Stable climate simulations using a realistic GCM with neural network parameterizations for atmospheric moist physics and radiation processes

Xin Wang[1], Yilun Han[2], Wei Xue[1], Guangwen Yang[1], Guang J. Zhang[3]

[1]Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China
[2]Department of Earth System Science, Tsinghua University, Beijing, 100084, China
[3]Scripps Institution of Oceanography, La Jolla, CA USA

Correspondence to: Wei Xue (xuewei@tsinghua.edu.cn), Yilun Han (hanyl16@mails.tsinghua.edu.cn)

Abstract. In climate models, subgrid parameterizations of convection and cloud are one of the main reasons for the biases in precipitation and atmospheric circulation simulations. In recent years, due to the rapid development of data science, Machine learning (ML) parameterizations for convection and clouds have been proven the potential to perform better than conventional parameterizations. At present, most of the existing studies are on aqua-planet and idealized models, and the problems of simulated instability and climate drift still exist. In realistic configurated models, developing a machine learning parameterization scheme remains a challenging task. In this study, a set of deep residual neural networks (ResDNNs) with strong nonlinear fitting ability is designed to emulate a superparameterization (SP) with different types of outputs. Sensitivity tests show that high accuracy is necessary to develop a stable ML parameterization. Trial-and-error is used to acquire the optimal ResDNN set for both high performance and long-term stability, named NN-Parameterization. In offline validation, NN-Parameterization emulates the SP results far better than the conventional subgrid parameterizations. Then, in the multi-year prognostic test, NN-Parameterization reproduces reasonable climate mean states but still with some biases. Most importantly, NN parameterization successfully reproduces the climate variability in a superparameterizated GCM, with an over 30-time faster running speed. Under real geographical boundary conditions, the hybrid ML-physical GCM well simulates the spatial distribution of boreal summer precipitation and significantly improves the frequency of precipitation extremes, which is largely underestimated in the Community Atmospheric Model version 5 (CAM5) with the horizontal resolution of 1.9°×2.5°. Furthermore, the hybrid ML-physical GCM simulates a stronger signal of the Madden-Julian oscillation with a more reasonable propagation speed than CAM5. This study is a pioneer to achieve multi-year stable climate simulations using a hybrid ML-physical GCM in actual land-ocean boundary conditions. It demonstrates the emerging potential for using machine learning parameterizations in climate simulations.

1 Introduction

The general circulation models (GCMs) have been widely used for studying climate variability, prediction and projections. Despite decades of GCM development, most GCMs still suffer from many systematic biases, especially at low latitudes. A prominent tropical bias in most current GCMs is the double intertropical convergence zone (ITCZ) syndrome, which is characterized by two parallel zonal bands of annual precipitation straddling the equator over the central and eastern Pacific (Lin, 2007; Zhang et al., 2019). Convectively coupled equatorial waves and the Madden-Julian Oscillation (MJO), featured by eastward propagating convective cloud clusters, are also not well simulated in GCMs (Ling et al., 2017; Cao and Zhang, 2017).

Many studies have attributed most of these biases to the imperfection of the parameterization schemes for atmospheric moist convection and cloud processes in current GCMs (Zhang and Song, 2010; Cao and Zhang, 2017; Song and Zhang, 2018; Zhang and Song, 2019). Cloud-related processes span a large range of spatial scales, from micron-scale cloud nucleation, meter-scale turbulence, to individual convective cells and organized convective systems, which are a few kilometers to hundreds of kilometers in size, and to tropical disturbances, which have a spatial scale of thousands of kilometers. They directly influence the radiation balance and hydrological cycle of the earth system and interact with the atmospheric circulation, affecting the transport and distribution of energy (Emanual et al., 1994). Therefore, it is very important to simulate the cloud and convection process in GCMs correctly. However, the current GCMs used for climate simulation have a horizontal resolution of ~100km and a vertical hydrostatic coordinate. Thus, in most GCMs, besides parameterized cloud microphysics, convection and its influence on the atmospheric circulation are represented by convective parameterization schemes, which are usually based on simplified theories, limited observations, and empirical relationships (Tiedtke, 1989; Zhang and McFarlane, 1995; Lopez-Gomez et al., 2020). Those schemes regard convective heat and moisture transport as the collective effects of idealized individual kilometer-scale convective cells. They cannot represent the effects of many complicated convective structures, including organized convective systems, leading to large uncertainties and biases in climate simulations (Bony et al., 2015).

Cloud Resolving Models (CRMs), on the other hand, have long been used to simulate convection. Because CRMs have higher horizontal and vertical resolutions and can explicitly resolve the thermodynamic processes in convection, they simulate convection more accurately, including convective organization (Feng et al., 2018). In recent years, CRMs have been used as superparameterization (SP) in low-resolution GCMs to replace conventional cumulus convection and cloud parameterization schemes. The commonly used SP model is the superparameterized version of the Community Atmosphere Model (SPCAM) developed by the National Center for Atmospheric Research (Grabowski and Smolarkiewicz, 1999; Grabowski, 2001, 2004; Khairoutdinov and Randall, 2001; Randall et al., 2003; Khairoutdinov et al., 2005). Compared with conventional cumulus convection and cloud parameterization schemes, SPCAM performs better in simulating mesoscale convective systems, diurnal cycles of precipitation, monsoons, precipitation frequency distribution, and MJOs (Khairoutdinov et al., 2005; Bretherton et al., 2014; Jiang et al., 2015; Jin et al., 2016; Kooperman et al., 2016). However, when using 2D CRM as SP, the improvement on climate mean states is not obvious (Khairoutdinov et al., 2005). Also, SPCAM requires far more computing resources than

the same resolution CAM in 1 to 2 orders of magnitude according to the resolution of the CRM subdomain. Thus, the use of SPCAM in long-term climate simulations and ensemble prediction is restricted by the current computing resource. Developing novel and computationally efficient schemes for high performance convection and cloud processes is still an open problem in GCM development.

In the last 5 years, the rapid development of machine learning (ML) technologies, especially deep learning technologies such as neural networks (NNs), has provided novel approaches to constructing parameterization schemes. Machine learning can identify and discover complex nonlinear relationships that exist in large data sets and model them. Several studies have used machine learning methods to develop convection and cloud parameterization schemes (e.g., Gentine et al., 2018; Rasp et al., 2018). These studies followed a similar approach. The first step is to derive a target dataset from a reference simulation, which is later used for machine learning models training. Then, the trained machine learning models are often evaluated offline against other independent reference simulations and finally implemented in a GCM to replace the conventional parameterization schemes.

Krasnopolsky et al. (2013) first proposed a proof-of-concept for developing convection parameterization based on the NN technique. Specifically, an ensemble of shallow NNs was applied to learn convective temperature and moisture tendencies, with training data from CRM simulations forced by observations in the tropical western Pacific. The resulting convective parameterization scheme was able to simulate the main features of cloud and precipitation in the NCAR CAM4 diagnostically. However, the key issue of prognostic validation in 3-D GCMs was not addressed. Recent studies have investigated ML parameterizations in prognostic mode in simplified aqua-planet GCMs. For example, Rasp et al. (2018) developed a deep fully connected NN (DNN) to predict convection and clouds, which was trained with the data from an aqua-planet SPCAM. The NN parameterization was then implemented in the corresponding aqua-planet CAM and produced multi-year prognostic results close to SPCAM. For this NN parameterization, Rasp (2020) found that minor changes, either to the training dataset or in the input/output vectors, can lead to model integration instabilities. Brenowitz and Bretherton (2019) fitted a DNN for convection and clouds to the coarse-grained data from a near-global aqua-planet cloud-resolving simulation using the System for Atmospheric Modeling (SAM). The NN scheme was then tested prognostically in a coarse-grid SAM. Their results showed that there were unphysical correlations learned by the network, and information in the upper levels from the input vector had to be removed to produce stable long-term simulations. Rather than using NNs, Yuval and O'Gorman (2020) used random forest to develop an ML parameterization based on the training data from a high-resolution idealized 3-D model with a setup of equatorial beta plane. They used two independent random forests to emulate different processes separately and ensured physical constraints by predicting subgrid fluxes instead of tendencies. Later, Yuval et al. (2021) completed the same task with NNs. Both works achieved stable simulations in coarse resolution aqua-planet GCMs. Brenowitz et al. (2020) proposed methods to interpret and stabilize ML parameterization of convection. In their work, a wave spectra analysis tool was introduced to explain why ML coupled GCMs blew up.

In real-world climate models with varied underlying surfaces, convection and clouds are more diverse under different climate backgrounds, which makes the task of developing ML-based parameterizations more complicated. A few early works

have shown the feasibility of using neural networks fitting cloud processes in real-world models. Han et al. (2020) used a 1-D deep residual convolutional neural network (ResNet) to emulate moist physics in SPCAM. This ResNet based parameterization fitted the targets with high accuracy and is successfully implemented in a single column model. Mooers et al. (2021) got a high-skill DNN via automated machine learning technique and forced an offline land model with DNN emulated atmospheric fields. However, neither of these studies have tested their NNs prognostically for long-term simulations. Similar to the idea of several NNs for different processes in Yuval and O'Gorman (2020), this study uses a set of NNs to emulate convection and cloud processes in SPCAM with an actual global land-ocean distribution. We use the residual connections in Han et al. (2020) to acquire super deep neural networks with great nonlinear fitting ability. Furthermore, we conduct systematic trial-and-error to filter out unstable NN parameterizations and get the best ResDNN set with both accuracy and long-term stability. The NN parameterization scheme is then implemented in the realistically configurated CAM to obtain long-term stable simulations. Technically, NNs are commonly implemented via high-level programming languages such as Python and deep learning libraries. However, GCMs are mainly written in Fortran, making integrating with deep learning algorithms inconvenient. Therefore, we introduce an NN-GCM coupling platform in which NN models and GCMs can interact through data transmission. This coupling strategy can facilitate the development of ML-physical hybrid models with high flexibility. Under real-geography boundary conditions, our work achieves more than 10-year stable climate simulations in Atmospheric Model Intercomparison Project (AMIP)-style experiments by using a hybrid ML-physical GCM. The simulation results may show some biases in climate mean fields but successfully reproduce variability in SPCAM. To our knowledge, this is the first time a decade-long stable real-world climate simulation is achieved with a NN-based parameterization.

The remainder of this paper is organized as follows. Section 2 briefly describes the model, the experiments, the DNN algorithm, and the DNN-GCM coupling platform. Section 3 analyses the simulation stability of NNCAM. Section 4 presents the offline validation of the DNN scheme, focusing on the output temperature and moisture tendencies. Results of multi-year simulations, employing the DNN parameterization scheme, are shown in Section 5. A summary and conclusions are presented in Section 6.

2 Methods and data

In this study, we choose SPCAM as the reference model to generate target simulations. A set of NNs is trained with the target simulation data using optimized hyperparameters. Then, they are organized as a subgrid physics emulator and implemented into the superparameterized version of Community Atmospheric Model (SPCAM), replacing both the CRM based SP and the radiation effects of the CRM. This NN-enabled GCM is referred to as NNCAM hereafter.

2.1 SPCAM setup and data generation

The GCMs used in this study are the CAM5.2 developed by the National Center for Atmospheric Research and its superparameterized version SPCAM (Khairoutdinov and Randall, 2001; Khairoutdinov et al., 2005). A complete description

of CAM5 is given by Neale et al. (2012). The dynamic core of CAM5 has a horizontal resolution of 1.9°×2.5° and 30 vertical levels with a model top at about 2 hPa. To represent moist processes, CAM5 adopts a plume-based treatment of shallow convection (Park and Bretherton, 2009), a mass-flux parameterization scheme for deep convection (Zhang and McFarlane, 1995), and an advanced two-moment representation of cloud microphysical processes (Morrison and Gettelman, 2008; Gettelman et al., 2010). In the AMIP experiments we conducted, CAM5 is coupled to a land surface model Community Land Model version 4.0 (Oleson et al., 2010) and uses prescribed sea surface temperatures and sea ice concentrations.

In this study, SPCAM is used to generate the training data. In SPCAM, a two-dimensional (2-D) CRM is embedded in each grid column of the host CAM as SP. The 2-D CRM has 32 grid points in the zonal direction and 30 vertical levels that are shared with the host CAM. The CRM handles convection and cloud microphysics to replace the conventional parameterization schemes, and the radiation is calculated on the CRM subgrids to include the cloud-radiation interaction at cloud scale (Khairoutdinov et al., 2005). Under realistic configuration, the planetary boundary layer process, orographic gravity wave drags, and the dynamic core are computed on the CAM grid. One conceptual advantage of using SPCAM as the reference simulation is that the subgrid and grid-scale processes are clearly separated, making it easy to define the parameterization task for an ML algorithm (Rasp, 2020).

