

From Physics-Based Parameterization to Data-Driven Emulation of Planetary Boundary Layer Turbulence in BAM- 1D

Antonio Vicente Pereira Neto

`crystone42@gmail.com`

Federal University of Rio de Janeiro

Enver Manuel Amador

National Institute for Space Research

Ramirez Gutierrez

National Institute for Space Research

5. Gutemberg Borges França

Federal University of Rio de Janeiro

4. Haroldo Fraga de Campos Velho

Instituto Nacional de Pesquisas Espaciais

José Cristiano Pereira

Federal University of Rio de Janeiro

Research Article

Keywords: planetary boundary layer, turbulence parameterization, surrogate modeling, neural network, BAM-1D, Holtslag–Boville scheme

Posted Date: October 19th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10620663/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Title

From Physics-Based Parameterization to Data-Driven Emulation of Planetary Boundary Layer Turbulence in BAM-1D.

Author's information

1. Antonio Vicente Pereira Neto
Applied Meteorology Laboratory, Federal University of Rio de Janeiro, Brazil.
e-mail: cristone42@gmail.com
ORCID: 0009-0006-5477-9293
2. Enver Manuel Amador Ramirez Gutierrez
Division of Weather and Climate Predictions, National Institute for Space Research, São José dos Campos, Brazil.
e-mail: enver.ramirez@inpe.br
ORCID: 0000-0002-4678-6382
3. Denis Magalhães de Almeida Eiras
São José dos Campos, Brazil.
e-mail: denis.eiras@inpe.br
4. Haroldo Fraga de Campos Velho
Coordination of Applied Research and Technological Development, National Institute for Space Research, São José dos Campos, Brazil.
e-mail: haroldo.camposvelho@inpe.br
ORCID: 0000-0003-4968-5330
5. Gutemberg Borges França
Applied Meteorology Laboratory, Federal University of Rio de Janeiro, Brazil.
e-mail: gutemberg@lma.ufrj.br
ORCID: 0000-0002-9661-1652
6. José Cristiano Pereira
National Laboratory for Scientific Computation.
e-mail: jpereira@lncc.br and josecristiano.pereira@ucp.br
ORCID: 0000-0002-2329-0560

Data Availability Statement: The data presented in this study is available on request from the corresponding author. The data are not publicly available due to be proprietary data of the National Institute for Space Research (INPE).

Acknowledgments: The authors are grateful to the National Institute for Space Research (INPE) for providing the data used in this research. Authors HFCV and GBF thank the National Council for Scientific and Technological Development (CNPq, Brazil) for the research activity project (Grants: 315349/2023-9 and , respectively).

Conflicts of Interest: The authors declare no conflict of interest.

Abstract

This study develops and evaluates a neural-network-based surrogate model for the Holtslag–Boville (HB) planetary boundary-layer turbulence parameterization implemented in BAM-1D, the one-dimensional version of the Brazilian Global Atmospheric Model. Rather than inferring turbulence directly from observations, the proposed approach emulates the nonlinear input–output relationships produced by the original HB scheme. A Multi-Layer Perceptron neural network (MLP-NN) was trained using approximately 174,000 atmospheric profiles (4.88 million vertical-level samples) from two years of BAM-1D simulations. The dataset represents the complete 28-level atmospheric column sampled every 6 min, while an independent 48-day dataset spanning different seasons was reserved for evaluation. Four independent MLP-NN models were trained to emulate the principal turbulence-related quantities generated by the HB scheme. Although counter-gradient terms showed lower coefficients of determination because of their smaller magnitudes and higher variability, their absolute errors and biases remained low. Online BAM-1D experiments demonstrated that replacing the HB parameterization preserved the temporal evolution of temperature, wind, and precipitation without numerical instabilities. Although the original HB scheme remains faster in the one-dimensional configuration, neural-network inference is well suited for future vectorized and parallel implementations. These results demonstrate that MLP-based surrogate modeling reproduces a physically consistent turbulence closure with high fidelity and numerical stability, providing a promising pathway toward hybrid physics–machine-learning parameterizations.

Suggested keywords: planetary boundary layer; turbulence parameterization; surrogate modeling; neural network; BAM-1D; Holtslag–Boville scheme.

1. Introduction

Numerical Weather Prediction (NWP) models simulate atmospheric dynamics by solving the Navier–Stokes equations while representing unresolved subgrid-scale processes through physical parameterizations (Kalnay, 2003). Among these processes, turbulence is particularly important because it cannot be explicitly resolved at typical model grid resolutions and therefore must be parameterized (Kalnay 2003; Stensrud 2007). Turbulence is a persistent and fundamental feature of the Planetary Boundary Layer (PBL), the lowest part of the atmosphere, whose depth and structure respond rapidly to surface heating, radiative cooling, wind shear, buoyancy, and atmospheric stability (Stull 1988; Garratt 1994).

Turbulent mixing within the PBL is a primary mechanism controlling the exchange of momentum, heat, moisture, and atmospheric constituents between the Earth's surface and the free atmosphere. These exchanges regulate near-surface temperature, humidity, wind, cloud formation, and pollutant dispersion, making an accurate representation of turbulence essential for numerical weather prediction and climate models (Stull 1988; Garratt 1994; Arya 2001). Turbulent mixing is driven by surface heat fluxes associated with daytime heating and nocturnal cooling. At night, rapid surface cooling suppresses turbulent kinetic energy, producing a stable boundary layer, whereas daytime surface heating enhances buoyancy-driven turbulence, promoting vigorous vertical mixing and the development of a convective boundary layer (Stull 1988).

The stability of the PBL is commonly diagnosed using the gradient Richardson number, which quantifies the balance between buoyancy and wind shear. Another widely used stability metric is the Monin–Obukhov length, which characterizes the relative importance of mechanical and buoyancy production of turbulence (Monin and Obukhov 1954; Stull 1988). In Reynolds-averaged atmospheric models, turbulent motions are represented by decomposing the flow into mean and fluctuating components. The resulting turbulent fluxes cannot be resolved explicitly and therefore require closure assumptions, forming the basis of most turbulence parameterizations (Reynolds 1895; Mellor and Yamada, 1982; Holtslag and Boville 1993).

A variety of turbulence parameterization approaches have been developed using different closure methods (Stull 1988; Campos Velho 2010). These approaches range from first-order schemes (e.g., Degrazia et al.

2000; Holtslag and Boville 1993), which relate turbulent fluxes directly to gradients of mean fluid quantities, to more complex higher-order closures (e.g., Mellor and Yamada 1982), where higher-order turbulent fluxes are parameterized. Among the first-order closure schemes implemented in BAM (The Brazilian Global Atmospheric Model), the Holtslag and Boville (HB) parameterization is used to represent turbulent transport within the PBL. BAM is a three-dimensional atmospheric model developed at the Brazilian National Institute for Space Research (INPE) for numerical weather prediction and climate applications (Figueroa et al. 2016).

Despite their usefulness and long operational history, standard (physics-based or fitting-curve) turbulence parameterizations have limitations, especially in regimes involving strong stability, rapid transitions, or nonlocal mixing (Stull 1988; Holtslag and Boville 1993; Mellor and Yamada 1982). These challenges have motivated the development of hybrid or data-driven methods that exploit high-resolution simulations (e.g., LES, DNS) and advances in machine learning (ML).

In recent years, machine learning (ML) has emerged as a promising approach for representing unresolved atmospheric processes in numerical weather prediction and climate models. Early studies demonstrated the potential of data-driven methods for atmospheric modeling and physical parameterizations (Dueben and Bauer 2018; Scher 2018; Rasp et al. 2018), while subsequent investigations showed that neural-network emulators can be successfully integrated into prognostic atmospheric models without compromising numerical stability (Brenowitz and Bretherton 2018; Schultz et al. 2021).

More recently, the rapid development of data-driven weather prediction systems, including GraphCast (Lam et al. 2023), NeuralGCM (Kochkov et al. 2024), and the Anemoi framework (Dramschi et al. 2024), has demonstrated that machine learning can accurately reproduce large-scale atmospheric dynamics and provide skillful medium-range weather forecasts. These developments highlight the increasing maturity of artificial intelligence techniques for Earth-system modeling and support the transition from purely research-oriented applications toward operational machine-learning forecasting systems that complement traditional physics-based numerical models.

In parallel, ML techniques have also been applied to emulate specific physical parameterizations, including

planetary boundary-layer turbulence, turbulent fluxes, vertical diffusivities, and planetary boundary-layer height (Yuval et al. 2021; Canché-Cab et al. 2024; Pereira Neto et al. 2026; Stapleton et al. 2025). Together, these studies demonstrate the potential of ML to reproduce complex turbulence processes and support the development of hybrid atmospheric models.

Motivated by these advances, the objective of this study is to develop, implement, and evaluate a machine-learning-based surrogate model for the HB planetary boundary layer turbulence parameterization in BAM-1D. A Multi-Layer Perceptron neural network (MLP-NN) is trained to emulate the nonlinear input–output relationships of the original HB scheme implemented in BAM-1D, the one-dimensional column version of the Brazilian Global Atmospheric Model (BAM) (Figueroa et al. 2016). Rather than predicting turbulence directly from observations, the proposed model learns the nonlinear relationships between atmospheric state variables and the turbulence quantities produced by the original parameterization.

The proposed methodology is evaluated through comparisons of turbulent diffusivities, counter-gradient terms, vertical profiles, temporal evolution of prognostic variables, and computational performance, allowing assessment of both the accuracy and numerical stability of the neural-network emulator during online integration into BAM-1D.

The remainder of this paper is organized as follows. Section 2 describes the materials and methods, including the BAM-1D model, the GoAmazon 2014/15 dataset, the neural-network architecture, and the proposed machine-learning methodology. Section 3 presents the results of the offline and online evaluations of the neural-network emulator. Section 4 discusses the main findings, their implications, and the limitations of the proposed approach and Finally, summarizes the main conclusions and outlines directions for future research.

2. Material and Methods

2.1. Dataset and Study Area

The atmospheric dataset used in this study was obtained from the GoAmazon 2014/15 experiment, conducted at Fazenda Exata near Manacapuru, Amazonas, Brazil ($03^{\circ}12'36''$ S, $60^{\circ}36'00''$ W; 60 m above sea level), approximately 66 km west of Manaus (Martin et al. 2016), as shown in Figure 1. The site is in central Amazonia, one of the most active convective regions in the tropics, where strong land–atmosphere interactions and intense boundary-layer processes play a central role in atmospheric variability.

