Accelerating Weather Forecasting: A Neural Network-Based Emulation of ISORROPIA

Georgios Evangelopoulos
Scientific Computing Center
Karlsruhe Institute of Technology
Karlsruhe, Germany
georgios.evangelopoulos@kit.edu

Gholamali Hoshyaripour
Institute of Meteorology & Climate Research
Karlsruhe Institute of Technology
Karlsruhe, Germany
ali.hoshyaripour@kit.edu

Jörg Meyer
Scientific Computing Center
Karlsruhe Institute of Technology
Karlsruhe, Germany
joerg.meyer2@kit.edu

Pankaj Kumar
Institute of Meteorology & Climate Research
Karlsruhe Institute of Technology
Karlsruhe, Germany
pankaj.kumar@kit.edu

Julia Bruckert
Institute of Meteorology & Climate Research
Karlsruhe Institute of Technology
Karlsruhe, Germany
julia.bruckert@kit.edu

Achim Streit
Scientific Computing Center
Karlsruhe Institute of Technology
Karlsruhe, Germany
achim.streit@kit.edu

***Abstract*—Atmospheric composition is an essential part of weather, climate and Earth system modeling. However, modeling atmospheric composition is a computationally expensive and time-consuming task that requires a significant amount of energy. As models scale to finer spatial and temporal resolutions, maintaining real-time performance becomes increasingly challenging. To address this, optimization and acceleration techniques are essential. One promising approach is the use of deep neural networks, which have demonstrated the capability to efficiently approximate complex systems with high accuracy. Predictions using these neural networks are notably faster compared to traditional methods, significantly reducing the computational burden. In this study, we present the development of a surrogate model designed to emulate ISORROPIA, a traditional model used for calculating the concentrations of chemical compounds in the ICON-ART (ICOsahedral Nonhydrostatic model with Aerosol and Reactive Trace gases) model. Specifically, ISORROPIA is an aerosol thermodynamic equilibrium model used by ART that requires substantial computational resources, occupying a significant portion of the overall calculation time, making it particularly well-suited for emulation. The methodology involved generating a comprehensive dataset using the traditional model, which served as the training data for the neural network. This dataset encompassed a wide range of chemical concentrations and conditions, ensuring the neural network could effectively learn the underlying patterns and relationships for real-life scenarios. A simple feedforward architecture was used and fine-tuned with the primary goal of maintaining a low approximation error while also striving to achieve the lowest possible inference timing. After training, the new neural network model was compared to ISORROPIA on ICON-ART simulation data. The results demonstrated that the neural network model successfully achieved the desired outcomes, maintaining low approximation error across the globe and efficient inference timing.**

***Index Terms*—Neural networks, Surrogate modeling, Atmospheric chemistry, ICON-ART**

I. Introduction

In order to properly forecast Earth's climate and weather, an accurate model is required first. The development of an accurate model is an active process that has been ongoing for generations, however, modeling such a complex, chaotic system remains a significant challenge. Thus, scientists have to compromise and use parameterizations and approximations in order to also account for the fact that simulations should complete within a reasonable time frame. Consequently, simulation speed and energy consumption are other focal points when optimizing the model, besides accuracy. A potential way to further enhance the model is by creating a surrogate model of it with neural networks [1]. The core idea behind this approach is that knowledge can be transferred from a large model to a smaller one with minimal loss in performance. The advantages of the smaller model include compatibility with less powerful hardware, faster inference times, and reduced energy consumption.

This is achievable by using neural networks, which are known to be universal function approximators under certain assumptions [2]. In general, neural networks have been used to classify images among thousands of classes [3], beat a champion of the game of Go [4] and very recently also model the weather and make predictions on it to a degree that the prediction was better than traditional models [5]. Several factors have enabled these advancements, but two are particularly noteworthy. First, new powerful hardware and software that is specifically designed for training and making predictions, have emerged. In particular, the new machine learning libraries have lowered the barrier to entry for researchers to perform training, making research more accessible. The second reason, is the abundance of data that is also accessible, and organized properly. Specifically for weather forecasting, the ERA5 dataset [6] by ECMWF was used as the basis for the breakthroughs in the field and played a big role in enabling the use of machine learning in the field. The need for high-quality data in large amounts cannot be overstated, as one of the main pitfalls in training neural nets is overfitting [7]. Overfitting is the phenomenon of having your trained model perform poorly on unseen data—in other words,

© 2025 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

G. Evangelopoulos *et al.*, "Accelerating Weather Forecasting: A Neural Network-Based Emulation of ISORROPIA," in *Proc. IEEE eScience 2025*, pp. 67–74, doi: 10.1109/eScience65000.2025.00017.

the model is unable to make predictions outside the problem space that was covered by the training data. It becomes clear that although these new data driven weather models perform better than the older traditional ones, they are not able to generalize as well. This limitation applies not only to the temporal scope of the models but also to the range of variables they predict.

In this work, taking into consideration that no universal dataset exists, to support the development of a foundation model that covers every demand, an approach is taken to create a surrogate model, called ANNISORROPIA, for one of the smaller scale submodules used in ICON-ART [8]. ICON (Icosahedral Nonhydrostatic) is a global numerical weather prediction model, and ART (Aerosol and Trace gases) is an extension to it that focuses on the atmospheric composition and the interactions of trace gases and aerosols. The submodule in question is called ISORROPIA [9], an aerosol thermodynamic equilibrium model that computes chemical concentrations. There are two main reasons ISORROPIA was picked. The first one is that it takes part in the most time-consuming simulations. The second is that it is also a standalone submodule that can function outside of ART. This makes data generation for training easily available.

