atmosphere

MDPI

Article

Stochastic Parameterization of Moist Physics Using Probabilistic Diffusion Model

Leyi Wang [1,2,*], Yiming Wang [3], Xiaoyu Hu [1,2], Hui Wang [1,2] and Ruilin Zhou [2]

[1] School of Mathematical Sciences, Peking University, Beijing 100871, China; huxy2023@pku.edu.cn (X.H.); hui.wang@pku.edu.cn (H.W.)
[2] Chongqing Research Institute of Big Data, Peking University, Chongqing 400031, China; ruilin.zhou@cqbdri.pku.edu.cn
[3] 2035 Future Laboratory, PIESAT Information Technology Co., Ltd., Beijing 100195, China; wangyiming_sz@piesat.cn
* Correspondence: lywang2237@pku.edu.cn

Abstract: Deep-learning-based convection schemes have garnered significant attention for their notable improvements in simulating precipitation distribution and tropical convection in Earth system models. However, these schemes struggle to capture the stochastic nature of moist physics, which can degrade the simulation of large-scale circulations, climate means, and variability. To address this issue, a stochastic parameterization scheme called DIFF-MP, based on a probabilistic diffusion model, is developed. Cloud-resolving data are coarse-grained into resolved-scale variables and subgrid contributions, which serve as conditional inputs and outputs for DIFF-MP. The performance of DIFF-MP is compared with that of generative adversarial networks and variational autoencoders. The results demonstrate that DIFF-MP consistently outperforms these models in terms of prediction error, coverage ratio, and spread–skill correlation. Furthermore, the standard deviation, skewness, and kurtosis of the subgrid contributions generated by DIFF-MP more closely match the test data than those produced by the other models. Interpretability experiments confirm that DIFF-MP's parameterization of moist physics is physically consistent.

Keywords: convection parameterization; diffusion model; generative model; machine learning

check for updates

Citation: Wang, L.; Wang, Y.; Hu, X.; Wang, H.; Zhou, R. Stochastic Parameterization of Moist Physics Using Probabilistic Diffusion Model. *Atmosphere* **2024**, *15*, 1219. https://doi.org/ 10.3390/atmos15101219

Academic Editors: Erick G. Sperandio Nascimento, Taciana Toledo De Almeida Albuquerque and Prashant Kumar

Received: 16 September 2024
Revised: 5 October 2024
Accepted: 8 October 2024
Published: 12 October 2024

1. Introduction

Convection plays a crucial role in atmospheric circulation, transferring heat from the Earth's surface to the upper atmosphere, creating a vertical air movement to mix air at different altitudes, driving cloud formation and extreme precipitation, and shaping the global mass, momentum, and energy budget. Meanwhile, current numerical models fail to explicitly resolve convection processes due to the resolution limits posed by computational constraints. Instead, they rely on convection parameterization schemes to estimate the influence of convection on resolved variables.

Traditional convection parameterization schemes, coming with error-prone empirical function forms and free parameters, may introduce significant errors when estimating heating, moistening, and precipitation rates at the grid scale [1,2], which translates into errors in continental precipitations and mesoscale convective systems [3,4]. Traditional schemes are also unable to accurately simulate the self-aggregation of convection [5,6]. They fail to capture the interactions between deep and shallow convection, as well as the interactions between convection, clouds, and large-scale atmospheric motions [2,4,5,7]—despite the critical role that these processes play in assessing climate sensitivity [5].

There is a growing interest in using high-fidelity data, including high-resolution simulations and observations, to calibrate existing convection schemes or develop data-driven simulators using deep neural networks. The latter provides more flexibility and a higher accuracy, promising a breaker to the deadlock of the convection parameterization problem [8,9]. Deep neural networks have been tested thoroughly in numerical models under aquaplanet or realistic geography setups, showing their potential to be the next-generation convection scheme [10–20]. Gentine et al. [9] and Rasp et al. [16] were among the first to explore deep-learning-based convection parameterization schemes as a replacement for super-parameterization in global climate models. Their findings showed that numerical models using deep learning schemes can simulate the Madden–Julian Oscillation and Kelvin waves with a reasonable amplitude and speed. Yuval and O'Gorman [19] applied a random forest model to parameterize subgrid contributions diagnosed from cloud-resolving data. Once integrated into the numerical model, their approach successfully reproduced the zonally averaged precipitation distribution, particularly capturing extreme precipitation events. Subsequent studies have focused on implementing deep learning convection schemes under realistic geographical conditions for real-event simulations [12,13,15,17,18].

