, D^m]^2) d\tau d\mu \right). \quad \text{(A.11)}$$

Our next task is to express the terms in the above equation through the integrated network residuals and the quadrature errors (A.1) and (A.2). First, notice that by (15) we have

$$\int_{\Gamma_0} \widehat{I^m}^2 d\mu = \int_{\Gamma^0} (R_\theta^{top,m})^2 d\mu.$$

Continuing with the Γ_{τ^*} term, we put $\tau = \tau^*$, and using (12), (13) and (16) we write

$$\widehat{I^m} = I^m - I_\theta^m = (1 + \delta_{m0}) \int_0^1 \rho^m(\mu, -\mu') I^m(\tau^*, -\mu') \mu' d\mu' - R_\theta^{g,m} - (1 + \delta_{m0}) \sum_{j=1}^{N_R} \tilde{w}_j \rho^m(\mu, -\tilde{\mu}_j) I_\theta^m(\tau^*, -\tilde{\mu}_j) \tilde{\mu}_j.$$

Adding and subtracting the term $(1 + \delta_{m0}) \int_0^1 \rho^m(\mu, -\mu') I_\theta^m(\tau^*, -\mu') \mu' d\mu'$ we get:

$$\widehat{I^m} = (1 + \delta_{m0}) \int_0^1 \rho^m(\mu, -\mu') \widehat{I^m}(\tau^*, -\mu') \mu' d\mu' - R_\theta^{g,m} + E_R[I_\theta^m, \rho^m].$$

Multiplying both sides with $\widehat{I^m}(\tau^*, \mu)$ and integrating over Γ_{τ^*} we obtain

$$\int_{\Gamma_{\tau^*}} \widehat{I^m}^2 d\mu \le \int_{\Gamma_{\tau^*}} (1 + \delta_{m0}) \int_0^1 \left| \rho^m(\mu, -\mu') \widehat{I^m}(\tau^*, -\mu') \widehat{I^m}(\tau^*, -\mu) \right| \mu' d\mu' d\mu + \int_{\Gamma_{\tau^*}} \left| \widehat{I^m} E_R[I_\theta^m, \rho^m] - \widehat{I^m} R_\theta^{g,m} \right| d\mu. \quad \text{(A.12)}$$

Similar manipulation as the one done for the J term gives the following bound for the reflection kernel:

$$\int_{\Gamma_{\tau^*}} (1 + \delta_{m0}) \int_0^1 \left| \rho^m(\mu, -\mu') \widehat{I^m}(\tau^*, -\mu') \widehat{I^m}(\tau^*, -\mu) \right| \mu' d\mu' d\mu \le M_R \int_{\Gamma_{\tau^*}} (\widehat{I^m})^2 d\mu. \quad \text{(A.13)}$$

Applying Lemma 1 for some $0 < \varepsilon_R < 1 - M_R$, we get:

$$\left| \int_{\Gamma_{\tau^*}} R_\theta^{g,m} \widehat{I^m} d\mu \right| \le \int_{\Gamma_{\tau^*}} \left\{ \frac{(R_\theta^{g,m})^2}{2\varepsilon_R} + \frac{\varepsilon_R \widehat{I^m}^2}{2} \right\} d\mu, \quad \text{(A.14)}$$

and

$$\left| \int_{\Gamma_{\tau^*}} E_R[I_\theta^m, D^m] \widehat{I^m} d\mu \right| \le \int_{\Gamma_{\tau^*}} \left\{ \frac{E_R[I_\theta^m, \rho^m]^2}{2\varepsilon_R} + \frac{\varepsilon_R \widehat{I^m}^2}{2} \right\} d\mu. \quad \text{(A.15)}$$

Combining (A.12), (A.13), (A.14) and (A.15), we get:

$$(1 - M_R - \varepsilon_R) \int_{\Gamma_{\tau*}} \widehat{I^m}^2 d\mu \le \int_{\Gamma_{\tau^*}} \frac{E_R[I_\theta^m, \rho^m]^2}{2\varepsilon_R} d\mu + \int_{\Gamma_{\tau^*}} \frac{(R_\theta^{g,m})^2}{2\varepsilon_R} d\mu.$$