One of the benefits of emulating submodules is that researchers are able to freely choose what version of the submodule to run, choosing the machine learning version introduces some approximation error, with speed being the trade-off. Another benefit of this approach is that it still allows the analysis of specific processes while accelerating simulation speed without relying on the creation of a single large model that encompasses everything.

This paper is structured as follows: In Section II a brief overview is given about neural networks and the ISORROPIA program. Section III reviews related work in the field of weather and climate simulation. Section IV outlines the methodology for emulating ISORROPIA, followed by experimental results and discussions in Section V. Finally, Section VI concludes with key insights and future directions.

II. Background

A. ISORROPIA

ISORROPIA [9] is a thermodynamic equilibrium model that predicts the concentration of different chemical species in the solid, liquid, and gas phases. It takes as input the concentrations of eight ions, specifically: sulfate, nitrate, chloride, sodium, ammonium, magnesium, calcium, and potassium as well as the relative humidity (RH) and temperature. Two types of problem cases can be defined: One is called “forward”, where it is assumed that the total gas and aerosol concentrations of sodium, ammonium, nitrate and sulfate are known. The other problem case is called “reverse”, and in that case, only the aerosol concentrations of those species are known. In addition, ISORROPIA allows the user to set the aerosol state to be thermodynamically stable so that salts precipitate if saturation is achieved or in the metastable state so that salts do not precipitate. This means that the concentrations of species in the solid phase are zero, except for calcium sulfate ($CaSO_4$), which, for simplification reasons, is assumed to be completely insoluble and sometimes appears even in metastable scenarios.

B. Neural Networks

Neural networks are universal function approximators [2] that can be used to solve a variety of tasks, such as classification and regression problems. They are inspired by biological neurons [10] with each neural neuron modifying the signal it receives based on weights, which are trainable parameters. A nonlinearity is introduced into the neuron to enrich its expressivity. By connecting neurons together, a network is created and as it grows in size, so does its capacity to represent complex functions.

Although there are many ways to train a network, this work focuses on supervised learning. It is called supervised learning because the ground truth is available and the network can directly learn from it. In order to track how well the model performs a loss function(also referred to as a cost function) is used. This function measures the distance between the predicted outcome and the ground truth with the gradients being shared across all neurons using an algorithm called backpropagation [11]. In simple terms, backpropagation transfers the gradients from the output layer to the input layer using the chain rule as the main component. From that point on, algorithms called optimizers [12], [13] are responsible for updating the weight values using the gradients given to them. The optimizers introduce extra hyperparameters to the problem with the most prominent one being the learning rate, which determines the magnitude of weight updates. A small learning rate leads to slower convergence and the possibility to be trapped in local minima. On the other hand, a big learning rate might lead to faster convergence, but it can also lead to the actual solution being constantly overshot. Therefore, it is common practice for the learning rate to be adjusted throughout training.

In order to avoid the network performing well on training data but poorly on unseen data, regularization is used. There are many ways to avoid this overfitting phenomenon, one such method is weight decay [14]. When using weight decay the magnitude of the weights is gradually reduced over time in order to avoid big values, the way this is done is through adding an extra term to the loss function that penalizes large weights. As a result, the optimizer besides minimizing the loss, also regulates the size of the weights.

III. Related Work

Weather and climate forecasting play an important role in daily life, from scheduling agricultural work to planning aviation operations. This, combined with the inherent difficulty of simulating chaotic systems, necessitates the daily use of significant computing power to meet the demand. As a result, advancements that would speed up or optimize the current models would be very beneficial and machine learning could be one way to achieve this. For example, in the field of medium-range weather forecasting, autoregressive data driven

models have shown the ability to outperform the IFS (Integrated Forecasting System) by ECMWF [15] when trained using the ERA5 dataset, which contains a wide range of atmospheric, land, and oceanic climate variables. In Pangu-Weather [16] a 3D Swin Transformer-based approach was used. The self-attention mechanism of the transformer [17] that was originally showcased as a tool for machines to translate text, was used with great success. Another approach, Graphcast [5], uses a Graph Neural Network (GNN) [18] based approach in an encode-process-decode pipeline [19]. In detail, a GNN encodes information from a grid around the globe to multiple meshes, with each mesh being of a different resolution, also called the multi-mesh. The information encoded into the multi-mesh is processed again by a GNN and finally the decoder GNN maps the processor output back to the original longitude-latitude grid, giving the final prediction. Both of these models are able to give predictions in only a few seconds, with Pangu-Weather being able to generate a 0.25° resolution 24 hour forecast in 1.4 seconds on a single NVIDIA TESLA V100 GPU and Graphcast being able to generate a 10-day forecast at 0.25° resolution on a single TPU v4 device.

