


RESEARCH ARTICLE

10.1029/2020MS002268

Key Points:

- A stochastic bin model of warm rain formation is added to a General Circulation Model at high computational cost
- Key warm rain metrics are improved, including precipitation onset and frequency
- Neural networks can efficiently replicate the results of a stochastic bin model rain scheme with low computational cost

Correspondence to:

A. Gettelman,
andrew@ucar.edu

Citation:

Gettelman, A., Gagne, D. J., Chen, C.-C., Christensen, M. W., Lebo, Z. J., Morrison, H., & Gantos, G. (2021). Machine learning the warm rain process. *Journal of Advances in Modeling Earth Systems*, 13, e2020MS002268. <https://doi.org/10.1029/2020MS002268>

Received 30 JUL 2020

Accepted 19 JAN 2021

Machine Learning the Warm Rain Process

A. Gettelman^{1,2} , D. J. Gagne¹ , C.-C. Chen¹ , M. W. Christensen^{2,3}, Z. J. Lebo⁴ , H. Morrison¹ , and G. Gantos¹

¹National Center for Atmospheric Research, Boulder, CO, USA, ²Department of Physics, Oxford University, Oxford, UK,

³Now at: Atmospheric Sciences and Global Change Division, Pacific Northwest National Laboratory, Richland, WA, USA, ⁴Department of Atmospheric Science, University of Wyoming, Laramie, WY, USA

Abstract Clouds are critical for weather and climate prediction. The multiple scales of cloud processes make simulation difficult. Often models and measurements are used to develop empirical relationships for large-scale models to be computationally efficient. Machine learning provides another potential tool to improve our empirical parameterizations of clouds. To explore these opportunities, we replace the warm rain formation process in a General Circulation Model (GCM) with a detailed treatment from a bin microphysical model that causes a 400% slowdown in the GCM. We analyze the changes in climate that result from the use of the bin microphysical calculation and find improvements in the rain onset and frequency of light rain compared to high resolution process models and observations. We also find a resulting change in the cloud feedback response of the model to warming, which will significantly impact the climate sensitivity. We then replace the bin microphysical model with several neural networks designed to emulate the autoconversion and accretion rates produced by the bin microphysical model. The neural networks are organized into two stages: the first stage identifies where tendencies will be nonzero (and the sign of the tendency), and the second stage predicts the magnitude of the autoconversion and accretion rates. We describe the risks of overfitting, extrapolation, and linearization by using perfect model experiments with and without the emulator. We can recover the solutions with the emulators in almost all respects, and get simulations that perform as the detailed model, but with the computational cost of the control simulation.

Plain Language Summary Cloud processes are perhaps the most critical and uncertain processes for weather and climate prediction. The complex nature of clouds and their variation at small spacial scales makes simulation of clouds very challenging. There exist many observations and detailed simulations of clouds that are used to develop and evaluate larger-scale models. Many times these models and measurements are used to develop empirical relationships for large-scale models to be computationally efficient. Machine learning provides another potential tool to improve our empirical parameterizations of clouds. We replace the warm rain formation process in an earth system model with an emulator. The emulator consists of multiple neural networks that predict whether specific tendencies will be nonzero and the magnitude of the nonzero tendencies. We describe the opportunity (massive speed up of cloud process calculations) and the risks of overfitting, extrapolation and linearization of a nonlinear problem by using perfect model experiments with and without the emulator.

1. Introduction

Clouds are one of the most critical yet uncertain aspects of weather and climate prediction. The complex nature of subgrid scale (SGS) cloud processes makes traceable simulation of clouds across scales difficult (or impossible). There exist many observations and detailed simulations of clouds that are used to develop and evaluate larger-scale models. Many times these models and measurements are used to develop empirical relationships for large-scale models to be computationally efficient, because using more detailed treatments is computationally prohibitive. Machine learning (ML) provides another potential tool to improve such parameterizations, by using detailed models either off-line or on-line and then building emulators for them to reduce simulation time (Krasnopolsky et al., 2005). Here we present a comprehensive investigation of replacing the warm rain formation process in an earth system model with on-line emulators that use detailed treatments from small-scale and idealized models to represent key cloud microphysical processes.

© 2021. The Authors.

This is an open access article under the terms of the Creative Commons Attribution-NonCommercial License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited and is not used for commercial purposes.

The warm rain formation process is critical for weather and climate prediction and governs the location, intensity, and duration of rainfall events, critical for weather and the hydrologic cycle. Rain formation also affects cloud lifetime and Cloud Radiative Effects (CRE), making it critical for predicting climate (Albrecht, 1989; Twomey, 1977). The specific process of rain formation is altered by the microphysical properties of clouds, making warm rain formation (with no ice involved) dependent on the size distribution of cloud drops, and thus ultimately susceptible to changes in the distribution of aerosol particles that act as Cloud Condensation Nuclei (CCN).

Ice of course will complicate the precipitation process. Supercooled liquid drops can exist, and these will either precipitate in a similar manner to warm precipitation or subsequently may freeze. Cloud droplets may also freeze and form ice crystals, which precipitate and collect liquid, freezing or riming as they fall. We will not concern ourselves in this work with processes involving (or potentially involving) ice. This of course is a critical issue for weather (Forbes & Ahlgren, 2014) and climate (Bodas-Salcedo et al., 2019; Gettelman et al., 2019) prediction, but is beyond the scope of this initial proof of concept.

The representation of rain formation in clouds involves the interaction of a population of hydrometeors. For warm clouds, the process is one of condensation, and then collision and coalescence, the latter usually calculated in models by solving the quasistochastic collection equation (Pruppacher & Klett, 1997). This treatment neglects correlations and fluctuations that impact collision/collection (Grabowski et al., 2019), and thus cannot capture stochastic impacts on rain formation like “lucky drops” that might be important for warm rain formation (Kostinski & Shaw, 2005; Wilkinson, 2016). We prefer the term “quasistochastic collection” rather than “stochastic collection” as the equation has sometimes been referred to previously. The quasistochastic collection process describes how each size particle interacts with other sizes. Quasistochastic collection can result in bimodal distributions of hydrometeors (or, at least, distributions that cannot be well represented by a single mode gamma function); these modes are usually termed “cloud” (small) and “rain” (large) drops. Inherently, the processes evolving cloud droplets and rain drops are different. For example, cloud droplets grow primarily by condensation whereas raindrops grow primarily by collision-coalescence. Moreover, sedimentation is in the Stokes’ regime for cloud droplets but not so for rain. Thus, in nearly all bulk microphysics schemes, the cloud and raindrop populations are modeled using separate distributions.

The quasistochastic collection process is computationally expensive to treat directly in large-scale global models for weather and climate prediction. It requires the precomputation of a collection kernel for how different sizes of hydrometeors will interact due to differential fall speeds, and it requires tracking populations of drops discretized by size bins. The tracking and advection of at least order 60 different bin quantities for liquid and ice combined makes bin schemes computationally expensive. Moreover, there is a conceptual mismatch in using detailed and computationally costly representations of microphysics in large-scale models that cannot resolve cloud- or mesoscale motions, although this may become less of a concern with increasing resolution of global models with even convection-permitting (order kilometer-scale) global simulations now becoming routine. Thus, traditionally, large-scale models with bulk microphysics have treated the quasistochastic collection process of warm rain formation in a heavily parameterized fashion (Khairoutdinov & Kogan, 2000; Seifert & Beheng, 2001). For conceptual simplicity, the process is often broken up into two processes: autoconversion and accretion. Autoconversion is the transition of cloud drops into rain as part of a cloud droplet distribution grows to large sizes. Accretion is the collection of cloud drops by larger rain drops falling through a cloud. Methods for determining autoconversion and accretion in bulk schemes vary widely. For instance, the earliest approaches were based on heuristics, requiring a threshold cloud water to be exceeded for autoconversion to commence, with “continuous collection” assumed for accretion (e.g., Kessler, 1969). More recent approaches have fit autoconversion and accretion rates to output from bin microphysics in large eddy simulation (LES) models (e.g., Khairoutdinov & Kogan, 2000; Kogan, 2013), or have incorporated theoretical aspects (e.g., Liu & Daum, 2004). Because they are the major loss mechanism for cloud water, different descriptions of these processes can result in very different model evolution and climates (e.g., Michibata & Takemura, 2015).

Because many existing formulations for autoconversion and accretion are simply empirical fits to data or other models, they are readily applicable to replacement with more sophisticated tools. Neural networks are multivariate function approximators that allow many more degrees of freedom than traditional polynomial or power-law methods, for example. They are usually trained on large data sets from observations or other

models. Neural networks were first used to emulate radiation parameterizations in weather (Chevallier et al., 2000) and climate (Krasnopol'sky et al., 2005) models and provided significant speedups with limited reductions in predictive accuracy. More recent work in this area has focused on emulating the effects of convection based on convection-resolving simulations (Brenowitz & Bretherton, 2018) or superparameterization (Gentine et al., 2018; Rasp et al., 2018) with promising emulation results limited by issues with numerical model stability when running the emulator for an extended time or when running the emulator outside the training climate. Stability issues for emulators have been discussed by Brenowitz and Bretherton (2019). Constraints on the neural network loss function, architecture, and inputs appear to assist in better performance (Beucler et al., 2020). Han et al. (2020) added energy constraints to the loss function as well. Other ML frameworks, such as random forests, feature architectures that inherently conserve energy and limit predictions to within the bounds of the training data, resulting in more stable simulations (Yuval & O'Gorman, 2020). Another path to greater numerical stability is to focus on emulating a smaller, but important, subset of the subgrid physical processes, which we investigate in this paper.

In this work, we replace the traditional empirically fit autoconversion and accretion rates in a GCM (following the approach of Khairoutdinov & Kogan, 2000) by solving the quasistochastic collection equation using a detailed bin microphysical model. This approach is implemented directly on-line in the GCM microphysics. The resulting code is too computationally expensive for practical simulations (as will be shown below), so we use a neural network to then emulate the code. We pose two hypotheses:

- Hypothesis 1: Simulating warm rain in a GCM by directly solving quasistochastic collection using a bin approach will greatly increase the computational cost, but will result in a qualitatively different warm rain formation processes and timing, as well as quantitatively different climate means and even emergent properties (e.g., aerosol-cloud interactions and cloud feedbacks)
- Hypothesis 2: ML Neural Network emulators can speed up the process and reproduce the qualitative and quantitative changes seen during testing of Hypothesis 1

The first hypothesis is independent of the neural network emulator, and tests whether there is sensitivity for climate to how the warm rain process is treated. The second question is a general test of the neural network emulator concept: can it work on-line in a “standard” climate simulation? Hypothesis 2 tests what is known as a “perfect model experiment”: can we reproduce the desired code from Hypothesis 1 with a neural network, since we have results of the code we are trying to reproduce.

The details of the model and methodology are discussed in Section 2. Results for emulator performance relative to the bin code are presented in Section 3. Simulation results replacing the existing autoconversion and accretion formulations are described in Section 4, including discussion of process rates, mean climate, and emergent properties. Discussion is in Section 5, and conclusions are in Section 6.

2. Methods

Here we describe the model used (Section 2.1), details of the quasistochastic collection treatment (Section 2.2) and details of the subsequent ML emulator methods (Section 3). Emulator and nonbase CAM code described here is archived at <https://github.com/NCAR/mlmicrophysics> and is based on the CAM6.0.000 code (see below).

2.1. Community Atmosphere Model Version 6

The Community Atmosphere Model version 6 (CAM6) is the atmospheric GCM component of the Community Earth System Model version 2 (Danabasoglu et al., 2020). CAM6 is more completely described in Gettelman et al. (2019) and Gettelman et al. (2020), here we repeat key warm cloud microphysical parameterization information. CAM6 features a two-moment stratiform cloud microphysics scheme (Gettelman & Morrison, 2015; Gettelman et al., 2015, hereafter MG2) with prognostic cloud liquid, cloud ice, rain, and snow hydrometeor classes. MG2 is coupled to a unified moist turbulence scheme, Cloud Layers Unified by Binormals (CLUBB), developed by Golaz et al. (2002) and Larson et al. (2002) and implemented in CAM by Bogenschutz et al. (2013). CLUBB handles stratiform clouds, boundary layer moist turbulence, and shallow

convective motions. CAM6 also has an ensemble plume mass flux deep convection scheme described by Zhang and McFarlane (1995) and Neale et al. (2008), which has very simple microphysics. Within the MG2 parameterization, the warm rain formation process is represented by expressions for autoconversion and accretion from (Khairoutdinov & Kogan, 2000, hereafter KK2000). KK2000 uses empirical power law fits to LES with bin-resolved microphysics to define:

$$\left(\frac{\partial q_r}{\partial t} \right)_{AUTO} = 13.5 q_c^{2.47} N_c^{-1.1} \quad (1)$$

$$\left(\frac{\partial q_r}{\partial t} \right)_{ACCRE} = 67(q_c q_r)^{1.15} \quad (2)$$

where q_c and q_r are mass mixing ratios for condensate and rain, and N_c is the number mixing ratio of condensate. For CAM6, the autoconversion rate exponent on N_c (-1.1) and prefactor (13.5) in Equation 1 have been adjusted from the original Khairoutdinov and Kogan (2000) scheme to better match observations (Gettelman et al., 2019). KK2000 also includes a source of drizzle drop number concentration from the autoconversion mass source assuming 25-micron radius drops and a corresponding sink of cloud drop number concentration from autoconversion and accretion mass.

2.2. Quasistochastic Collection

We replace the KK2000 process rate equations with an estimate of the quasistochastic collection process from the Tel Aviv University (TAU) bin microphysical model. The TAU model uses a “bin” or “sectional” approach, where the drop size distribution is resolved into 35 size bins. It differs from most other microphysical codes in that it solves for two moments of the drop size distribution in each of these bins. This allows for an accurate transfer of mass between bins and alleviates anomalous drop growth (Tzivion et al., 1987). The original components were developed by Tzivion et al. (1987, 1989) and Feingold et al. (1988), with later applications and development documented in Reisin et al. (1996), Stevens et al. (1996), Tzivion et al. (1999), Yin et al. (2000), Harrington et al. (2000), and Lebo and Seinfeld (2011). Note that process rates are one aspect of bin microphysics, another critical aspect being that bin schemes prognose multiple microphysical variables and thus evolve hydrometeor size distributions with many degrees of freedom. On the other hand, MG2 represents the drop size distribution with only four degrees of freedom, corresponding to two bulk prognostic variables each for cloud and rain. Here we will employ the bin approach to obtain autoconversion and accretion rates, but will retain the bulk MG2 approach with four prognostic microphysical variables to evolve cloud and rain.