2.2 NN-Parameterization

2.2.1 Data sets

The NN-Parameterization is a deep learning emulator of the SP and its cloud-scale radiation effects in SPCAM. Therefore, the inputs of this emulator are borrowed from the SP input variables such as the grid-scale state variables and forcings, including specific humidity q_v, temperature T, largescale water vapor forcing $\left(\frac{\partial q_v}{\partial t}\right)_{ls}$ and temperature forcing $\left(\frac{\partial T}{\partial t}\right)_{ls}$. Additionally, we select surface pressure P_s and solar insolation (SOLIN) at the top of the model from the radiation module. The outputs of NN-Parameterization are subgrid-scale tendencies of moisture $\left(\frac{\partial q_v}{\partial t}\right)$ and of temperature $\left(\frac{\partial T}{\partial t}\right)$ at each model level as well as net shortwave and longwave radiative fluxes at both the surface and the TOA. This heating is composed of moist heating in the SP and the GCM-grid-averaged SP radiative heating. Also, it is important to include direct and diffuse downwelling solar radiation fluxes as output variables to force the coupled land surface model. Specifically, they are solar downward visible direct to surface (SOLS), solar downward near infrared direct to surface (SOLL), solar downward visible diffuse to surface (SOLSD), and solar downward near infrared diffuse to surface (SOLLD). In the end, the precipitation is derived from column integration of predicted moisture tendency to keep basic water conservation.

The large-scale forcings are commonly not included in previous studies with aqua-planet configuration. However, under realistic configuration, such forcings are composed of the dynamics and the planetary boundary layer diffusion, thereby carrying critical information about the complex background circulations and surface condition. Similarly, those downwelling solar radiation fluxes with separation of direct versus diffusion records the received solar energy by the coupled surface model

with different land cover types and processes (Mooers et al., 2021). If not included, the land surface is not heated up by the sun, therefore, seriously weaking the sea and land breeze and monsoon circulations.

Table 1 lists the input and output variables and their normalization factors. There are 30 model levels for each profile variables. Therefore, the input vector consists of 122 elements for 4 profile variables and 2 scalars, while the 68-element output vector is made of 2 profiles and 8 scalars. All input and output variables are normalized to ensure that they are in the same magnitude before they are put into the NN-parameterization for training, testing, model prognostic validation. The normalization factor for each variable shown in the supplemented codebase is determined by the maximum of its absolute values.

The training dataset used by all considered NNs is 40% temporally random sampled from the 2-year SPCAM simulation from January 1, 1997 to December 31, 1998. Notably, random sampling is only done in the time dimension but not in latitude and longitude, including all 13,824 samples from global grid points for each selected time step. To avoid any mix or temporal connection between the training set and offline validation set, we random sample 40% timesteps from the SPCAM simulation in the year 2000 as offline validation set in the sensitivity test.

2.2.2 A ResDNN Set

In the development of the NN-Parameterization scheme, it is found that when different variables are used as the output of the neural network, the training difficulty is quite different. Especially, the neural network's ability to fit the radiation heating and scalar fluxes is significantly stronger than the tendencies variables. This is also found in Gentine et al. (2018), in which the coefficient of determination (R^2) of radiative heating tendency is higher than that of moisture tendency at most model levels. We believe that using a single NN with one target to train all variables, i.e., moisture tendency, temperature tendency, and radiation fluxes, inevitably causes mutual interference. Since gradient descending is applied to optimize the network in training, mutual interference between different targets is expected to cause the cancel out of gradient directions used for descending (Crawshaw et al., 2020; Zhang and Yang., 2021) and ultimately affect the convergence of the network. We use different neural networks to train the tendency of moisture and temperature, and radiation fluxes, respectively. By doing so, we avoid the gradient cancellation between multiple targets and improve the convergence speed and fitting accuracy when training the network. As described in Section 3.1, when using the same network configuration, radiation fluxes are trained much easier with higher accuracy than tendencies of moisture and temperature. We admit that putting heating and moistening rate in two different NNs arbitrarily cut physical connections between them. But this separation is surely doing training more easily in the developing stage.

In this study, to mimic the column-independent SP and its radiation effects, the input and output of NN-Parameterization have to be both 1-D vectors. This means that the data input and output of NN-Parameterization are much simpler than those in the existing mainstream machine learning problems, such as image recognition and text-speech recognition, so it is impossible to apply most of the existing complex neural networks directly. Taking the convolutional neural network CNN as an example, the study of Albawi et al. (2017) shows that CNN has more advantages than DNN in the learning of large-scale

images. The problem we face is that the input is a 122-dimensional vector stitched by multiple different physical quantities with only 4 30-element 1D profiles plus 2 scalars, which cannot meet the requirements of "large-scale" (generally at least 32×32 two-dimensional images). So, there is no need to use CNN. Hornik et al. (1989) proved that a single-layer neural network can approximate any function. Although the problem that NN-Parametrization needs to deal with is highly nonlinear, from the point of view of machine learning, it is essentially a mapping problem from a 122-element 1D-vector to a 1D-vector with a length of 68. According to the universal approximation theorem, DNN is feasible. Therefore, when constructing NN-Parametrization, we first tried to use DNN for fitting, and introduced residual connections to extend DNN to ResDNN.

After numerous experiments, we got the best hyperparameters of DNN and ResDNN. When training a Fully connected DNN, the hidden layer width of the network should be set to 512, and the network depth should not exceed 7, otherwise it will affect the convergence of the DNN. In order to make the neural network capture more non-linear information, enhance the fitting ability. We introduce skip connections to extend the 7-layer DNN to 14-layer ResDNN. The network structure of ResDNN is shown in Figure 1. In the training process, both DNN and ResDNN use an initial learning rate of 0.001 and a learning rate decaying strategy as cosine annealing (Loshchilov et al., 2016) without dropout and L2 regularization. Adam (Kingma and Ba, 2014) is chosen as the optimizer to minimize the mean squared errors (MSEs). The results in Figure 2 show that ResDNN can fit data is significantly better than DNN, with details described in Section 3.1. At the same time, the sensitivity tests in section 3 also prove that no DNN model can ensure the stable simulation of NNCAM. So, we chose ResDNNs sets as stable candidates to build NN-Parameterization. After obtaining all well-fit ResDNN sets, the next step is to couple the candidates into NNCAM one by one for prognostic tests and find sets that can support stable simulation. To complete this extremely challenging task, we have more than 50 prognostic tests. All experiments and analyses on stability will be introduced in section 3 as well.

2.2.3 Implementation of NN-Parameterization

The NN-Parameterization is implemented into SPCAM to replace both the CRM based superparameterization and its radiation effects on the basis of coarse grid average. In the beginning of each timestep, NNCAM calls the NN-Parameterization and predict the moisture tendency $\left(\frac{\partial q_v}{\partial t}\right)$, the temperature tendency $\left(\frac{\partial T}{\partial t}\right)$ and radiation fluxes. Then the DNN predictions are returned to NNCAM, updating the model states and fluxes. Additionally, the surface total precipitation is derived from column integration of the predicted moisture tendency. The near-surface conditions of the atmosphere and downwelling radiation fluxes are transferred to the land surface model. After the coupling of the land surface model and the prescribed SST, the host CAM5 performs the planetary boundary layer diffusion and let its dynamic core complete a timestep integration (Figure 1). In the next timestep, the dynamic core returns the new model states to the NN-Parameterization as inputs again. During the whole process, NN-Parameterization and GCM will constantly update each other's status. How to couple the NN Parameterization with GCM and run efficiently and effectively is the key to the implementation of NNCAM. To solve these problems, we develop the NN-GCM coupler that integrates NNs into NNCAM, which will be introduced in the following section.

2.3 The NN-GCM Coupler

Deep learning research mainly uses machine learning frameworks based on Python interfaces to train neural network models and deploy them through C++ or Python programs. While GCM is mainly developed in Fortran, it is a very challenging work to call a neural network model based on Python/C++ interface in GCM codes written in Fortran. Solving the problem of code compatibility between NN and GCM can significantly help develop NN based Parameterizations for climate models.

To implement a NN based Parameterization in current climate model which is mostly developed in Fortran, many researchers try to get the network parameters (e.g., weight, bias) from the machine learning models and implement the NN models (e.g., DNNs) with hard coding in Fortran. At runtime, NNCAM will call a NN parametrization as a function (Rasp et al., 2018; Brenowitz and Bretherton, 2019). Recently, some researchers have developed a Fortran-neural network interface that can be used to deploy DNNs into GCMs (Ott et al., 2020). This interface can import neural network parameters from outside of Fortran program, and the Fortran-based implementation ensures that it can be flexibly deployed in GCMs. However, embedding a NN parameterization in NNCAM is still a troublesome task with no existing coupling framework to support many of the latest network structures. This problem will restrict developers from building more powerful NNs and deploying them in NNCAM.

We develop the coupler to bridge NN-Parameterization with the host CAM5. Through this coupler, the neural network can communicate with the dynamic core and other physical schemes in NNCAM in each time step. When NNCAM is running, as shown in ① in Figure 5, the coupler receives the state and forcing output from dynamic core in Fortran based CAM5. For each input variable, we use the native MPI interface in CAM5 to gather the data of all processes to the master process into a tensor. Then, as shown in ② of Figure 5, the coupler will transmit the gathered tensor through the data buffer to the NN-Parameterization running on the same node as the master process. The NN-Parameterization gets the input, infers the outputs, and transmits them back to the coupler. As shown in ③ of Figure 5, the coupler will first write these tendencies and radiation fluxes back to the master process and then broadcast the data to CAM5 processes running on the computing nodes through the MPI transmission interface. Therefore, other parameterizations get the predictions from NN-Parameterization to complete the follow-up procedures (④ in Figure 5).

In practice, the NN-GCM Coupler introduces a data buffer that supports system-level interface, which is accessible by both Fortran based GCM and Python based NN without supplementary foreign codes. This can avoid code compatibility issues when building Machine Learning coupled numerical models. It supports all mainstream machine learning frameworks, including native PyTorch and TensorFlow. Based on the coupler, one can efficiently and flexibly deploy the Deep Learning Model in NNCAM, and can even take advantage of the latest developed neural networks.

All neural network models deployed through NN-GCM Coupler can support GPU accelerated inference to achieve excellent computing performance. In this study, we ran SPCAM and NNCAM on 192 CPU cores. NNCAM also used 2 GPUs for acceleration. During the NNCAM runtime, each time step of NNCAM requires NN-Parameterization to complete an inference and conduct data communication with NNCAM. This is a typical high-frequency communication scenario. We

evaluated the amount of data (about 20MB for CAM5 with the horizontal resolution of 1.9°×2.5°) that needs to be transmitted for each communication, and determined to establish a data buffer on a high-speed solid-state drive to ensure a balance of performance and compatibility. It takes about $1x10^{-2}$ seconds to access the data buffer in each time step, which is enough to support the efficient simulation of NNCAM. The Simulation Years per Day (SYPD) of NNCAM based on NN-GCM Coupler has an impressive performance improvement, when using 192 Intel CPU cores, the SYPD of SPCAM is 0.3, the SYPD of CAM5 is 20, and the SYPD of NNCAM is 10. It is worth noting that, NNCAM based on NN-GCM coupler uses an additional GPU to accelerate NN-Parameterization. When NN-GCM Coupler is not used, NN-Parameterization is implemented by Fortran and accelerated by Fortran-based Math Kernel Library, the SYPD is 1.5.

3 A Road to Stability

3.1 Sensitivity Tests and Trial-and-error

To develop a stable NN parameterization, we propose a ResDNN set, where each neural network is responsible for predicting a class of variables (see section 2.2.2). One may wonder whether the ResDNN architecture is necessary and whether offline accuracy of NNs matters in online stability. This section tries to deal with the questions via a series of sensitivity tests.

To prove the necessity of the ResDNN architecture, we use the 7-layer DNN as the control group. We do not include other types of ML architecture, since random forest is less likely to perform as accurately as neural networks and cannot be implemented in GPUs (Yuval et al., 2021), and 1D CNN is not widely used in other studies except Han et al. (2020) with unknown prognostic performance.

The prognostic tests of NN parameterization begin at 1998-01-01 as a startup. As initialization, calling the SP in SPCAM at the first step is required to generate the correct largescale forcings as the input for NN parameterizations. In the sensitivity test, we freeze the ResDNN for the 8 radiation fluxes to simplify the neural network choices, since their offline validation is extremely accurate with R^2 above 0.98 over 50 training epochs (Figure 2b). Different from the accurately trained radiation fluxes, the tendencies of temperature and moisture are less accurate and can hypothetically affect the prognostic performance. To evaluate the tendency of moistening and heating in one metric, we introduce the MSE of moist static energy changing rate ($dh = C_p dT + L_v dq_v$) as:

$$MSE_h = \left\| \frac{1}{g}(dh_{NN} - dh_{SPCAM})\Delta p \right\|_2, \quad (1)$$

where g is the gravity constant, C_p refers to the heat capacity of air, L_v is the latent heat of water vapor, and Δp is the layer thickness. Multiple ResDNN pairs for dq_v and dT and DNN pairs are trained from 5 epochs to 50 epochs, carrying different offline validation accuracy.

Figure 4 shows the offline validation MSE_h versus the prognostic steps. First, DNN parameterizations (blue triangles) are systematic less accurate than ResDNN ones (blue dots and black inverted triangles), which is consistent with Figure 2a.