Central Amazonia has a humid tropical climate, with annual precipitation exceeding 2,300 mm and a marked seasonal rainfall cycle associated with the South American monsoon system (Marengo et al. 2001; Fisch et al. 2004). These seasonal variations encompass highly convective wet-season conditions and comparatively more stable dry-season regimes (Zhuang et al. 2017), providing a broad range of atmospheric states for evaluating turbulence parameterizations.

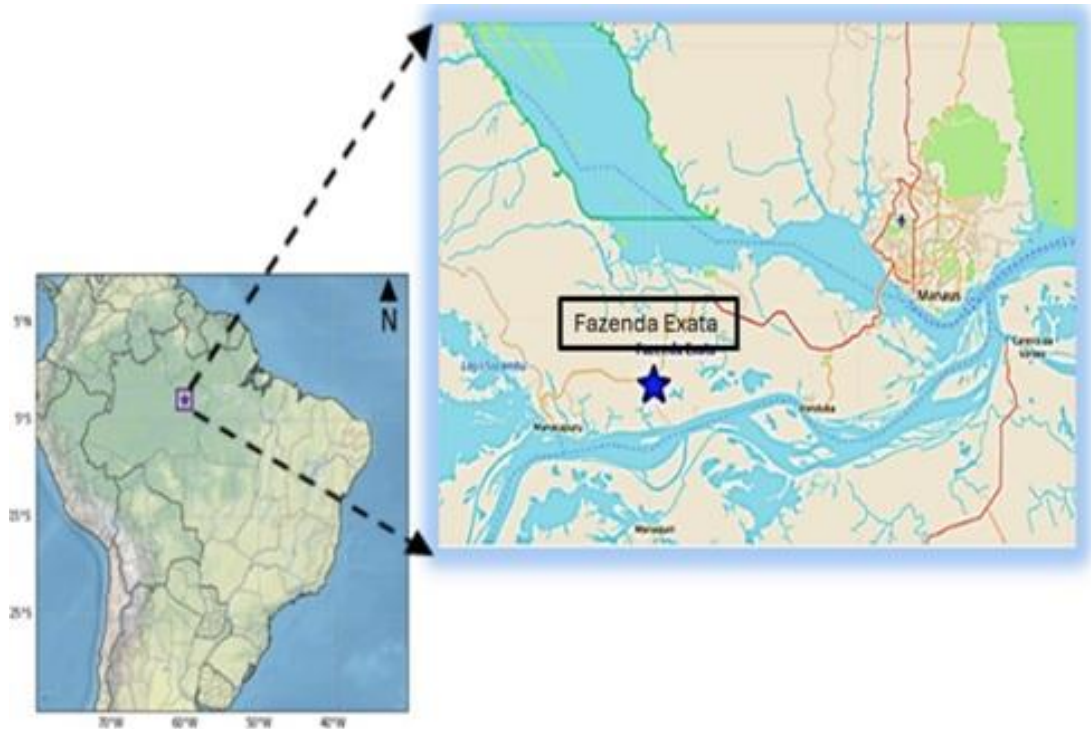


Fig 1. Map of the location of the Exata farm (Municipality of Manacapuru-AM).

The GoAmazon campaign was conducted within the Atmospheric Radiation Measurement (ARM) program and provided thermodynamic profiles, surface meteorological observations, surface energy fluxes, cloud measurements, and information on boundary-layer evolution (Mather and Voyles, 2013; Martin et al., 2017). In this study, the observations were not used directly as neural-network targets. Instead, they

supplied the initial conditions and large-scale forcing required by BAM-1D. The resulting BAM-1D simulations, performed with the original HB turbulence scheme, were subsequently used to construct the machine-learning dataset.

2.2. BAM-1D Atmospheric Model

The numerical experiments were conducted using BAM-1D, a one-dimensional column version of the Brazilian Global Atmospheric Model (BAM), which simulates the vertical evolution of the atmosphere at a fixed horizontal location while retaining the physical parameterizations and numerical integration framework of the full three-dimensional model (Figueroa et al. 2016). This configuration provides a controlled framework for evaluating individual physical parameterizations by isolating local atmospheric responses.

For the present experiments, BAM-1D was initialized and forced using observational data collected during the GoAmazon 2014/15 campaign. The initial atmospheric state consisted of vertical profiles of temperature, specific humidity, and zonal and meridional wind components, while the large-scale forcing included tendencies of temperature, moisture, vertical velocity, and horizontal wind. The principal physical parameterizations adopted in the simulations are summarized in Table 1.

Table 1. Physical parameterizations used in the BAM-1D simulations.

Physical Process	Reference
Surface process	Mellor Yamada (1982)
Planetary boundary layer	Holtslag Boville (1993)
Deep convection	Grell and Dévényi (2002)
Shallow convection	Tiedtke (1989)
Long and shortwave radiation	RRTMG; Iacomo et al. (2008)

Surface turbulent fluxes are represented with the Mellor–Yamada formulation, together with Monin–Obukhov similarity relationships for near-surface exchange. The PBL is represented by the HB scheme, a first-order K-theory closure that uses stability-dependent eddy diffusivities for momentum and heat,

includes nonlocal counter-gradient transport under convective conditions, diagnoses PBL height using a bulk Richardson-number criterion, and applies background free-tropospheric diffusion. The HB scheme constitutes the reference parameterization emulated and evaluated throughout this study. Deep convection follows the Grell and Dévényi (2002) ensemble formulation, shallow convection follows the Tiedtke (1989) mass-flux scheme, and radiative transfer is represented by RRTMG (Iacono et al. 2008).

2.3. Generation of the Machine-Learning Dataset

BAM-1D simulations covering 2014–2015 were first performed using the original HB turbulence parameterization. At each model time step and vertical level, atmospheric predictors available within BAM-1D and the turbulence-related quantities produced by the HB scheme were stored. The machine-learning task therefore consisted of reproducing the nonlinear input–output relationships of the HB scheme rather than inferring turbulence directly from observations. The complete workflow is in Figure 2.

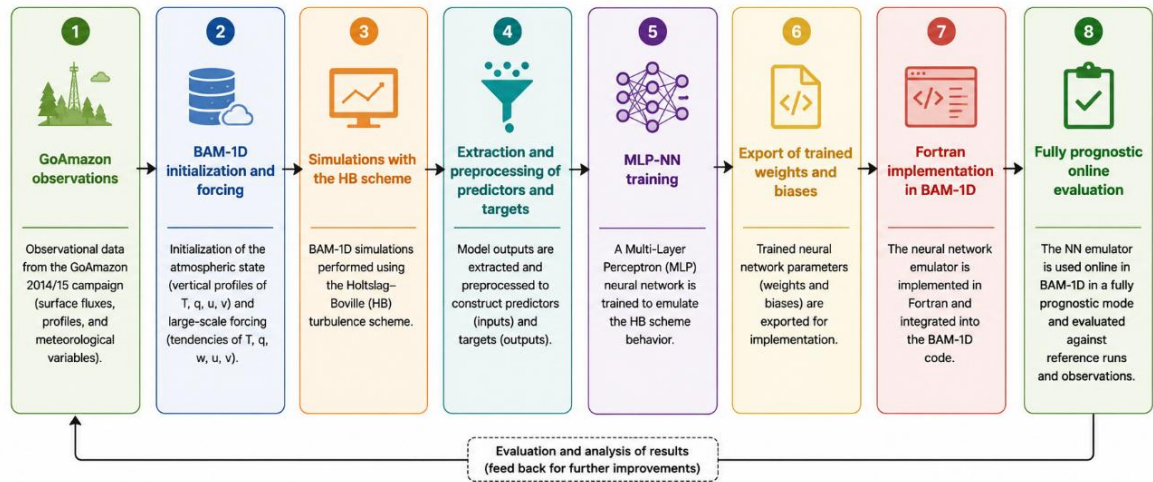


Fig 2. Schematic workflow of the proposed methodology, from GoAmazon-based BAM-1D simulations to MLP-NN training and fully prognostic online evaluation.

Four independent neural networks were trained, one for each target quantity: momentum diffusivity (k_{vm} , $m^2 s^{-1}$), heat diffusivity (k_{vh} , $m^2 s^{-1}$), the momentum counter-gradient transport term (c_{gs} , $cg/flux$), and the heat counter-gradient transport term (c_{gh} , $K m^{-1}$).

2.4. Neural-Network Architecture

The surrogate model is a feed-forward Multi-Layer Perceptron neural network (MLP-NN) with 15 input neurons, one hidden layer containing 10 neurons, and one output neuron (Figure 3). Sigmoid activation functions were used in both the hidden and output layers. Separate networks with the same general architecture were trained for *kvm*, *kvh*, *cgs*, and *cgh*.

The network architecture was selected through a series of exploratory experiments in which the number of hidden neurons, learning rate, momentum coefficient, activation function, and stopping criteria were systematically varied. Architectures with fewer hidden neurons exhibited reduced predictive accuracy, whereas larger networks produced only marginal improvements while increasing computational cost and the risk of overfitting. Among the tested configurations, a single hidden layer with 10 neurons provided the best compromise between accuracy, computational efficiency, and generalization performance on the independent validation dataset.

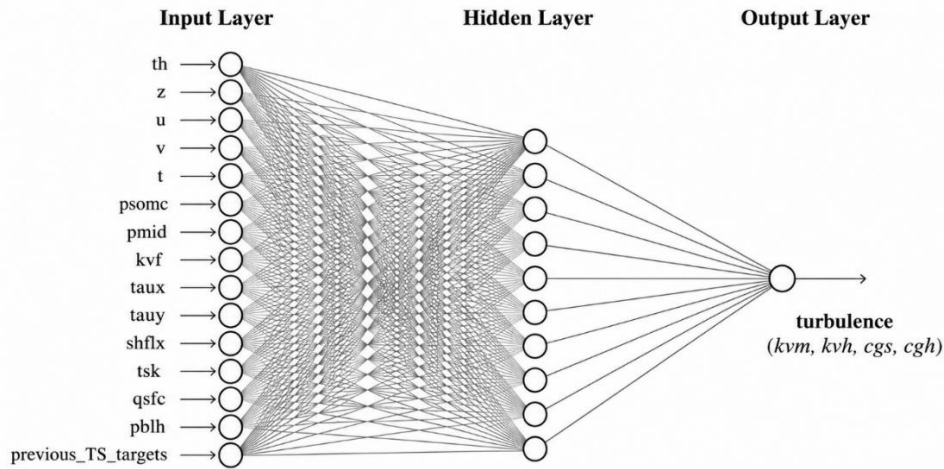


Fig 3. Neural-network architecture was adopted in this study. MLP-NN architecture with a single hidden layer composed of 10 neurons. Adapted from Haykin (2009).