The method of application in CAM is as follows. First, we discretize the MG2 bulk size distributions for cloud liquid and rain into number concentrations in individual bins. Cloud liquid and rain are put in the same continuous distribution of 35 mass-doubling bins for the TAU code. Then we use this as input to the TAU code for quasistochastic collection, assigning the mass variables (which is needed since TAU is a two-moment bin scheme predicting number and mass in each bin) based on the mean bin mass. The quasistochastic collection code has 60 substeps in the 1800s GCM time step, effectively a 30s timestep to evolve the distributions. This was found to yield similar results to smaller timesteps (5s) but with more computational efficiency. The result is a revised set of 35 mass bins with the number in each bin. We then find a local minimum in the distribution of final drop number across bins: this is always found in the case where there is rain and condensate present after the application of the collection kernel. The minimum is typically between 40 and 100 microns diameter. Not using a fixed value of the minimum eliminates a set parameter, and since a minimum can almost always be found, the separation is related to the physical distribution of drops. This minimum is used to divide the bins into cloud liquid and rain. The total number and mass of cloud liquid and rain is defined, and tendencies calculated as the final total mass and number resulting from the quasistochastic collection calculation minus the initial mass and number divided by the 1800s GCM time step. This estimated quasistochastic collection tendency is then directly applied instead of the KK2000 accretion and autoconversion tendencies in the code. Nothing else is changed.

MG2 couples the KK2000 process rates to a SGS distribution of cloud water (Morrison & Gettelman, 2008, Equation 8), but this is not done with the TAU bin code. In CAM6 and most other models, the impact of the SGS distribution is modeled using a linear scaling factor for the autoconversion and accretion rates, with different scaling factors for each. While a linear scaling could be applied to the TAU process rates, it is not possible to do this in a way consistent with MG2 because the approach using TAU does not separate autoconversion and accretion. We therefore neglect this adjustment for the TAU rates, but this study is a proof of concept for applying ML and it should not be considered as a final model tuning. Including the impacts of SGS distributions of cloud and rain should be considered in the future. This could potentially be done, for example, by explicitly integrating TAU process rates over the SGS distribution of cloud water to obtain a training data set for ML (see Discussion in Section 5). Note that SGS adjustments to autoconversion and accretion are performed on the control simulation. We have also run and analyzed a control simulation removing these SGS adjustments on autoconversion and accretion to understand their effects on the control simulation.

The code is also set up to simulate the accretion and autoconversion rates from MG2 on the same state, and this is saved as a diagnostic. This allows a direct comparison of the original MG2 KK2000 tendency (autoconversion + accretion) with the stochastic collection tendency from the TAU code.

2.3. Simulations

CAM6 is run in a standard $0.9^\circ \times 1.25^\circ$ (latitude and longitude) configuration with 32 levels in the vertical. Boundary conditions are climatological averages of Sea Surface Temperatures (SSTs), greenhouse gases and emissions of aerosols, and precursors appropriate for 1990–2010 (i.e., averaged around 2000). To build a training data set for the emulator, we output the instantaneous inputs and outputs from the quasistochastic collection code, which consists of the input state and tendencies of mass and number mixing ratios, along with air pressure and temperature. The advantage of this method is we can efficiently generate independent 4D samples (space and time) for training the emulator, of whatever size necessary. We sample the instantaneous output every 25 h over a 2-year period, so that the local time precesses through the diurnal cycle over a month. This method is run for two years to generate approximately 700 different timesteps over different seasons and times of day at 192 lat \times 288 lon \times 32 levels or 1.8 million samples per time step (about 1.2 billion samples total). We note that many samples have zero condensate at or above them (about 60% cloud cover, and 1/3 of the column levels for liquid, so approximately 20% of samples have column liquid). Note that zero samples are important as well for training the neural network since zero liquid cases need to be filtered out. The first simulated year of the model run (hours 861 through 8,800) are used for training, and the second year (hours 8,856 through 17,466) are used for offline validation.

Simulations for evaluation were run for one year with 3 h instantaneous output frequency. We then ran further simulations for 9 years to estimate long-term climate impacts. To estimate anthropogenic Aerosol-Cloud Interactions (ACI), identical simulations were conducted with aerosol and precursor emissions only set back to 1850 “preindustrial” conditions. To estimate cloud feedbacks, simulations with the same forcing but with SSTs increased uniformly by +4 K were conducted following Cess (1987).

Table 1 defines the different codes used for these simulations. The control CAM6 code uses KK2000 for autoconversion and accretion. TAU-Bin replaces KK2000 autoconversion and accretion with the stochastic collection code from the TAU bin microphysics code as described in Section 2.2. TAU-ML replaces this code with an emulator based on training data from TAU-Bin simulations as described in Section 3. ML-NoFixer is the TAU-ML code with limiters on the emulator tendencies removed. Cntl-NoEnh is the CAM6 control code with no SGS variability enhancement of rain formation to better match the TAU simulations (see below).

2.4. Emulation

The special properties of the microphysics data necessitate a more complex emulator structure than the single neural network emulator employed by others (e.g., Krasnopolksy et al., 2005; Rasp et al., 2018). Most of the microphysics inputs span many orders of magnitude. The tendencies are either zero or are exponentially

Table 1
Description of Different Code Types for Simulations

Name	Description
Control-KK2000	CAM6 model with KK2000
Cntl-NoEnh	CAM6-KK2000 with no enhancement due to SGS variability
TAU-Bin	CAM6 with KK2000 replaced with warm rain from bin
TAU-ML	TAU-Bin replaced with neural network emulator from TAU-Bin
ML-NoFixer	TAU-ML with mass fixer removed

Abbreviations: CAM6, Community Atmosphere Model version 6; ML, machine learning; SGS, subgrid scale; TAU, Tel Aviv University.

distributed. The N_r tendency is exponentially distributed in both the positive and negative directions. A single fully connected neural network of sufficient width and depth can approximate any continuous function according to the universal approximation theorem (Pinkus, 1999), but microphysics tendencies exhibit tendencies of zero when clouds and precipitation are not occurring. In order to account for these complexities, the data are transformed and two to three neural networks are used to predict the sign and magnitude of each tendency.

The input variables, preprocessing, and neural network setup is shown in Figure 1. The input variables are summary statistics describing the cloud liquid and rain water distributions in each grid cell along with air density and the fraction of the grid volume occupied by cloud water and precipitation. Mixing ratio and number concentration should be sufficient statistics to describe the particle size distributions, but we found that including slope, intercept, and spectral width parameters further improved the offline validation results. The distribution statistics and output tendencies are all log-transformed. Negative tendencies are first multiplied by -1 before being log-transformed. All inputs and outputs are normalized by subtracting the training set mean and dividing by the training set standard deviation.

The ML emulator system consists of three classifier neural networks to predict whether each tendency is nonzero and four regression neural networks to predict the magnitude of the tendencies as indicated in Figure 1. The classifier networks for q_r and N_c predict the tendency to be either zero or nonzero, but the N_r classifier network predicts whether the tendency is negative, zero, or positive since self-collection results in a negative N_r tendency and autoconversion results in a positive N_r tendency. The classifier neural networks are trained on all grid cells with either q_c or q_r greater than $10^{-18} \text{ kg kg}^{-1}$, and the regressor networks are trained only on samples with a nonzero tendency. Classifier neural networks are evaluated on all validation set samples, but regressor neural networks are only evaluated on validation samples with nonzero tendencies in the control run.

Each neural network consists of four fully connected hidden layers with 60 neurons in each layer, and Rectified Linear Unit (ReLU) activation functions (Nair & Hinton, 2010). Each network is trained for 10 epochs (passes through the training data) with a batch size of 4,096 examples. The Adam optimizer (Kingma & Ba, 2015) is used with a learning rate of 0.001. Categorical cross-entropy (Goodfellow et al., 2016) is used as the loss function for the classifier neural networks, and mean squared error is used for the regression neural networks. Ridge, or L2 norm regularization (Tikhonov, 1963), of the hidden layer weights with a penalty weight λ of 10^{-4} helps constrain the magnitude of the neural network weights, so that the training process will produce a more robust model even with correlated inputs. The neural networks are trained with Tensorflow (Abadi et al., 2016). The weights are saved to an intermediate netCDF file format that is then read into CESM and run during model integration using a custom-built Fortran 90 neural network inference module. The Fortran inference code can support densely connected neural networks with an arbitrary number of layers. The code is available as noted above.

As noted above, training data for the networks are based on simulations of CAM with the TAU bin code, using individual timestep samples in space and time as individual training events, for about 250 million individual samples with clouds. With this size of training data, there was little sensitivity to the number of time samples.

3. Results-Emulator Performance

First we describe some basic metrics of emulator performance before we analyze results. The first metric is timing. We have analyzed timing statistics from simulations with the CAM6 control code, the TAU quasistochastic collection code, and simulations where the TAU results have been replaced with the ML emulator (TAU-ML). We use standard CESM timing metrics for the total atmosphere model cost of 9 years, 1° (~100 km) horizontal resolution simulation. All simulations were performed using the same number of tasks and layout on the same supercomputer (taking about one wall clock hour per simulated year). Three separate control simulations estimate a standard deviation on the timing numbers between runs of the same code at about $\pm 5\%$. We find that using the TAU approach for autoconversion and accretion running in CAM results in a model run time over four times (+410%) longer than the control code. The emulator used to replace the TAU code in MG2 (TAU-ML) runs 8% slower than the control case. This is just beyond one standard deviation of the speed of individual years from the control and TAU-ML simulations, but may not be strictly significant.

Next we analyze if the emulator reproduces the TAU process rates it is designed to reproduce. The emulator produces a q_c , N_c , and N_r tendency, with q_r being the negative of the q_c tendency. Figure 2 illustrates a scatter plot of the rates between the emulated code and the underlying bin code. Focusing on the neural network versus the TAU bin code it is trained on (Figure 2, top row), the emulator does an excellent job of reproducing the training data (Table 2). Most of the density is on the 1:1 line and coefficients of determination are 0.98–1.00. Extreme values are asymmetric, indicating that the emulator does not reproduce all the extremes exactly. We define “extremes” as values of the tendency ratio $r = \text{TAU-ML}/\text{TAU}$ beyond a specific value. Table 3 illustrates that extremes beyond $r < 0.5$ or $r > 2$ occur less than 10% of the time for all tendencies except the positive N_r tendency. There is skewness toward smaller extremes (i.e., TAU-ML tendency extremes less than TAU are more frequent).

The slight asymmetries for extremes can be seen in Figure 3 as a difference in frequency for large tendency values. The PDF of the bin code is the purple line and the emulator PDF the blue. Here it is clear the emulator slightly narrows the distribution of process rates, with lower frequency of extreme high values. The distribution is narrower on the high end for dN_c/dt (Figure 3f), and positive dN_r/dt (Figure 3h).

Next we look at the difference between the emulator (TAU-ML) and the TAU bin code. Since we use different simulations, we evaluate it based on monthly means at each grid location on the planet. Figure 4 illustrates the ratio of dq_c/dt between the emulator and the bin code. As expected, emulated tendencies on average are within $\pm 20\%$ of the bin code. Blank regions in the upper troposphere and at high latitudes (Figure 4a) and in the tropics (Figure 4b) have values $< 1 \times 10^{-9} \text{ kg kg}^{-1} \text{ s}^{-1}$ (a few parts per million per minute). Lowest correspondence is in the deep tropics. This may be due to the prominence of deep convection and limited regions that meet the criteria there. But in most of the regions with significant tendencies, the process rate ratios are within $\pm 20\%$. The trend toward lower tendencies in the emulator at higher altitudes is not significant, as there is very little liquid above 700 hPa. At lower altitudes (Figure 4b), there is slight trend toward higher tendencies in the emulator (blue colors) in regions of low clouds in the E. Pacific and Indian Ocean, as well as over N. Hemisphere mid latitude land. This does not seem to be due to any significant difference in the thermodynamics between the simulations.

There are small differences in the mean state that can be seen in mean thermodynamic profiles of temperature, cloud fraction, cloud liquid, and rain. Figure 5 indicates that the TAU (orange) and TAU-ML (Green) differ only in cloud liquid and pretty much throughout the profile, peaking in the cloud layer. This occurs mostly outside of shallow cloud regions where clouds are deeper, but occurs in most regions.

Next we examine individual process rates from the warm cloud microphysics in different regions and compare the TAU and TAU-ML codes (Figure 6). Here we use instantaneous output from the simulations. TAU (red lines) is the bin or emulator tendency for autoconversion and accretion. Over the S. Ocean (65°S–50°S, 0–360°E), in mostly supercooled liquid clouds (Gettelman et al., 2020), the emulator (TAU-ML, Figure 6b), has nearly 50% less loss of condensate to precipitation near 800 hPa than the TAU bin code it is representing at the top of shallow cloud layers. Over the Subtropical Atlantic near Barbados (10°S–25°N, 290–320°E), the TAU-ML emulator (Figure 6e) has slightly more loss at the top of the shallow cloud layer, and less below,

Table 2
Machine Learning Emulator Classifier Performance on the Offline Validation Set for Each Tendency

Tendency	Accuracy	True positive rate	True negative rate
q_r	0.984	0.981	0.987
N_c	0.993	0.996	0.989
$+N_r$	0.985	0.967	0.989
$-N_r$	0.985	0.991	0.981

for very little change in the vertical column. Note that none of the other process rates are significantly impacted by the shift between the TAU code and the emulator (TAU-ML). We have looked at the mean thermodynamic profiles of temperature, cloud water and precipitation between the simulations. In the Barbados region are no significant differences between the TAU and TAU-ML emulator simulations. Over the S. Ocean, there is a slight tendency for more cloud water in the TAU-ML simulation, consistent with the reduced loss process. This might be a feedback that enhances the differences between the simulations.

When the TAU-ML emulator is trained on present day climate, it will occasionally produce tendencies that would result in a negative mass or number to either cloud or rain when applied to the model state in the TAU-ML simulations. We have built the model to check for nonconservation of the TAU-ML emulator results, and correct (“fix”) the mass if necessary. This is done by ensuring that the final masses and number concentrations of both cloud and rain are positive or zero, and reducing the tendencies accordingly. The fixer is not needed for TAU-Bin or KK2000 (control) simulations, and is in addition to constraints on the sum of all the process rate tendencies. The fixer is used in all TAU-ML simulations with present day or perturbed climate. In present day simulations (the climate where the emulator is trained), the fixer is not necessary for stability, but if we run the TAU-ML emulator with a perturbed simulation (e.g., SST+4K), then the model will crash without the fixer. It is an interesting research question regarding the level of stability and how to insure stability in the emulator.