One limitation the aforementioned data driven approaches have is that their prediction errors accumulate quickly making their results become unreliable past around 10 days and the longer the simulation lasts, the less natural the prediction looks, which is not the case for the traditional state-of-the-art methods that are more constrained. This limitation can be bypassed by employing a hybrid model that takes advantage of existing traditional models and applying machine learning algorithms to enhance them. This philosophy was used in NeuralGCM (Neural General Circulation Model) [20] that uses a differentiable dynamical core for the dynamical equations and a learned physics module multi-layer perceptron, which follows the encode-process-decode pipeline. The dynamical core simulates large scale fluid motions and thermodynamics, including gravity and Coriolis force, and the multi-layer perceptron predicts the effect of unresolved processes, such as cloud formation, radiative transport, and precipitation. NeuralGCM achieves comparable performance to the previous models for the medium-range forecast category, but its strength is that it performs multiyear simulations albeit at a lower resolution. In comparison to the aforementioned models, it can perform a 70,000-day simulation in 24 hours at 1.4° resolution on a TPU v4 device.

For machine learning-based surrogate models of aerosol and chemical tracers, in [21] using simulations from ICON-ART, an LSTM-based surrogate model called ICONET was built to simulate trace gases. The model receives as input the temperature, pressure, cosine of the solar zenith angle, and the volume mixing ratios of 12 trace gases and outputs the volume mixing ratios of the trace gases at the next time step. ICONET was compared to the ICON-ART chemistry on the same hardware while coupled to ICON-ART and showed a speedup of 3.1, with generally good accuracy for most trace gases.

TABLE I
ISSORROPIA INPUT VALUE RANGES

Input	Value Range
$[Na]^+$	[0, 20] μg/m^3
$[SO_4]^{2-}$	[0, 10] μg/m^3
$[NH_4]^+$	[0, 50] μg/m^3
$[NO_3]^-$	[0, 30] μg/m^3
$[Cl]^-$	[0, 20] μg/m^3
$[Ca]^{2+}$	[0, 10] μg/m^3
$[K]^+$	[0, 10] μg/m^3
$[Mg]^{2+}$	[0, 10] μg/m^3
Relative Humidity	[0, 1]
Temperature	[230, 320] K

TABLE II
LOG-NORMAL SAMPLING PARAMETERS

Input	Mean	Standard Deviation	Upper Limit
$[Na]^+$	0	1.29	30 μg/m^3
$[SO_4]^{2-}$	0	0.99	20 μg/m^3
$[NH_4]^+$	0	1.68	60 μg/m^3
$[NO_3]^-$	0	1.46	40 μg/m^3
$[Cl]^-$	0	1.29	30 μg/m^3
$[Ca]^{2+}$	0	0.99	20 μg/m^3
$[K]^+$	0	0.99	20 μg/m^3
$[Mg]^{2+}$	0	0.99	20 μg/m^3

IV. METHODOLOGY

A. Dataset Generation

To begin, a dataset is necessary for the training to take place. Generating training samples using ISORROPIA requires two components. The first is the configuration of the solution control file. In that control file, the parameters are primarily kept at their default values, except for the accuracy of the activity coefficients, the method of their calculation, and the flag to enforce machine precision for mass conservation. Specifically, the accuracy is changed from 5×10^{-2} to 1×10^{-4} with calculations performed online rather than using lookup tables. This change is made to generate higher-quality training data, which is also why mass conservation is enforced at machine precision. The second requirement is a file that will contain the problem specifications and the inputs from which the output values will be generated. For the problem specifications, the parameters are set to the forward metastable case since ICON-ART assumes that aerosols exist in either the liquid or gas state and lastly, concentrations are computed in μg/m^3. The outputs to be predicted are 19 and can be seen in Table III. Generating the input sample must be done carefully as it is important for the input values to be realistic and also broad enough to cover most plausible scenarios found in nature. After gathering information from previous ICON-ART simulations and existing literature [22], [23], a data range was established as shown in Table I. Ideally, values would be sampled from a uniform distribution, but because the combinations are infinite and it is known that in most cases the concentration of the chemical elements around the globe is relatively low, a log-normal distribution was used. The parameters picked can be

TABLE III
NEURAL NETWORK OUTPUT DESCRIPTIONS

Code Name	Description
GNH3	Gas phase NH_3
GHCL	Gas phase HCl
GHNO3	Gas phase HNO_3
CCASO4	Solid phase $CaSO_4$
HLIQ	Aqueous phase H^+
NALIQ	Aqueous phase Na^+
NH4LIQ	Aqueous phase NH_4^+
CLLIQ	Aqueous phase Cl^-
SO4LIQ	Aqueous phase SO_4^{2-}
HSO4LIQ	Aqueous phase HSO_4^-
NO3LIQ	Aqueous phase NO_3^-
CaLIQ	Aqueous phase Ca^{2+}
KLIQ	Aqueous phase K^+
MgLIQ	Aqueous phase Mg^{2+}
NH4AER	NH_3
CLAER	HCl
NO3AER	HNO_3
WATER	Aerosol water content
LMASS	Aerosol aqueous phase mass

seen in Table II and they were set in a way that there is a 99% probability that the value sampled will be from the expected range. Values higher than the expected range were allowed with the idea that it would help the model to perform better if it would ever have to predict them, still a margin of 10 μg/m^3 was allowed with values higher than that were clipped to the upper limit. Finally, 2 million synthetic training samples were generated by feeding these sampled inputs into ISORROPIA. As a last step, 10% of the samples were reserved for validation and the 90% remaining for training. The validation dataset consists of a set of samples that are not used for training. The goal is to monitor the performance of the neural network on both datasets simultaneously and if the metrics on the training set continue to improve while the opposite is observed on the validation dataset, this indicates overfitting and an inability to generalize, prompting the termination of the training process.