2.5. Predictor and Target Variables

The atmospheric predictors were selected because they represent the principal thermodynamic, dynamic, and surface controls on boundary-layer turbulence and are directly available within BAM-1D. Fourteen

atmospheric predictors were combined with one recursive predictor: the corresponding target quantity estimated at the previous model time step. This recursive input provides temporal memory and supports a smoother evolution of the predicted turbulence quantities during online integration.

Table 2 – Variables used as inputs for neural-network training and their corresponding quantities in the HB turbulence parameterization.

Name variable	BAM	Name variables	BAM
Potential temperature	th	Surface u stress	taux
Height above surface	z	Surface y stress	tauy
Zonal wind Component	u	Surface temperature	tsk
Meridional wind Component	v	Surface specific Humidity	qsfc
Temperature (t - used for density)	t	Pressure	psomc
Midpoint pressures	pmid	Free atmospheric eddy diffusivity	kvf
Surface heat flux	shflx	Boundary-Layer height	pblh

For each network, the fifteenth input was the previous-time-step value of the corresponding target variable. Thus, the complete input vector comprised 14 atmospheric predictors plus one recursive target predictor. The use of a recursive predictor is central to the online configuration because it allows the emulator to represent temporal persistence while remaining coupled to the evolving BAM-1D atmospheric state.

2.6. Neural-Network Development in GNU/Octave

GNU/Octave was used as the computational environment for neural-network implementation and model development. Although the original MLP software package includes the Multi-Particle Collision Algorithm (MPCA) as an optimization procedure for neural-network hyperparameters (Luz et al. 2008; Anochi and Campos Velho 2014; Anochi et al. 2015; Torres et al. 2015), only the standalone MLP module was employed in this study. The network architecture and training hyperparameters were determined through exploratory experiments in GNU/Octave by testing different learning rates, momentum coefficients,

numbers of hidden neurons, activation functions, and stopping criteria. The detailed results of these tuning experiments are omitted because they are beyond the scope of this work.

2.7. Dataset Preparation and Training Strategy

All BAM-1D simulations performed between January 2014 and December 2015 were combined to construct the machine-learning dataset. The model employs the complete atmospheric column composed of 28 vertical levels with a temporal resolution of 6 min. After merging the simulations, the resulting database contained 4,878,776 vertical-level samples, corresponding to approximately 174,242 atmospheric profiles. To ensure temporal independence between model development and evaluation, the last two days of each month were excluded from the development dataset and reserved exclusively for independent testing. Consequently, 4,569,656 samples were used for model development, whereas 309,120 samples (48 independent days) were retained for testing.

Because turbulent mixing is physically restricted to the PBL, only samples satisfying $z \leq \text{PBLH}$ were retained for network training. This filtering step removed atmospheric levels above the diagnosed PBL, reducing the training dataset while preserving only the physically relevant samples used to emulate the Holtslag–Boville parameterization.

Preliminary training experiments using only the original filtered dataset resulted in unsatisfactory predictive performance, indicating that the available training samples did not provide sufficient variability for the MLP-NN to accurately learn the nonlinear relationships associated with the HB parameterization. To improve network generalization and training stability, a data augmentation strategy was therefore applied to the filtered dataset by generating five additional realizations using additive Gaussian noise with a relative amplitude of 2% of the standard deviation of each predictor variable. A fixed random seed (42) was adopted to ensure reproducibility. This procedure increased the diversity of the training samples while preserving their physical characteristics, resulting in a final training dataset containing 3,428,076 samples.

The development dataset was subsequently divided into 70% for training and 30% for validation. Predictors and target variables were normalized to the $[0, 1]$ interval using min–max scaling. Network training was

performed in GNU/Octave using the backpropagation algorithm with a learning rate of 0.3, a momentum coefficient of 0.2, and a maximum of 5,000 epochs. Separate neural networks were trained for each target variable (k_{vm} , k_{vh} , c_{gs} , and c_{gh}). The final architecture, consisting of 15 input neurons, one hidden layer with 10 neurons, and one output neuron, was selected as a compromise between predictive accuracy, numerical robustness, and computational efficiency. The independent test dataset was not used during architecture selection, hyperparameter tuning, or validation. After training, the optimized weights and biases were exported and integrated into the BAM-1D Fortran implementation.

2.8. Online Integration into BAM-1D

After training, the optimized weights and biases were exported from GNU Octave and implemented in a dedicated Fortran routine within the BAM-1D turbulence module. During model integration, BAM-1D supplies the current atmospheric predictors and the previous-time-step turbulence estimate to the relevant network, which returns the turbulence quantity required by the prognostic equations. The four neural networks therefore replace the corresponding HB calculations for k_{vm} , k_{vh} , c_{gs} , and c_{gh} while preserving the remaining model structure and time-integration procedure.

The evaluation was conducted through fully prognostic online simulations rather than solely through standalone offline predictions. This configuration allows the emulator outputs to interact continuously with the atmospheric state and enables direct assessment of accumulated feedback, temporal consistency, and numerical stability (Brenowitz and Bretherton 2018; Yuval et al. 2021).

The experiments were designed to assess three complementary aspects: (i) the fidelity with which the emulator reproduces HB-generated turbulence quantities, (ii) the preservation of numerical stability throughout model integration, and (iii) the influence of the surrogate parameterization on the temporal evolution of temperature, wind components, and precipitation. This framework enables a consistent evaluation of accuracy, computational performance, and online dynamical compatibility within BAM-1D.

3. Results

3.1. Turbulence-Profile Emulation Performance

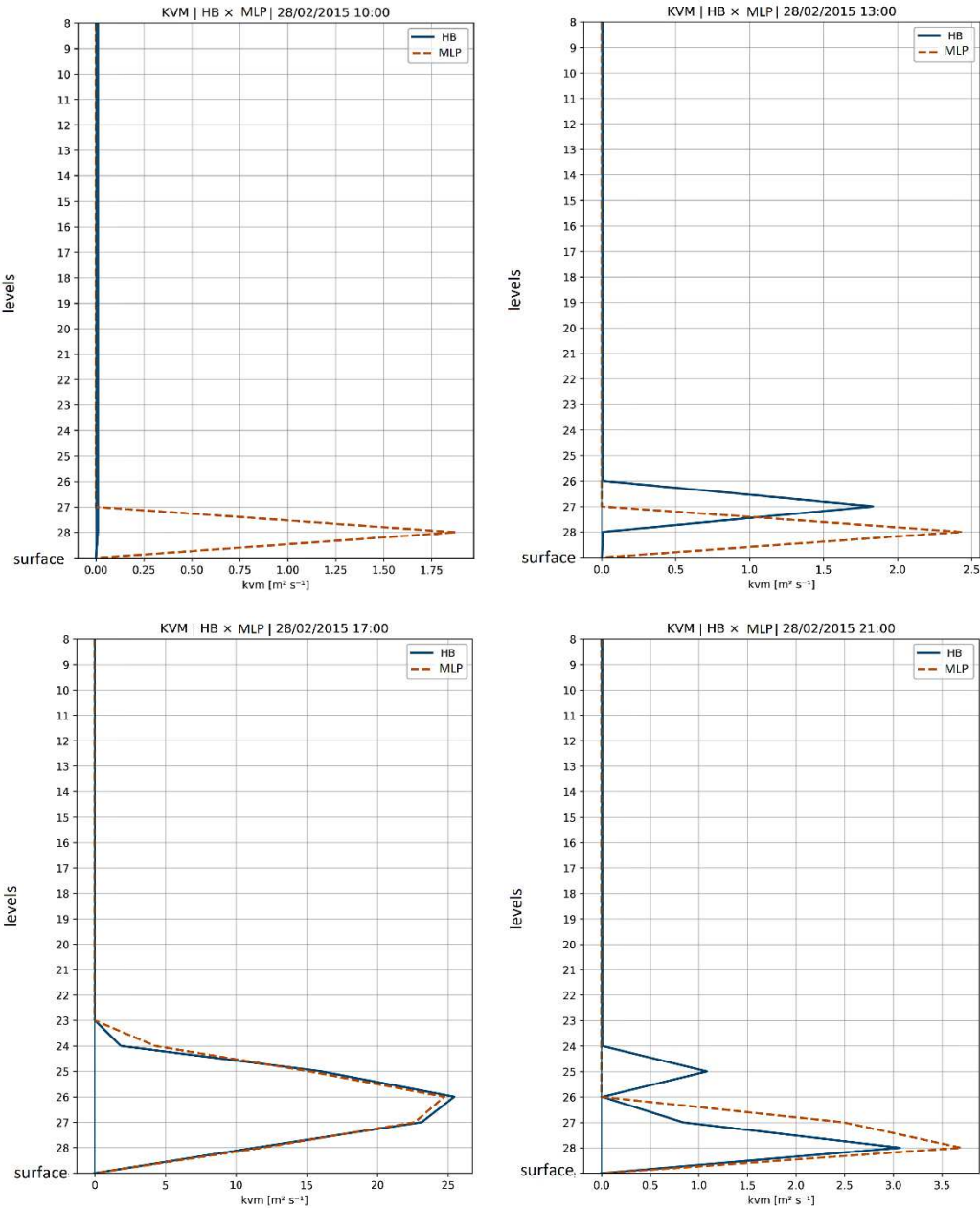
The turbulence variables generated by the HB scheme were initially evaluated through direct comparisons of vertical profiles. The analysis focused on the four variables predicted by the surrogate model: momentum diffusivity (kvm), heat diffusivity (kvh), momentum counter-gradient transport (cgs), and heat counter-gradient transport (cgh). Representative periods of the diurnal cycle (10, 13, 17, and 21 UTC) were analyzed to evaluate model performance under stable nocturnal, convective daytime, and transition boundary-layer conditions.

Several independent testing days covering different atmospheric and seasonal conditions were analyzed during the evaluation phase. Since all cases exhibited similar behavior, only the results for 28 February 2015 are presented here as a representative example of the emulator performance. The remaining cases showed the same overall agreement between the neural-network emulator and the original Holtslag–Boville (HB) parameterization, indicating that the proposed methodology is robust under different atmospheric conditions.