285 They cannot run stably in prognostic tests with the best DNN parameterization to sustain half a year of simulation. For the ResDNNs, the less well-trained ones with high MSE crash for a shorter simulation period than DNNs. However, when the offline MSE decreases to a certain level (e.g., 290 W^2/m^4), some of the ResDNN parameterizations are stable for extreme long-term simulations, while others remain unstable.

Generally, a NN parameterization that can support long-term integration should have both good generalization abilities
290 and high accuracy for training and validation. Above all, sufficient accuracy is necessary for all neural networks. From Figure 4, it can be interpreted that a vague threshold exists in the validation MSE. ResDNNs can be trained for higher accuracy since they are much deeper than DNNs with much higher model capacity. So, they are more competent than DNNs in this job. On the other hand, studies showed that high-capacity models are harder to train and more likely to overfit (Goodfellow et al., 2016). Thus, the prognostic stability differences between less well-trained ResDNNs and the well-trained ones are drastic
295 compared with DNNs. Also, some overly trained ResDNNs with lowest validation loss are speculated to overfit. Those overfitting models are less likely to generalize to unknown backgrounds caused by accumulated errors in the ML-GCM system, ending up model crashes. However, those are just intuitive experiences but not guarantee ways for stability.

In the time evolution of the global averaged total energy (Figure 5). The system energy grows exponentially and then blows up for unstable ResDNN parameterizations (the red and orange lines). In contrast, the stable ones can keep the total
300 energy at a certain level and reproduce the annual cycle fluctuations in SPCAM. Among the stable ResDNN schemes, some can get nearly a perfect reproduction of the total energy evolution of SPCAM (the blue line), while some inaccurately simulate the climate state with a large deviation (green line). Therefore, among the accurate ResDNN parameterizations (e.g., offline validation $MSE_h < 290\ W^2/m^4$), we still have to use the trial-and-error to filter out unstable ones and then select the best ResDNN pair for moistening and heating rate that can reduplicate the total energy time evolution of SPCAM with the least
305 deviation. We name this best ResDNN pair together with the ResDNN in charge of radiation fluxes the NN-Parameterization. This NN-GCM coupled model is called NNCAM and is later evaluated for climate mean states and variability.

3.2 Gravity Wave Diagnosis

It is still a question of why unstable NN parameterizations blow up models. The fast-growing energy of the unstable runs indicates a possible underlying unrealistic energy amplifying mechanism in the NN-GCM coupled system. Brenowitz et al.
310 (2020) offered interpretations. When an unstable NN parameterization is coupled with dynamics, it tends to amplify any unrealistic perturbation caused by emulation errors and pass it to the entire system through gravity waves. In contrast, the stable NN parameterizations tend to dump all the perturbs quickly. This is true in our study with realistic configuration. Such unstable gravity waves are observed in the prognostic simulation of an unstable ResDNN (the red line in Figure 5). The animation in Movie S1 records the first unrealistic wave and Movie S2 documents more intense waves afterward with a
315 perfectly round shape. Brenowitz et al. (2020) also introduced an analysis tool that calculates wave energy spectra of a hierarchy model that couples the linear response functions (LRF) of a NN parameterization to a simplified two-dimensional linear dynamic system, where perturbations can propagate in 2D gravity waves. We apply the tool in our study and detect

similar results of unstable mode for the unstable ResDNN with positive energy growth rate across all wave numbers at phase speed between 5 m/s to 20 m/s (Figure S1b). While the stable ResDNN shows a stable mode with the growth rate of nearly all wave numbers and phases below zero (Figure S1a).

4 Offline Validation of NN-Parameterization

Before evaluating the prognostic results, demonstration of offline performance with geographic information is needed for the following purposes: 1) To show how well our NN-Parameterization emulates the SP in realistic configuration compared with baseline CAM5 physics and with previous studies. 2) To reveal the strength and weakness of NN emulations with correct input, give clues to the analysis of prognostic results in the following section. We performed offline testing with a realistically configurated SPCAM from January 1st 1999 to December 31st 2000, where NN-Parameterization is diagnostically run paralleled to the SP, and so does the CAM5 physics. The results over the entire second year of the period are chosen for evaluation, completely independent from the training dataset. Following the conventions in Han et al. (2020) and Mooers et al. (2021), we choose mean fields and coefficient of determination (R^2) as the two metrics for evaluations.

The mean diabatic heating and drying rates produced by convection and large-scale condensation in SPCAM and NN-Parameterization are in close agreement. Figure 6 shows the latitude-height cross-sections of the annual mean heating and moistening rates in SPCAM and the corresponding NN-Parameterization. At 5 °N, SPCAM shows maximum latent heating in the deep troposphere, corresponding to deep convection at the ITCZ. In the subtropics, there is heating and moistening in the lower troposphere, corresponding to stratocumulus and shallow convection in the subtropics. In the midlatitudes, there is a secondary heating maximum below 400 hPa due to midlatitude storm tracks. All these features are well reproduced by NN-Parameterization. Note that in the midtroposphere, the ITCZ peak in the drying rates is slightly weaker in NN-Parameterization compared with that of SPCAM (Figure 6c and 6d).

In addition to the mean fields, the high prediction skill of NN-Parameterization is also shown in the spatial distribution of R^2. To demonstrate R^2 for the 3D variables such as diabatic heating and moistening, same as Mooers et al. (2020), zonal averages are calculated in advance before R^2 calculation for each location in the pressure-latitude cross-section. For diabatic heating, R^2 is above 0.7 over the entire mid to low troposphere and the high skill regions with R^2 greater than 0.9 concentrates in low levels but are extended to mid-troposphere in storm tracks (Figure 7a). As for the moistening rate, the high skill zones concentrate in the mid to upper troposphere (Figure 7b), leaving low skill areas below. Those regions with low accuracy are generally located in the mid to low troposphere in tropics and subtropics, corresponding to deep convection at ITCZ and shallow convection in subtropics. Nonetheless, the tendencies from diagnostic CAM5 parameterization hardly draw any similarity to those simulated by the SP except for a few locations in the mid to upper troposphere in tropics and polar regions (Figure 7c & 7d).

The global distribution of R^2 for the precipitation predictions is shown in Figure 8. Our NN-Parameterization shows a great prediction skill globally, especially in the midlatitude storm tracks. The prediction skill is relatively low in many areas

between 30°S to 30°N and some midlatitude continents (Figure 8a), in particular, not ideal in the ITCZ deep convection regions. Moreover, for shallow convection in Subtropical Eastern Pacific and Subtropical Eastern Atlantic, the precipitation prediction skill hits bottom, corresponding to the subtropical low skill zones for moistening rate (Figure 6b). On the other hand, the total precipitation simulated by CAM5 parameterizations is much less analogous to the SP than NN-Parameterization with a systematically lower accuracy globally. CAM5 precipitation can reach a relatively high accuracy along the mid-latitude storm tracks but fail most regions in the tropics.

Generally, NN-Parameterization performs far better than CAM5 parameterization in the 1-year offline testing and shows similar accuracy as the DNN in Mooers et al. (2020). The real-geography data can significantly decrease the emulation skill of a deep learning model (Mooers et al., 2021), where the convection backgrounds are much more complex with meridional and zonal asymmetric and seasonal variated circulations, not to mention the orograph and various types of underlying land surface. In that case, the ResDNN is a valuable NN architecture that can bring good performance as the automated hyperparameter tuning algorithm without searching for hundreds of NN candidates. Still, our NN-Parameterization is exposed to low accuracy predictions in subtropical shallow convection areas, a great challenge for machine learning emulation of moistening rate and precipitation. In those regions, the local variance/std is close to zero. But the NNs in our study are trained in the loss function of mean squared error, which is not sensitive to small values.

5 Long-term Prognostic Validation

NN-Parameterization is selected for best prognostic performance in Section 3.1. It is coupled in the realistic configurated SPCAM to replace the SP and its cloud-scale radiation effects. This coupled model is called NNCAM afterwards and is compared with SPCAM and CAM5. All three model starts at January 1st 1998 as start up. They are all run for 6 years with the first year for spin up and the next 5 years from January 1st 1999 to December 31st 2003 for evaluation and comparison. Later, the simulation of NNCAM is extended for another 5 years to December 31st 2008 to show its stability. Due to excessive computing resources consumption, the simulation of SPCAM is not get extended. In analysis of prognostic results, the following are selected for demonstration of climatology and variability: multi-year mean fields of temperature and humidity, precipitation, precipitation frequency distribution, and the Madden Julian Oscillation.

5.1 Climatology

5.1.1 Vertical profiles of temperature and humidity

In this section, we first evaluate the vertical structure of the mean temperature and humidity. Figure 9 shows the zonally averaged vertical profiles of air temperature and specific humidity as simulated by the NNCAM and the CAM5, in contrast to the SPCAM simulations. Overall, the NNCAM simulate reasonable thermal and moisture structure. However, it is shown that NNCAM has some biases in mean fields of temperature and humidity, which is shown as larger root mean squared errors (RMSEs) or larger differences than CAM5 (Figure S2). The larger deviations are temperature biases in the tropopause, where

the cold-point region is thinner and warmer in NNCAM than in SPCAM and CAM5. In addition, there are cold biases above 200 hPa and warm biases blow over polar regions in NNCAM. For the humidity field, there are slight dry biases over the equator and wet biases elsewhere in NNCAM. Even with the biases, the climate mean states are consistent with those in the last 5-year simulation for NNCAM (Figure S3), which indicates almost no climate drift in the long-term simulation.

5.1.2 Precipitation

Figure 10 shows the spatial distributions of winter (December-January-February) and summer (June-July-August) mean precipitation simulated by SPCAM, NNCAM, and CAM5. The SPCAM simulation results are regarded as reference precipitation. In SPCAM (Figure 10a and 10b), massive precipitation can be found in regions of Asian monsoon and midlatitude storm tracks over the northwest Pacific and Atlantic oceans. In the tropics, the primary peaks of rainfall are in the eastern Indian Ocean and Maritime Continent regions. Furthermore, two zonal precipitation bands are located at 0°–10°N in the equatorial Pacific and Atlantic oceans, constituting the northern ITCZ. The southern South Pacific Convergence Zone (SPCZ) is mainly located around 5°S–10°S near the western Pacific warm pool region and experiences a southeast tilt as it extends eastward into the central Pacific. The main spatial patterns of SPCAM precipitation climatology are properly reproduced by both NNCAM and CAM5. In NNCAM, strong rainfall centers are well simulated over the tropical land regions over Maritime Continent, the Asian monsoon region, and South America and Africa (Figure 10c and 10d). In addition, the heavy summertime precipitation over the Northwestern Pacific simulated by SPCAM is well represented in NNCAM (Figure 10a and 10c). In CAM5, there is too little precipitation over that area (Figure 10e). Moreover, NNCAM can maintain the spatial pattern and global average of precipitation in the next 5-year simulation, reassuring its long-term stability (Figure S4).

Generally, NNCAM draws more similarity to SPCAM than CAM5 in spatial distribution of summertime multiyear precipitation with smaller RMSE and global averaged biases. However, in the difference plot (Figure 11), NNCAM moderately underestimates precipitation along the equator, Indian monsoon region, and maritime continent in summer (Figure 11a). In boreal winter, NNCAM simulates a weak and excessively separated SPCZ from ITCZ, with both precipitation centers shifting away from each other. As a result, we detect underestimation in the equatorial regions of the maritime continent as well as the SPCZ but overestimation on the north of the equator in the West Pacific (Figure 11b), which makes NNCAM less resemble SPCAM than CAM5 in this season. This simulation biases in NNCAM are speculated linked to the weaker drying tendencies of the ITCZ midtroposphere from the NN parameterization and low accuracy of NNCAM predictions in tropics.

5.2 Variability

5.2.1 Frequency Distribution of Precipitation

Moreover, NNCAM shows better performance in simulating precipitation extremes. Figure 12 shows the probability densities function of simulated daily precipitation in the tropics (30°S−30°N) with a precipitation intensity interval of 1 mm day^{-1}. In CAM5, heavy precipitation events exceeding 20 mm day^{-1} are greatly underestimated. In addition, light to moderate

precipitation events between 2−20 mm day^{-1} are overestimated with an unreal probability peak around 10 mm day^{-1} in CAM5, which is a typical simulation bias found in simulations with parameterized convection but not in explicitly resolved convections (Holloway et al., 2012). Compared with CAM5, the spectral distribution of precipitation in NNCAM is much closer to SPCAM. heavy rainfall events are substantially enhanced, and the overestimated precipitation occurrence between 2−20 mm day^{-1} is reduced with no spurious peak around 10 mm day^{-1}.