B. Data Preprocessing

In machine learning training, it is recommended to normalize the input data [11]. This is because larger values disproportionately influence gradient updates and, as a result, not all features are treated with equal importance. For this work, the operational value ranges are known, so min-max scaling normalization can be applied so that the smallest input value is 0 and the largest one is 1. Before that, one additional preprocessing step is performed: all input values smaller than 10^{-10} μg/m^3 are set to zero, as such small concentrations are considered negligible for this work. This also addresses one more case in ISORROPIA, which occasionally outputs negative concentration values. After contacting one of the authors of the ISORROPIA publication, this was confirmed to be a known bug, and it was suggested that the negative values to be set to zero. The training and testing data can be found at [24].

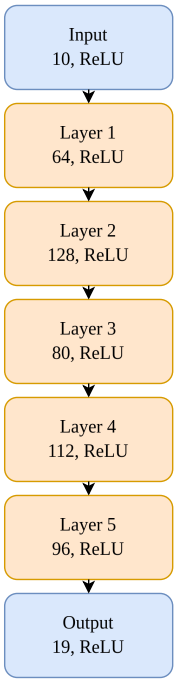

Fig. 1. Architecture of ANNISORROPIA. In each layer, the number of nodes and the activation function used, is shown.

C. Architecture and Hyperparameters

For the neural network architecture, a simple feedforward neural network was used. The relatively low dimensionality of the problem allows for a simple implementation with a low number of parameters, which is desirable for achieving maximum speed-up. For the training, the AdamW [25] optimizer was used. AdamW is a variant of the Adam optimizer with weight decay. The purpose of weight decay during training is to hinder the model from overfitting as much as possible and as a result to generalize better. Furthermore, as an activation function to introduce nonlinearities, the rectified linear unit (ReLU) was used, which is a function that outputs 0 for negative inputs and behaves like a linear function otherwise [26]. Notably, using a ReLU activation at the output ensures concentrations cannot be negative. For the number of hidden layers as well as the number of nodes per layer a random hyperparameter search was conducted, which showed that networks with around 5 layers tend to perform the best and the best architecture can be seen in Fig. 1.

Having established the architecture's shape and size, the last thing remaining is the learning rate schedule for the optimizer which will be done based on training epochs (where one epoch is a complete pass through all training samples). At the beginning, this value is set to 1×10^{-3}, then after 25 epochs of training if the loss error has not improved by at least 1×10^{-2}, the learning rate gets halved up to a minimum learning rate of 1×10^{-6}. Finally, if no improvement is observed over 25 epochs the training terminates and the weights that performed best on the validation dataset are saved.

TABLE IV
NEURAL NETWORK: TRAINING R^2 SCORES

Code Name	R^2 Score
GNH3	0.9999
GHCL	0.9981
GHNO3	0.8704
CCASO4	0.9999
HLIQ	0.9999
NALIQ	0.9989
NH4LIQ	0.9987
CLLIQ	0.9977
SO4LIQ	0.9915
HSO4LIQ	0.9775
NO3LIQ	0.9960
CaLIQ	0.9997
KLIQ	0.9985
MgLIQ	0.9450
NH4AER	0.9623
CLAER	0.9977
NO3AER	0.9960
WATER	0.9883
LMASS	0.9891

D. Metrics

The loss function used is the mean absolute error (MAE), which is defined as follows:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{1}$$

where y_i is the true value, $\hat{y}_i$ is the predicted value, and n is the number of samples.

In this case, the mean absolute error refers to the prediction difference of concentrations, making the results more interpretable for domain experts.

Additionally, the R^2 score was also used. This metric, also known as the coefficient of determination, evaluates how well the regression model fits the data. A value of 1 indicates a perfect fit, a value of 0 means the model performs no better than predicting the mean of the target values, and a negative value indicates that the model performs worse than simply predicting the mean. The R^2 score is defined as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{2}$$

where y_i is the true value, $\hat{y}_i$ is the predicted value, $\bar{y}$ is the mean of the true values, and n is the number of samples.

E. Technical Implementation

The model was trained using Keras [27] with Tensorflow [28] as the backend. Meanwhile, the training and evaluation of the model took place on the HPC center HoreKa in Karlsruhe. For the training one node was used were one node consists of 4 NVIDIA A100 GPUs with 40GB of memory and two CPU sockets containg two Intel Xeon Platinum 8368. ISORROPIA was compiled using the 8.5 version of the gfortran compiler.

V. RESULTS AND DISCUSSION

In this section, the results of the training are showcased and analyzed based on the metrics that were mentioned in Section 4. In addition, a comparison will be made to see how fast each model can make predictions. Lastly, data from an ICON-ART simulation will be used for a one time step comparison between ANNISORROPIA and ISORROPIA.

A. Training Results

In Table IV we can see that the R^2 scores for all 19 outputs are very high which means that this architecture is sufficiently accurate to capture the synthetic data used for training, with a slightly lower score for the gas phase HNO3 output.

The mean absolute error on Fig. 2 shows that an accuracy of nearly one decimal place is achieved on average, but there are a few outputs with a bigger μg/m^3 error. Specifically, the outputs GHNO3, MgLIQ, WATER, and LMASS have a greater than 0.1 mean error. In Fig. 3 the standard deviation of the prediction error is shown. Generally the deviation is low for most chemical species besides the outputs that were highlighted already in the previous paragraph for having a higher mean absolute error. Considering the skewness of the input data and the outliers that this would cause, a larger value for the standard deviation than the mean was expected. To further investigate the outliers, a boxplot was created in Fig. 4, that showed the relative error for each prediction based on its actual value. In some cases, for the water and liquid total concentrations there are outliers that cause a relative error of almost 10^{10}. Looking closer at those cases, it happens when the true values are on the order of 10^{-9}. Across all predictions, the percentage of instances where the relative error for the water aerosol concentration is higher than 0.25 is 4.32% and respectively for the total liquid concentration the result is 2.8% of the samples. The training results show that ANNISORROPIA is able to make reasonably accurate predictions for all 19 outputs, but there are cases that the neural network misjudges the presence of water.