Returning to the central inequality (A.11), we get:

$$2(1 - M_S - \varepsilon_S) \int_S (\widehat{I^m})^2 d\tau d\mu \le \int_{\Gamma_0} (R_\theta^{top,m})^2 d\mu + \frac{1}{2\varepsilon_R(1 - M_R - \varepsilon_R)} \int_{\Gamma_{\tau^*}} \left\{ E_R[I_\theta^m, \rho^m]^2 + (R_\theta^{g,m})^2 \right\} d\mu + \frac{1}{\varepsilon_S} \int_S \left\{ (R_\theta^{int,m})^2 + (E_S[I_\theta^m, D^m])^2 \right\} d\tau d\mu. \quad \text{(A.16)}$$

The loss error terms may be viewed as quasi Monte-Carlo quadrature estimates of the error integrated across the domain. The bound on the integration error is determined by the discrepancy of the nodes where the integration occurs and the variation of the function integrated [24]. Initially, we will examine the discrepancy of the quadrature nodes. We adhere to the notation presented in [25] for defining Star Discrepancy, which assesses the distribution of the quadrature nodes within the domain.

Definition 1 (*Star Discrepancy*)**.** Let $\{x_1, \dots, x_n\} \subset [0,1]^d$ be a sequence of points. The *star discrepancy* $D^*(x_1, \dots, x_n)$ of this sequence is defined as:

$$D^*(x_1, \dots, x_n) = \sup_{R \subset [0,1]^d; R = \prod_{i=1}^d [0, b_i]} \left| \frac{|\{x_i \mid x_i \in R\}|}{n} - \lambda(R) \right|,$$

where R is a d-dimensional rectangle anchored at the origin, with $R = \prod_{i=1}^d [0, b_i]$, and λ denotes the Lebesgue measure of R.

In our method, we utilize Sobol points as quadrature nodes, a popular selection originally described in [26]. The Star Discrepancy of these points is notably bounded by $O(\frac{\log^d n}{n})$, where d represents the dimension of the integration domain. In Monte Carlo theory, the variation of the function being integrated is evaluated using the Hardy-Krause variation, which is defined as follows in [24]:

Definition 2. Let $f \in C^d([0,1]^d)$. Then the *variation* $V(f)$ of f in the sense of Hardy and Krause is defined in the following way:

$$V_{HK}(f) := \sum_{k=1}^{d} \sum_{1 \le i_1 < \cdots < i_k \le d} V^{(k)}\left(f; i_1, \dots, i_k\right)$$

$$V^{(k)}\left(f; i_1, \dots, i_k\right) := \int_0^1 \cdots \int_0^1 \left| \frac{\partial^k f|_{i_1, \dots, i_k}}{\partial x_{i_1} \cdots \partial x_{i_k}} \right| \mathrm{d}x_{i_1} \cdots \mathrm{d}x_{i_k}$$

$$f|_{i_1, \dots, i_k}(x_{i_1}, \dots, x_{i_k}) := f(\xi), \text{ where } \xi_i = \begin{cases} x_i & \text{for } i \in i_1, \dots, i_k \\ 0 & \text{else.} \end{cases}$$

Using both the Star Discrepancy and the Hardy-Krause variation we can bound the error of the integral estimate using the Koksma-Hlawka inequality.

Theorem 2 (*Koksma-Hlawka Inequality*)**.** *Let $f : [0,1]^d \to \mathbb{R}$. If $V_{HK}(f) < \infty$, then for any set of points $\{x_1, \dots, x_n\} \subset [0,1]^d$, the following inequality holds:*

$$\left| \frac{1}{n} \sum_{i=1}^{n} f(x_i) - \int_{[0,1]^d} f(x) \, dx \right| \le V_{HK}(f) D^*(x_1, \dots, x_n).$$

As the training points are the Sobol quadrature points, we realize that the training errors $\mathcal{E}^{top,m}, \mathcal{E}^{g,m}, \mathcal{E}^{int,m}$ given in (18) are the quasi