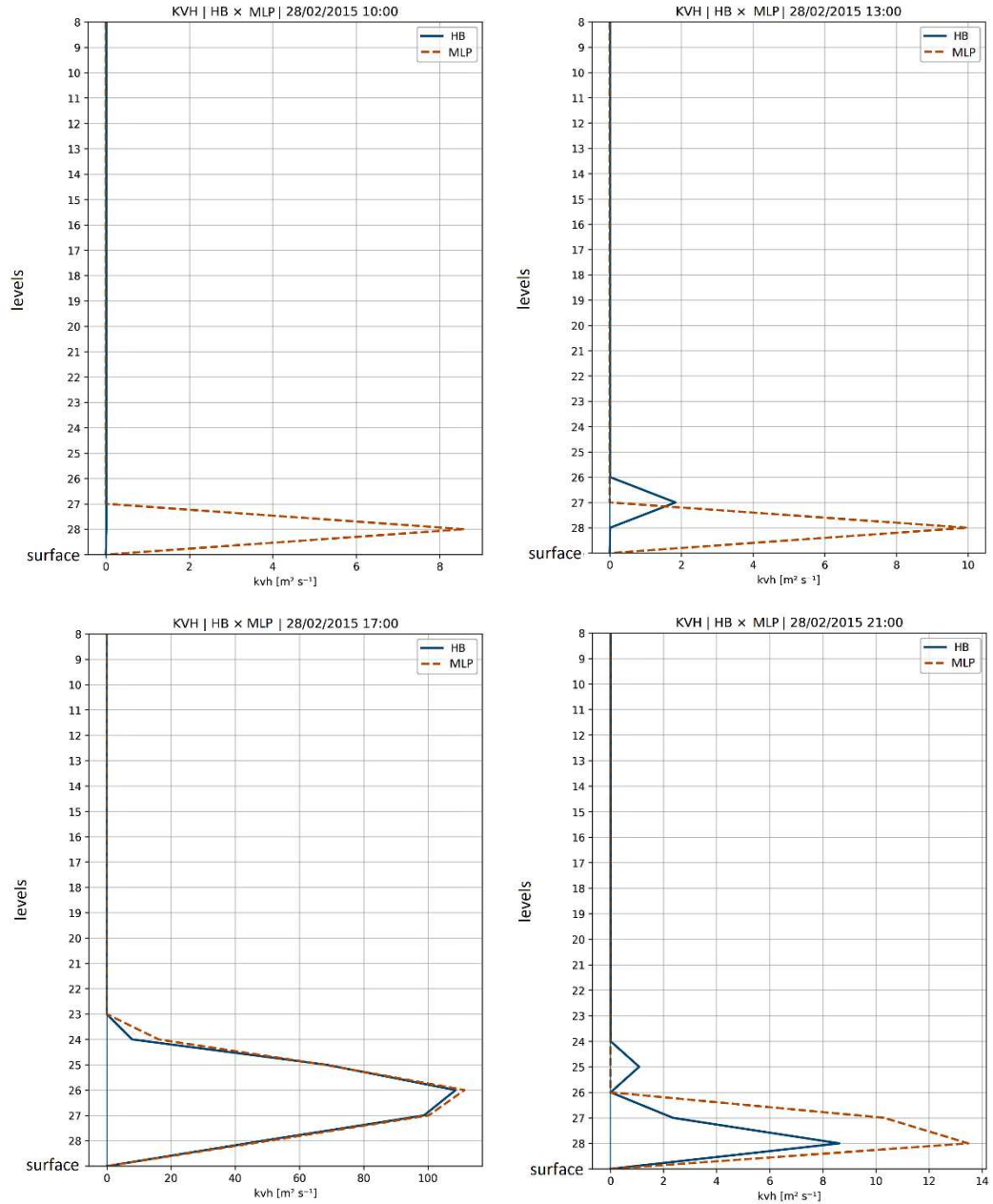
Figure 4 compares the vertical profiles of momentum diffusivity (kvm) and heat diffusivity (kvh) generated by the original HB scheme and by the neural-network emulator (MLP). Overall, the emulator accurately reproduced both the magnitude and the vertical structure of the turbulent diffusivity fields throughout the analyzed diurnal cycle.

During stable nocturnal periods, both approaches generated the expected weak-mixing profiles characterized by low diffusivity values. Small localized differences were observed near the surface, where turbulence is highly sensitive to local gradients and atmospheric stability. Nevertheless, the overall vertical structure remained highly consistent between the two methods.

During daytime convective conditions, when turbulent mixing was strongest and the planetary boundary layer was fully developed, the MLP-NN accurately reproduced the intensity and vertical extent of the HB-generated diffusivity profiles. Likewise, during transition periods, both approaches captured the progressive weakening of turbulence associated with the evening stabilization of the boundary layer.



(a)

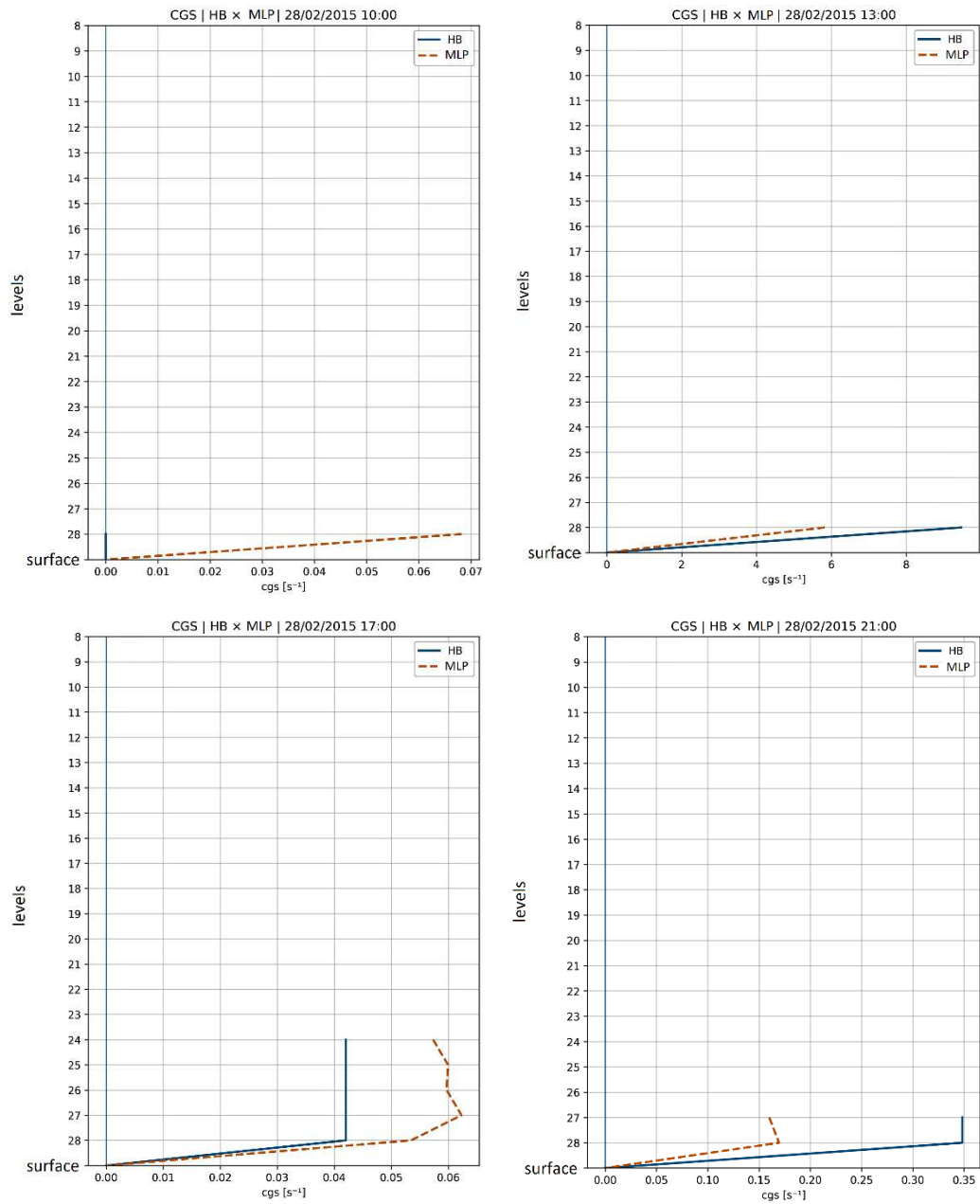


(b)

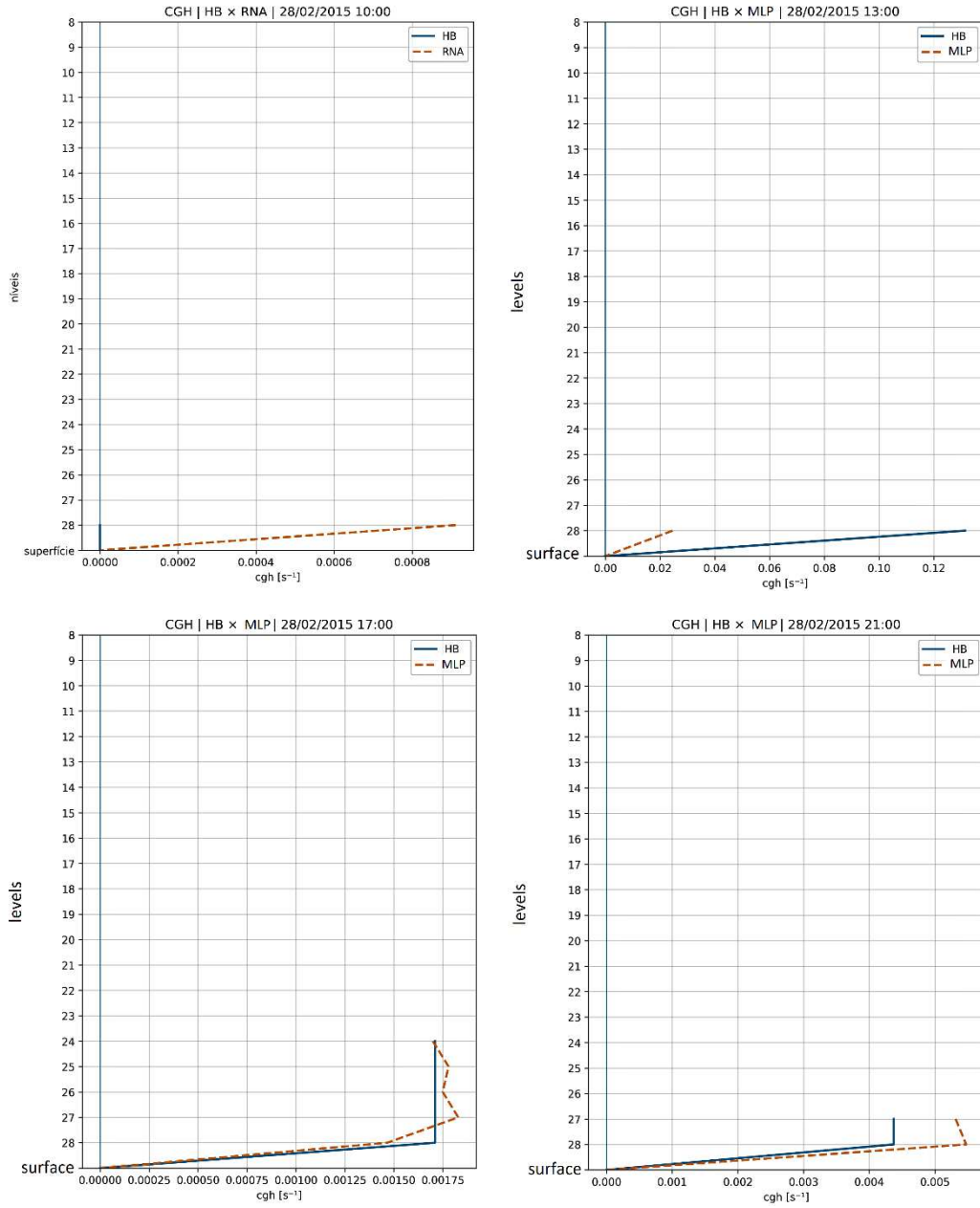
Fig 4. Comparison between the original HB turbulence parameterization and its neural-network emulator (MLP) for the turbulent diffusivity coefficients on 28 February 2015. (a) Momentum diffusivity (k_{vm} , $m^2 s^{-1}$) and (b) heat diffusivity (k_{vh} , $m^2 s^{-1}$). In both panels, the subplots are arranged in chronological order: upper left (10 UTC), upper right (13 UTC), lower left (17 UTC), and lower right (21 UTC). The vertical axis represents the model levels.