Having analyzed the accuracy of the network, a speed comparison was conducted to measure how long it takes to make predictions for all the samples created. For testing both models, all 76 cores across the two Intel Xeon Platinum 8368 were used. Although it is common to make predictions with a GPU for neural networks, in this case it was deemed more appropriate to not use one, as ICON-ART does not support their use, as of the time of writing. An added benefit of this approach is that the comparison results are easier to interpret as the same hardware was used for both models. It should be noted that ISORROPIA was executed as a compiled Fortran executable, while ANNISORROPIA used the Keras library in Python for the predictions. The results show that after five runs for each model, ISORROPIA took 315.01 ± 0.50 seconds and ANNISORROPIA took 0.52 ± 0.05 seconds. This means that for this test case the neural network was approximately 606x faster.

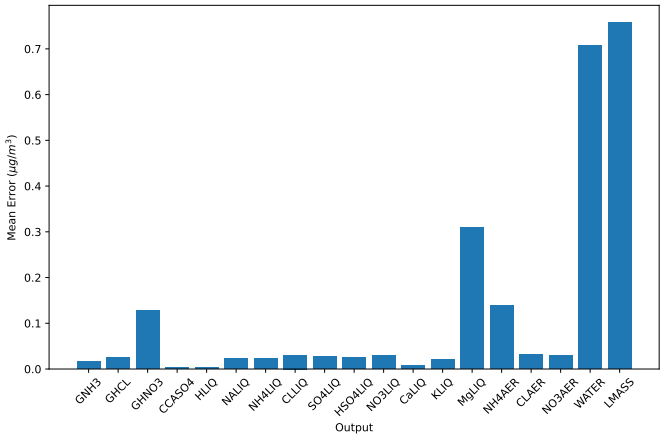

Fig. 2. Absolute mean error of the neural network for the training phase, for each individual output.

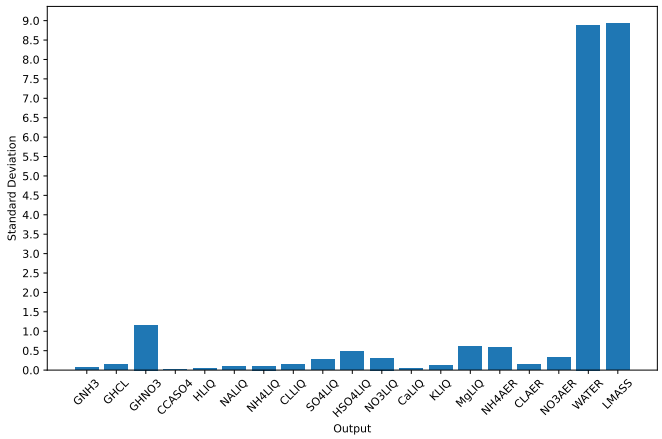

Fig. 3. Standard deviation of the error from the neural network predictions during the training phase.

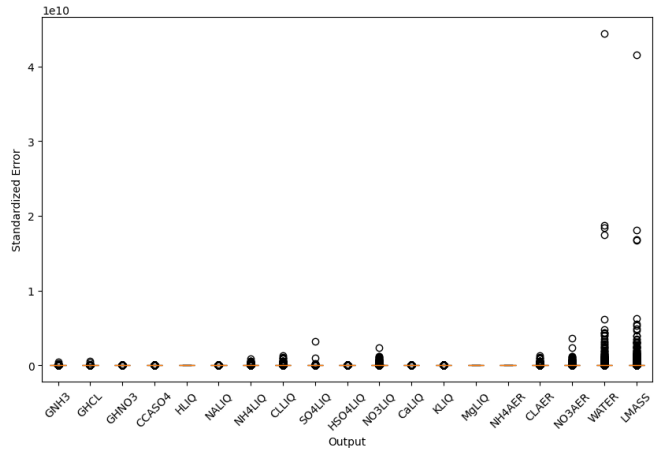

Fig. 4. Each sample's standardized prediction error for the training phase.

B. ICON-ART Test Case

To further investigate the efficiency of the new model, simulation data was taken from ICON-ART and used as input for ANNISSOROPIA and ISORROPIA. The simulation data is found on a global grid with 90 height levels (level 90 being the lowest level) and for each height level, it contains 259,920 data points, consisting of 720 longitude and 361 latitude levels. Because the relative humidity data were missing from the simulation results and ISORROPIA requires that data, three test cases were created where the relative humidity was set to 0.4, 0.6 and 0.8. This test suite, with the varying relative humidity values makes it easier to gain better insight into how the model performs across different humidity levels. For the comparison, altitude levels 85, 80 and 75 were selected, which correspond to distances of approximately 436, 1,255 and 2,432 meters above ground, respectively. The reason these altitudes were chosen is that the presence of aerosols was more prominent at those levels compared to higher altitudes. One last thing to note about this comparison is that in all test samples the values of calcium, potassium and magnesium are zero as they were not relevant for the simulation. This provides a valuable test case for evaluating how ANNISORROPIA handles unseen data, as there were no samples where all these values were zero.