5.2.2 The MJO

The MJO is a crucial tropical intraseasonal variability at the time scale of 20–100 days (Wheeler and Kiladis, 1999). Figure 13 presents the wavenumber and frequency spectra for equatorial precipitation daily anomalies from SPCAM, NNCAM, and CAM5 in 4 consecutive boreal winter from 1999 to 2003. SPCAM shows widespread power signals over zonal number of 1-4 and periods between 20-100 plus a peak around at zonal numbers of 1−3 and periods of 70-100-day for eastward propagation (Figure 13a). Similarly, in NNCAM, there is a spectral peak at the wavenumbers of 1−2 and periods of 50-80 day for east propagation (Figure 13b), exhibiting intense intraseasonal signals. For CAM5 (Figure 13c), the spectral power is concentrated around 30-day and more extended periods (greater than 80 days) at wavenumber 1 for eastward propagation. In addition, CAM5 also shows signals of westward propagation around 30-day period. Compared with CAM5, NNCAM shows stronger intraseasonal power and resembles SPCAM better. To quantify this similarity, we calculate the coefficient of determination R^2 of the precipitation spectrum in NNCAM and CAM5, using the spectrum in SPCAM as the target value. The precipitation spectrum R^2 in NNCAM (0.51) is much higher than that in CAM5 (0.40).

The MJO is characterized by the eastward propagation of deep convective structures along the equator. Generally, it generally forms over the Indian Ocean, strengthens over the Pacific, and weakens in the eastern Pacific due to interaction with cooler SSTs (Madden and Julian, 1972). Figure 14 presents the longitude-time lag evolution of 10°S−10°N meridional averaged daily anomalies of intraseasonal (filtered with 20-100 day bandpass) precipitation and 200 hPa zonal wind (U200) in boreal winter. The results show that both SPCAM and NNCAM reasonably reproduce the eastward propagating convection from the Indian Ocean across the Maritime Continent to the Pacific (Figure 14a and 14b), confirmed by both precipitation field and U200 field. Therefore, we conclude that NNCAM captures the key MJO propagation simulated in SPCAM. In contrast, the time lag plot of CAM5 depicts an unpleasant west propagation. Same as the precipitation spectrum, R^2 of the time lag coefficient is shown to quantify the resemblance. The time lag coefficient of U200 in NNCAM is much closer to SPCAM than CAM5, with a way higher R^2, indicating that the NN-Parameterization successfully emulates the convection variability of the SP and reflects it in the dynamic fields.

6 Summary and Conclusions

This study investigates the potential of deep neural network based parameterizations in SPCAM to reproduce long-term climatology and climate variability. We present NN-Parameterization, a ResDNN set, to emulate the SP with a 2D CRM and

its cloud scale radiation in effects in a realistic configurated SPCAM with true land-ocean distribution and orography. The input variables to the NN-Parameterization include specific humidity, temperature, largescale water vapor and temperature forcings, surface pressure and solar insolation. The output variables of the NN-Parameterization consist of the subgrid tendencies of moisture and temperature, net radiation fluxes at the top of the model and surface, and solar radiation fluxes down to the surface. We proposed a set of 14-layer deep residual neural networks in which each NN is in charge of one type of output variable. With such a design, we gain the best emulation accuracy for each predictor. Via a systematic trial-and-error searching procedure, we are able to firstly select sets of ResDNNs that support stable prognostic climate simulations and then choose the best set with lowest climate errors as the formal NN-Parameterization. Moreover, a mechanism of unreal perturbation amplification is found in GCM simulations with unstable NN parameterizations with the spectrum diagnostic tool invented in Brenowitz et al. (2020).

The offline test shows the great skills of the NN-Parameterization in emulating the SP outputs and its cloud scale radiation effects in SPCAM. The overall diabatic heating and drying rates in the NN-Parameterization and SPCAM are in close agreement. When implemented in the host SPCAM to replace its time-consuming SP and its radiation effects, the NN-Parameterization successes in an extensive long-term stable prognostic simulation and predicts reasonable mean vertical structures in temperature and humidity, and the precipitation distributions. Compared with the SPCAM target simulation, NNCAM still produces some biases in mean fields, such as a warmer troposphere over polar regions and tropopause and strong precipitation underestimation in equatorial regions. On the other hand, the better climate variability in SPCAM over CAM5 is well learned by our NN-Parameterization and reproduced in NNCAM with better frequency in extreme rainfall, similar MJO spectrum and propagation direction and speed. Although with the biases in climate states so far, NNCAM can still be regarded as the first attempt to prognostically couple a NN-based parameterization in realistic configurated 3D GCM.

Many previous studies have well-studied machine learning parameterizations implemented in aqua-planet configurated 3D GCM. Some faced instability in coupled simulations (Brenowitz and Bretherton, 2019), while some succeeded in long-term stable prognostic simulations with deep fully-connected neural networks (Rasp et al., 2018; Yuval et al., 2021) as well as random forest (Yuval and O'Gorman, 2020). In contrast to aqua-planet simulations, the spatial heterogeneity is prominent over land in GCMs which are configurated with real-geography boundary conditions. In this case, a plain fully connected neural networks the SP output (Mooers et al., 2021). The convection, clouds, and the interacted radiation of the CRM together with real-geography boundary conditions are without doubt far more complicated than in idealized models. To meet the new demand under realistic configuration, we design ResDNN with sufficient depth to further improve the nonlinear fitting ability of NN-Parameterization. With the skip connections, the 7-layer DNN models can be extended to 14 layers, therefore, significant improving offline accuracy. In the prognostic tests, a few ResDNN parameterizations can support long term stable run, while all DNN parameterizations are so far found unstable.

Trial-and-error is still the only way to find stable NN parameterizations. So far, we have not come up with an a priori mothed that guaranteed stability. However, we do find some clues in the sensitivity tests. We believe sufficient offline accuracy is essential for online stability by confirming all inaccurate NN parameterizations unstable. On the other hand, some highly

accurate ones still crash the prognostic simulation, where we find rapid increasing total energy. This mechanism is that unstable NNs cannot damp neural network emulation errors but amplify and propagate them to the entire system through gravity waves.

The prognostic biases in mean fields in speculated as a result of by the combined effect of the emulation errors of all the NN-Parameterization prediction fields. Further study is required. Still, it can be related to the spatially non-uniform accuracy of NN-Parameterization, such as relatively low fitting accuracy in tropical deep convective regions and shallow subtropical convection and stratiform cloud regions. Such problems have also been reported in previous studies (Gentine et al., 2018; Mooers et al., 2021). We believe that a NN parameterization with heterogeneous characteristics across different regions, rather than a globally uniform scheme, can further improve the fitting accuracy in this tropical and subtropical region.

Embedding deep neural networks into Fortran based atmospheric models is still a handicap. Before this study, researchers mainly used hard coding to build neural networks (Rasp et al., 2018; Brenowitz and Bretherton, 2019). An easier way is to use Fortran based neural network libraries that can flexibly import network parameters (Ott et al., 2020). These methods have successfully implemented NN in GCM, but they can only support dense layer based NN. As a result, developers cannot take advantage of the most advanced neural network structures such as convolution, shortcut, self-attention, variational autoencoder, etc., to build powerful DNN based Parameterizations. In this research, through NN-GCM Coupler, NN-Parameterization can support the mainstream GPU-enabled machine learning frameworks. Thanks to the simple and effective implementation of the DNN-GCM Coupler, our NNCAM achieves 30 times SYPD compared to SPCAM by using a ResDNN set in NN-Parameterization, although these DNNs are much deeper than the previous state-of-the-art fully-connected NNs in this field.

Code and data availability. The original training and testing data can be accessed at https://doi.org/10.5281/zenodo.5625616. The source codes of SPCAM version 2 and NNCAM have been archived, and made publicly available for downloading from https://doi.org/10.5281/zenodo.5596273.

Author contributions. XW trained the deep learning model, constructed the DNN-GCM Coupler, performed the NNCAM and CAM5 experiments, and wrote the main part of the paper. YH conducted the SPCAM simulations, offered valuable suggestions on the development of the NN parameterization, and participated in the writing of the paper and revision. WX supervised this work, provided critical comments on this work and participated in the writing of the paper. GJZ provided key points for this research and participated in the revision of the paper. GWY supported this research and gave important opinions. All the authors discussed the model development and the results.

Competing interests. The authors declare no conflict of interest.

Acknowledgements. This work is partially supported by National Key R&D Program of China (grant no. 2017YFA0604500), and the National Natural Science Foundation of China (grant no. 42130603). Dr. Yilun Han is supported by National Key R&D Program of China (grant no. 2017YFA0604000). We thank Prof. Yong Wang for his guidance on SPCAM simulations

and valuable discussions on this work. We also thank Prof. Yixiong Lu for providing professional advice on the evaluation of the simulation results of NNCAM.

References

Albawi, S., Mohammed, T. A., and Al-Zawi, S.: Understanding of a convolutional neural network, 2017 International Conference on Engineering and Technology (ICET), 21-23 Aug. 1-6, 10.1109/ICEngTechnol.2017.8308186, 2017.

Bony, S., Stevens, B., Frierson, D. M. W., Jakob, C., Kageyama, M., Pincus, R., Shepherd, T. G., Sherwood, S. C., Siebesma, A. P., Sobel, A. H., Watanabe, M., and Webb, M. J.: Clouds, circulation and climate sensitivity, Nature Geoscience, 8, 261-268, 10.1038/ngeo2398, 2015.

Brenowitz, N. D. and Bretherton, C. S.: Spatially Extended Tests of a Neural Network Parametrization Trained by Coarse-Graining, Journal of Advances in Modeling Earth Systems, 11, 2728-2744, 10.1029/2019ms001711, 2019.

Brenowitz, N. D., Beucler, T., Pritchard, M., and Bretherton, C. S.: Interpreting and Stabilizing Machine-Learning Parametrizations of Convection, Journal of the Atmospheric Sciences, 77, 4357-4375, 10.1175/jas-d-20-0082.1, 2020.

Bretherton, C. S., Blossey, P. N., and Stan, C.: Cloud feedbacks on greenhouse warming in the superparameterized climate model SP-CCSM4, Journal of Advances in Modeling Earth Systems, 6, 1185-1204, https://doi.org/10.1002/2014MS000355, 2014.

Cao, G. and Zhang, G. J.: Role of Vertical Structure of Convective Heating in MJO Simulation in NCAR CAM5.3, Journal of Climate, 30, 7423-7439, 10.1175/jcli-d-16-0913.1, 2017.

Crawshaw, M.: Multi-task learning with deep neural networks: A survey, arXiv preprint arXiv:2009.09796, 2020.

Emanuel, K. A., David Neelin, J., and Bretherton, C. S.: On large-scale circulations in convecting atmospheres, Quarterly Journal of the Royal Meteorological Society, 120, 1111-1143, 10.1002/qj.49712051902, 1994.

Gentine, P., Pritchard, M., Rasp, S., Reinaudi, G., and Yacalis, G.: Could machine learning break the convection parameterization deadlock?, Geophysical Research Letters, 45, 5742-5751, 2018.

Gettelman, A., Liu, X., Ghan, S. J., Morrison, H., Park, S., Conley, A. J., Klein, S. A., Boyle, J., Mitchell, D. L., and Li, J.-L. F.: Global simulations of ice nucleation and ice supersaturation with an improved cloud scheme in the Community Atmosphere Model, Journal of Geophysical Research: Atmospheres, 115, https://doi.org/10.1029/2009JD013797, 2010.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.

Grabowski, W. W.: Coupling Cloud Processes with the Large-Scale Dynamics Using the Cloud-Resolving Convection Parameterization (CRCP), Journal of the Atmospheric Sciences, 58, 978-997, 10.1175/1520-0469(2001)058<0978:Ccpwtl>2.0.Co;2, 2001.

Grabowski, W. W.: An Improved Framework for Superparameterization, Journal of the Atmospheric Sciences, 61, 1940-1952, 10.1175/1520-0469(2004)061<1940:Aiffs>2.0.Co;2, 2004.

Grabowski, W. W. and Smolarkiewicz, P. K.: CRCP: a Cloud Resolving Convection Parameterization for modeling the tropical convecting atmosphere, Physica D: Nonlinear Phenomena, 133, 171-178, https://doi.org/10.1016/S0167-2789(99)00104-9, 1999.

Han, Y., Zhang, G. J., Huang, X., and Wang, Y.: A Moist Physics Parameterization Based on Deep Learning, Journal of Advances in Modeling Earth Systems, 12, e2020MS002076, 10.1029/2020ms002076, 2020.

He, K., Zhang, X., Ren, S., and Sun, J.: Deep residual learning for image recognition, The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, Nevada, June 26 - July 1, 770-778, 2016.

Holloway, C. E., Woolnough, S. J., and Lister, G. M. S.: Precipitation distributions for explicit versus parametrized convection in a large-domain high-resolution tropical case study, Quarterly Journal of the Royal Meteorological Society, 138, 1692-1708, https://doi.org/10.1002/qj.1903, 2012.

Hornik, K., Stinchcombe, M., and White, H.: Multilayer feedforward networks are universal approximators, Neural Networks, 2, 359-366, https://doi.org/10.1016/0893-6080(89)90020-8, 1989.