The key difference in this work from past work such as Rasp et al. (2018) is that we apply a local correction to the tendencies to ensure that mass is conserved and number concentrations are physical (positive, or zero when mass is zero). This ensures that global total energy and moisture are constrained. Local mass and energy correction is standard practice in cloud microphysics schemes (e.g., Morrison et al., 2005). Most cloud microphysics schemes and process parameterization methods suffer from similar problems related to extrapolation and overfitting, even if a simple empirical function is used. Cloud microphysics has numerical issues simply related to the scales involved (the combination of drop numbers on the order of 10^8 and masses of order 10^{-6} can start to strain the numerical precision of models). In addition, a local correction is already applied to the sum of the process rates in MG2 since the microphysical rates are time split (all processes operate simultaneously on the same state).

Figure 7 illustrates the frequency of occurrence of the mass fixer in the TAU-ML control simulation. The highest frequency occurs in the S. Hemisphere subtropics close to the surface, with another lower peak frequency in the deep tropics at 800 hPa (Figure 7a). Figure 7 shows annual means, but seasonal means are similar: the peak remains in the S. in all seasons. Figure 7b indicates that the peak fixer invocation occurs nearly 1/3 of the time in regions of trade cumulus clouds in the subtropics. These are regions with small grid box average liquid water content due to small cloud fractions, but potentially high in-cloud water contents. We return to this later when looking at derived properties of the system. It is hypothesized that we could eliminate the need for the fixer by including perturbed climate samples in the training data.

We have also analyzed the correction relative to the mean tendency value and frequency of occurrence. The mean correction is quite small (<5% of the tendency) and peaks in the dry regions of the subtropical free troposphere above the boundary layer. Where there is significant cloud mass and the fixer frequency is greater than 0.04 in Figure 7a, the fixer is typically less than 0.5% of the mean tendency.

4. Results-Model Output

Having established that the emulator works to reproduce the results of the bin code, we now discuss the differences between the CAM base code (CAM6) and the code that directly solves the quasistochastic collection equation (TAU). We analyze first process rates and key metrics of warm rain formation, then the mean state climate, and finally emergent properties such as aerosol forcing and cloud feedbacks. We also assess whether there are further differences between TAU and TAU-ML simulations.

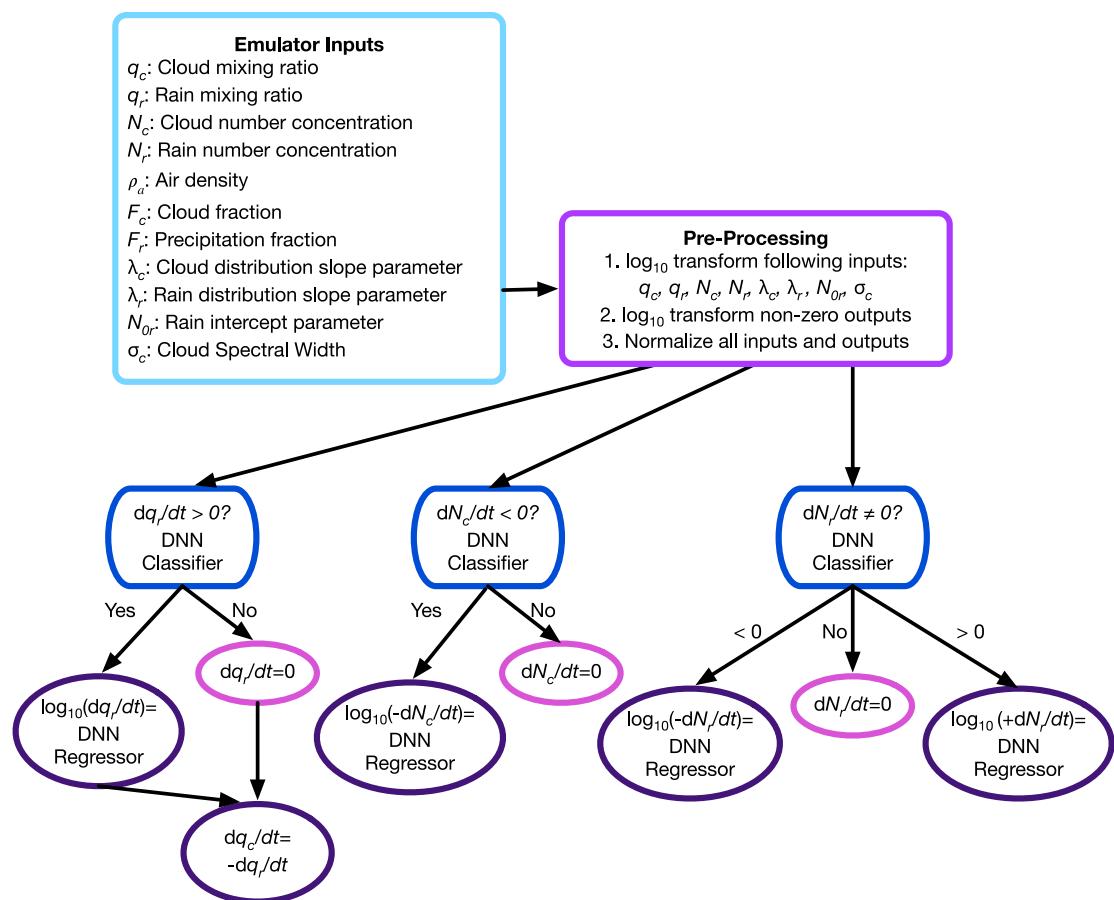


Figure 1. Flow diagram describing the preprocessing and machine learning pipeline for predicting each microphysical tendency.

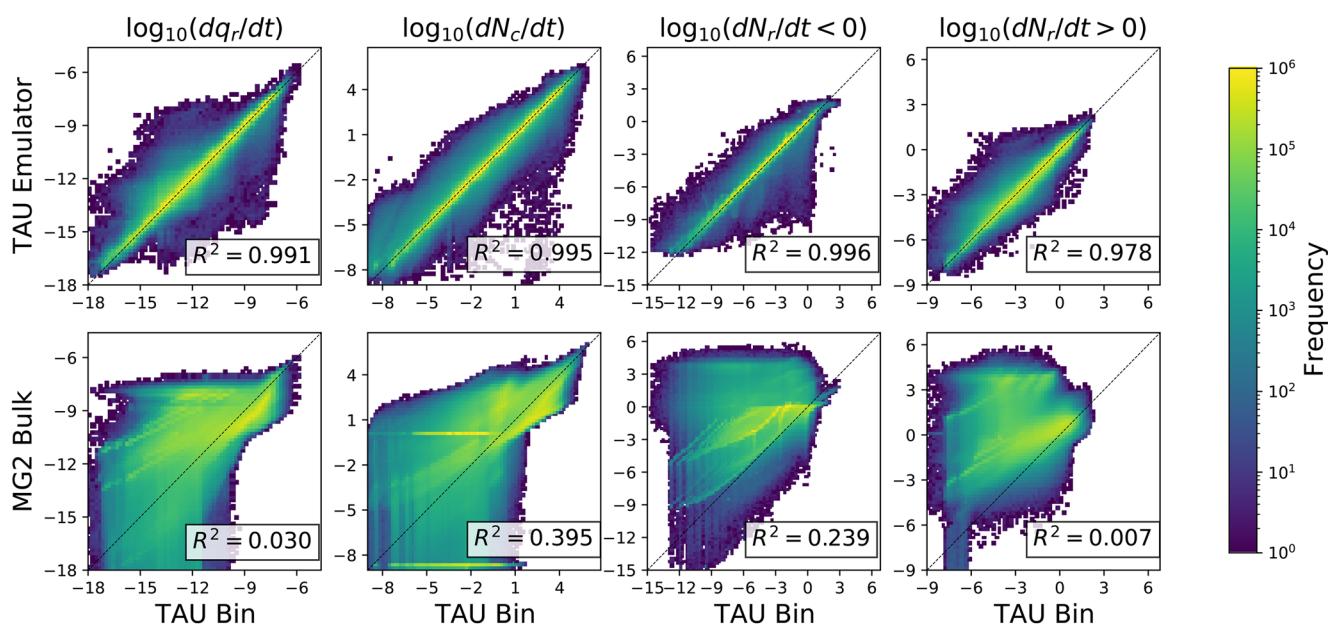


Figure 2. Frequency plots of the logarithm of TAU bin rates (horizontal) versus Neural Network Emulator (Top row) and MG2 Bulk Scheme (Bottom Row). Shown (left to right) are the rain mass tendency (dq_r/dt), condensate number tendency (dN_c/dt), negative rain number tendency ($dN_r/dt < 0$) and positive rain number tendency ($dN_r/dt > 0$). Coefficient of determination (R^2) shown on the plots. TAU, Tel Aviv University.

Table 3

TAU-ML Tendency Extremes: Fraction of Points in Testing Data Set Outside of a Specified Ratio $r = \text{TAU-ML}/\text{TAU}$

Tendency	$r < 0.1$	$r < 0.5$	$r > 2$	$r > 10$	$r < 0.1 \text{ or } r > 10$
q_r	0.04	0.49	0.43	0.03	0.01
N_c	0.03	0.82	0.12	0.01	0.01
$+N_r$	0.12	0.53	0.30	0.05	0.01
$-N_r$	0.01	0.29	0.68	0.02	0.00

Abbreviations: ML, machine learning; TAU, Tel Aviv University.

4.1. Process Rates

Figure 6 illustrates the liquid process rates in the TAU (a and d) and TAU ML (emulator) simulations (b and e), and their difference (c and f), as discussed above. In each of the two simulations, the MG2 control case autoconversion and accretion rates from Khairoutdinov and Kogan (2000) are also run on the same state to generate tendencies. These are not applied to the model evolution (prognostic tendencies) but are saved diagnostically. They represent, however, how the Khairoutdinov and Kogan (2000) autoconversion and accretion would have responded to the same model state, so are valuable for comparison. Over the S. Ocean (Figures 6a and 6b), the TAU code (and its emulator, TAU-ML) produces a larger loss of water than the KK2000 scheme (MG2, blue). The vertical structure is similar. However, over the subtropics around Barbados (Figures 6d and 6e), the Bin code and its emulator both produce much lower rates of condensate loss than MG2, and only in the upper regions of the clouds, with a peak at ~ 850 hPa, rather than closer to 900 hPa found in KK2000 (Figures 6d and 6e, difference between red and blue lines). This has implications for the overall climate simulations, as we will see below.

Figure 8 is similar to Figure 4, except it compares the TAU bin process rates for autoconversion and accretion to those produced by MG2 in the same simulation. As in Figure 6, the process rates are calculated in the same simulation on the same state, where “MG2” is KK2000 and “TAU_Bin” replaces this with the direct calculation of quasistochastic collection. The figure uses monthly means, but instantaneous fields yield the same results. Consistent with Figure 6, in the extra-tropics (such as the S. Ocean), the TAU Bin process rates are more negative (larger loss) than MG2, while in the subtropics, the rates are typically less

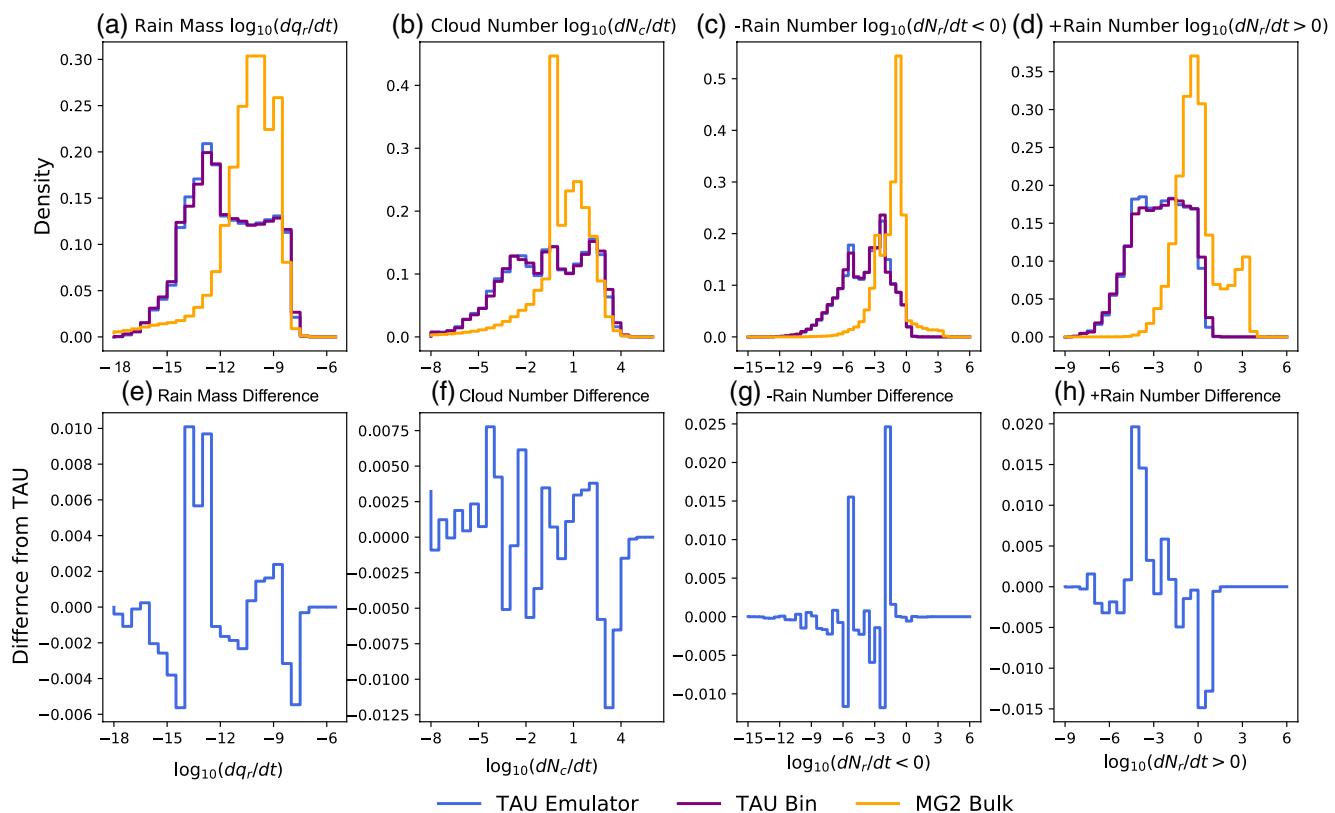


Figure 3. Probability distributions of the logarithm of process rates. (a) Rain mass tendency (dq_r/dt), (b) cloud condensate number tendency (dN_c/dt), (c) negative rain number tendency ($dN_r/dt < 0$), and (d) positive rain number tendency ($dN_r/dt > 0$). The TAU bin distribution is shown in purple, the emulator (TAU-ML) solid blue and the MG2 bulk autoconversion + accretion tendencies in orange. The bottom panels (e–h) show the difference of the ML emulator from TAU on the same horizontal scale. ML, machine learning; TAU, Tel Aviv University.