Comparing the R^2 values between Table V, Table VI, and Table VII, ANNISORROPIA scores very well for the majority of the outputs, especially the predictions for the total water concentration. There are also a couple of trends worth pointing out. Specifically, the predictions of a few outputs tend to become less accurate the lower the relative humidity gets and they also get worse as the distance from the ground increases. Specifically, the outputs of the gas phase HNO3, the solid phase CaSO4 and aqueous phase HSO4. Upon further inspection, what usually happens in these incorrect predictions, is that ANNISORROPIA predicts the presence of a species when the actual concentration is zero, or fails to predict it when it should be present.

C. Discussion

The feedforward architecture used was able to fit properly on the synthetic data used for its training. This is very encouraging because the bigger the network, the longer predictions take and the more energy is consumed. Another added benefit of the small size is that training and tuning only require a few hours and do not depend on state-of-the-art hardware to do so. This aligns well with the idea that if a researcher needs to use it for a more specialized case it would be easy to retrain. Even though ANISSOROPIA performs well there are a few points that warrant further attention. One point is that the model misjudges the existence of species in a few rare cases. How common this issue is remains unclear. It will be tested in future work that aims to integrate ANNISORROPIA in ICON-ART as an alternative to ISORROPIA when a faster simulation speed is required. One possible remedy for this issue is to use a specialized loss function that has an extra term which penalizes such errors, taking a similar approach to [21]. The

second point that requires attention is the fact that predictions seem to be slightly worse the higher the altitude and the lower the relative humidity. This trend is currently attributed to the difficulty of predicting values close to zero. Therefore, using ANNISORROPIA at higher altitudes, for example in the stratosphere, would require further fine-tuning, which because of the small size of the network, should be very fast to train.

Overall, considering the 606x the speed-up achieved, this is a promising way to attempt to accelerate the performance of preexisting models by creating surrogate models of them with the help of machine learning. A closer look at Fig. 5 shows that ANNISORROPIA is able to capture the concentration of the species around the globe, but in Fig. 6 it is shown that it can also introduce slight misjudgments globally. Nonetheless, it is important to emphasize that such models will continue to improve the more data are gathered and larger datasets are assembled, making this approach applicable across various fields.

TABLE V
R^2 VALUES FOR TEST PREDICTIONS AT RH = 0.8

Output	Level 85	Level 80	Level 75
GNH3	0.9992	0.9932	0.9781
GHCL	0.9995	0.9966	0.9847
GHNO3	0.9769	0.7230	0.2107
CCASO4	0.9819	0.8200	-0.4147
HLIQ	1.0000	1.0000	0.9999
NALIQ	0.9994	0.9982	0.9948
NH4LIQ	0.9997	0.9979	0.9910
CLLIQ	0.9999	0.9997	0.9993
SO4LIQ	0.9995	0.9934	0.9750
HSO4LIQ	0.9879	0.8457	0.5754
NO3LIQ	0.9974	0.9675	0.8599
CaLIQ	1.0000	1.0000	0.9998
KLIQ	0.9990	0.9910	0.9464
MgLIQ	1.0000	1.0000	1.0000
NH4AER	0.9960	0.9549	0.7622
CLAER	0.9999	0.9997	0.9993
NO3AER	0.9973	0.9665	0.8574
WATER	0.9998	0.9994	0.9986
LMASS	0.9998	0.9994	0.9986

TABLE VI
R^2 VALUES FOR TEST PREDICTIONS AT RH = 0.6

Output	Level 85	Level 80	Level 75
GNH3	0.9987	0.9896	0.9691
GHCL	0.9993	0.9948	0.9793
GHNO3	0.9440	0.4736	0.0659
CCASO4	0.9733	0.7169	-1.2772
HLIQ	1.0000	0.9999	0.9999
NALIQ	0.9997	0.9991	0.9975
NH4LIQ	0.9992	0.9952	0.9804
CLLIQ	0.9999	0.9997	0.9993
SO4LIQ	0.9994	0.9936	0.9778
HSO4LIQ	0.9699	0.6776	0.3464
NO3LIQ	0.9959	0.9454	0.7191
CaLIQ	1.0000	0.9997	0.9991
KLIQ	0.9996	0.9984	0.9933
MgLIQ	1.0000	1.0000	1.0000
NH4AER	0.9918	0.9194	0.6575
CLAER	0.9999	0.9997	0.9993
NO3AER	0.9958	0.9437	0.7141
WATER	0.9997	0.9992	0.9974
LMASS	0.9998	0.9994	0.9982

TABLE VII
R^2 VALUES FOR TEST PREDICTIONS AT RH = 0.4

Output	Level 85	Level 80	Level 75
GNH3	0.9982	0.9869	0.9611
GHCL	0.9986	0.9874	0.9361
GHNO3	0.9112	0.3320	0.0257
CCASO4	0.9705	0.6896	-1.3913
HLIQ	1.0000	0.9999	0.9999
NALIQ	0.9996	0.9988	0.9967
NH4LIQ	0.9984	0.9901	0.9646
CLLIQ	0.9998	0.9994	0.9981
SO4LIQ	0.9985	0.9911	0.9780
HSO4LIQ	0.9514	0.5560	0.2339
NO3LIQ	0.9951	0.9401	0.5494
CaLIQ	0.9995	0.9951	0.9644
KLIQ	0.9997	0.9992	0.9961
MgLIQ	1.0000	1.0000	1.0000
NH4AER	0.9877	0.8887	0.6125
CLAER	0.9998	0.9994	0.9981
NO3AER	0.9950	0.9382	0.5395
WATER	0.9997	0.9989	0.9972
LMASS	0.9998	0.9992	0.9976

VI. CONCLUSION

Over the years hardware has consistently improved. This in combination with the abundance of data that is generated and collected makes machine learning a valuable tool to solve a variety of problems. In particular, neural networks offer promising opportunities to accelerate long-term climate simulations.