However, the lack of stochasticity in neural-network-based convection schemes can negatively impact the performance of numerical models. Gentine et al. [9] found that the heating and moistening tendencies predicted by deep neural networks exhibited a reduced variability below 700 hPa. Similarly, Rasp et al. [16] demonstrated that, in multi-year simulations, the standard deviation of convective heating tendencies below 700 hPa was significantly lower after incorporating the same neural network into the numerical model. In contrast, traditional stochastic parameterization schemes have been shown to improve ensemble prediction skills, as well as the simulation of the Madden–Julian Oscillation, climate means, and variability [21–23]. It is anticipated that the stochastic parameterization of moist physics using neural networks could further enhance their performance in numerical models.

A generative model is a type of deep learning model grounded in probability theory, which maps a known prior distribution to a target distribution, making it well-suited for stochastic parameterization. Classic generative models include the variational autoencoder (VAE) [24] and generative adversarial network (GAN) [25]. GANs have been explored for the stochastic parameterization of convection, subgrid stress in ocean models, and stochastic tendencies in the Lorenz-96 model [26–32]. However, VAEs are hindered by issues such as blurry generated samples and posterior collapse [33,34], while GANs face challenges with training instability and mode collapse [35,36]. Although efforts have been made to stabilize GAN training and improve the diversity and quality of generated samples [33,35–38], achieving a good performance still requires significant case-specific expertise when using GANs and VAEs.

In recent years, a new family of generative models, probabilistic diffusion models (PDMs) [39], has garnered significant attention. PDMs serve as the foundation for well-known text-to-image models such as Stable Diffusion [40] and Dall-E [41]. They divide the generative process into a series of relatively simple denoising tasks, representing a paradigm shift that makes PDMs easier to train and less prone to mode collapse. [39,42,43]. PDMs have demonstrated a strong superiority in producing high-quality generative samples, further solidifying their prominence in the field [44].

In this study, we develop a stochastic parameterization scheme for moist physics based on a probabilistic diffusion model (DIFF-MP). Cloud-resolving global simulations are coarse-grained into resolved variables and subgrid contributions to form the training data for DIFF-MP. A key limitation of PDMs is their slow inference speed. To address this, we adapt the approach of Chen et al. [45], training DIFF-MP on a range of noise levels in a stochastic manner to generalize for larger denoising steps, thereby accelerating the process. Classifier-free guidance [46] is employed during inference to fine-tune the influence of conditional information by fusing the denoised latents of both conditioned and unconditioned models, further enhancing DIFF-MP's performance. DIFF-MP is then compared to GANs and VAEs on test data for the stochastic parameterization of moist physics. Finally, we explore the physical interpretability of DIFF-MP.

We find that DIFF-MP consistently outperforms the other two models across seven key criteria for the stochastic parameterization of moist physics. This study is one of the first to explore the potential of PDMs in stochastic parameterization, and it is expected to inspire further research into the broader application of PDMs in numerical model development and atmospheric sciences.

This study is organized as follows. Section 2 provides details on the cloud-resolving data and scale separation techniques used for generating the training data. It also outlines the training process for DIFF-MP, including methods for accelerating DIFF-MP and improving its performance through classifier-free guidance. Section 3 presents the performance of DIFF-MP on the test data, comparing it with baseline models, including GANs and VAEs, and discusses the interpretability of DIFF-MP. Finally, Section 4 concludes the study and offers perspectives for future research.

2. Methodology

2.1. Training Data Preprocessing

The training data are derived from high-fidelity cloud-resolving simulations using the Global–Regional Integrated Forecast System (GRIST) [47,48], which is formulated on primitive equations. The GRIST employs a structured Delaunay–Voronoi grid [49] for its horizontal mesh, making it well-suited for global simulations due to its isotropic properties. High-fidelity data from high-resolution simulations are commonly used as training data for machine-learning-based parameterization schemes, as they explicitly resolve most atmospheric motions, including convection.

The simulations span the following four periods: 1–20 October 1988; 1–20 January 1998; 1–20 April 2005; and 10–29 July 2013, covering four seasons with ENSO and MJO events of varying intensities. La Niña and El Niño were particularly strong in 1988 and 1998, respectively, while MJO events were prominent in 1988, 1998, and 2005. These periods were selected to provide diverse training data. The horizontal resolution of the simulations was 5 km, sufficient to resolve deep convection. There were 30 vertical levels below 20 km, with additional levels in the boundary layer. The initial conditions were interpolated from ECMWF Reanalysis v5 (ERA5) [50], and the boundary conditions (sea surface temperature) were updated every 24 h. The GRIST employed the Yonsei University (YSU) [51] scheme for boundary-layer parameterization, the Noah-MP land surface model for surface–atmosphere fluxes, the WRF single-moment 6-class scheme (WSM6) [52] for microphysics, and the RRTMG schemes [53] for shortwave and longwave radiation. The model outputs were saved every hour.