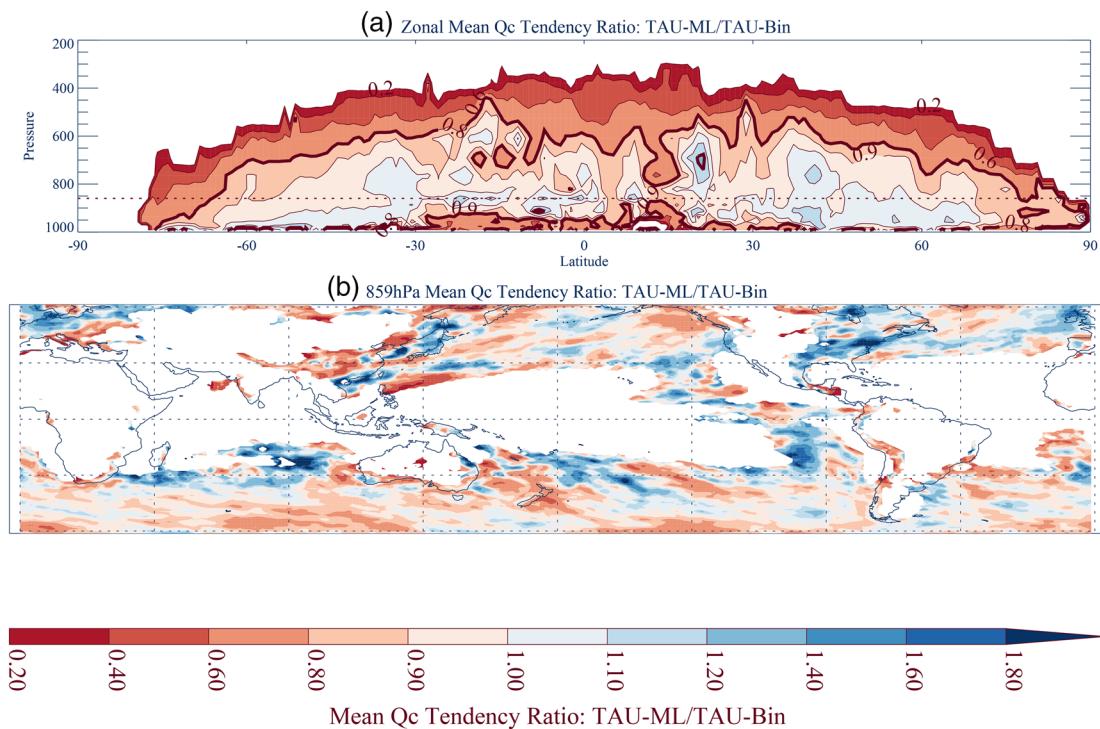


Figure 4. Ratio of q_c tendency (dq_c/dt) between the TAU-ML code and TAU bin code it is designed to reproduce. (a) Zonal mean latitude-height, (b) horizontal map from 60S-60 N at the 859 hPa level near the top of the planetary boundary layer. Blank regions have tendencies less than $1 \times 10^{-9} \text{ kg kg}^{-1} \text{ s}^{-1}$. ML, machine learning; TAU, Tel Aviv University.

negative (smaller loss), except in regions with high water content over the N. E. and S. E. Pacific, and S. E. Atlantic. The results are consistent with Figure 3, top row, where the MG2 control case has slightly more frequent high process rates ($1 \times 10^{-9} \text{ kg kg}^{-1} \text{ s}^{-1}$), and fewer moderate rates ($1 \times 10^{-12} \text{ kg kg}^{-1} \text{ s}^{-1}$) than the bin scheme. Note that the TAU bin code has significantly less change in rain number than the MG2 KK2000 autoconversion and accretion for either the negative (Figure 3, bottom row second from left) or the positive (Figure 3, bottom row, right) cases.

We have also analyzed the onset of precipitation by looking at the average rain rate as a function of drop effective radius and Liquid Water Path (LWP). Rosenfeld et al. (2012), Figure 1, illustrate for a LES that significant rain rates are rarely seen for an effective radius (Re) < 15 microns, a result they note is also found in observations. This is only for one set of cases, but Figure 9a illustrates that with the KK2000 scheme in MG2 there are significant rain rates for high LWP but small effective radius. The TAU Bin (Figure 9b) and emulated TAU Bin (Figure 9c) simulations do not show this behavior: they have much lower rain rates for high LWP but $Re < 15$ microns, in better agreement with observations and LES.

We have also examined the diurnal cycle of precipitation in the different simulations. The TAU bin and emulator code show no significant changes from the base code in the diurnal cycle of precipitation. There are small changes over ocean, but they are not significant. Over land there are no changes, likely because over land the diurnal cycle is mostly dominated by the deep convection parameterization which is not directly affected by the TAU bin code.

We do see changes in the intensity of precipitation. Figure 10 illustrates the frequency of occurrence of different large-scale (nonconvective) rain rates in the simulations, based on 3-hourly large-scale precipitation averages over a full 2 years. The plot includes warm rain over the whole planet (all regions and regimes). The shaded region represents one standard deviation of precipitation values over a month (most months are similar globally) in each intensity bin, based on 3-hourly samples. Shown are the control case with KK2000 (blue), the TAU bin code (orange), the emulated code (green), and the emulator without the mass fixer (red). The histogram of counts has been normalized into frequency (integral of 1). In general the bin code

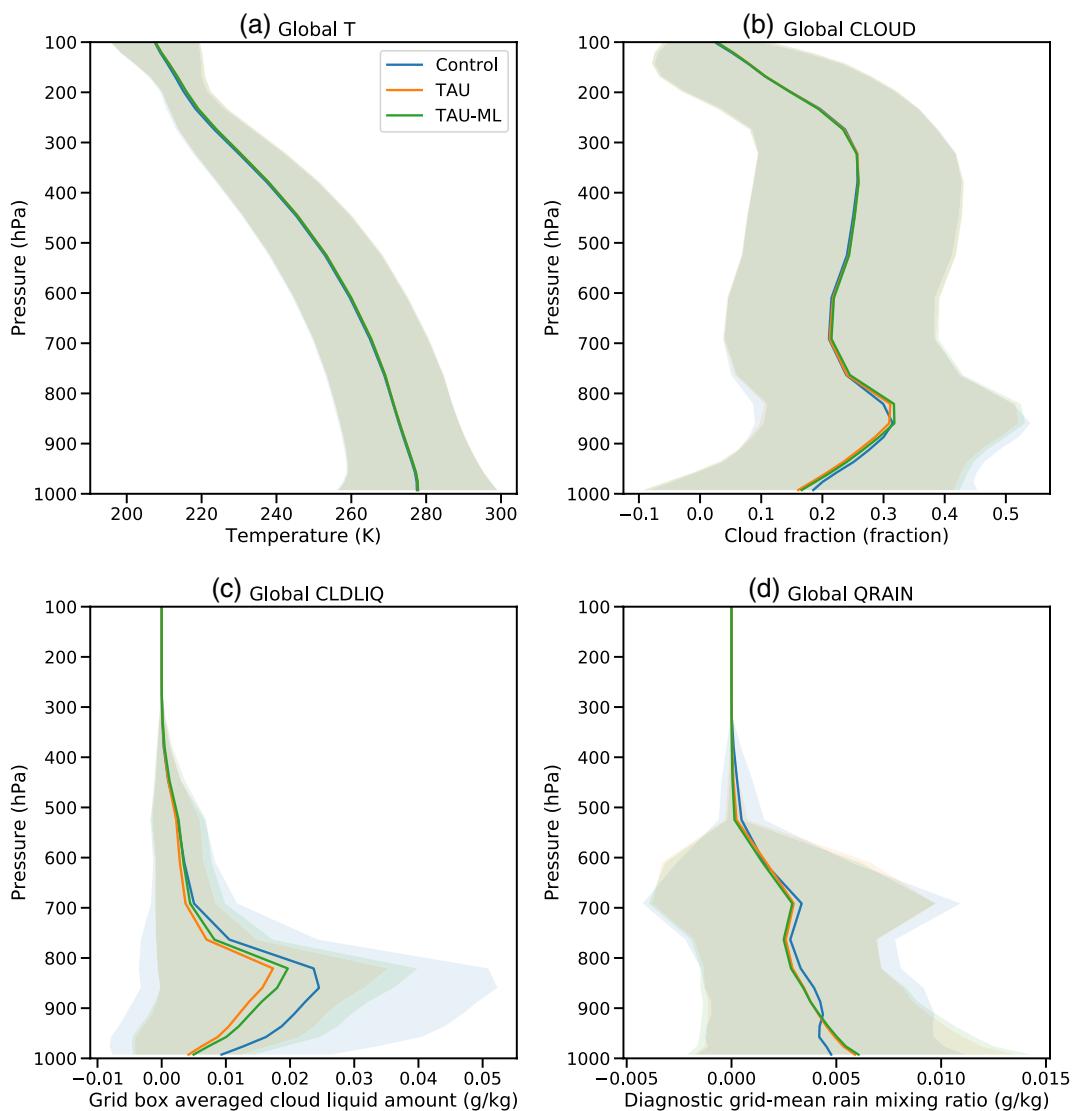


Figure 5. Annual mean global profiles of (a) Temperature, (b) Cloud fraction (CLOUD), (c) Cloud liquid (CLDLIQ) mixing ratio (g/kg), and (d) Rain (QRAIN) mixing ratio (g/kg) from control (blue), TAU (orange) and TAU-ML (green). Shading shows standard deviation of the mean profiles. ML, machine learning; TAU, Tel Aviv University.

produces higher large-scale rain rates at large values (>200 mm/day) than the control case with KK2000. The emulator code produces a lower frequency of occurrence of these values, but only when the mass fixer is applied. Without the fixer there are significant anomalies in the ML code, and the emulator produces a small frequency of very high precipitation values.

These results hold across regimes, and across perturbed climates. We examined individual regions with different regimes. The general trend in different regions and regimes is similar to the global picture in Figure 10: there are more frequent high precipitation values without the mass fixer, and the TAU-ML code with the mass fixer produces slightly lower frequency of the highest precipitation than the nonemulated TAU code. We have also examined precipitation intensity in perturbed climate runs, specifically the SST4K simulations. All simulations produce higher frequency of high precipitation in a warmer climate, as expected from the increased specific humidity. Similar to present day climate in Figure 10, the TAU and TAU-ML emulator simulations have higher frequency of large precipitation intensity than the control. The TAU-ML simulation has slightly lower frequency at high precipitation rates than the TAU simulation, similar to present day, but within the spread of variability of the TAU simulation.

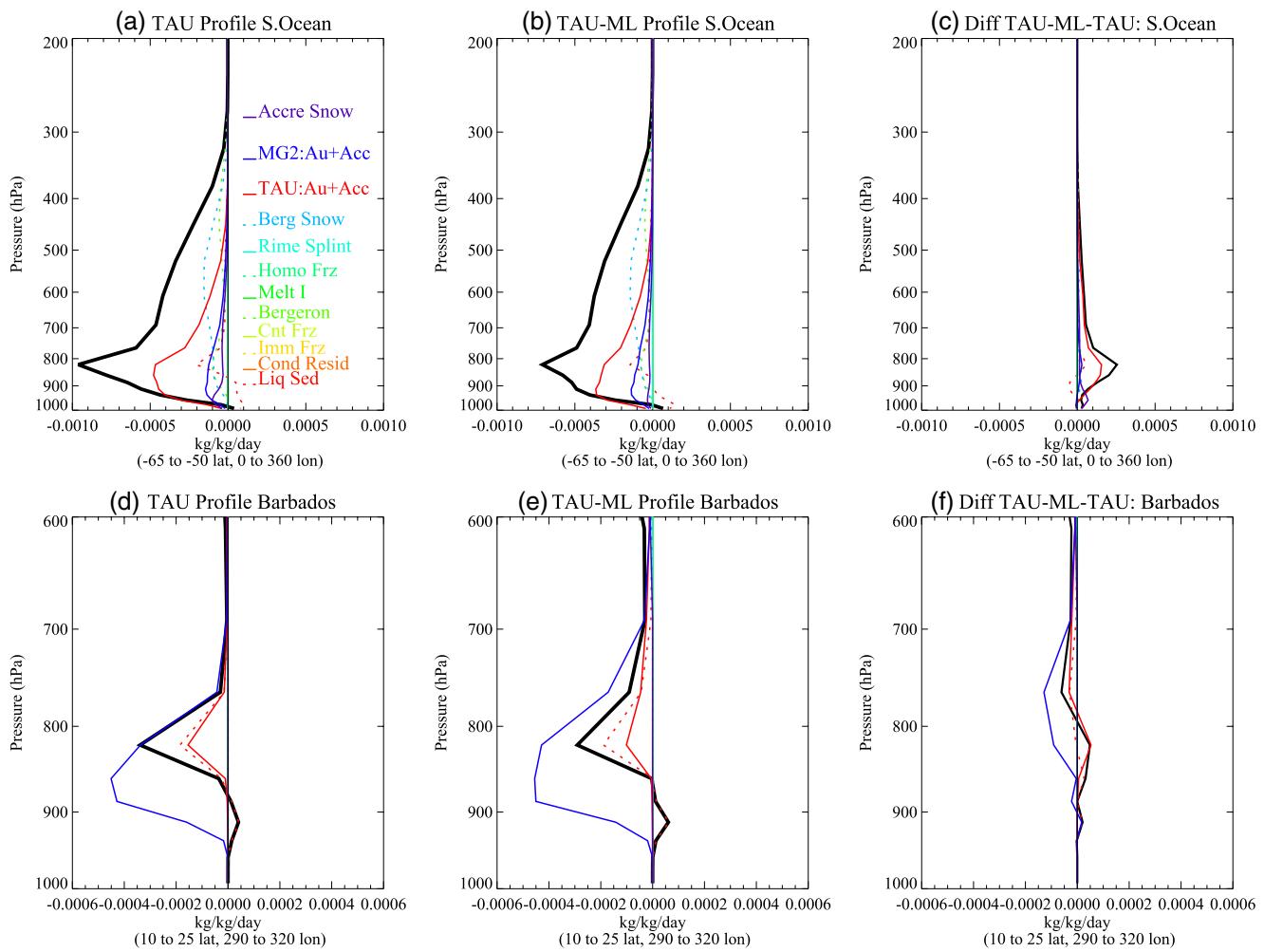


Figure 6. Mean process rates in the S. Ocean (65°S – 50°S , 0 – 360°E , Top: (a–c)) and the ocean region around Barbados (10°S – 25°N , 290 – 320°E , Bottom: (d–f)) regions as defined in the text. Process rates from the TAU bin code (a, d Left), Emulated code (TAU-ML b, e center) and the difference (TAU-ML minus TAU c, f right). Total of all rates is the thick black line. ML, machine learning; TAU, Tel Aviv University.