Motivated by this potential, this work introduced a surrogate model, ANISSORROPIA, which was trained to emulate one of the submodules found in the global atmospheric model ICON-ART. By emulating a submodule, a researcher has the option to choose between the original version and a faster, machine learning-based alternative that introduces a small approximation error to the prediction, but performs the predictions much faster. This hybrid modular approach allows for a lot of flexibility and leverages existing infrastructure. ANNISORROPIA was trained using synthetic data and was able to emulate ISORROPIA's outputs for a rich range of input combinations. The neural network also demonstrated robustness when making predictions for unseen data taken from an ICON-ART simulation, but a few cases were observed were the model misjudged the actual concentration of a few select chemical species.

As the next step, the network will be integrated into the ICON-ART codebase and further testing will be conducted to evaluate the overall acceleration when used alongside other submodules. Special attention will be given to assessing how the added approximation affects the stability of the long-lasting simulations. Furthermore, additional submodules will be emulated, aiming to achieve an even greater speed-up. This will also open up opportunities to explore different network architectures in combination with potential constraints to help preserve physical properties. Overall, there are several promising directions to consider further that could enhance weather and climate modeling.

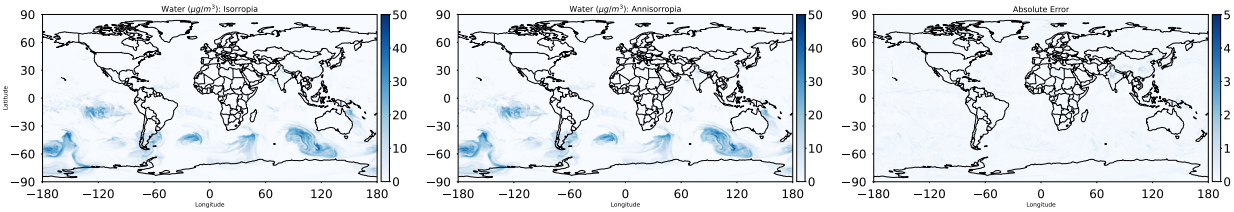

Fig. 5. A comparison of ISORROPIA's water aerosol concentration prediction on the left, and ANNISORROPIA's prediction in the middle, with the absolute error being shown on the right. This test case is for RH=0.6 at height level 80. The maximum plot value shown, is lowered on the right plot to improve the visibility of the prediction difference.

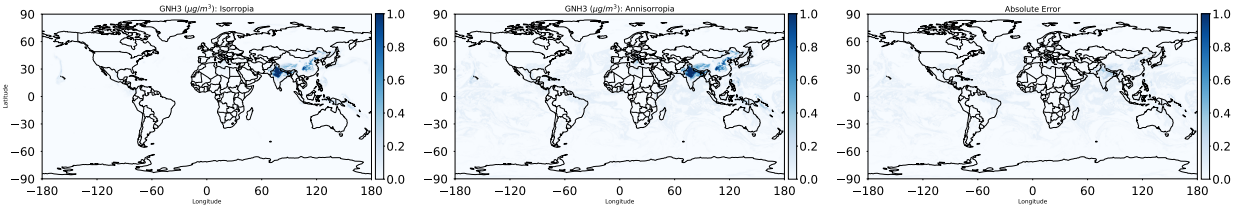

Fig. 6. A comparison of ISORROPIA's gas phase NH_3 concentration prediction on the left, and ANNISORROPIA's prediction in the middle, with the absolute error being shown on the right. This test case is for RH=0.6 at height level 80.

ACKNOWLEDGMENT

The authors gratefully acknowledge the computing time provided on the high-performance computer HoreKa by the National High-Performance Computing Center at KIT (NHR@KIT). This center is jointly supported by the Federal Ministry of Education and Research and the Ministry of Science, Research and the Arts of Baden-Württemberg, as part of the National High-Performance Computing (NHR) joint funding program (https://www.nhr-verein.de/en/our-partners). HoreKa is partly funded by the German Research Foundation (DFG). The authors are also very thankful to Prof. Athanasios Nenes and Dr. Christos Fountoukis for providing the source code of ISORROPIA and answering questions that arose during the development of ANNISORROPIA.

REFERENCES

[1] G. Hinton, O. Vinyals, and J. Dean, "Distilling the Knowledge in a Neural Network," arXiv preprint arXiv:1503.02531, 2015.

[2] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals and Systems*, vol. 2, no. 4, pp. 303–314, Dec. 1989.

[3] H. Pham, Z. Dai, Q. Xie, and Q. V. Le, "Meta Pseudo Labels," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Nashville, TN, USA: IEEE, Jun. 2021, pp. 11 552–11 563.

[4] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016.