Monte-Carlo quadrature errors for $(R_\theta^{g,m})^2, (R_\theta^{top,m})^2$, and $(R_\theta^{int,m})^2$, respectively. Hence by the Koksma-Hlawka inequality we obtain the following estimates:

$$\int_{\Gamma_{\tau^*}} (R_\theta^{g,m})^2 d\mu \le \mathcal{E}^{g,m} + V_{HK}\left((R_\theta^{g,m})^2\right) \frac{\log N_g}{N_g} \tag{A.17}$$

$$\int_{\Gamma_0} (R_\theta^{top,m})^2 d\mu \le \mathcal{E}^{top,m} + V_{HK}\left((R_\theta^{top,m})^2\right) \frac{\log N_{top}}{N_{top}}, \tag{A.18}$$

$$\int_S (R_\theta^{int,m})^2 d\mu d\tau \le \mathcal{E}^{int,m} + V_{HK}\left((R_\theta^{int,m})^2\right) \frac{\log {N_{int}}^2}{N_{int}}. \tag{A.19}$$

Next, we consider the following theorem for the quadrature error of single-variable integration, cf. [27, page 146]

Theorem 3. *Let $f \in C^{2N}([a,b])$, and define the 1D Gaussian quadrature error over N points as*

$$E = \int_a^b f(x)\,dx - \sum_{i=1}^{N} w_i f(x_i),$$

where w_i are the weights and x_i are the nodes used in the Gaussian quadrature. Then, the error E is given by

$$E = \frac{(b-a)^{2N+1}(N!)^4}{(2N+1)[2N!]^3} f^{(2N)}(\xi),$$

for some ξ in the interval (a,b).

For readability, we introduce the following notation:

$$S_N = \frac{(N!)^4}{(2N!)^3} \implies \ln S_N = 4\ln(N!) - 3\ln(2N!).$$

Using Stirling's approximation, $\ln(N!) \approx N\ln(N) - N + O(\ln N)$, we can express $\ln S_N$ as:

$$\ln S_N = 4\{N\ln(N) - N + O(\ln N)\} - 3\{2N\ln(2N) - 2N + O(\ln(2N))\},$$

which simplifies to:

$$\ln S_N = -2N\ln(N) + 2N + O(\ln N).$$

Given that $\ln S_N$ varies as $-2N\ln(N) + 2N$ up to $O(\ln N)$, we can approximate S_N with the bound:

$$S_N \le \tilde{M} N \left(\frac{e}{N}\right)^{2N},$$

where $\tilde{M}$ is a constant.

Upon applying Theorem 3 and the bound on S_N to the terms $E_S[I_\theta^m, D^m]$ and $E_R[I_\theta^m, \rho^m]$, we derive the following bounds:

$$\int_S E_S[I_\theta^m, D^m]^2 d\tau d\mu \le \bar{C}_S \left(\frac{2e}{N_S}\right)^{N_S}, \tag{A.20}$$

$$\int_{-1}^{0} E_R[I_\theta^m, \rho^m]^2 d\mu \le \bar{C}_R \left(\frac{e}{N_R}\right)^{4N_R} \tag{A.21}$$

where $\bar{C}_S = C(|\tau^*|, \|I_\theta^m\|, \|D^m\|)$ and $\bar{C}_R = C(\|I_\theta^m\|_{\Gamma^{\tau^*}}, \|\rho^m\|_{\Gamma^{\tau^*}})$ depend on the norms of the network prediction function and the initial data.

By assigning all bounds back into (A.16), we obtain the following estimate for the error of the network prediction:

$$\mathcal{E}_G^m = \int_S \widehat{I^m}^2 d\tau d\mu \le C\left[\mathcal{E}^{top,m} + \mathcal{E}^{g,m} + \mathcal{E}^{int,m} + \frac{\log N_g}{N_g} + \frac{\log N_{top}}{N_{top}} + \frac{\log^2 N_{int}}{N_{int}} + \left(\frac{e}{N_S}\right)^{-4N_S} + \left(\frac{e}{N_R}\right)^{-4N_R}\right]. \tag{A.22}$$

The constant C is defined as:

$$C = \frac{1}{2(1-M_S-\varepsilon_S)} \max\left\{\frac{1}{\varepsilon_S}\bar{C}_S, \frac{1}{2\varepsilon_R(1-M_R-\varepsilon_R)}\bar{C}_R, V_{HK}\left((R_\theta^{top,m})^2\right), \frac{1}{\varepsilon_R(1-M_R-\varepsilon_R)} V_{HK}\left((R_\theta^{g,m})^2\right), \frac{1}{\varepsilon_S} V_{HK}\left((R_\theta^{int,m})^2\right), \frac{1}{\varepsilon_S}\right\}. \tag{A.23}$$

Appendix B. ε- Young's inequality

Here we prove Lemma 1 for completeness.