Finally, we examine the frequency of occurrence of surface precipitation in Figure 11, and compare this to data from CloudSat. CloudSat-retrieved precipitation is averaged over 1° regions at monthly intervals based on the 2C-Rain (Lebsack & L'Ecuyer, 2011) and 2C-Snow Profile (Wood et al., 2014) data sets. CloudSat is upscaled by aggregating profiles along the orbit at a native resolution of 1.75–100 km to match the model resolution. If at least one of the profiles has a precipitation rate greater than a threshold (we have looked at 0.01, 0.05, and 0.3 mm/h) the whole upscaled bin is considered precipitating. This is then aggregated to $1^{\circ} \times 1^{\circ}$ regions. Both day and nighttime retrievals are combined. The method is identical to Stephens et al. (2010). Instantaneous precipitation output is binned to the 100 km Cloudsat Resolution. The intensity threshold for grid box averaged precipitation is reduced by a factor of 100 (model)/1.75 km (CloudSat). This results in 0.01 mm/h in CloudSat corresponding to 0.004 mm/d (0.01 mm/h \times 1/58 \times 24 h/d). We have compared 0.01, 0.05, and 0.3 mm/h from cloudsat (0.004, 0.02, and 0.12 mm/d from the model). All thresholds have the same characteristics, with slightly different magnitudes (lower thresholds have higher frequency).

CAM has too frequent total precipitation over the ocean when compared to observations from CloudSat (Figure 11a). This is common with many other models, and with earlier versions of CAM (Stephens et al., 2010). In the control simulation, the average frequency is 0.8 for total precipitation (Figure 11a), and only slightly less for large-scale precipitation, (Figure 11b), which includes shallow convection in CAM6. This implies little additional frequency for convective precipitation, peaking at perhaps 0.2. The TAU code and the TAU-ML emulator significantly reduce the frequency of large-scale precipitation in the tropics

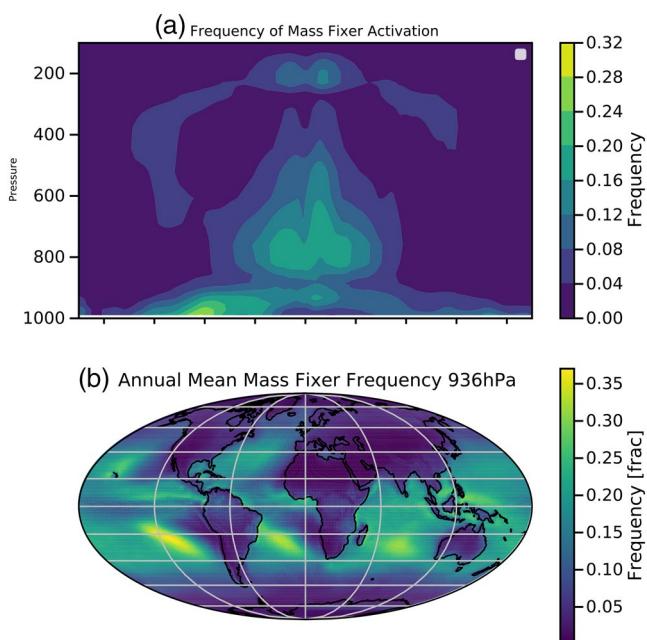


Figure 7. (a) Zonal mean annual frequency of occurrence of the mass fixer in the TAU-ML code control case. (b) Horizontal map of annual mean frequency of occurrence of the mass fixer in the TAU-ML code at 936 hPa. ML, machine learning; TAU, Tel Aviv University.

and subtropics between 45°S and 45°N, reducing it from nearly 0.8 to 0.2 in good agreement with CloudSat. Convective precipitation frequency is then larger (0.4). Note that CloudSat data includes convective and large-scale precipitation, so we expect the observed frequency to be higher than the large-scale frequency alone. The mass fixer (ML-NoFixer) does not change these results. There is still too frequent large-scale precipitation in the storm tracks in all simulations. By this metric, differences with CloudSat frequency are similar across different thresholds, differing only in slight shifts in frequency (higher frequency for lighter thresholds, and vice versa).

4.2. Mean Climate

Next we turn to analysis of the mean climate in the simulations, and compare the TAU-Bin code and the emulator with each other and the control model with KK2000. Note that the control model with KK2000 does include SGS variability corrections, which the TAU and TAU-ML simulations do not. Figure 12 provides an overview of the mean state climate, focusing on clouds and radiation in the simulations. We have added, where available, observations from the NASA Clouds in the Earth Radiant Energy System (CERES) Energy Balance Adjusted Flux (EBAF) product (Loeb et al., 2018), version 4.1. We focus on our two hypotheses in the questions above. First we explore whether the emulator (TAU-ML) produces the same mean climate as the TAU Bin code it is trying to emulate, and second how the TAU Bin and/or emulator climate differs from the control climate with KK2000.

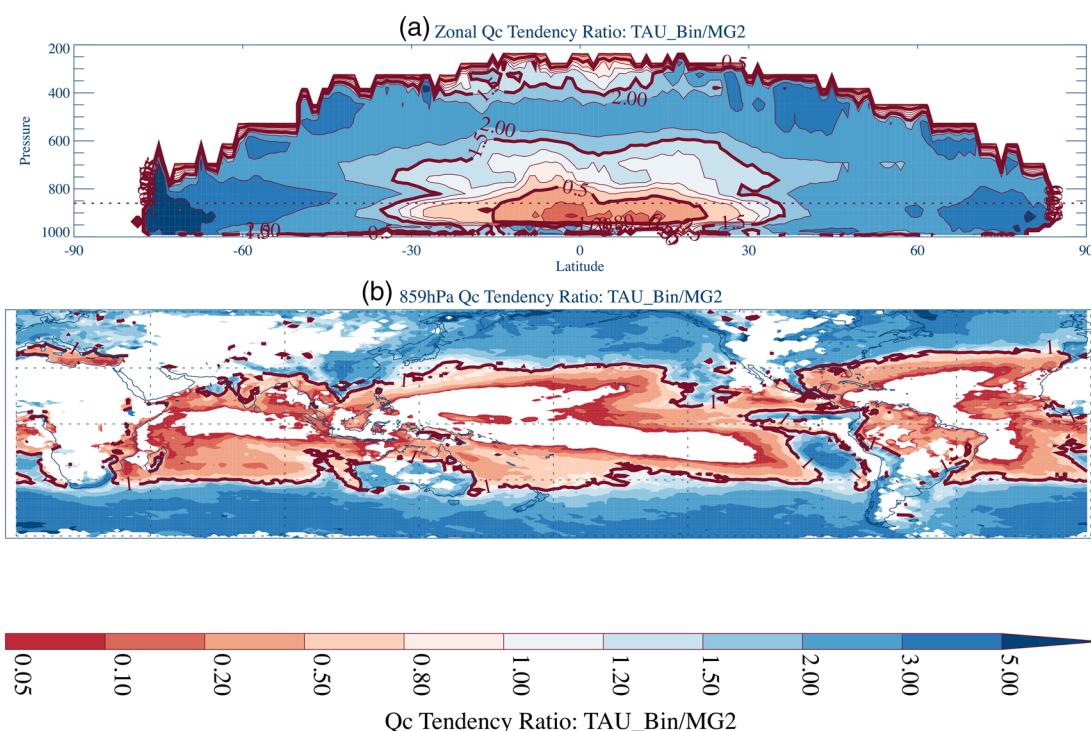


Figure 8. Ratio of Q_c tendency (dQ_c/dt) between the TAU bin code and MG2 code it is designed to reproduce. (a) Zonal mean latitude-height, (b) horizontal map from 60°S–60°N at the 859 hPa level. TAU, Tel Aviv University.

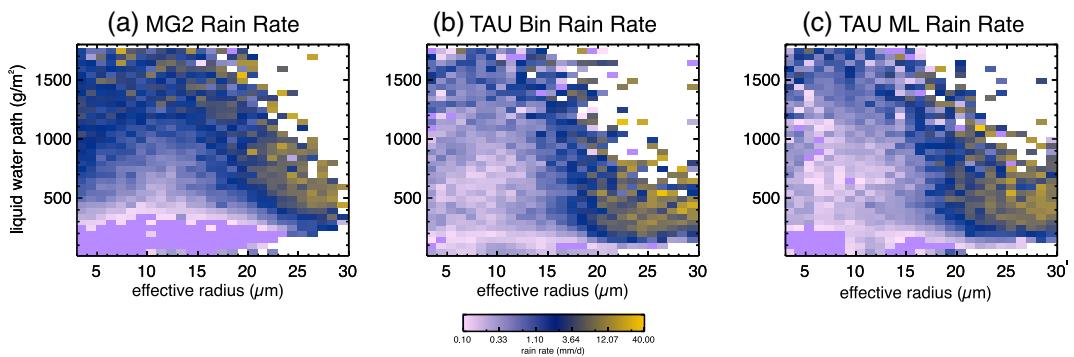


Figure 9. Contoured frequency distributions of rain rate plotted against effective radius (microns) and liquid water path (g/m^2) for MG2 control (a), TAU bin (b), and TAU-ML emulator simulations (c). ML, machine learning; TAU, Tel Aviv University.

The emulator is trained on instantaneous output from 2 years of data of the TAU Bin code. To evaluate the emulator, we run a further 7 years of the TAU Bin code, and compare this to 9 years with the emulator. Results indicate that the emulator has 10%–15% more LWP than the TAU Bin code at most latitudes outside the tropics (Figure 12a). This is not due to any vertical shifting of clouds, but likely a feedback resulting from slight differences in the mean process rates (Figure 6). The higher LWP is associated with slightly larger drop sizes (Figure 12d), but the same number concentration (Figure 12e) and cloud fraction (Figure 12c). Thus the change is solely in the mass of liquid, not its number. The increased LWP then results in 10% higher cloud optical depth (Figure 12f) in the storm tracks, and a corresponding small difference in SW Cloud Radiative Effect (CRE, Figure 12g). There is no change in Ice Water Path (Figure 12b) or in the LW CRE (Figure 12h). This is consistent with Figure 4 which illustrates that most of the atmosphere has a ratio of TAU-ML emulator to TAU Bin code q_c tendencies of slightly less than 1, resulting in less loss of water and more remaining cloud liquid. It is also seen in the vertical structure of process rates over the S. Ocean in Figure 6c, showing less q_c tendency.

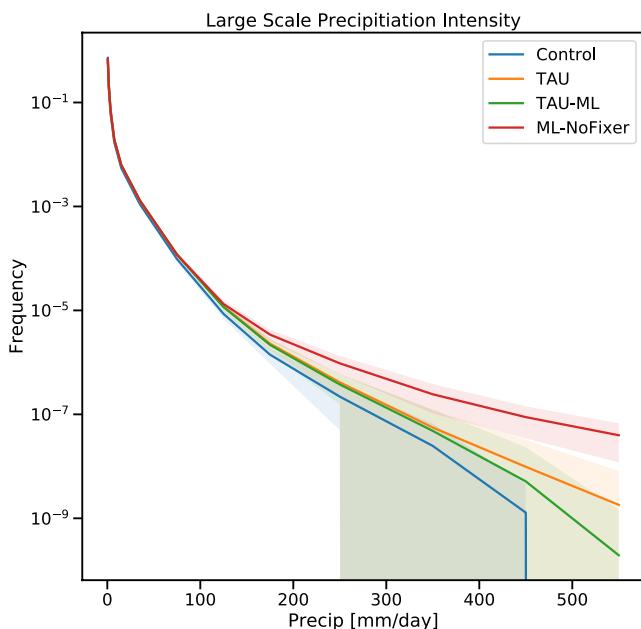


Figure 10. Frequency of occurrence of different rain rates (mm/day) in the simulations, based on three hourly precipitation averages for warm rain anywhere on the planet (global). The shaded region represents one standard deviation of the monthly frequency (typically 240 time samples) from each simulation. Shown are the control case with KK2000 (blue), the TAU bin code (orange), the TAU-ML emulated code (green), and the emulator without the mass fixer (ML-NoFixer, red). ML, machine learning; TAU, Tel Aviv University.

Figure 12 also compares the emulator code with a simulation of the emulator run without the mass fixer for present day climate. These simulations are not identical, but Figure 12 indicates that their mean climates are very similar: adding the mass fixer does not change climate. There do not appear to be appreciable differences in the S. Hemisphere subtropical regions where the mass fixer is most active. Only a few differences appear in cloud optical depth (Figure 12f) at high latitudes, likely from cases with low liquid water. Next we compare the TAU Bin code (and TAU-ML emulator) results to the control climate with KK2000 autoconversion and accretion. The code with KK2000 (Control) has much higher LWP in the storm track latitudes of 30°–60°N and S (Figure 12a) with the same cloud fraction in the storm tracks (Figure 12c). Note that this is not due to any SGS enhancement in process rates, as will be discussed below. In the subtropical S. Hemisphere (30°–15°S), there is similar or less LWP and lower cloud fraction in the control case than the TAU cases, and similarly lower cloud fraction in the control case in the N. Hemisphere subtropics. Effective radius is smaller with the TAU Bin or TAU-ML code (Figure 12d), while number concentration is not substantially different between any of the cases (Figure 12e).