Jiang, X., Waliser, D. E., Xavier, P. K., Petch, J., Klingaman, N. P., Woolnough, S. J., Guan, B., Bellon, G., Crueger, T., DeMott, C., Hannay, C., Lin, H., Hu, W., Kim, D., Lappen, C.-L., Lu, M.-M., Ma, H.-Y., Miyakawa, T., Ridout, J. A., Schubert, S. D., Scinocca, J., Seo, K.-H., Shindo, E., Song, X., Stan, C., Tseng, W.-L., Wang, W., Wu, T., Wu, X., Wyser, K., Zhang, G. J., and Zhu, H.: Vertical structure and physical processes of the Madden-Julian oscillation: Exploring key model physics in climate simulations, Journal of Geophysical Research: Atmospheres, 120, 4718-4748, 10.1002/2014jd022375, 2015.

Khairoutdinov, M., Randall, D., and DeMott, C.: Simulations of the Atmospheric General Circulation Using a Cloud-Resolving Model as a Superparameterization of Physical Processes, Journal of the Atmospheric Sciences, 62, 2136-2154, 10.1175/jas3453.1, 2005.

Khairoutdinov, M. F. and Randall, D. A.: A cloud resolving model as a cloud parameterization in the NCAR Community Climate System Model: Preliminary results, Geophysical Research Letters, 28, 3617-3620, 10.1029/2001gl013552, 2001.

Kingma, D. P. and Ba, J.: Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980, 2014.

Kooperman, G. J., Pritchard, M. S., Burt, M. A., Branson, M. D., and Randall, D. A.: Robust effects of cloud superparameterization on simulated daily rainfall intensity statistics across multiple versions of the Community Earth System Model, Journal of Advances in Modeling Earth Systems, 8, 140-165, 10.1002/2015ms000574, 2016.

Krasnopolsky, V. M., Fox-Rabinovitz, M. S., and Belochitski, A. A.: Using ensemble of neural networks to learn stochastic convection parameterizations for climate and numerical weather prediction models from data simulated by a cloud resolving model, Advances in Artificial Neural Systems, 2013, 5, 2013.

Lin, J.-L.: The Double-ITCZ Problem in IPCC AR4 Coupled GCMs: Ocean–Atmosphere Feedback Analysis, Journal of Climate, 20, 4497-4525, 10.1175/jcli4272.1, 2007.

Ling, J., Li, C., Li, T., Jia, X., Khouider, B., Maloney, E., Vitart, F., Xiao, Z., and Zhang, C.: Challenges and Opportunities in MJO Studies, Bulletin of the American Meteorological Society, 98, ES53-ES56, 10.1175/bams-d-16-0283.1, 2017.

Lopez-Gomez, I., Cohen, Y., He, J., Jaruga, A., and Schneider, T.: A Generalized Mixing Length Closure for Eddy-Diffusivity Mass-Flux Schemes of Turbulence and Convection, Journal of Advances in Modeling Earth Systems, 12, e2020MS002161, https://doi.org/10.1029/2020MS002161, 2020.

Loshchilov, I. and Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts, arXiv preprint arXiv:1608.03983, 2016.

Madden, R. A. and Julian, P. R.: Description of Global-Scale Circulation Cells in the Tropics with a 40–50 Day Period, Journal of Atmospheric Sciences, 29, 1109-1123, 10.1175/1520-0469(1972)029<1109:Dogscc>2.0.Co;2, 1972.

Mooers, G., Pritchard, M., Beucler, T., Ott, J., Yacalis, G., Baldi, P., and Gentine, P.: Assessing the Potential of Deep Learning for Emulating Cloud Superparameterization in Climate Models With Real-Geography Boundary Conditions, Journal of Advances in Modeling Earth Systems, 13, e2020MS002385, https://doi.org/10.1029/2020MS002385, 2021.

Morrison, H. and Gettelman, A.: A New Two-Moment Bulk Stratiform Cloud Microphysics Scheme in the Community Atmosphere Model, Version 3 (CAM3). Part I: Description and Numerical Tests, Journal of Climate, 21, 3642-3659, 10.1175/2008jcli2105.1, 2008.

Neale, R. B., Chen, C.-C., Gettelman, A., Lauritzen, P. H., Park, S., Williamson, D. L., Conley, A. J., Garcia, R., Kinnison, D., and Lamarque, J.-F.: Description of the NCAR community atmosphere model (CAM 5.0), NCAR Technical Note, 1, 1-12, 2012.

Oleson, K. W., Lawrence, D. M., Gordon, B., Flanner, M. G., Kluzek, E., Peter, J., Levis, S., Swenson, S. C., Thornton, E., and Feddema, J.: Technical description of version 4.0 of the Community Land Model (CLM), NCAR Technical Note, 2010.

Ott, J., Pritchard, M., Best, N., Linstead, E., Curcic, M., and Baldi, P.: A Fortran-Keras Deep Learning Bridge for Scientific Computing, Scientific Programming, 2020, 8888811, 10.1155/2020/8888811, 2020.

Park, S. and Bretherton, C. S.: The University of Washington Shallow Convection and Moist Turbulence Schemes and Their Impact on Climate Simulations with the Community Atmosphere Model, Journal of Climate, 22, 3449-3469, 10.1175/2008jcli2557.1, 2009.

Randall, D., Khairoutdinov, M., Arakawa, A., and Grabowski, W.: Breaking the Cloud Parameterization Deadlock, Bulletin of the American Meteorological Society, 84, 1547-1564, 10.1175/bams-84-11-1547, 2003.

Rasp, S.: Coupled online learning as a way to tackle instabilities and biases in neural network parameterizations: general algorithms and Lorenz 96 case study (v1.0), Geosci. Model Dev., 13, 2185-2196, 10.5194/gmd-13-2185-2020, 2020.

Rasp, S., Pritchard, M. S., and Gentine, P.: Deep learning to represent subgrid processes in climate models, Proceedings of the National Academy of Sciences, 115, 9684-9689, 10.1073/pnas.1810286115, 2018.

Song, X. and Zhang, G. J.: The Roles of Convection Parameterization in the Formation of Double ITCZ Syndrome in the NCAR CESM: I. Atmospheric Processes, Journal of Advances in Modeling Earth Systems, 10, 842-866, https://doi.org/10.1002/2017MS001191, 2018.

Tiedtke, M.: A Comprehensive Mass Flux Scheme for Cumulus Parameterization in Large-Scale Models, Monthly Weather Review, 117, 1779-1800, 10.1175/1520-0493(1989)117<1779:Acmfsf>2.0.Co;2, 1989.

Wheeler, M. and Kiladis, G. N.: Convectively Coupled Equatorial Waves: Analysis of Clouds and Temperature in the Wavenumber–Frequency Domain, Journal of the Atmospheric Sciences, 56, 374-399, 10.1175/1520-0469(1999)056<0374:Ccewao>2.0.Co;2, 1999.

Yuval, J. and O'Gorman, P. A.: Stable machine-learning parameterization of subgrid processes for climate modeling at a range of resolutions, Nature Communications, 11, 3295, 10.1038/s41467-020-17142-3, 2020.

Yuval, J., O'Gorman, P. A., and Hill, C. N.: Use of Neural Networks for Stable, Accurate and Physically Consistent Parameterization of Subgrid Atmospheric Processes With Good Performance at Reduced Precision, Geophysical Research Letters, 48, e2020GL091363, https://doi.org/10.1029/2020GL091363, 2021.

Zhang, G. J. and McFarlane, N. A.: Sensitivity of climate simulations to the parameterization of cumulus convection in the Canadian climate centre general circulation model, Atmosphere-Ocean, 33, 407-446, 10.1080/07055900.1995.9649539, 1995.

Zhang, G. J. and Song, X.: Convection Parameterization, Tropical Pacific Double ITCZ, and Upper-Ocean Biases in the NCAR CCSM3. Part II: Coupled Feedback and the Role of Ocean Heat Transport, Journal of Climate, 23, 800-812, 10.1175/2009jcli3109.1, 2010.

Zhang, G. J., Song, X., and Wang, Y.: The double ITCZ syndrome in GCMs: A coupled feedback problem among convection, clouds, atmospheric and ocean circulations, Atmospheric Research, 229, 255-268, https://doi.org/10.1016/j.atmosres.2019.06.023, 2019.

Zhang, Y. and Yang, Q.: A Survey on Multi-Task Learning, IEEE Transactions on Knowledge and Data Engineering, 1-1, 10.1109/TKDE.2021.3070203, 2021.

Table 1. Input and output variables. For inputs, $\boldsymbol{q_v(z)}$ denotes the vertical profile of water vapor. $\boldsymbol{T(z)}$is the profile of temperature, and $\boldsymbol{dq_{v\ l.s.}(z)}$and $\boldsymbol{dT_{l.s.}}$ are the large scale forcing of water vapor and temperature, respectively. $\boldsymbol{P_s}$ is the surface pressure and *Solin* is the TOA solar insolation. For outputs, $\boldsymbol{dq_v(z)}$ and $\boldsymbol{dT(z)}$ are the tendencies of water vapor and temperature due to moist physics and radiative processes calculated by the NN-Parameterization. The net longwave and shortwave fluxes at the surface and the TOA are surface net longwave flux (FLNS), surface net shortwave flux (FLNT), TOA net longwave flux (FLNT), and TOA net shortwave fluxes (FSNT). The 4 downwelling solar radiation including solar downward visible direct to surface (SOLS), solar downward near infrared direct to surface (SOLL), solar downward visible diffuse to surface (SOLSD), and solar downward near infrared diffuse to surface (SOLLD) are shortwave radiation fluxes reaching the surface.

Inputs	Outputs
$q_v(z)$, $T(z)$, $dq_{vls}(z)$, $\mathrm{d}T_{ls}(z)$, P_s, *Solin*	$dq_v(z)$, $dT(z)$, *FLNS, FSNS, FLNT, FSNT, SOLS*, *SOLL*, *SOLSD*, *SOLLD*

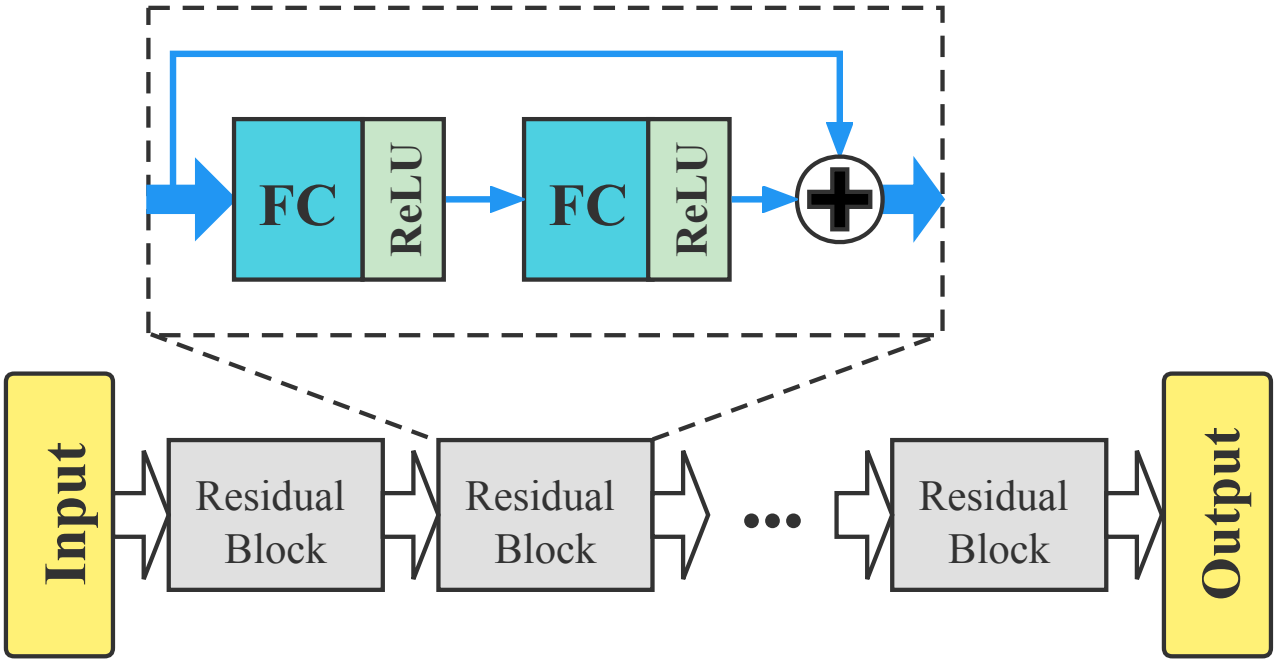


Figure 1. Schematic showing the structure of ResDNN. It consists of 7 residual blocks, each of which (shown in dashed box) contains two 512 node-wide dense (fully-connected) layers with a ReLU as activation, and a layer jump. The input and output are discussed in section 2.2.1.