Proof of Lemma 1. Let $a, b > 0$, since a square is always non-negative we get:

$$0 \le (a-b)^2$$

$$0 \le a^2 - 2ab + b^2$$

$$2ab \le a^2 + b^2$$

$$ab \le \frac{a^2}{2} + \frac{b^2}{2}$$

Now let there be $\varepsilon > 0$, by applying the last inequality for $a' = \frac{a}{\sqrt{\varepsilon}}$ and $b' = \sqrt{\varepsilon} b$ we get

$$ab \le \frac{a^2}{2\varepsilon} + \frac{\varepsilon b^2}{2}. \quad \square$$

Data availability

Data will be made available on request.

References

[1] Core Writing Team, Lee H, Romero J, editors. Climate Change 2023: Synthesis Report. Contribution of Working Groups I, II and III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change. Geneva, Switzerland: IPCC; 2023, p. 35–115. http://dx.doi.org/10.59327/IPCC/AR6-9789291691647.

[2] Bilal M, Nichol JE, Bleiweiss MP, Dubois D. A simplified high resolution MODIS Aerosol Retrieval Algorithm (SARA) for use over mixed surfaces. Remote Sens Environ 2013;136:135–45. http://dx.doi.org/10.1016/j.rse.2013.04.014, URL https://www.sciencedirect.com/science/article/pii/S0034425713001417.

[3] Lu S, Landgraf J, Fu G, van Diedenhoven B, Wu L, Rusli SP, et al. Simultaneous retrieval of trace gases, aerosols, and cirrus using RemoTAP—The global orbit ensemble study for the CO2M mission. Front Remote Sens 2022;3. http://dx.doi.org/10.3389/frsen.2022.914378, URL https://www.frontiersin.org/articles/10.3389/frsen.2022.914378.

[4] Segal-Rozenhaimer M, Miller DJ, Knobelspiesse K, Redemann J, Cairns B, Alexandrov MD. Development of neural network retrievals of liquid cloud properties from multi-angle polarimetric observations. J Quant Spectrosc Radiat Transfer 2018;220:39–51. http://dx.doi.org/10.1016/j.jqsrt.2018.08.030, URL https://www.sciencedirect.com/science/article/pii/S0022407318304667.

[5] Stamnes K, Tsay S-C, Wiscombe W, Jayaweera K. Numerically stable algorithm for discrete-ordinate-method radiative transfer in multiple scattering and emitting layered media. Appl Opt 1988;27(12):2502–9. http://dx.doi.org/10.1364/AO.27.002502, URL https://opg.optica.org/ao/abstract.cfm?URI=ao-27-12-2502.

[6] Mayer B. Radiative transfer in the cloudy atmosphere. EPJ Web Conf 2009;1:75–99. http://dx.doi.org/10.1140/epjconf/e2009-00912-1.

[7] Xu F, Davis AB, West RA, Esposito LW. Markov chain formalism for polarized light transfer in plane-parallel atmospheres, with numerical comparison to the Monte Carlo method. Opt Express 2011;19(2):946–67. http://dx.doi.org/10.1364/OE.19.000946, URL https://opg.optica.org/oe/abstract.cfm?URI=oe-19-2-946.

[8] Xu F, Davis AB, Diner DJ. Markov chain formalism for generalized radiative transfer in a plane-parallel medium, accounting for polarization. J Quant Spectrosc Radiat Transfer 2016;184:14–26. http://dx.doi.org/10.1016/j.jqsrt.2016.06.004, URL https://www.sciencedirect.com/science/article/pii/S0022407316301753.

[9] Mishra S, Molinaro R. Physics informed neural networks for simulating radiative transfer. Journal of Quantitative Spectroscopy and Radiative Transfer 2021;270:107705. http://dx.doi.org/10.1016/j.jqsrt.2021.107705, URL https://www.sciencedirect.com/science/article/pii/S0022407321001989.

[10] Riganti R, Negro LD. Auxiliary physics-informed neural networks for forward, inverse, and coupled radiative transfer problems. Applied Physics Letters 2023;123(17):171104. http://dx.doi.org/10.1063/5.0167155, arXiv:https://pubs.aip.org/aip/apl/article-pdf/doi/10.1063/5.0167155/18187189/171104_1_5.0167155.pdf.