The high-resolution data had to be preprocessed into subgrid contributions and large-scale resolved variables to form the training dataset. Subgrid processes, including microphysics and subgrid vertical transports, are the outputs of DIFF-MP. These consist of the subgrid vertical transports of heat and water vapor, as well as the following four outputs from the WSM6 scheme: temperature tendency ($Tend_{T-mp}$), water vapor tendency ($Tend_{q_v-mp}$), cloud water (q_c), and cloud ice mixing ratios (q_i). Rain, snow, graupel mixing ratios, and subgrid vertical transports of q_c and q_i were excluded, as they are negligible. The following six variables are selected as input conditions for DIFF-MP: temperature (T), water vapor mixing ratio (q_v), surface pressure (P_s), sensible heat flux (SHF), latent heat flux (LHF), and shortwave radiation at surface ($SOLIN$). The preprocessing steps for the high-resolution data are outlined as follows.

A random set of points on the high-resolution Delaunay–Voronoi grid, which are seamlessly connected, were selected for coarse graining and subgrid diagnostics. These points are labeled as $P_1, P_2, P_3, \ldots, P_n$ and were arranged to approximate a regular hexagon or pentagon as closely as possible. The coarse-grained variable is denoted as $\overline{a}$ for any variable a in the high-resolution data.

$$\overline{a} = \frac{1}{n} \sum_{P_i \in \{P_1,\, P_2,\, P_3,\, \ldots,\, P_n\}} a_{P_i}. \tag{1}$$

Through coarse graining (Equation (1)), we can obtain averaged variables for DIFF-MP's conditional inputs and outputs, except for the subgrid contributions of T and q_v. If the difference between a and $\overline{a}$ is a', then the subgrid vertical flux of a is,

$$\overline{a'w'} = \frac{1}{n} \sum_{P_i \in \{P_1,\ P_2,\ P_3,\ \ldots,\ P_n\}} a'_{P_i} \cdot w'_{P_i}. \tag{2}$$

The tendency due to the subgrid vertical flux of a is,

$$Tend_{a-flux} = -\frac{\partial \overline{a'w'}}{\partial z}. \tag{3}$$

According to the definition, the subgrid contributions for T and q_v are,

$$Tend_{q_v-sgs} = \overline{Tend_{q_v-mp}} + Tend_{q_v-flux} = \overline{Tend_{q_v-mp}} - \frac{\partial \overline{{q_v}'w'}}{\partial z}, \tag{4}$$

$$Tend_{T-sgs} = \overline{Tend_{T-mp}} + Tend_{T-flux} = \overline{Tend_{T-mp}} - \frac{\partial \overline{T'w'}}{\partial z}. \tag{5}$$

The reason why the subgrid contributions are formulated as presented above is explained in the Supplementary Materials. The conditional input and output variables for DIFF-MP are shown in Table 1.

Table 1. The conditional input and output variables of machine learning schemes in this study. Level numbers for each variable are also presented.

Conditional Input	Level Number	Output	Level Number
Temperature	30	Subgrid tendencies for T	30
Water vapor mixing ratio	30	Subgrid tendencies for q_v	30
Surface pressure	1	Cloud water mixing ratio	30
Sensible heat flux	1	Cloud ice mixing ratio	30
Latent heat flux	1		
Shortwave radiation at surface	1		

This study considers the following four resolutions: 120 km, 60 km, 30 km, and 15 km, corresponding to 576, 144, 36, and 9 points, respectively, as defined in Equations (1) and (2). The number of training samples was the same for all four resolutions. The conditional input for DIFF-MP consisted of four surface-layer variables, each of which was vertically duplicated 30 times to align with the vertical profiles of T and q_v, forming the input matrix. T and P_s were normalized by subtracting the 0.05 quantile and dividing by the difference between the 0.95 quantile ($a_{0.95}$) and 0.05 quantile ($a_{0.05}$),

$$a_{norm} = \frac{a - a_{0.05}}{a_{0.95} - a_{0.05}}. \tag{6}$$

The other conditional input and output variables were normalized by dividing by $a_{0.95}$,

$$a_{norm} = \frac{a}{a_{0.95}}. \tag{7}$$

After normalization, samples containing output variables larger than 3.0 were replaced with neighboring samples below this threshold to exclude abnormally high values. Each of the four simulations covered 20 days, with the first day being discarded due to model spin-up. The subsequent 13 days were used for training, the 4 following days for validation, and the final 2 days for testing. Data from all four periods were randomly mixed, resulting in 51,118,080 training samples, 15,728,640 validation samples, and 7,864,320 testing samples.