[5] R. Lam, A. Sanchez-Gonzalez, M. Willson, P. Wirnsberger, M. Fortunato, F. Alet, S. Ravuri, T. Ewalds, Z. Eaton-Rosen, W. Hu, A. Merose, S. Hoyer, G. Holland, O. Vinyals, J. Stott, A. Pritzel, S. Mohamed, and P. Battaglia, "Learning skillful medium-range global weather forecasting," *Science*, vol. 382, no. 6677, pp. 1416–1421, 2023.

[6] H. Hersbach, B. Bell, P. Berrisford, G. Biavati, A. Horányi, J. Muñoz Sabater, J. Nicolas, C. Peubey, R. Radu, I. Rozum, D. Schepers, A. Simmons, C. Soci, D. Dee, and J.-N. Thépaut, "ERA5 hourly data on single levels from 1940 to present." [Online]. Available: https://cds.climate.copernicus.eu/datasets/reanalysis-era5-single-levels?tab=overview

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, F. Pereira, C. J. Burges, L. Bottou, and K. Q. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012.

[8] J. Schröter, D. Rieger, C. Stassen, H. Vogel, M. Weimer, S. Werchner, J. Förstner, F. Prill, D. Reinert, G. Zängl, M. Giorgetta, R. Ruhnke, B. Vogel, and P. Braesicke, "ICON-ART 2.1: a flexible tracer framework and its application for composition studies in numerical weather forecasting and climate simulations," *Geoscientific Model Development*, vol. 11, no. 10, pp. 4043–4068, 2018.

[9] C. Fountoukis and A. Nenes, "ISORROPIA II: a computationally efficient thermodynamic equilibrium model for K^+–Ca^{2+}–Mg^{2+}–NH_4^+–Na^+–SO_4^{2-}–NO_3^-–Cl^-–H_2O aerosols," *Atmospheric Chemistry and Physics*, vol. 7, no. 17, pp. 4639–4659, 2007.

[10] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, Dec. 1943.

[11] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient BackProp," in *Neural Networks: Tricks of the Trade: Second Edition*, G. Montavon, G. B. Orr, and K.-R. Müller, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 9–48.

[12] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proceedings of the 30th International Conference on Machine Learning*, May 2013, pp. 1139–1147.

[13] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv preprint arXiv:1412.6980*, Jan. 2017.

[14] A. Krogh and J. Hertz, "A Simple Weight Decay Can Improve Generalization," in *Advances in Neural Information Processing Systems*, J. Moody, S. Hanson, and R. P. Lippmann, Eds., vol. 4. Morgan-Kaufmann, 1991.

[15] C. D. Roberts, R. Senan, F. Molteni, S. Boussetta, M. Mayer, and S. P. E. Keeley, "Climate model configurations of the ECMWF Integrated Forecasting System (ECMWF-IFS cycle 43r1) for HighResMIP," *Geoscientific Model Development*, vol. 11, no. 9, pp. 3681–3712, 2018.

[16] K. Bi, L. Xie, H. Zhang, X. Chen, X. Gu, and Q. Tian, "Accurate medium-range global weather forecasting with 3D neural networks," *Nature*, vol. 619, no. 7970, pp. 533–538, Jul. 2023.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.

[18] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How Powerful are Graph Neural Networks?" *arXiv preprint arXiv:1810.00826*, Feb. 2019.
[19] T. Pfaff, M. Fortunato, A. Sanchez-Gonzalez, and P. W. Battaglia, "Learning Mesh-Based Simulation with Graph Networks," *arXiv preprint arXiv:2010.03409*, Jun. 2021.
[20] D. Kochkov, J. Yuval, I. Langmore, P. Norgaard, J. Smith, G. Mooers, M. Klöwer, J. Lottes, S. Rasp, P. Düben, S. Hatfield, P. Battaglia, A. Sanchez-Gonzalez, M. Willson, M. P. Brenner, and S. Hoyer, "Neural general circulation models for weather and climate," *Nature*, vol. 632, no. 8027, pp. 1060–1066, Aug. 2024.
[21] E. Azmi, J. Meyer, M. Strobl, M. Weimer, and A. Streit, "Approximation and Optimization of Global Environmental Simulations with Neural Networks," in *Proceedings of the Platform for Advanced Scientific Computing Conference*, ser. PASC '23. New York, NY, USA: Association for Computing Machinery, Jun. 2023, pp. 1–11.
[22] J. H. Seinfeld and S. N. Pandis, *Atmospheric Chemistry and Physics: From Air Pollution to Climate Change*, 3rd ed. Hoboken, NJ, USA: Wiley, 2016.
[23] J. Bruckert, L. Hirsch, Á. Horváth, R. A. Kahn, T. Kölling, L. O. Muser, C. Timmreck, H. Vogel, S. Wallis, and G. A. Hoshyaripour, "Dispersion and Aging of Volcanic Aerosols After the La Soufrière Eruption in April 2021," *Journal of Geophysical Research: Atmospheres*, vol. 128, no. 8, p. e2022JD037694, 2023.
[24] G. Evangelopoulos, "Annisorropia training and test data," Aug. 2025. [Online]. Available: https://doi.org/10.5281/zenodo.16791151
[25] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," *arXiv preprint arXiv:1711.05101*, Jan. 2019.
[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 2017.
[27] F. Chollet *et al.*, "Keras," https://keras.io, 2015.
[28] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," 2015. [Online]. Available: https://www.tensorflow.org/