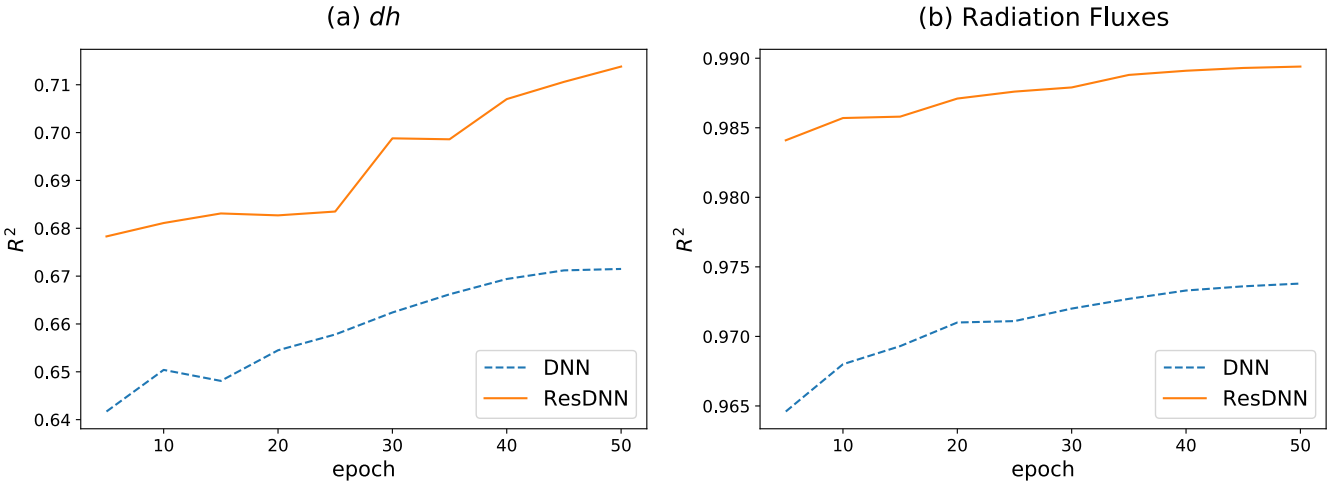


Figure 2. Fitting accuracies (R^2) of both the proposed ResDNN (orange solid lines) and DNN (blue dashed lines) for different targets. (a) shows the R^2 of moist static energy changing rate (dh) versus training epochs and (b) shows the fitting accuracy of the average R^2 over the 8 radiation fluxes. Note: Spatial averaging of MSE is performed before calculating R^2.

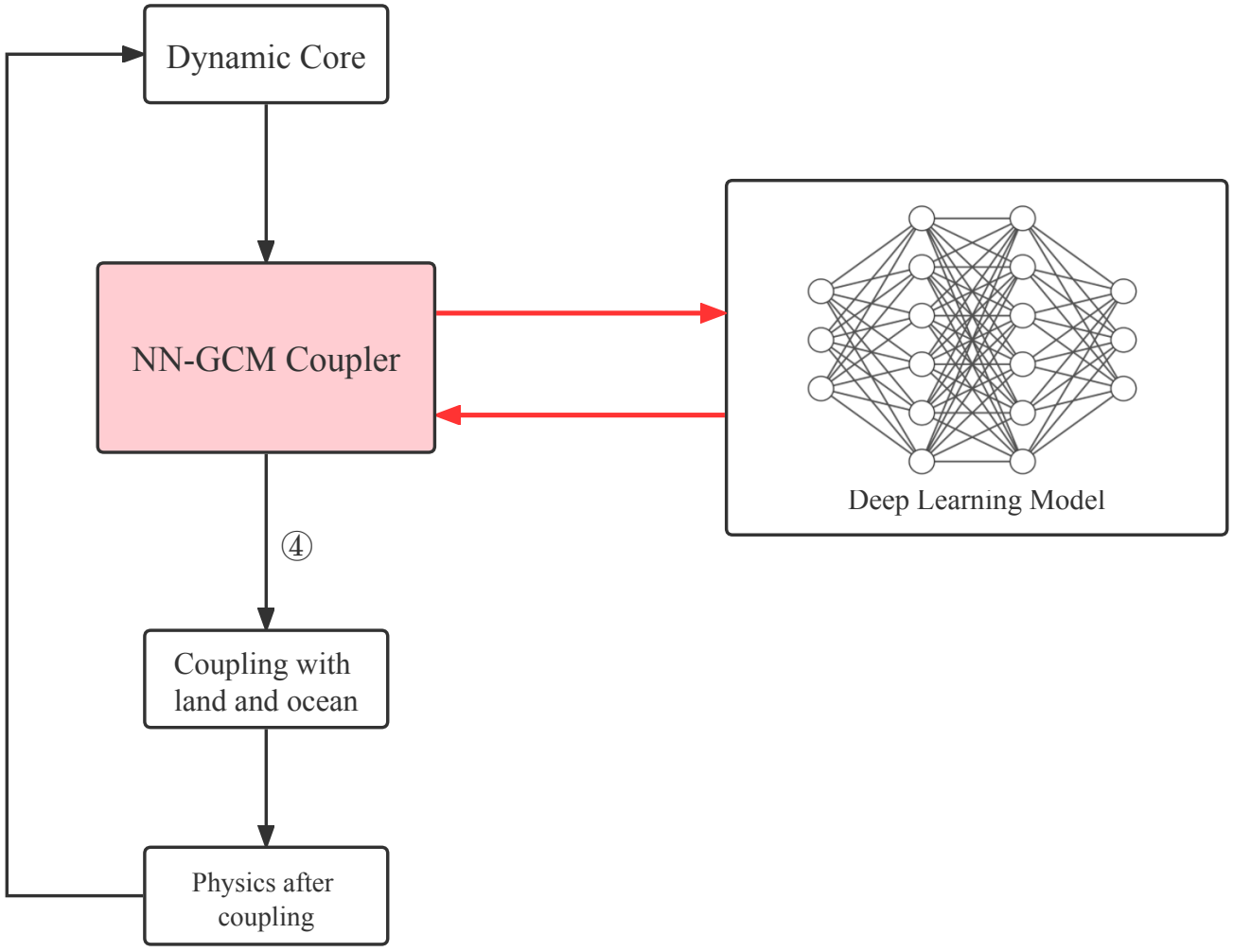


Figure 3. A flow chart of NNCAM including NN-GCM Coupler. NNCAM runs in the direction of the arrow, and each box represents a module. Among them, DNN-GCM Coupler is indicated by light red. NN-Parameterization is shown in the sub-figure on the right. Note: ① represents the dynamic core transmits data to NN-GCM Coupler; ② and ③ represent the data communication between NN-GCM Coupler and NN-Parameterization; ④ represents the host GCM accepts the result from NN-Parameterization.

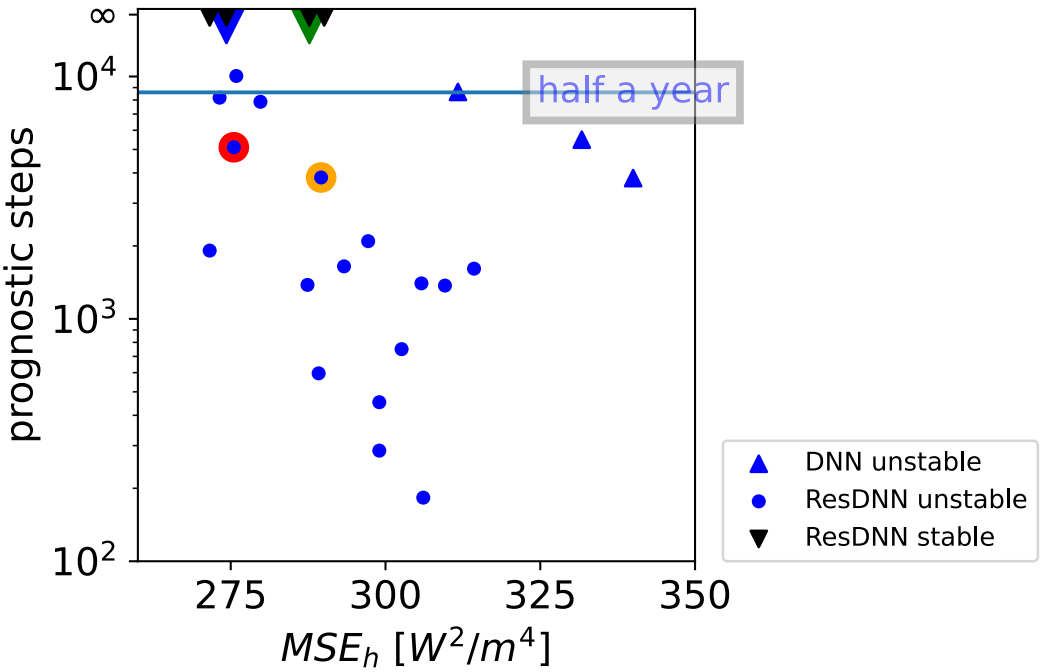


Figure 4. The offline moist static energy mean square error vs. prognostic steps. The black reversed triangles are stable NN coupled prognostic simulations lasting more than 10 years, blue ones are unstable simulations, and the blue triangles are for DNNs. The marked dots with colored outline are later exhibited in Figure 5 for time evolution of global averaged energy.

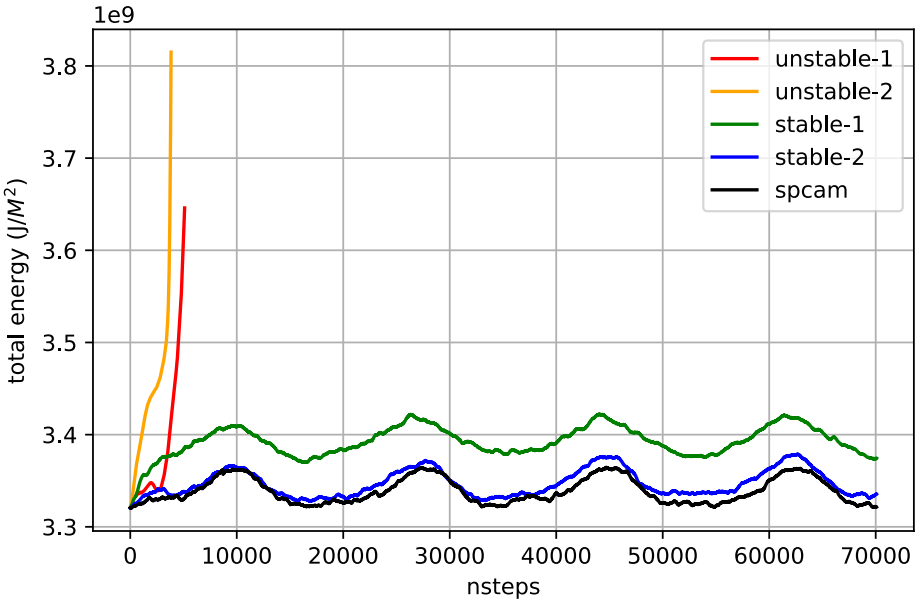


Figure 5. Time evolution of global averaged column integral total energy of NNCAM with different ResDNN parameterizations (marked with the same colors in Figure 4) and SPCAM target (the black line): Blue for stable and accurate ResDNN, green for a stable but deviated ResDNN, orange and red lines for unstable ResDNN.

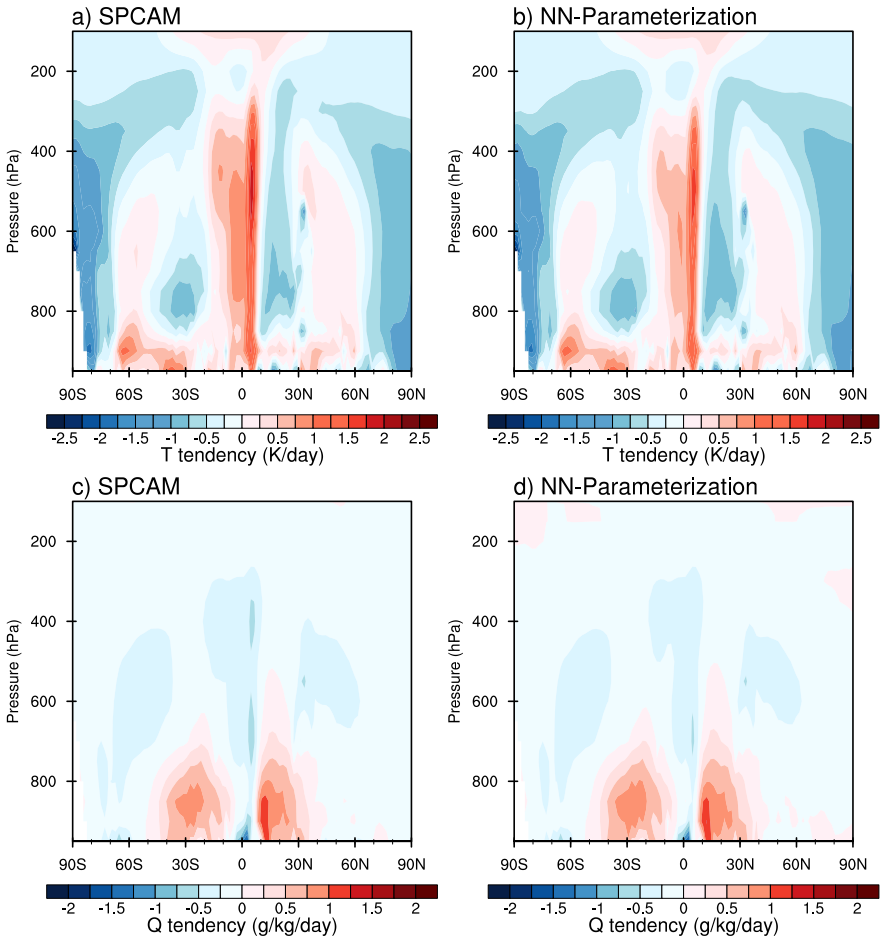


Figure 6. Latitude-pressure cross sections of annual and zonal mean heating (top) and moistening (bottom) from moist physics during the year 2000 for (a, c) SPCAM simulations, and (b, d) offline test by the NN-Parameterization.