These results demonstrate that the neural-network emulator preserves the principal characteristics of the HB turbulence parameterization across a wide range of atmospheric conditions, providing a physically coherent representation of the vertical turbulent structure while maintaining consistency throughout the

diurnal cycle. Figure 5 compares the vertical profiles of the counter-gradient transport terms for momentum (cgs) and heat (cgh) generated by the original HB scheme and the neural-network emulator (MLP). Overall, the emulator successfully reproduced the magnitude and vertical structure of both variables throughout the analyzed diurnal cycle. Although small differences are observed at some levels, particularly near regions of stronger gradients, the general behavior of the profiles remains consistent with that produced by the original parameterization. These results indicate that the neural-network emulator can capture the main characteristics of the counter-gradient transport terms under different boundary-layer conditions.



(a)



(b)

Fig 5. Comparison between the original HB turbulence parameterization and its neural-network emulator (MLP) for the counter-gradient transport terms on 28 February 2015. (a) Momentum counter-gradient term (CGS, CG flux) and (b) heat counter-gradient term (CGH, K m^{-1}). In both panels, the subplots are arranged in chronological order: upper left (10 UTC), upper right (13 UTC), lower left (17 UTC), and lower right (21 UTC). The vertical axis represents the model levels.

3.2 Statistical Evaluation of Turbulence Variables

While the profile comparisons presented in Section 3.1 provide a qualitative assessment of the neural-network emulator, a quantitative evaluation is required to measure its ability to reproduce the turbulence-related quantities generated by the HB scheme. This assessment was performed using the independent testing dataset and standard statistical metrics, including the coefficient of determination (R^2), root mean square error (RMSE), mean absolute error (MAE), and bias.

Table 3 summarizes the statistical performance obtained for the four target variables predicted by the MLP-NN: momentum diffusivity (kvm), heat diffusivity (kvh), momentum counter-gradient transport (cgs), and heat counter-gradient transport (cgh).

Table 3. Consolidated statistical metrics for the turbulent coefficients estimated by the MLP-NN in comparison with the HB scheme.

Variable	R^2	RMSE	MAE	BIAS
kvm ($\text{m}^2 \text{s}^{-1}$)	0.99513	0.90991	0.55579	0.31107
kvh ($\text{m}^2 \text{s}^{-1}$)	0.99441	5.65054	2.92264	-0.94419
cgs (cg/flux)	0.73446	0.37352	0.066239	0.058306
cgh (k/m)	0.80786	0.001613	0.000543	0.000475

The results indicate agreement between the neural-network emulator and the original HB parameterization for the turbulent diffusivity coefficients. Both kvm and kvh exhibit high coefficients of determination and low error values, demonstrating that the MLP-NN successfully reproduced the dominant nonlinear relationships governing turbulent momentum and heat transport within the boundary layer.

The counter-gradient transport terms (cgs and cgh) presented comparatively lower coefficients of determination. Nevertheless, the corresponding absolute errors and biases remained small. This behavior reflects the substantially smaller magnitudes and greater relative variability of these quantities compared with the diffusivity coefficients. Consequently, small absolute deviations may translate into larger relative statistical differences without necessarily implying a deterioration of the physical representation.

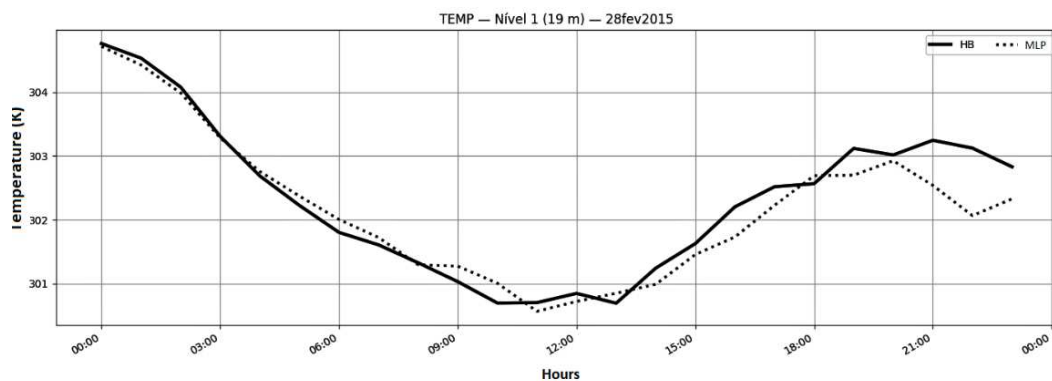
3.3. BAM-1D Simulations

The performance of the neural-network turbulence emulator was further evaluated through fully prognostic BAM-1D simulations. In addition to reproducing the turbulence-related quantities generated by the original HB scheme, the impact of the surrogate parameterization on the evolution of key atmospheric variables was examined. The analysis focused on temperature, wind components, and precipitation, which provided an integrated assessment of the atmospheric response to turbulent transport and the overall consistency of the emulator within the model framework.

3.3.1 Temperature Profiles

The comparison of absolute temperature profiles indicates that the MLP-NN emulator successfully reproduced the thermodynamic evolution generated by the original HB scheme. Figure 6 presents the simulated temperature fields for representative atmospheric conditions during the evaluation period. Overall, the neural-network parameterization preserved the diurnal temperature cycle, including nocturnal cooling, daytime warming, and the development of the convective boundary layer. The agreement between the HB scheme and the MLP-NN was particularly strong during the February and October cases, with both approaches producing similar temperature distributions throughout the atmospheric column.

The June case exhibited larger localized differences, especially near the surface and within intermediate levels. Nevertheless, the overall temporal evolution and vertical thermal structure remained consistent between the two configurations throughout the simulation period. These results indicate that the MLP-NN emulator preserves the principal characteristics of the atmospheric temperature evolution simulated by BAM-1D under distinct atmospheric conditions.