We have added for reference (where available) observations from the NASA CERES EBAF product (Loeb et al., 2018), which indicate that the reduction in LWP may be too large (though this is uncertain from satellite observations), while the changes to subtropical cloud fraction may be an

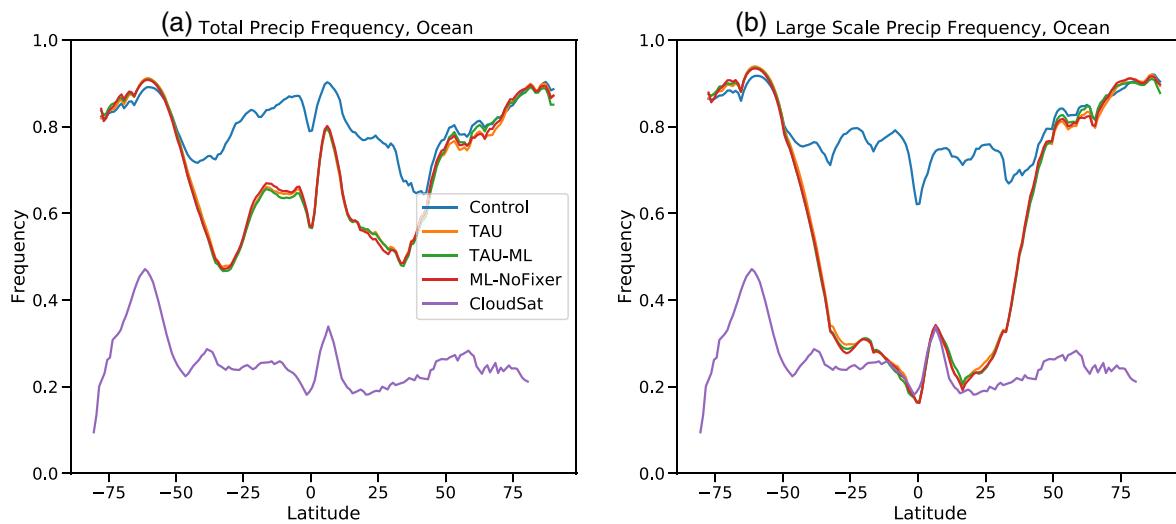


Figure 11. Annual mean frequency of occurrence of (a) total (large scale + deep convection) and (b) large-scale (stratiform) precipitation greater than 0.05 mm/h for CloudSat, and a model gridbox average of 0.02 mm/d. Shown are the control case with KK2000 (blue), the TAU bin code (orange), the TAU-ML emulated code (green), the emulator without the mass fixer (ML-NoFixer, red) and CloudSat observations (purple). CloudSat precipitation is obtained from 2C-Rain-Profile and 2C-Snow-Profile products as described in the text and frequency calculations follow Stephens et al. (2010). ML, machine learning; TAU, Tel Aviv University.

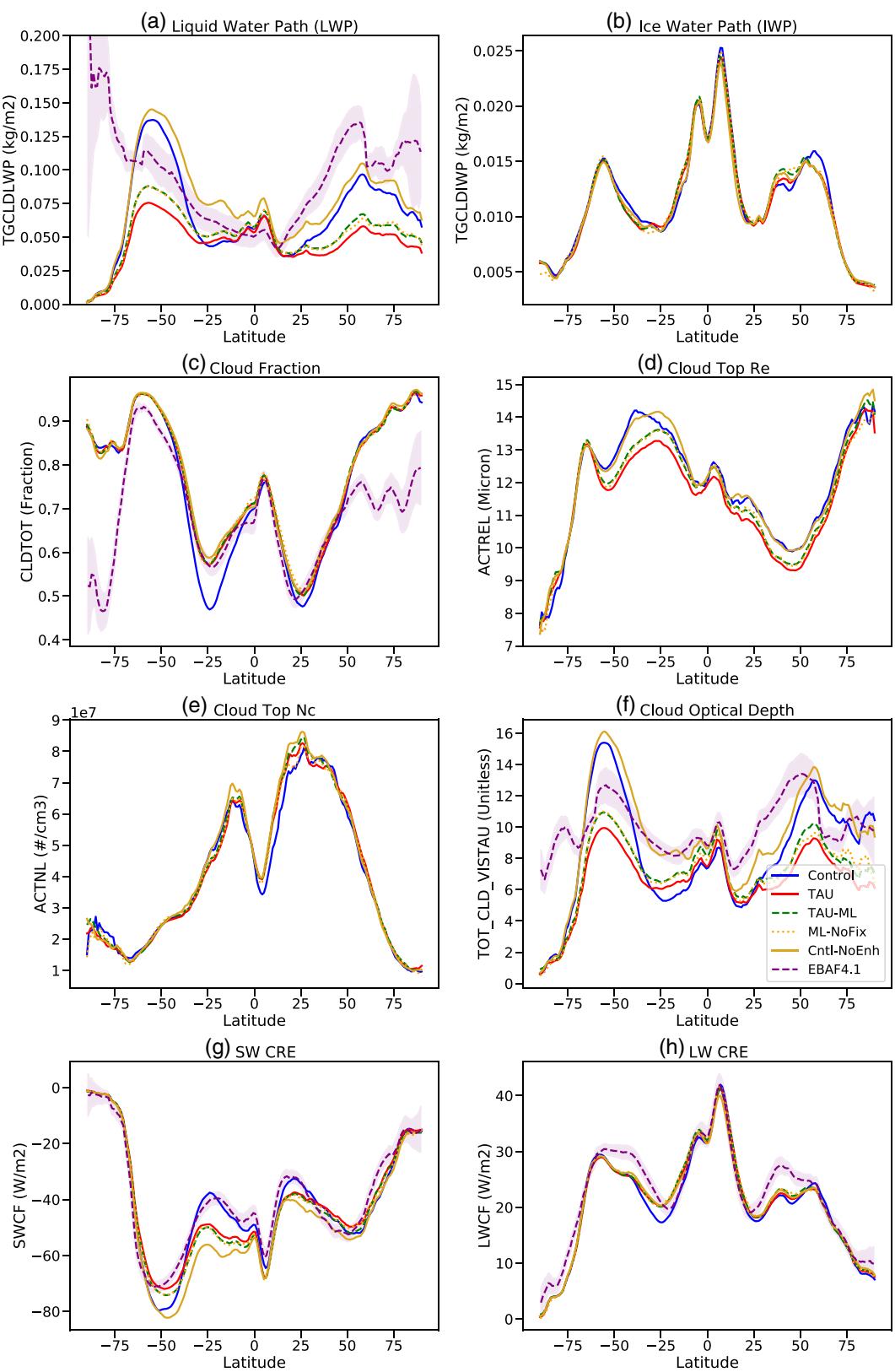
improvement. The impact of these microphysical changes is a reduction in cloud optical depth in the storm tracks, and increases in cloud optical depth in the subtropics with the Bin code. This is an improvement in the subtropics, and creates similar biases of opposite sign to the control code in the extratropics (Figure 12f). The overall radiative impact is mostly on SW CRE (Figure 12g), with improvements at high latitudes, and a degradation with too much cloud forcing in the subtropics compared to observations.

It is interesting that increased optical depth (but still lower than CERES) and increased cloud fraction (similar to CERES) at low latitudes yield CRE that is too strong (more negative) in the TAU BIN and TAU-ML emulator simulations. This likely results from higher cloud frequency of occurrence (not just area fraction) in these regions which is evident in the higher frequency of occurrence of precipitation seen in Figure 11, and seems due perhaps to convective precipitation and convective water. Note that we do not necessarily expect a “better climate” with the TAU code than the control model, since we have made major changes to the microphysics scheme and not made an effort to work to compensating biases in other parts of the cloud microphysics, macrophysics, or the turbulence scheme.

The differences between control and TAU simulations are not due to neglecting SGS variability in autoconversion and accretion. The Cntl-NoEnh simulation without these effects has increased LWP from the Control CAM6 (Figure 12a), indicating that including SGS variability (increasing the loss rate with an enhancement factor) DECREASES LWP as expected. If applied to the TAU or TAU-ML code, this would decrease the LWP further from the control. Reducing the loss process and increasing LWP increases cloud fraction in the Cntl-NoEnh simulation (Figure 12c), with resulting higher optical depth (Figure 12f) and larger magnitude SW CRE (Figure 12g) and LW CRE (Figure 12h). So Inclusion of SGS variability in the TAU warm rain process rates as an enhancement might be expected to reduce the magnitude of the cloud radiative effect, potentially bringing the TAU and TAU-ML simulations closer to observations for the SW cloud radiative effects (Figure 12g). Thus inclusion of SGS adjustments to the process rates might be a strategy for matching observations with the TAU and TAU-ML warm rain process rates.

4.3. Emergent Properties

The warm rain formation process is critical for the mean state of clouds. It may be also critical for the response of clouds to perturbations. A very important global response is the response of cloud to changes in aerosols that nucleate cloud drops, called ACI. ACI result when changes in aerosols affect CCN and hence cloud drop number. This results in significant radiative perturbations and adjustments of cloud



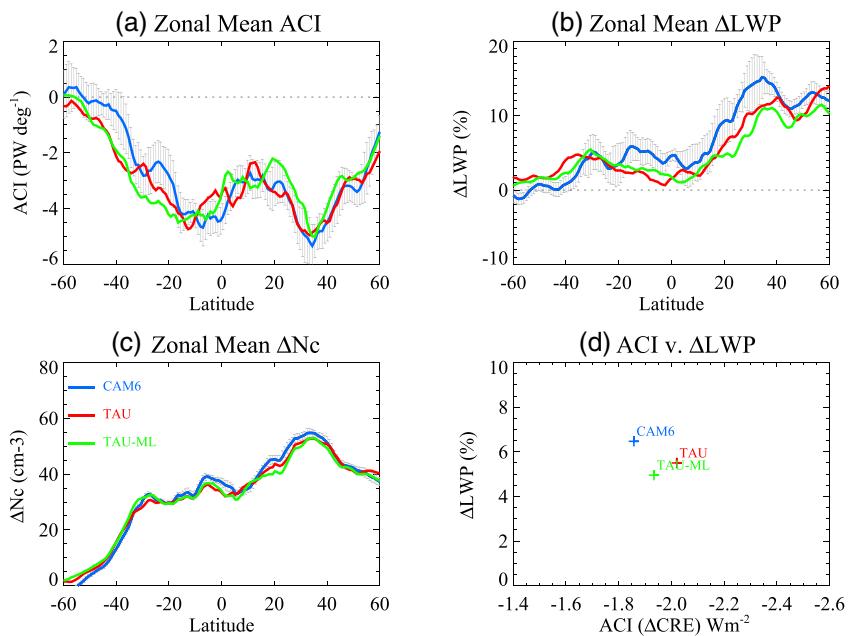


Figure 13. (a) Zonal mean Aerosol Cloud Interactions (ACI) in PW deg^{-1} , (b) zonal mean percent change in LWP, (c) zonal mean change in droplet number concentration (N_c), and (d) global-average ACI (Wm^{-2}) v. percent change in LWP. Simulations shown: CAM6 control (blue solid), TAU bin code (red solid), and TAU-ML emulator (green solid). Error bars indicate one standard deviation of the annual means at each latitude. CAM6, Community Atmosphere Model version 6; ML, machine learning; TAU, Tel Aviv University.

microphysical processes; see Bellouin et al. (2020) for a review. ACI are the largest uncertainty in historical and present anthropogenic climate forcing (Boucher et al., 2013). Second, we look at cloud feedbacks, the response of cloud radiative effects to surface temperature changes (Gettelman & Sherwood, 2016; Stephens, 2005), which are the largest uncertainty in understanding the sensitivity of climate to forcing (Boucher et al., 2013).

Figure 13 illustrates the magnitude of ACI from the different simulations. The unit of peta-watts ($1 \text{ PW} = 10^{15} \text{ W}$) per degree of latitude in Figure 13 is essentially area weighted around a latitude circle (dividing by $\text{m}^2 \text{ deg}^{-1}$ yields Wm^{-2}). ACI are calculated by running another simulation with forcing identical to the baseline cases discussed above, but with aerosol emissions set to 1850 values. All other forcings remain at 2000 levels. The differences of the two simulation climates are the ACI. The total ACI defined as the change in net cloud radiative effect for the Control (CAM6) KK2000, TAU bin, and emulated (TAU-ML) simulations are -1.85 , -1.95 , and -2.0 Wm^{-2} , respectively. Differences result from slightly higher ACI in the TAU and TAU-ML code in the S. Hemisphere (Figure 13a). The change in LWP (Figure 13b) is slightly lower in the N. Hemisphere in the TAU Bin and emulator code, while changes in drop number concentration are nearly identical (Figure 13c). On the whole, these differences are not significantly different from each other, given the variance of annual radiation differences by latitude. The shaded region is one standard deviation of annual means at each latitude overlaid on the CAM6 control simulation.

Finally, we examine cloud feedbacks. Feedbacks are estimated as the radiative kernel adjusted change in cloud radiative effect as defined by Soden et al. (2008), with kernels from Shell et al. (2008) used as in Gettelman and Sherwood (2016). For this analysis, we use a uniform +4 K perturbation in the sea surface

Figure 12. Zonal mean climatologies from Control (blue solid), Control without SGS enhancement (Cntl-NoEnh, yellow solid), TAU bin (red solid), TAU-ML (green dash), the emulator without the mass fixer (ML-NoFix, orange dotted) and CERES EBAF4.1 satellite observations (purple dash). (a) Liquid Water Path (LWP), (b) Ice Water Path (IWP), (c) Cloud fraction, (d) Cloud Top Effective Radius (Re), (e) Cloud Top Droplet number concentration (N_c), (f) Cloud optical Depth, (g) Shortwave Cloud Radiative Effect (SW CRE), and (h) Longwave Cloud Radiative Effect (LW CRE). Shading shows ± 1 standard deviation of monthly anomalies for CERES data. ML, machine learning; TAU, Tel Aviv University.