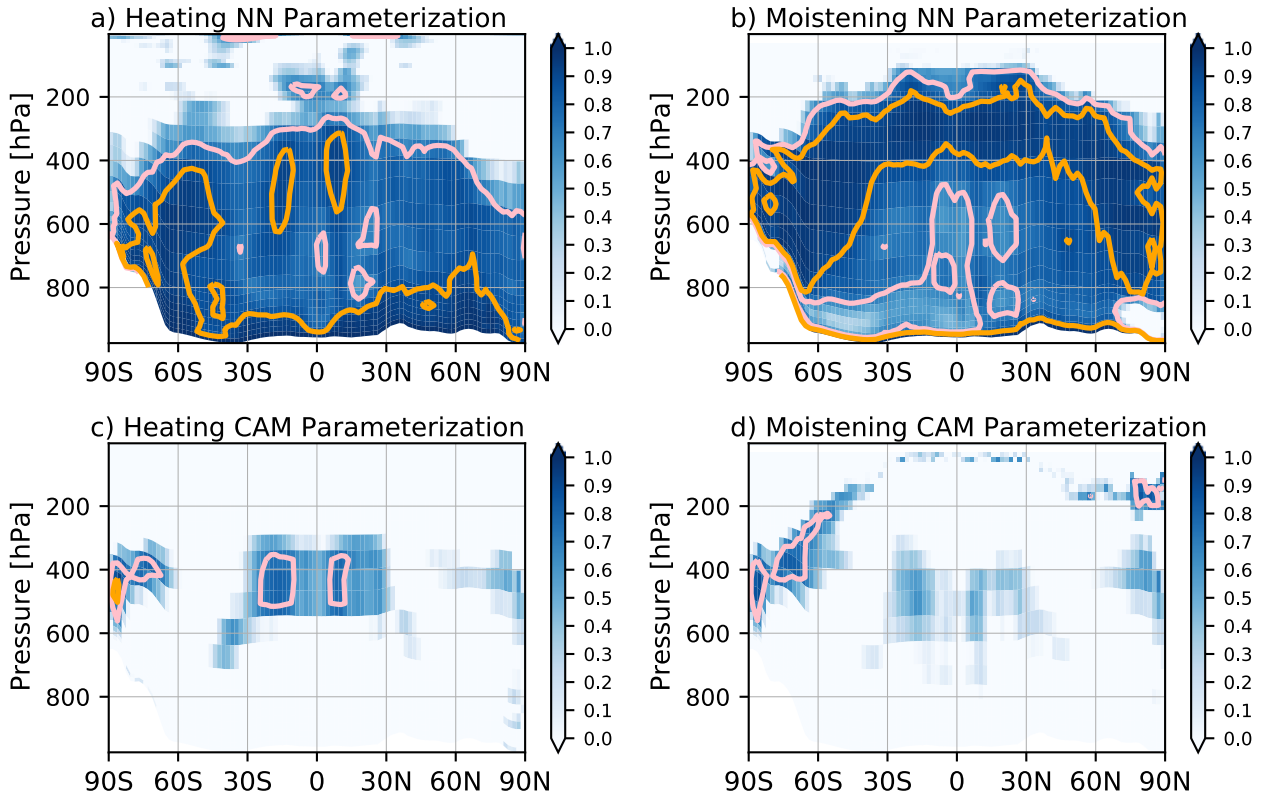
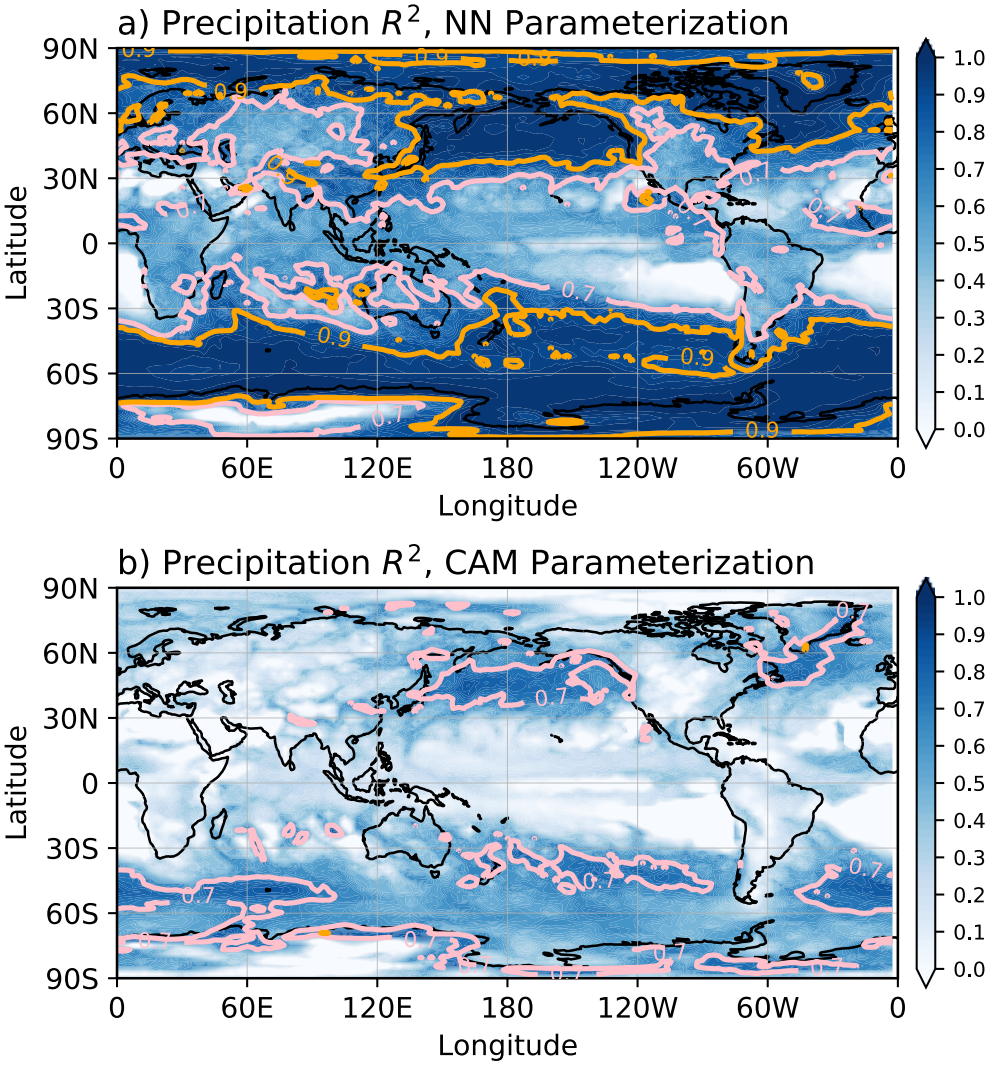


Figure 7. Latitude-pressure cross sections of coefficient of determination (R^2) for zonal averaged heating (left panels) and moistening (right panels). They are predicted by (a & b) NN-Parameterization in the offline one-year SPCAM run, and (c & d) by offline CAM5 parameterizations. Both are evaluated at 30-min timestep interval. Note: areas where R^2 is greater than 0.7 are contoured in pink and those greater than 0.9 are contoured in orange.



685

Figure 8. Latitude-pressure cross sections of coefficient of determination (R^2) for the derived precipitation predicted by NN-parameterization (a) and total precipitation from CAM5 parameterization (b) in the offline one-year SPCAM run. The predictions and SPCAM targets are in 30min timestep interval. Note: areas where R^2 is greater than 0.7 are contoured in pink and those greater than 0.9 are contoured in orange.

690

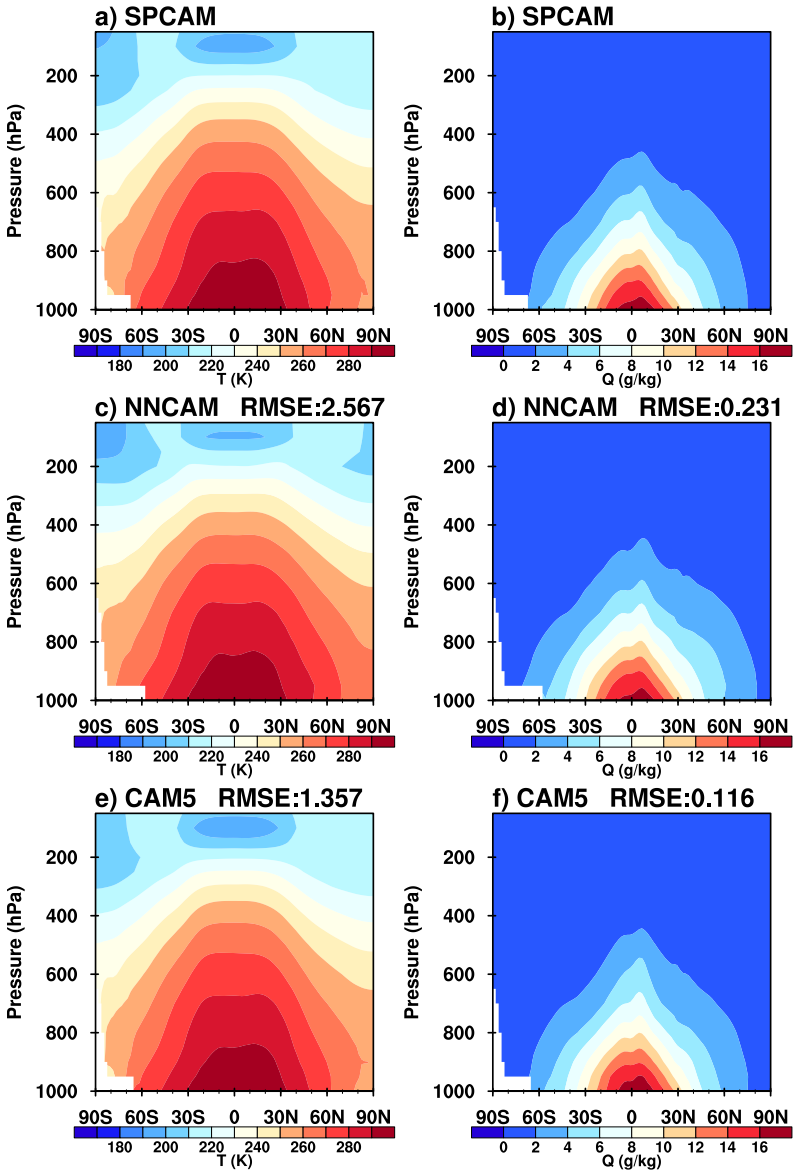


Figure 9. Latitude-pressure cross sections of annual and zonal mean temperature (left panels) and specific humidity (right panels) from (a, b) SPCAM (1999−2003), (c, d) NNCAM (1999−2003), and (e, f) CAM5 (1999−2003).