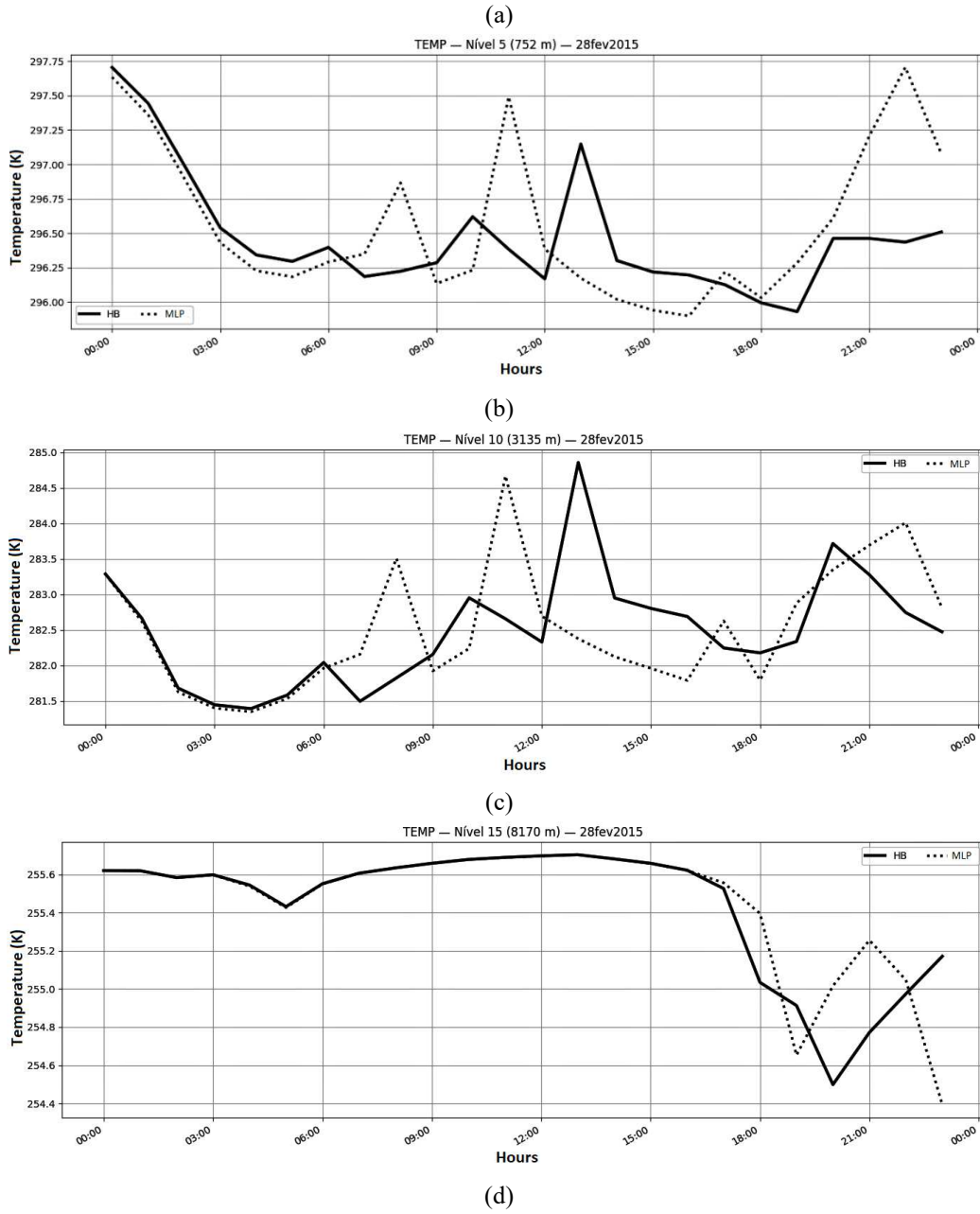


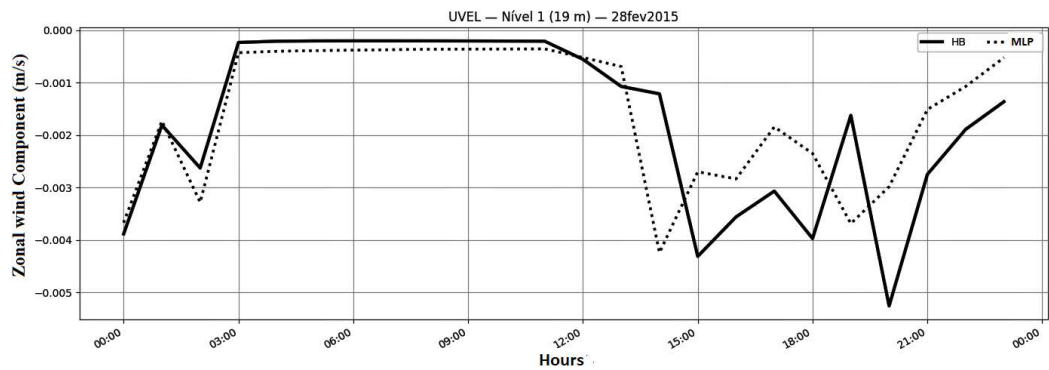
Fig 6. Comparison of the diurnal cycle of absolute temperature (TEMP) simulated by BAM-1D using the original HB scheme and its neural-network emulator (MLP-NN). The panels represent different vertical levels of the model, with the corresponding heights indicated in the title of each subplot.

3.3.2. Wind Components

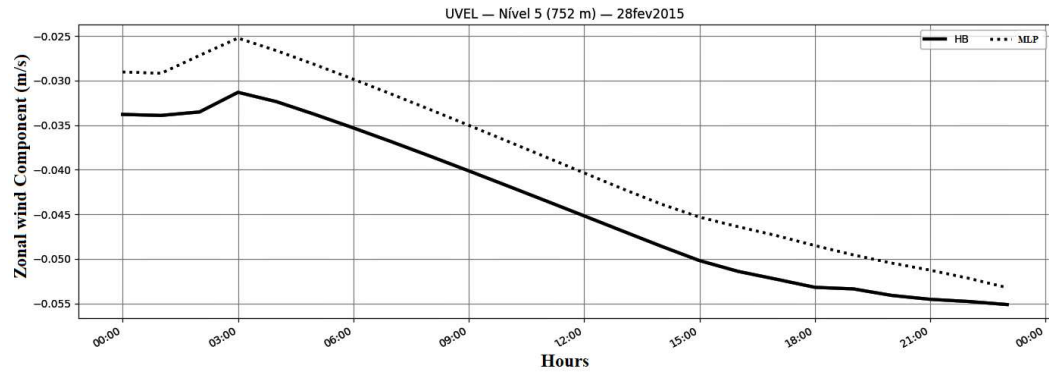
The zonal (u) and meridional (v) wind components were also compared to evaluate the impact of the neural-network turbulence parameterization on the simulated atmospheric circulation. Figure 7 shows the temporal

evolution of the wind field generated by BAM-1D using the HB scheme and the MLP-NN emulator. In general, both approaches reproduced similar temporal patterns and vertical structures throughout the analyzed periods.

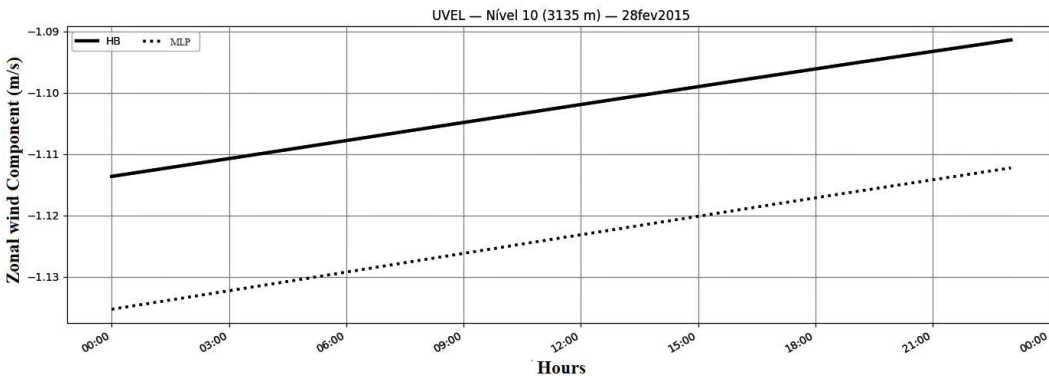
The agreement was strongest at intermediate and upper boundary-layer levels, where both parameterizations produced comparable wind distributions. Larger differences were observed near the surface, particularly during periods of enhanced turbulence and stronger wind shear. Despite these differences, the overall temporal evolution of the wind components remained consistent across all analyzed cases.



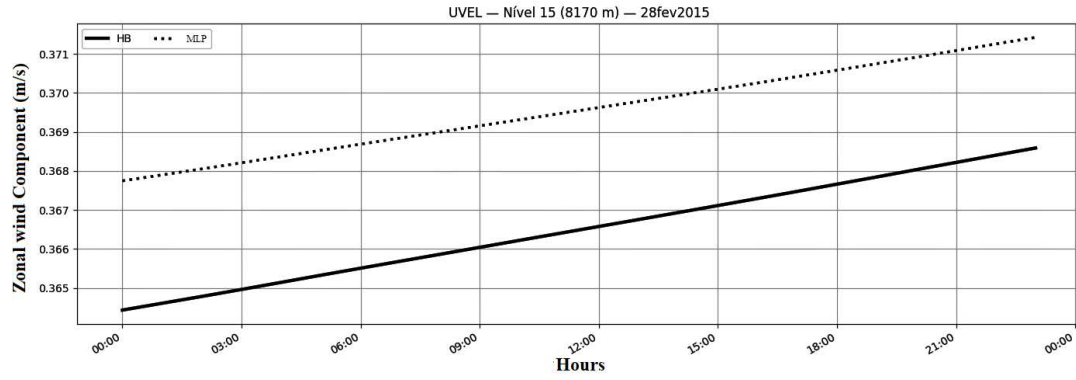
(a)



(b)



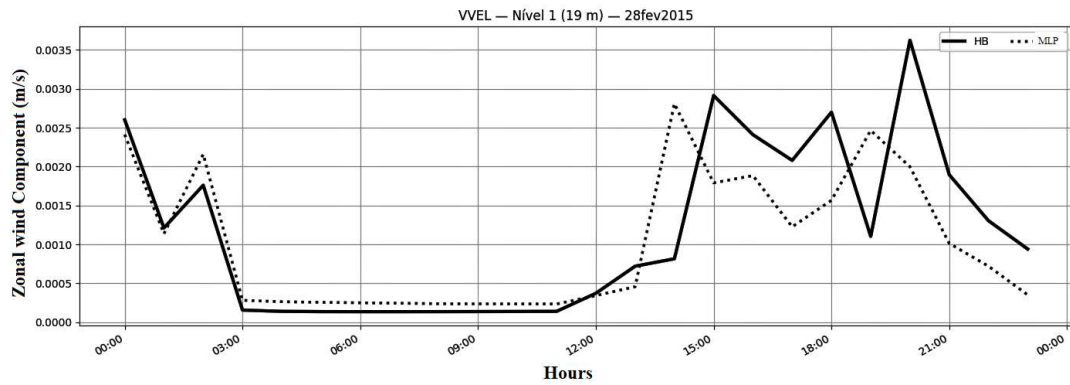
(c)



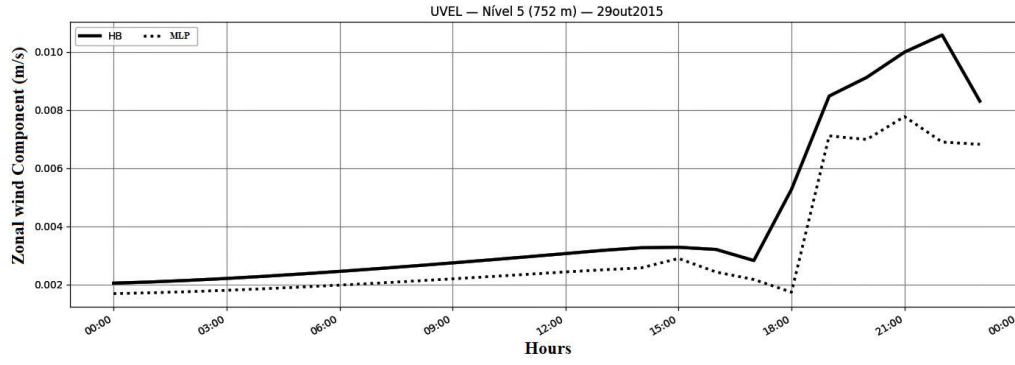
(d)

Fig 7. Comparison of the diurnal cycle of zonal wind component (UVEL) simulated by BAM-1D using the original HB scheme and its neural-network emulator (MLP-NN). The panels represent different vertical levels of the model, with the corresponding heights indicated in the title of each subplot.