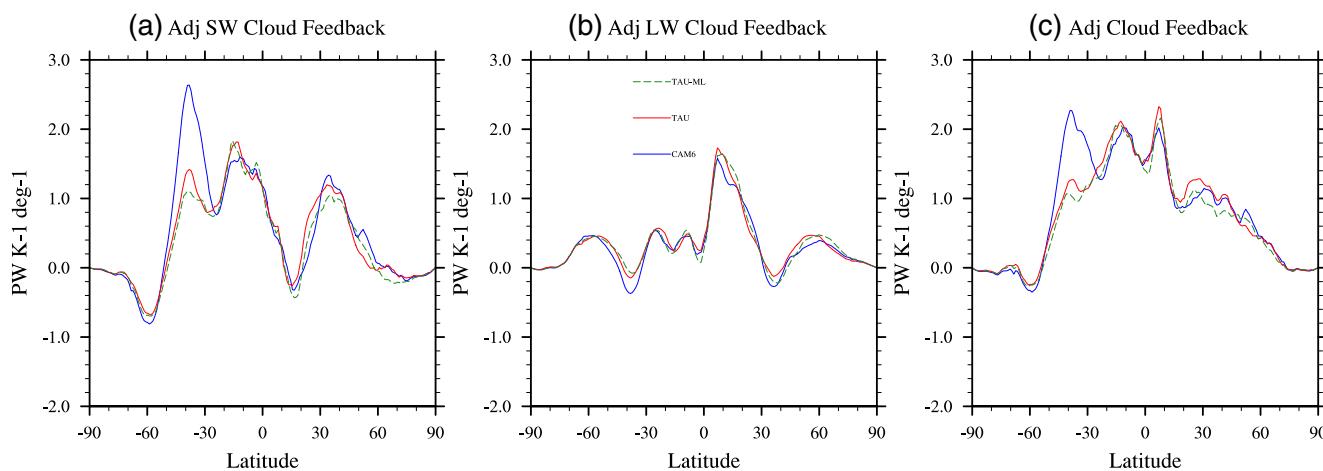


Figure 14. Zonal mean kernel adjusted cloud feedbacks. (a) SW, (b) LW, and (c) Net. Simulations: CAM6 (blue solid), TAU bin (red solid), and TAU ML emulator (Green Dash). CAM6, Community Atmosphere Model version 6; ML, machine learning; TAU, Tel Aviv University.

temperature from the control case, a standard metric used in many studies following Cess (1987). Figure 14 illustrates the results for (a) SW, (b) LW, and (c) net cloud feedback. Feedbacks are comparable, except over the S. Ocean where the control CAM6 case with KK2000 has much higher positive SW feedbacks, resulting in a significantly higher net cloud feedback. The globally integrated zonal mean net cloud feedback is $0.81 \text{ Wm}^{-2}\text{K}^{-1}$ for CAM6, $0.77 \text{ Wm}^{-2}\text{K}^{-1}$ for TAU Bin, and $0.70 \text{ Wm}^{-2}\text{K}^{-1}$ for TAU ML.

The bin code and emulator have significantly lower SW cloud feedbacks over the S. Ocean. This region was identified by Gettelman et al. (2019) as a region where cloud feedbacks increased significantly in CAM6 due to better representation of cloud phase as supercooled liquid clouds. This removed a negative cloud phase feedback (Tan et al., 2016) and increased cloud feedback and climate sensitivity. The feedback in the emulator code is reduced because the ice fraction extends lower in the atmosphere over the S. Ocean. The lower liquid water path (Figure 12a) with the same ice water path (Figure 12b) increases the ice fraction and results in reduced positive cloud feedback. Note that the overall radiative effect in the S. Ocean is improved in the TAU code simulations (Figure 12g), indicating this change might reduce biases and improve realism. Comparisons between CAM6 and in situ aircraft observations over the S. Ocean by Gettelman et al. (2020) indicate that CAM6 has too little ice, indicating that the increased ice fraction may be more realistic.

5. Discussion

Here we discuss the mechanics of the emulator, the mass fixer, and some of the issues related to SGS variability.

The neural network emulator is able to capture the TAU results, with the use of three different networks and significant regularization, as well as a need to capture additional input variables for the network *not* included in the physical routine. This indicates that the network requires some linearization and minimization of nonlinearities by separating the networks with a classifier. It also indicates that the network requires additional information, likely because with the current model input and output structure, the neural networks can only learn part of the physics. There are 11 inputs and 3 outputs and we use 5 neural networks with ~ 60 thousand weights. Such a high capacity can solve complex tasks, but they may overfit (Goodfellow et al., 2016). This is a common problem with neural networks which is currently an active area of research in the literature. It may be that extended training data sets are needed, or likely that different methods should be used to incorporate physical processes in the loss function following (Beucler et al., 2020). However, our networks are quite successful at running in real time within a full Earth System Model framework.

Without these advances, it is not surprising that a mass fixer is needed for the emulator. CAM6 uses double precision values, and the microphysics has lots of cases where potentially large values of number and small values of mass are present. This is due to the conceptual design of the approach: we treat the emulator as

another potentially error prone parameterization operating in a process split framework (with multiple processes seeing the same state), and limit it accordingly with mass and energy constraints. Many other microphysical process rates have similar constraints. The fixer is invoked mostly in regions of very small mass, which is also not surprising. However, the necessity of the fixer for perturbed climates is interesting. It is occurring perhaps because the subtle state of different regimes is shifting. Because the emulator trained here includes additional inputs that are not input to the actual routine (like pressure), it may be that the emulator is overfit to the process, and shifting climates are resulting in extrapolation. This indicates that the emulator is not capturing the TAU model physics well. Recent work has provided some ideas for paths forward to reduce or eliminate the need for such external constraints and to improve the ability to “capture” the physical constraints internally. Brenowitz and Bretherton (2019) suggest that potentially limiting the emulator to certain vertical levels to reduce the emulator locking onto correlations may be appropriate. Han et al. (2020) suggest methods to start including energy conservation in the loss function. These ideas should be incorporated in future studies with GCMs.

We note that in this first proof of concept treatment of ML for the warm rain process, we have neglected the SGS variability that is treated in the existing bulk scheme. The different performance of the MG and TAU schemes is not due to the SGS variability, as removal of the variability results in a small increase in LWP, while the TAU scheme without added SGS variability has much less water, implying that adding SGS variability would reduce LWP further. In principle the ML-derived process rates could be combined with the SGS representation of cloud water variability (assuming a gamma function of cloud water mixing ratio). However, the ML-derived rates do not lend themselves to analytic integration over these gamma functions unlike the KK2000 process rates. Numerical integration could be done, but this is far too computationally costly to be practical. Another possible approach would be to fit power law rates to the ML-derived rates and then use the exponent parameter in the fit to derive the SGS enhancement factor as is done currently with the KK2000 rates. However, this also seems cumbersome. Finally, ML could itself be used to develop an enhancement factor consistent with the gamma SGS distribution of cloud water mixing ratio within grid cells. Again, such efforts are beyond the scope of this “proof of concept” study.

6. Conclusions

Returning to our hypotheses, first, the TAU-ML emulator code is able to reproduce almost all the metrics of the TAU bin formulation and recover almost all the computational cost penalty. There are some differences in mean climate, mostly in liquid water path. The differences between TAU-ML and TAU result from the emulator underestimating the mean process rates (Figures 4 and 6), so the LWP is slightly higher in the emulator code. A mass fixer was found to be needed to get the code to run stably for perturbed climates. The fixer is active in subtropical regions with low mean liquid water path. This does not appear to impact any of the climate results or emergent properties like cloud feedbacks or aerosol-cloud interactions. However, the emulator does have excessive precipitation intensity if the mass fixer is not applied. We note that mass fixers are currently applied to other process rates in the code as well, since the microphysical process rates are process split (they operate on the same state and are then combined). It is further hypothesized (but left for future work) that training on a combined data set including perturbed climate simulations might reduce or eliminate the need for the fixer. Thus the emulator may be a useful option for simulating process rates, with the noted caveats above.

Second, the use of the TAU bin quasistochastic collection process results in a different climate simulation from the base code with KK2000. The overall climate shows more significant biases than the well tuned and adjusted code with KK2000, but there are several interesting and important features. First, the onset of precipitation is improved significantly, and mostly occurs only when mean drop size is large, in agreement with bin microphysics in LES models and observations (Rosenfeld et al., 2012). Second, the frequency of occurrence of large-scale precipitation in the subtropics and tropics is significantly lower with the TAU code or TAU-ML emulator of the code than the control case (Figure 11). This is a substantial improvement and reduces a longstanding and common model bias. It is likely coupled to the precipitation onset being limited to large mean drop size. Third, the bin code modifies the intensity of precipitation for extreme but low-frequency events. The emulator is able to reproduce all of these differences with the control simulation.

Finally, the mean state in the subtropics degrades (with respect to radiative effects) but it improves over the S. Ocean. This is due mostly to improvements with lower LWP, resulting in a different balance of ice and liquid in S. Ocean cloud systems.

There are two important conclusions related to emergent properties. First, the change in S. Ocean clouds results in a significant drop in cloud feedback strength, which would significantly impact the climate sensitivity. CAM6 has been seen to have little ice when compared in detail to in-situ observations of S. Ocean supercooled liquid clouds (Gettelman et al., 2020), so this may be an improvement. Further analysis here would be very useful and critical for understanding cloud feedbacks.

Second, the cloud radiative effects of aerosols (ACI) are virtually unchanged in the simulations, despite very different warm rain process rates and a very different representation of the interaction of drop number with rain formation. Recent work has focused on the importance of rain formation and autoconversion and accretion as governing ACI (Bellouin et al., 2020; Gettelman, 2015). However, this work shows little sensitivity of ACI to the formulation of autoconversion and accretion. CAM6 still sees a large LWP response (cloud adjustment) to increased aerosols, despite a very different dependence of the warm rain process on drop number. This argues that the onset of precipitation with drop number and the frequency of precipitation (where differences are seen) are not important for ACI. It may also be that the cloud adjustments to aerosols might be damped by other buffering processes (Stevens & Feingold, 2009), such as interactions with turbulence or entrainment. Recently, Karset et al. (2020) linked drop number to turbulent entrainment in CAM5 and did not find significant sensitivity of ACI to parameterized turbulent entrainment. Further investigation of the interaction of turbulence and cloud macrophysics with aerosols is warranted, as cloud adjustments to aerosols and ACI in CAM6 do not seem sensitive to the microphysical representation of the warm rain process. This result should be tested in other modeling systems.

Finally, it is important to note there is uncertainty in the bin microphysics calculations. Different bin schemes will do different things, depending on the specified collection efficiencies, numerics, bin resolution, etc. This needs to be kept in mind when thinking of bin schemes as a “benchmark.” Moreover, an increasing number of studies employ Lagrangian particle-based microphysics schemes, which use a much different approach to evolve drop size distributions than bin microphysics. These schemes represent cloud hydrometeors by a collection of “superdroplets” that move along Lagrangian flow trajectories, and they address some conceptual and practical limitations of bin microphysics (Grabowski et al., 2019).

In summary, ML emulators do appear to provide useful speedups with accurate representations of complex microphysical processes that can be used to provide insight into important uncertainties in climate models. Current methods do require bespoke development of emulators, and also require more inputs that would seem to be required to accurately predict and simulate model evolution. Advances in ML applied to geophysical problems which include more of the physics of the problem have the potential to change that, and will be pursued further. We are also pursuing a follow on study to examine the stability and sensitivity of the emulators in more detail.

Data Availability Statement

The emulator and nonbase CAM code described here is archived at <https://github.com/NCAR/mlmicrophysics>. It is also available through <https://doi.org/10.5281/zenodo.4110959>.

Acknowledgments

The National Center for Atmospheric Research is sponsored by the United States National Science Foundation. Simulation output is available on the Earth System Grid. We also thank G. Feingold and collaborators for developing and maintaining the TAU microphysics code, which can be obtained at <https://www.esrl.noaa.gov/csl/staff/graham.feingold/code/>. We thank the two anonymous reviewers for their comments to improve the manuscript.

References

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). *Tensorflow: A system for large-scale machine learning* (pp. 265–283). Proceedings of 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16). Retrieved from <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf>
- Albrecht, B. A. (1989). Aerosols, cloud microphysics and fractional cloudiness. *Science*, 245, 1227–1230.
- Bellouin, N., Quaas, J., Gryspenert, E., Kinne, S., Stier, P., Watson-Parris, D., et al. (2020). Bounding global aerosol radiative forcing of climate change. *Reviews of Geophysics*, 58(1), e2019RG000660. <https://doi.org/10.1029/2019RG000660>
- Beucler, T., Pritchard, M., Gentine, P., & Rasp, S. (2020). Towards physically-consistent, data-driven models of convection. In Accepted for oral presentation at the 2020 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). <https://arxiv.org/abs/2002.08525>