[11] Stamnes K, Thomas GE, Stamnes JJ. Radiative Transfer in the Atmosphere and Ocean. 2nd ed.. Cambridge University Press; 2017.

[12] Raissi M, Perdikaris P, Karniadakis G. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J Comput Phys 2019;378:686–707. http://dx.doi.org/10.1016/j.jcp.2018.10.045, URL https://www.sciencedirect.com/science/article/pii/S0021999118307125.

[13] Lu L, Jin P, Pang G, Zhang Z, Karniadakis GE. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. Nature Machine Intelligence 2019;3:218–29, URL https://api.semanticscholar.org/CorpusID:233822586.

[14] Lu Y, Wang L, Xu W. Solving multiscale steady radiative transfer equation using neural networks with uniform stability. Res Math Sci 2022;9(3):45. http://dx.doi.org/10.1007/s40687-022-00345-z.

[15] Raynaud G, Houde S, Gosselin FP. ModalPINN: An extension of physics-informed Neural Networks with enforced truncated Fourier decomposition for periodic flow reconstruction using a limited number of imperfect sensors. J Comput Phys 2022;464:111271. http://dx.doi.org/10.1016/j.jcp.2022.111271, URL https://www.sciencedirect.com/science/article/pii/S0021999122003333.

[16] Wang S, Sankaran S, Wang H, Perdikaris P. An expert's guide to training physics-informed neural networks. 2023, arXiv:2308.08468.

[17] Wiscombe W. The Delta–M method: Rapid yet accurate radiative flux calculations for strongly asymmetric phase functions. J Atmos Sci 1977;34(9):1408–22.

[18] Nakajima T, Tanaka M. Algorithms for radiative intensity calculations in moderately thick atmospheres using a truncation approximation. J Quant Spectrosc Radiat Transfer 1988;40(1):51–69. http://dx.doi.org/10.1016/0022-4073(88)90031-3, URL https://www.sciencedirect.com/science/article/pii/0022407388900313.

[19] Emde C, Buras-Schnell R, Kylling A, Mayer B, Gasteiger J, Hamann U, et al. The libRadtran software package for radiative transfer calculations (version 2.0.1). Geosci Model Dev 2016;9(5):1647–72. http://dx.doi.org/10.5194/gmd-9-1647-2016, URL https://gmd.copernicus.org/articles/9/1647/2016/.

[20] Hess M, Koepke P, Schult I. Optical properties of aerosols and clouds: The software package OPAC. Bull Am Meteorol Soc 1998;79(5):831–44.

[21] Models of aerosols, clouds, and precipitation for atmospheric propagation studies. In: AGARD. 1990.

[22] Liu DC, Nocedal J. On the limited memory BFGS method for large scale optimization. Math Program 1989;45(1):503–28. http://dx.doi.org/10.1007/BF01589116.

[23] Rahman H, Pinty B, Verstraete MM. Coupled surface-atmosphere reflectance (CSAR) model: 2. Semiempirical surface model usable with NOAA advanced very high resolution radiometer data. J Geophys Res: Atmos 1993;98(D11):20791–801.

[24] Caflisch RE. Monte Carlo and quasi-Monte Carlo methods. Acta Numer 1998;7:1–49. http://dx.doi.org/10.1017/S0962492900002804.

[25] Niederreiter H. Low-discrepancy and low-dispersion sequences. J Number Theory 1988;30(1):51–70. http://dx.doi.org/10.1016/0022-314X(88)90025-X, URL https://www.sciencedirect.com/science/article/pii/0022314X8890025X.

[26] Sobol' I. On the distribution of points in a cube and the approximate evaluation of integrals. USSR Comput Math Math Phys 1967;7(4):86–112. http://dx.doi.org/10.1016/0041-5553(67)90144-9, URL https://www.sciencedirect.com/science/article/pii/0041555367901449.

[27] Kahaner D, Moler CB, Nash SG, Forsythe GEGE. Numerical methods and software / David Kahaner, Cleve Moler, Stephen Nash. In: Numerical methods and software. Prentice hall series in computational mathematics, Englewood Cliffs, N.J.: Prentice-Hall; 1989.