695

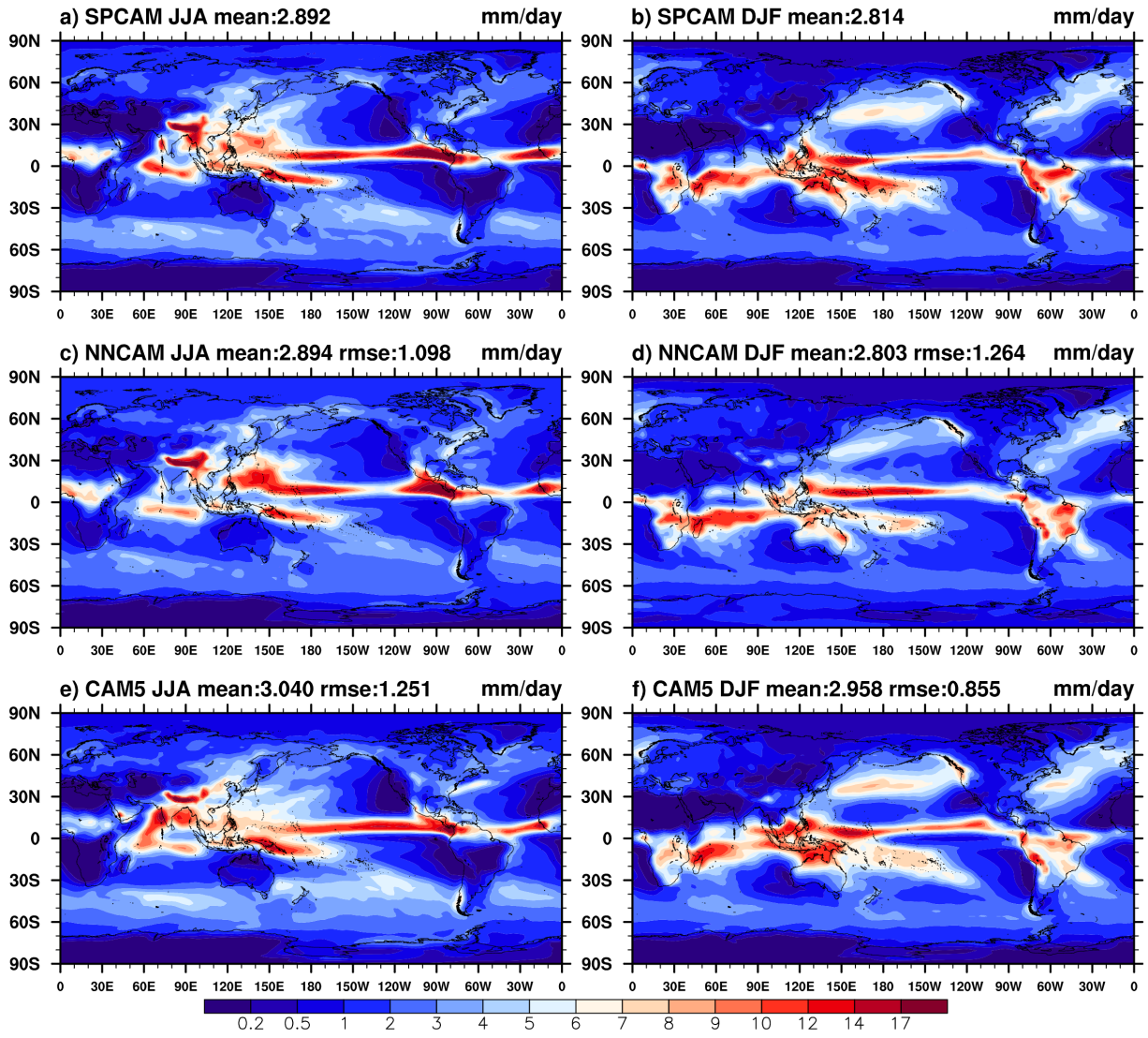


Figure 10. The mean precipitation rate (mm day^{-1}) of June-July-August (left panels) and December-January-February (right panels) for (a, b) SPCAM (1999−2003), (c, d) NNCAM (1999−2003), and (e, f) CAM5 (1999−2003).

700

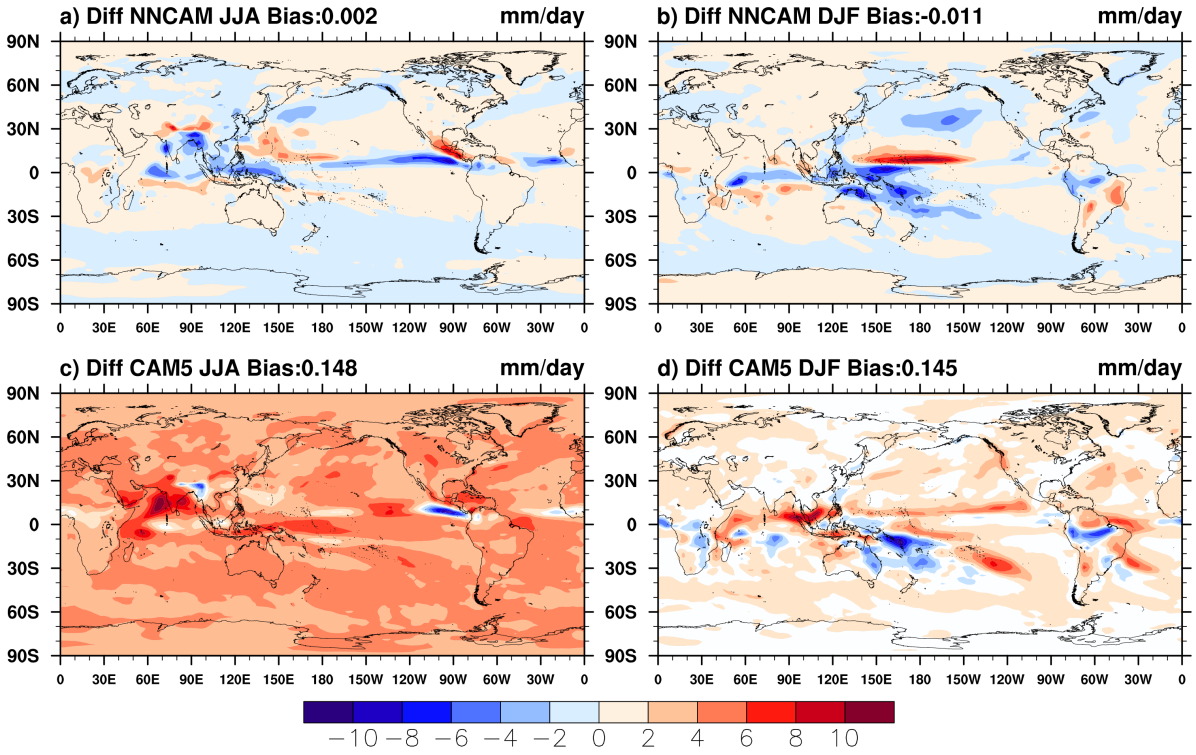


Figure 11. Global distribution of precipitation difference averaged over boreal summer (left panels) and winter (right panels) between NNCAM and SPCAM (a & b) and between CAM5 and SPCAM (c & d).

705

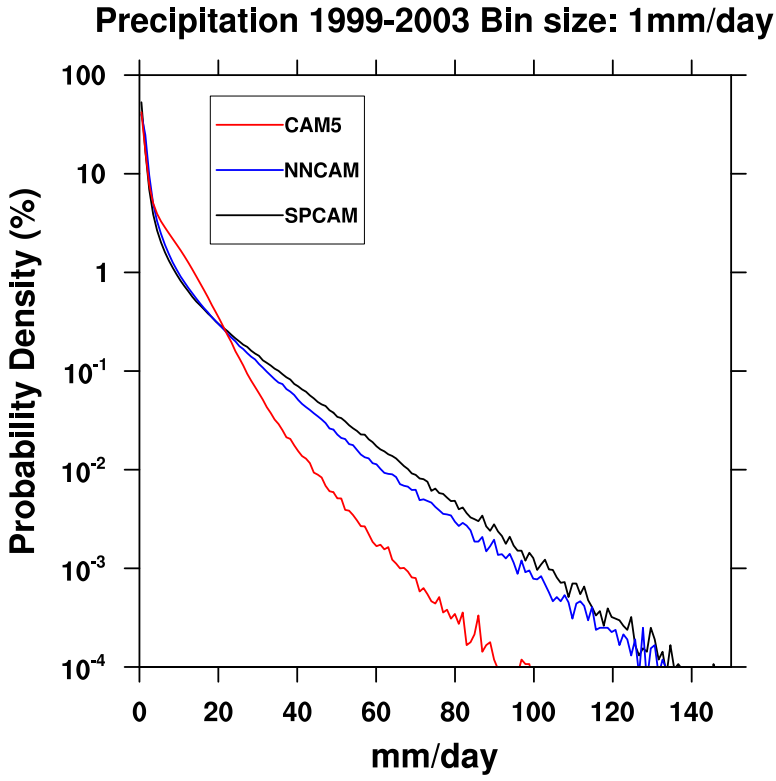


Figure 12. Probability densities of daily mean precipitation in the tropics (30°S−30°N) from the three model simulations. Black, blue and red solid lines denote SPCAM, NNCAM and CAM5, respectively.

710

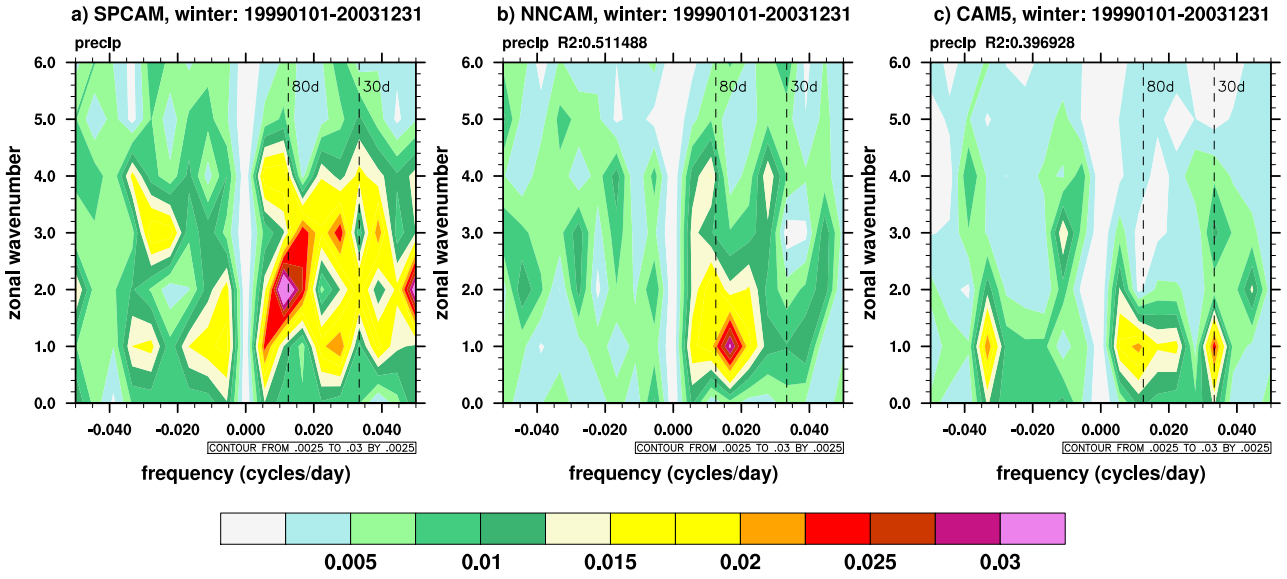


Figure 13. The wavenumber-frequency spectra of 10°S−10°N daily precipitation anomalies for (a, b) SPCAM, (c, d) NNCAM, and (e, f) CAM5 simulations for boreal winter.

715

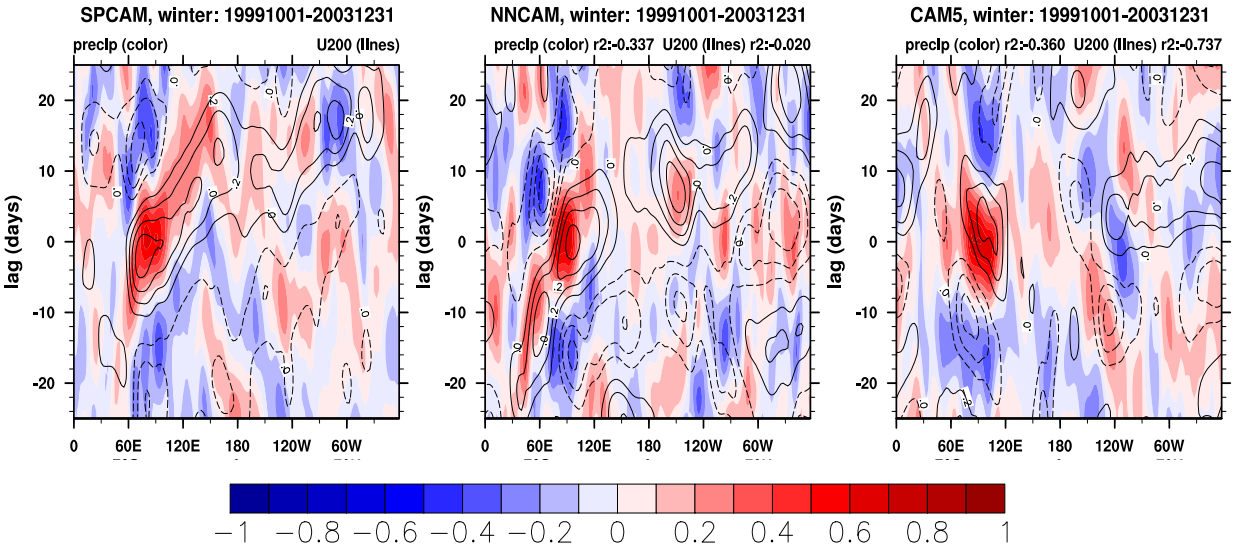


Figure 14. Longitude-time evolution of lagged correlation coefficient for the 20-100 day band-pass-filtered precipitation anomaly (averaged over 10°S−10°N) against regionally averaged precipitation (shaded) and zonal wind at 200hPa (contoured) over the equatorial eastern Indian Ocean (80E−100°E, 10°S−10°N).