- Bodas-Salcedo, A., Mulcahy, J. P., Andrews, T., Williams, K. D., Ringer, M. A., Field, P. R., & Elsaesser, G. S. (2019). Strong dependence of atmospheric feedbacks on mixed-phase microphysics and aerosol-cloud interactions in HadGEM3. *Journal of Advances in Modeling Earth Systems*, 11(6), 1735–1758. <https://doi.org/10.1029/2019MS001688>
- Bogenschutz, P. A., Gettelman, A., Morrison, H., Larson, V. E., Craig, C., & Schanen, D. P. (2013). Higher-order turbulence closure and its impact on Climate Simulation in the Community Atmosphere Model. *Journal of Climate*, 26(23), 9655–9676. <https://doi.org/10.1175/JCLI-D-13-00075.1>
- Boucher, O., Randall, D., Artaxo, P., Bretherton, C., Feingold, G., Forster, P., et al. (2013). Clouds and aerosols. In T. F. Stocker, D. Qin, G.-K. Plattner, M. Tignor, S. K. Allen, J. Boschung, et al. (Eds.), *Climate change 2013: The physical science basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change* (pp. 571–657). Cambridge University Press.
- Brenowitz, N. D., & Bretherton, C. S. (2018). Prognostic validation of a neural network unified physics parameterization. *Geophysical Research Letters*, 45(12), 6289–6298. <https://doi.org/10.1029/2018GL078510>
- Brenowitz, N. D., & Bretherton, C. S. (2019). Spatially extended tests of a neural network parametrization trained by coarse-graining. *Journal of Advances in Modeling Earth Systems*, 11(8), 2728–2744. <https://doi.org/10.1029/2019MS001711>
- Cess, R. D. (1987). Exploratory studies of cloud radiative forcing with a general circulation model. *Tellus*, 39A, 460–473.
- Chevallier, F., Morcrette, J.-J., Chéruy, F., & Scott, N. A. (2000). Use of a neural-network-based long-wave radiative-transfer scheme in the ECMWF atmospheric model. *Quarterly Journal of the Royal Meteorological Society*, 126(563), 761–776. <https://doi.org/10.1002/qj.49712656318>
- Danabasoglu, G., Lamarque, J.-F., Bacmeister, J., Bailey, D. A., DuVivier, A. K., Edwards, J., et al. (2020). The Community Earth System Model version 2 (CESM2). *Journal of Advances in Modeling Earth Systems*, 12(2), e2019MS001916. <https://doi.org/10.1029/2019MS001916>
- Feingold, G., Tzivion, S., & Leviv, Z. (1988). Evolution of raindrop spectra. Part I: Solution to the stochastic collection/breakup equation using the method of moments. *Journal of the Atmospheric Sciences*, 45(22), 3387–3399. [https://doi.org/10.1175/1520-0469\(1988\)045<3387:EORSP>2.0.CO;2](https://doi.org/10.1175/1520-0469(1988)045<3387:EORSP>2.0.CO;2)
- Forbes, R. M., & Ahlgrimm, M. (2014). On the representation of high-latitude boundary layer mixed-phase cloud in the ECMWF global model. *Monthly Weather Review*, 142(9), 3425–3445. <https://doi.org/10.1175/MWR-D-13-00325.1>
- Gentine, P., Pritchard, M., Rasp, S., Reinaudi, G., & Yacalis, G. (2018). Could machine learning break the convection parameterization deadlock? *Geophysical Research Letters*, 45(11), 5742–5751. <https://doi.org/10.1029/2018GL078202>
- Gettelman, A. (2015). Putting the clouds back in aerosol–cloud interactions. *Atmospheric Chemistry and Physics*, 15(21), 12397–12411. <https://doi.org/10.5194/acp-15-12397-2015>
- Gettelman, A., Bardeen, C. G., Mccluskey, C. S., Järvinen, E., Stith, J., Bretherton, C., et al. (2020). Simulating observations of Southern Ocean clouds and implications for climate. *Journal of Geophysical Research: Atmospheres*, 125(21), e2020JD032619. <https://doi.org/10.1029/2020JD032619>
- Gettelman, A., Hannay, C., Bacmeister, J. T., Neale, R. B., Pendergrass, A. G., Danabasoglu, G., et al. (2019). High climate sensitivity in the Community Earth System Model version 2 (CESM2). *Geophysical Research Letters*, 46(14), 8329–8337. <https://doi.org/10.1029/2019GL083978>
- Gettelman, A., & Morrison, H. (2015). Advanced two-moment bulk microphysics for global models. Part I: Off-line tests and comparison with other schemes. *Journal of Climate*, 28(3), 1268–1287. <https://doi.org/10.1175/JCLI-D-14-00102.1>
- Gettelman, A., Morrison, H., Santos, S., Bogenschutz, P., & Caldwell, P. M. (2015). Advanced two-moment bulk microphysics for global models. Part II: Global model solutions and aerosol–cloud interactions. *Journal of Climate*, 28(3), 1288–1307. <https://doi.org/10.1175/JCLI-D-14-00103.1>
- Gettelman, A., & Sherwood, S. C. (2016). Processes responsible for cloud feedback. *Current Climate Change Reports*, 2, 1–11. <https://doi.org/10.1007/s40641-016-0052-8>
- Golaz, J.-C., Larson, V. E., & Cotton, W. R. (2002). A PDF-based model for boundary layer clouds. Part II: Model results. *Journal of the Atmospheric Sciences*, 59, 3552–3571.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*: MIT Press.
- Grabowski, W. W., Morrison, H., Shima, S.-I., Abade, G. C., Dziekan, P., & Pawlowska, H. (2019). Modeling of cloud microphysics: Can we do better? *Bulletin of the American Meteorological Society*, 100(4), 655–672. <https://doi.org/10.1175/BAMS-D-18-0005.1>
- Han, Y., Zhang, G. J., Huang, X., & Wang, Y. (2020). A moist physics parameterization based on deep learning. *Journal of Advances in Modeling Earth Systems*, 12(9), e2020MS002076. <https://doi.org/10.1029/2020MS002076>
- Harrington, J. Y., Feingold, G., & Cotton, W. R. (2000). Radiative impacts on the growth of a population of drops within simulated summertime arctic stratus. *Journal of the Atmospheric Sciences*, 57(5), 766–785. [https://doi.org/10.1175/1520-0469\(2000\)057<0766:RIOTGO>2.0.CO;2](https://doi.org/10.1175/1520-0469(2000)057<0766:RIOTGO>2.0.CO;2)
- Karset, I. H. H., Gettelman, A., Storelvmo, T., Alterskjær, K., & Berntsen, T. K. (2020). Exploring impacts of size-dependent evaporation and entrainment in a global model. *Journal of Geophysical Research: Atmospheres*, 125(4), e2019JD031817. <https://doi.org/10.1029/2019JD031817>
- Kessler, E. (1969). On the distribution and continuity of water substance in atmospheric circulations. In E. Kessler (Ed.), *On the distribution and continuity of water substance in atmospheric circulations* (pp. 1–84). Boston, MA: American Meteorological Society. https://doi.org/10.1007/978-1-935704-36-2_1
- Khairoutdinov, M. F., & Kogan, Y. (2000). A new cloud physics parameterization in a large-eddy simulation model of marine stratocumulus. *Monthly Weather Review*, 128, 229–243.
- Kingma, D. P., & Ba, J. (2015). *Adam: A method for stochastic optimization*. Proceedings of the 3rd international conference for learning representations. San Diego, CA. Retrieved from <https://arxiv.org/abs/1412.6980>
- Kogan, Y. (2013). A cumulus cloud microphysics parameterization for cloud-resolving models. *Journal of the Atmospheric Sciences*, 70(5), 1423–1436. <https://doi.org/10.1175/JAS-D-12-0183.1>
- Kostinski, A. B., & Shaw, R. A. (2005). Fluctuations and luck in droplet growth by coalescence. *Bulletin of the American Meteorological Society*, 86(2), 235–244. <https://doi.org/10.1175/BAMS-86-2-235>
- Krasnopol'sky, V. M., Fox-Rabinovitz, M. S., & Chalikov, D. V. (2005). New approach to calculation of atmospheric model physics: Accurate and fast neural network emulation of longwave radiation in a climate model. *Monthly Weather Review*, 133(5), 1370–1383. <https://doi.org/10.1175/MWR2923.1>
- Larson, V. E., Golaz, J.-C., & Cotton, W. R. (2002). Small-scale and mesoscale variability in cloudy boundary layers: Joint probability density functions. *Journal of the Atmospheric Sciences*, 59(24), 3519–3539. [https://doi.org/10.1175/1520-0469\(2002\)059<3519:SSAMVI>2.0.CO;2](https://doi.org/10.1175/1520-0469(2002)059<3519:SSAMVI>2.0.CO;2)
- Lebo, Z. J., & Seinfeld, J. H. (2011). Theoretical basis for convective invigoration due to increased aerosol concentration. *Atmospheric Chemistry and Physics*, 11(11), 5407–5429. <https://doi.org/10.5194/acp-11-5407-2011>

- Lebsack, M. D., & L'Ecuyer, T. S. (2011). The retrieval of warm rain from CloudSat. *Journal of Geophysical Research*, 116, D20209. <https://doi.org/10.1029/2011JD016076>
- Liu, Y., & Daum, P. H. (2004). Parameterization of the Autoconversion process. Part I: Analytical formulation of the Kessler-type parameterizations. *Journal of the Atmospheric Sciences*, 61(13), 1539–1548. [https://doi.org/10.1175/1520-0469\(2004\)061\(1539:POTAPI\)2.0.CO;2](https://doi.org/10.1175/1520-0469(2004)061(1539:POTAPI)2.0.CO;2)
- Loeb, N. G., Doelling, D. R., Wang, H., Su, W., Nguyen, C., Corbett, J. G., et al. (2018). Clouds and the Earth's Radiant Energy System (CERES) Energy Balanced and Filled (EBAF) Top-of-Atmosphere (TOA) edition-4.0 data product. *Journal of Climate*, 31(2), 895–918. <https://doi.org/10.1175/JCLI-D-17-0208.1>
- Michibata, T., & Takemura, T. (2015). Evaluation of autoconversion schemes in a single model framework with satellite observations. *Journal of Geophysical Research: Atmospheres*, 120(18), 9570–9590. <https://doi.org/10.1002/2015JD023818>
- Morrison, H., Curry, J. A., Sjude, M. D., & Zuidema, P. (2005). A new double-moment microphysics parameterization for application in cloud and climate models. Part II: Single column modeling of arctic clouds. *Journal of the Atmospheric Sciences*, 62, 1678–1693.
- Morrison, H., & Gettelman, A. (2008). A new two-moment bulk stratiform cloud microphysics scheme in the NCAR Community Atmosphere Model (CAM3), Part I: Description and Numerical Tests. *Journal of Climate*, 21(15), 3642–3659.
- Nair, V., & Hinton, G. E. (2010). *Rectified linear units improve restricted Boltzmann machines* (pp. 807–814). Proceedings of the 27th international conference on international conference on machine learning, Omnipress, Madison, WI.
- Neale, R. B., Richter, J. H., & Jochum, M. (2008). The impact of convection on ENSO: From a delayed oscillator to a series of events. *Journal of Climate*, 21, 5904. <https://doi.org/10.1175/2008JCLI2244.1>
- Pinkus, A. (1999). Approximation theory of the MLP model in neural networks. *Acta Numerica*, 8, 143–195. <https://doi.org/10.1017/S0962492900002919>
- Pruppacher, H. R., & Klett, J. D. (1997). *Microphysics of clouds and precipitation* (2nd ed.): Kluwer Academic.
- Rasp, S., Pritchard, M. S., & Gentine, P. (2018). Deep learning to represent subgrid processes in climate models. *Proceedings of the National Academy of Sciences of the United States of America*, 115(39), 9684–9689. <https://doi.org/10.1073/pnas.1810286115>
- Reisin, T., Levin, Z., & Tzivion, S. (1996). Rain production in convective clouds as simulated in an axisymmetric model with detailed microphysics. Part I: Description of the model. *Journal of the Atmospheric Sciences*, 53(3), 497–519. [https://doi.org/10.1175/1520-0469\(1996\)053\(0497:RPICCA\)2.0.CO;2](https://doi.org/10.1175/1520-0469(1996)053(0497:RPICCA)2.0.CO;2)
- Rosenfeld, D., Wang, H., & Rasch, P. J. (2012). The roles of cloud drop effective radius and LWP in determining rain properties in marine stratocumulus. *Geophysical Research Letters*, 39, L13801. <https://doi.org/10.1029/2012GL052028>
- Seifert, A., & Beheng, K. D. (2001). A double-moment parameterization for simulating autoconversion, accretion and selfcollection. *Atmospheric Research*, 59–60, 265–281.
- Shell, K. M., Kiehl, J. T., & Shields, C. A. (2008). Using the radiative Kernel technique to calculate climate feedbacks in NCAR's community atmosphere model. *Journal of Climate*, 21, 2269–2282. <https://doi.org/10.1175/2007JCLI2044.1>
- Soden, B. J., Held, I. M., Colman, R., Shell, K. M., Kiehl, J. T., & Shields, C. A. (2008). Quantifying climate feedbacks using radiative Kernels. *Journal of Climate*, 21(14), 3504–3520. <https://doi.org/10.1175/2007JCLI2110.1>
- Stephens, G. L. (2005). Cloud feedbacks in the climate system: A critical review. *Journal of Climate*, 18(2), 237–273.
- Stephens, G. L., L'Ecuyer, T., Forbes, R., Gettelman, A., Golaz, J.-C., Bodas-Salcedo, A., et al. (2010). Dreary state of precipitation in global models. *Journal of Geophysical Research*, 115, D24211. <https://doi.org/10.1029/2010JD014532>
- Stevens, B., & Feingold, G. (2009). Untangling aerosol effects on clouds and precipitation in a buffered system. *Nature*, 461(7264), 607–613. <https://doi.org/10.1038/nature08281>
- Stevens, B., Feingold, G., Cotton, W. R., & Walko, R. L. (1996). Elements of the microphysical structure of numerically simulated nonprecipitating stratocumulus. *Journal of the Atmospheric Sciences*, 53(7), 980–1006. [https://doi.org/10.1175/1520-0469\(1996\)053\(0980:EOMS\)2.0.CO;2](https://doi.org/10.1175/1520-0469(1996)053(0980:EOMS)2.0.CO;2)
- Tan, I., Storelvmo, T., & Zelinka, M. D. (2016). Observational constraints on mixed-phase clouds imply higher climate sensitivity. *Science*, 352(6282), 224–227. <https://doi.org/10.1126/science.aad5300>
- Tikhonov, A. N. (1963). Solution of incorrectly formulated problems and the regularization method. *Soviet Mathematics*, 4, 1035–1038.
- Twomey, S. (1977). The influence of pollution on the shortwave albedo of clouds. *Journal of the Atmospheric Sciences*, 34(7), 1149–1152.
- Tzivion, S., Feingold, G., & Levin, Z. (1987). An efficient numerical solution to the stochastic collection equation. *Journal of the Atmospheric Sciences*, 44(21), 3139–3149. [https://doi.org/10.1175/1520-0469\(1987\)044\(3139:AENSTT\)2.0.CO;2](https://doi.org/10.1175/1520-0469(1987)044(3139:AENSTT)2.0.CO;2)
- Tzivion, S., Feingold, G., & Levin, Z. (1989). The evolution of raindrop spectra. Part II: Collisional collection/breakup and evaporation in a rainshaft. *Journal of the Atmospheric Sciences*, 46(21), 3312–3328. [https://doi.org/10.1175/1520-0469\(1989\)046\(3312:TEORSP\)2.0.CO;2](https://doi.org/10.1175/1520-0469(1989)046(3312:TEORSP)2.0.CO;2)
- Tzivion, S., Reisin, T. G., & Levin, Z. (1999). A numerical solution of the kinetic collection equation using high spectral grid resolution: A proposed reference. *Journal of Computational Physics*, 148(2), 527–544. <https://doi.org/10.1006/jcph.1998.6128>
- Wilkinson, M. (2016). Large deviation analysis of rapid onset of rain showers. *Physical Review Letters*, 116(1), 018501. <https://doi.org/10.1103/PhysRevLett.116.018501>
- Wood, N. B., L'Ecuyer, T. S., Heymsfield, A. J., Stephens, G. L., Hudak, D. R., & Rodriguez, P. (2014). Estimating snow microphysical properties using collocated multisensor observations. *Journal of Geophysical Research: Atmospheres*, 119(14), 8941–8961. <https://doi.org/10.1002/2013JD021303>
- Yin, Y., Levin, Z., Reisin, T. G., & Tzivion, S. (2000). The effects of giant cloud condensation nuclei on the development of precipitation in convective clouds—A numerical study. *Atmospheric Research*, 53(1), 91–116. [https://doi.org/10.1016/S0169-8099\(99\)00046-0](https://doi.org/10.1016/S0169-8099(99)00046-0)
- Yuval, J., & O'Gorman, P. A. (2020). Stable machine-learning parameterization of subgrid processes for climate modeling at a range of resolutions. *Nature Communications*, 11(1), 3295. <https://doi.org/10.1038/s41467-020-17142-3>
- Zhang, G. J., & McFarlane, N. A. (1995). Sensitivity of climate simulations to the parameterization of cumulus convection in the Canadian Climate Center general circulation model. *Atmosphere-Ocean*, 33, 407–446.