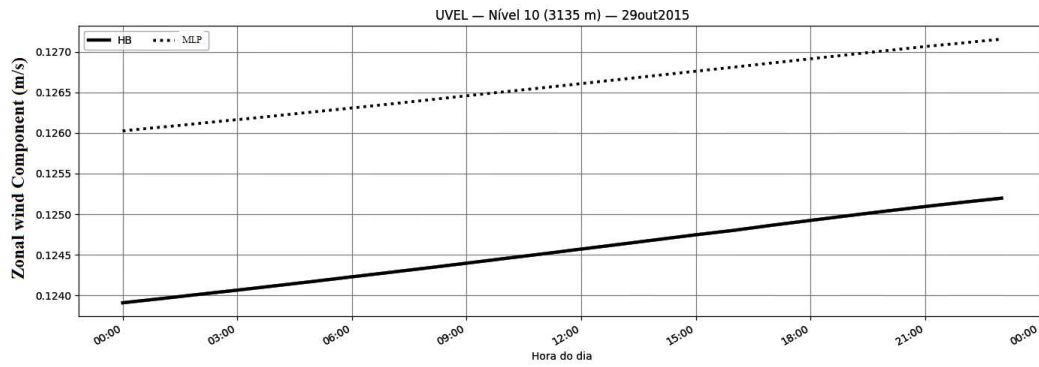
The February case exhibited the greatest variability, whereas the June case presented the closest agreement between the two parameterizations. The October case displayed intermediate behavior. Overall, the results indicate that the neural-network emulator preserves the principal characteristics of the atmospheric flow simulated by the original turbulence parameterization.



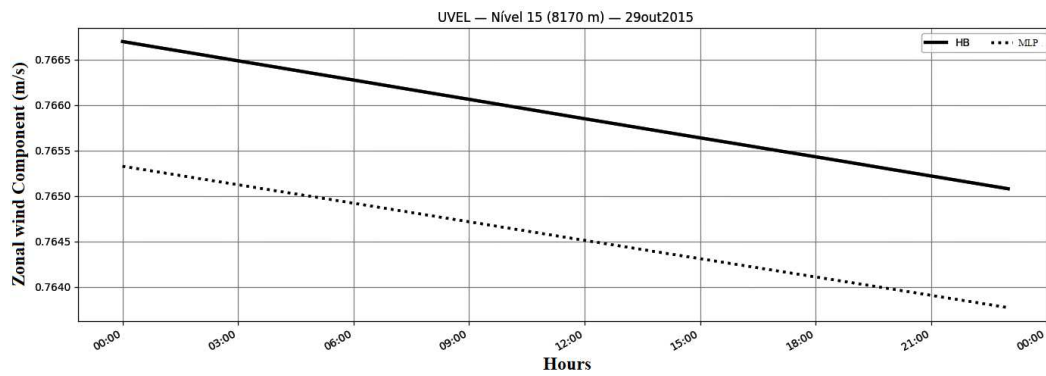
(a)



(b)



(c)



(d)

Fig 8. Comparison of the diurnal cycle of absolute temperature (VVEL) simulated by BAM-1D using the original HB scheme and its neural-network emulator (MLP-NN). The panels represent different vertical levels of the model, with the corresponding heights indicated in the title of each subplot.

3.4. Impact on Precipitation

Precipitation was analyzed as an integrated atmospheric response variable influenced by modifications in turbulent transport processes. Figure 9 compares the precipitation time series generated by BAM-1D using the original HB parameterization and the MLP-NN emulator. For analyzed cases, the neural-network surrogate reproduced the principal precipitation events identified in the reference simulations. The timing of precipitation occurrence and the general magnitude of rainfall peaks were preserved throughout the evaluation period.

The February case exhibited greater variability and intermittency, reflecting the more active convective conditions characteristic of the wet season. The June case showed stronger agreement between the two parameterizations, whereas the October case was characterized by weak and intermittent precipitation.

Across all cases, both approaches produced similar precipitation evolution without the occurrence of unrealistic oscillations or discontinuities during model integration. These results demonstrate that the neural-network emulator preserves the main temporal characteristics of the precipitation fields generated by BAM-1D under different atmospheric regimes.

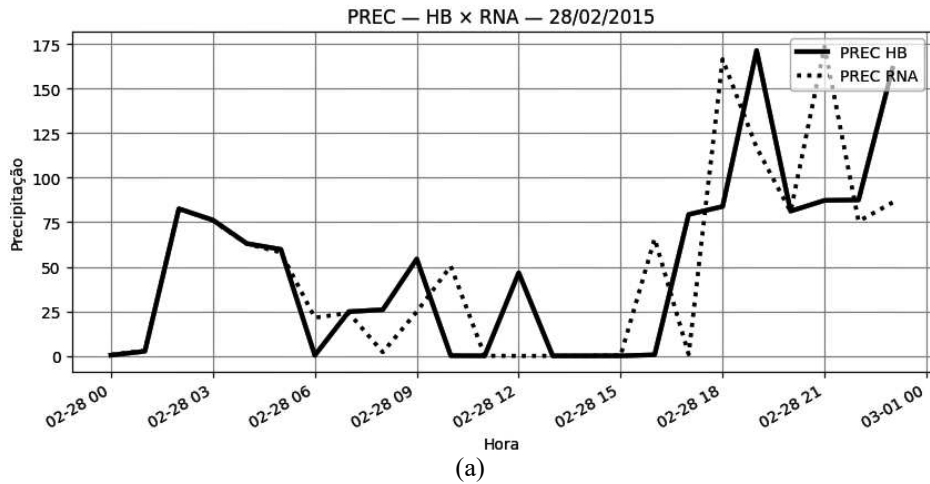


Fig 9. Time series of precipitation simulated by BAM-1D using the original HB turbulence parameterization and its MLP-NN emulator on 28 February 2015.

3.5. Computational Performance and Scalability

The computational cost of the neural-network emulator was compared with that of the original HB turbulence parameterization. The average execution time of the HB routine was approximately 10^{-6} s per call, whereas the MLP-NN inference required approximately 10^{-5} s per call in the BAM-1D configuration. Although the neural-network implementation was slower than the original turbulence routine in the present one-dimensional framework, both approaches remained computationally inexpensive relative to the total model integration cost.

The computational structure of the MLP-NN consists primarily of matrix multiplications and activation-function evaluations associated with a forward propagation step. In contrast, the HB scheme involves physically based calculations dependent on atmospheric stability and turbulence-regime conditions. The execution-time comparison therefore indicates that both approaches are suitable for operational integration within BAM-1D, while exhibiting distinct computational characteristics.

4. Discussion and Conclusions

The results demonstrate that the proposed Multi-Layer Perceptron neural network (MLP-NN) emulates the behavior of the HB scheme within the BAM-1D framework. The emulator reproduced the turbulence-related quantities generated by the original scheme, particularly the momentum and heat diffusivities (kvm and kvh), which exhibited the highest levels of agreement in both the profile analyses and statistical evaluation. Although the counter-gradient transport terms (cgs and cgh) showed lower coefficients of determination, their absolute errors and biases remained small, indicating that the neural network preserved the overall behavior of the original parameterization.

The analyses performed with the neural-network emulator integrated into BAM-1D further demonstrated that replacing the HB scheme did not introduce numerical instabilities, unrealistic oscillations, or physically inconsistent behavior. The simulated evolution of temperature, wind components, and precipitation remained in close agreement with the reference simulations, indicating that the surrogate model is dynamically compatible with the host atmospheric model under fully prognostic conditions.

These findings are consistent with recent studies investigating machine-learning emulation of physical parameterizations in Earth-system models (Brenowitz and Bretherton 2018; Yuval et al. 2021). More broadly, the present work contributes to the growing development of hybrid physics–machine-learning frameworks. Unlike fully data-driven forecasting systems such as GraphCast, NeuralGCM, and ECMWF’s Anemoi, the proposed approach preserves the physical structure of the numerical model while replacing only a specific subgrid-scale process. This strategy provides a controlled pathway for integrating machine-learning techniques into operational atmospheric models while maintaining physical interpretability and model consistency.

The computational analysis showed that the original HB scheme remains faster than the neural-network emulator in the current one-dimensional BAM-1D configuration. However, the MLP-NN consists primarily of deterministic matrix operations that are naturally compatible with vectorization, parallel execution, and hardware acceleration. Consequently, the relative computational cost observed here may not reflect the performance achievable in future large-scale implementations using modern high-performance computing architectures.

Some limitations should also be acknowledged. The emulator was trained using simulations generated from atmospheric conditions sampled during the GoAmazon 2014/15 experiment in central Amazonia. Therefore, its applicability outside the range of conditions represented in the training dataset remains to be evaluated. In addition, the neural network was designed to reproduce the behavior of the HB scheme rather than to improve its physical formulation, implying that structural limitations of the original parameterization are inherently inherited by the emulator.

Overall, the results demonstrate that neural-network-based surrogate modeling can reproduce an operational turbulence parameterization with high fidelity while preserving numerical stability and physical consistency during model integration. The proposed framework provides a practical pathway toward the development of hybrid physics–machine-learning parameterizations and contributes to ongoing efforts to incorporate artificial intelligence techniques into next-generation atmospheric models. Future work should extend the methodology to additional climatic regimes, evaluate its performance within the three-

dimensional BAM configuration, and investigate the emulation of other physical parameterizations, including convection, radiation, and cloud microphysics.

Declarations

Author Contributions: All authors contributed to the study's conception and design. A.V.P.N performed material preparation, data collection, pre-processing, and analysis, did the experiments, and wrote the first draft of the manuscript. All authors commented on previous versions of the manuscript. All authors read and approved of the final manuscript.

References

Anochi JA, Campos Velho HF (2014) Optimization of Feedforward Neural Network by Multiple Particle Collision Algorithm. In: 2014 IEEE Symposium on Foundations of Computational Intelligence (FOCI), pp 128–134. IEEE. <https://doi.org/10.1109/FOCI.2014.7007817>

Anochi JA, Campos Velho HF, Furtado HCM, Luz EFP (2015) Self-configuring Two Types of Neural Networks by MPCA. *Journal of Mechanics Engineering and Automation* 5:112–120.

Arya PS (2001) *Introduction to Micrometeorology*. 2nd edn. Academic Press, San Diego. (International Geophysics Series, Vol. 79).

Brenowitz CD, Bretherton CS (2018) Prognostic validation of a neural network unified physics parameterization. *Geophysical Research Letters*, 45(12), 6289–6298. <https://doi.org/10.1029/2018GL078510>

Campos Velho HF (2010) *Modelagem Matemática em Turbulência Atmosférica*. *Notas em Matemática Aplicada*, vol. 48. Sociedade Brasileira de Matemática Aplicada e Computacional, São Carlos. <https://doi.org/10.5540/001.2012.0048.01>.

Canché-Cab L, San-Pedro L, Ali B, Rivero M, Escalante M (2024). The atmospheric boundary layer: A review of current challenges and a new generation of machine learning techniques. *Artificial Intelligence Review*, **57**, 331. <https://doi.org/10.1007/s10462-024-10962-5>

Degrazia GA, Anfossi D, Carvalho JC, Mangia C, Tirabassi T, Campos Velho HF (2000) Turbulence parameterisation for PBL dispersion models in all stability conditions. *Atmospheric Environment* 34(21):3575–3583. [https://doi.org/10.1016/S1352-2310\(00\)00116-3](https://doi.org/10.1016/S1352-2310(00)00116-3)

Dueben PD, Bauer P (2018) Challenges and design choices for global weather and climate models based on ML. *Geoscientific Model Development*, p. 3999-4009. <https://doi.org/10.5194/gmd-11-3999-2018>

Dramsch J, Raoult B, Chantry M, García T, Denby L, Prill F, Sokka N, Vocino A, Wijnands J, Nipen T, Osuna C, Akodad S, Van Ginderachter M, Van den Bleeken D (2024) Introducing Anemoi: A New Collaborative Framework for Machine Learning Weather Forecasting. *ECMWF Newsletter*, No. 181, Autumn 2024. <https://www.ecmwf.int/en/newsletter/181/news/introducing-anemoi-new-collaborative-framework-ml-weather-forecasting>

Figueroa SN, Bonatti JP, Kubota PY, Grell GA, Morrison H, Barros SR, Fernandez JP, Ramirez E, Siqueira L, Luzia G, Others (2016) The Brazilian global atmospheric model (BAM): performance for tropical rainfall forecasting and sensitivity to convective scheme and horizontal resolution. *Weather and Forecasting*, v. 31, 1547–1572. <https://doi.org/10.1175/WAF-D-16-0062.1>

Fisch G, Tota J, Machado LAT, Silva Dias MAF, Lyra RFF, Nobre CA, Dolman AJ, Gash JHC (2004) The convective boundary layer over pasture and forest in Amazonia. *Theoretical and Applied Climatology*, 78(1–3), 47–59. <https://doi.org/10.1007/s00704-004-0043-x>

Grell GA, Dévényi D (2002) A generalized approach to parameterizing convection combining ensemble and data assimilation techniques. *Geophysical Research Letters*, 29(14), 1693. <https://doi.org/10.1029/2002GL015311>

616

617 Haykin S (2009) Neural Networks and Learning Machines - Third Edition. Pearson Prentice Hall. 978-0-
618 13-147139-9

619

620 Holtslag AAM, Boville BA (1993) Local versus non-local boundary-layer diffusion in a global climate
621 model. *Journal of Climate*, American Meteorological Society, v. 6, n. 10, p. 1825–1842.

622

623 Iacono MJ, Delamere JS, Mlawer EJ, Shephard MW, Clough SA, Collins WD (2008) Radiative forcing
624 by long-lived greenhouse gases: Calculations with the AER radiative transfer models. *Journal of*
625 *Geophysical Research: Atmospheres*, 113(D13), D13103. <https://doi.org/10.1029/2008JD009944>

626

627 Kalnay E (2003) Atmospheric Modeling, Data Assimilation and Predictability. Cambridge University
628 Press, Cambridge, United Kingdom. <https://doi.org/10.1017/CBO9780511802270>

629 Kochkov D, Yuval J, Langmore I, Norgaard P, Smith J, Mooers G, Klöwer M, Lottes J, Rasp S,
630 Düben P, Hatfield S, Battaglia P, Sanchez-Gonzalez A, Willson M, Brenowitz C, Behrens J,
631 Pritzel A, Lam R, Rabinowitz N, Mohamed S, Gong Y, Fortunato M, Alet F, George D, Kumaran
632 D, Leskovec J, Sellwood M, Hassabis D, Pfaff T (2024) Neural General Circulation Models for
633 Weather and Climate. *Nature* 632:1060–1066. <https://doi.org/10.1038/s41586-024-07744-y>

634

635 Lam R, Sanchez-Gonzalez A, Willson M, Wirsberger P, Fortunato M, Alet F, Ravuri S, Ewalds T, Eaton-
636 Rosen Z, Hu W, Meroze A, Hoyer S, Holland G, Vinyals O, Veeling B, Battaglia P, Hassabis D (2023)
637 GraphCast: Learning Skillful Medium-Range Global Weather Forecasting. *Science* 382(6677):1416–1421.
638 <https://doi.org/10.1126/science.adi2336>

639

640 Luz EFP, Becceneri JC, Campos Velho HF (2008) A New Multi-Particle Collision Algorithm for
641 Optimization in a High-Performance Environment. *Journal of Computational Interdisciplinary Sciences*
642 1(1):3–10.

643

Mather JH, Voyles JW (2013) The ARM Climate Research Facility: A Review of Structure and Capabilities. *Bulletin of the American Meteorological Society*, 94(3), 377–392. <https://doi.org/10.1175/BAMS-D-11-00218.1>

Marengo JA, Liebmann B, Kousky VE, Filizola NP, Wainer IC (2001) On the Onset and End of the Rainy Season in the Brazilian Amazon Basin. *Journal of Climate*, 14(5), 833–852. [https://doi.org/10.1175/1520-0442\(2001\)014<0833:OTOAEO>2.0.CO;2](https://doi.org/10.1175/1520-0442(2001)014<0833:OTOAEO>2.0.CO;2)

Martin ST, Artaxo P, Machado LAT, Manzi AO, Souza RAF, Schumacher C, Wang J, Andreae MO, Barbosa HMJ, Fan J, Fisch G, Goldstein AH, Guenther A, Jimenez JL, Poschl U, Silva Dias MA, Smith JN, Wendisch M (2016) Introduction: Observations and Modeling of Green Ocean Amazon (GOAmazon 2014/15). *Atmospheric Chemistry and Physics*, v. 15, p. 4785–4797. <https://doi.org/10.5194/acp-16-4785-2016>

Martin ST, Artaxo P, Machado L, Manzi AO, Souza RAF, Schumacher C, Wang J, Biscaro T, Brito J, Calheiros A, Jardine K, Medeiros A, Portela B, de Sá SS, Adachi K, Aiken AC, Albrecht R, Alexander L, Andreae MO, Barbosa HMJ, Buseck P, Chand D, Comstock JM, Day DA, Dubey M, Fan J, Fast J, Fisch G, Fortner E, Giangrande S, Gilles M, Goldstein AH, Guenther A, Hubbe J, Jensen M, Jimenez JL, Keutsch FN, Kim S, Kuang C, Laskin A, McKinney K, Mei F, Miller M, Nascimento R, Pauliquevis T, Pekour M, Peres J, Petäjä T, Pöhlker C, Pöschl U, Rizzo L, Schmid B, Shilling JE, M. A. Silva Dias MA, Smith JN, Tomlinson JM, Tóta J, Wendisch M (2017): The Green Ocean Amazon Experiment (GoAmazon2014/5) Observes Pollution Affecting Gases, Aerosols, Clouds, and Rainfall over the Rain Forest. *Bulletin of the American Meteorological Society*, v. 98(5), 981–997. <https://doi.org/10.1175/BAMS-D-16-00221.1>

Mellor GL, Yamada T (1982) Development of a turbulence closure model for geophysical fluid problems. *Reviews of Geophysics and Space Physics*, 20(4), 851–875. <https://doi.org/10.1029/RG020i004p00851>.

Monin AS, Obukhov AM (1954) Basic laws of turbulent mixing in the surface layer of the atmosphere. *Trudy Geofizicheskogo Instituta AN SSSR* 24(151):163–187 (in Russian).

Pereira Neto AV, Campos Velho HF, França GB, Pereira JC (2026) Neural Network for
 Estimating the Planetary Boundary Layer Height. *Meteorology and Atmospheric Physics*, 138,
 31. <https://doi.org/10.1007/s00703-026-01121-2>

Rasp S, Pritchard MS, Gentine P (2018) Deep Learning to Represent Subgrid Processes in
 Climate Models. *Proceedings of the National Academy of Sciences of the United States of
 America (PNAS)* 115(39):9684–9689. <https://doi.org/10.1073/pnas.1810286115>

Reynolds O (1895) On the dynamical theory of incompressible viscous fluids and the determination of the
 criterion. *Philosophical Transactions of the Royal Society of London A* **186**:123–164.
<https://doi.org/10.1098/rsta.1895.0004>

Scher S (2018) Toward Data-Driven Weather and Climate Forecasting: Approximating a Simple General
 Circulation Model with Deep Learning. *Geophysical Research Letters*, 45(12), 616–624.
<https://doi.org/10.1029/2018GL080704>

Schultz MG, Betancourt C, Gong B, Kleinert F, Langguth M, Leufen LH, Mozaffari A, Stadtler S (2021)
 Can deep learning beat numerical weather prediction? *Phil. Trans. R. Soc. A* 379: 20200097.
<https://doi.org/10.1098/rsta.2020.0097>

Stull RB (1988) *An Introduction to Boundary Layer Meteorology*. Springer Science & Business Media.
<https://doi.org/10.1007/978-94-009-3027-8>

Tiedtke M (1989) A Comprehensive Mass Flux Scheme for Cumulus Parameterization in Large-
 Scale Models. *Monthly Weather Review*, 117(8), 1779–1800. [https://doi.org/10.1175/1520-0493\(1989\)117<1779:ACMFSF>2.0.CO;2](https://doi.org/10.1175/1520-0493(1989)117<1779:ACMFSF>2.0.CO;2)

Torres RH, Luz EFP, Campos Velho HF (2015) Multi-Particle Collision Algorithm with Reflected Points. *Proceeding Series of the Brazilian Society of Applied and Computational Mathematics*, 3(1). <https://doi.org/10.5540/03.2015.003.01.0433>.

Yuval J, O’Gorman PA, & Hill C. (2021). Use of neural networks for stable and physically consistent parameterization of subgrid processes in climate models. *Journal of Advances in Modeling Earth Systems*, 13, e2021MS002477. <https://doi.org/10.1029/2021MS002477>

Zhuang Y, Fu R, Marengo JA, Wang H (2017) Seasonal Variation of Shallow-to-Deep Convection Transition and Its Link to the Environmental Conditions over the Central Amazon. *Journal of Geophysical Research: Atmospheres*, 122, 2649–2666. <https://doi.org/10.1002/2016JD025993>