To expedite the validation and testing of DIFF-MP, only a random subset of 10,000 validation samples and 1,600,000 testing samples was used.

2.2. DIFF-MP, Inference Acceleration, and Classifier-Free Guidance

2.2.1. DIFF-MP

Figure 1 illustrates the preprocessing of high-resolution data and how DIFF-MP stochastically parameterizes moist physics. DIFF-MP operates through forward and reverse diffusion processes. In the forward diffusion process, target data are progressively corrupted by Gaussian noise at each step until they become complete noise (represented by the blue arrows in Figure 1). Training data for DIFF-MP are generated during this forward process. At each step, both the added Gaussian noise and the target data fused with noise (denoised latents $\boldsymbol{x}_t$ in Figure 1) are saved. DIFF-MP is then trained to reverse this forward process, generating target data step-by-step in the reverse process (red arrows in Figure 1). DIFF-MP predicts the noise added to the denoised latents at each time step based on conditional inputs and the denoised latent itself, and then subtracts the predicted noise to recover the target data. A detailed explanation of PDMs' mathematical derivations, training, and sampling algorithms is provided in the Supplementary Materials.

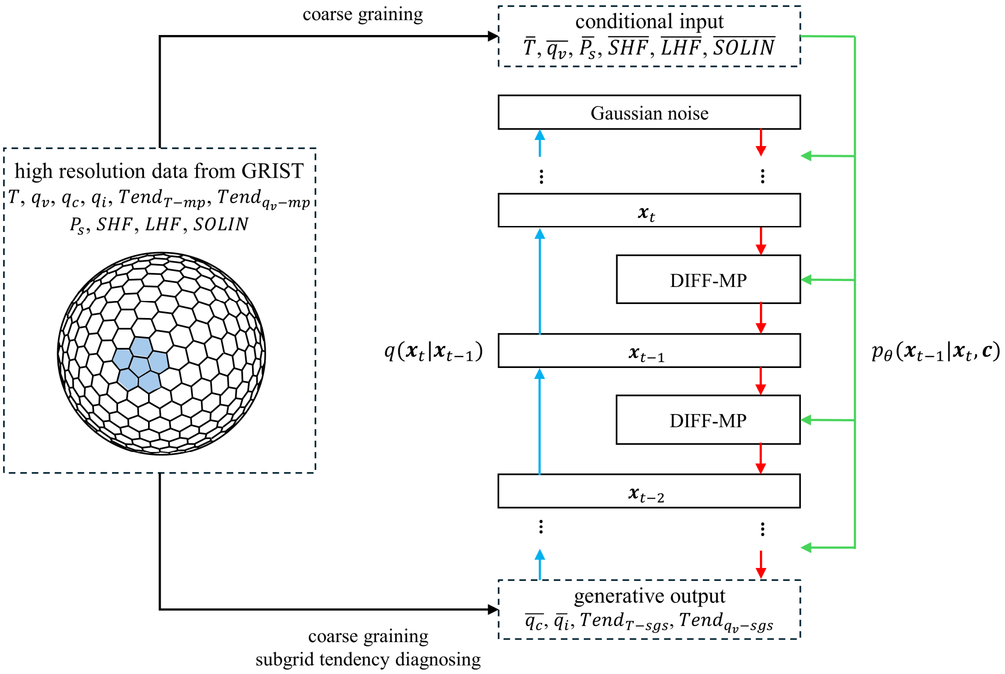
coarse graining
conditional input
$\overline{T}, \overline{q_v}, \overline{P_s}, \overline{SHF}, \overline{LHF}, \overline{SOLIN}$
Gaussian noise
high resolution data from GRIST
$T, q_v, q_c, q_i, Tend_{T-mp}, Tend_{q_v-mp}$
$P_s, SHF, LHF, SOLIN$
$\boldsymbol{x}_t$
DIFF-MP
$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$
$\boldsymbol{x}_{t-1}$
$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{c})$
DIFF-MP
$\boldsymbol{x}_{t-2}$
generative output
$\overline{q_c}, \overline{q_i}, \overline{Tend_{T-sgs}}, \overline{Tend_{q_v-sgs}}$
coarse graining
subgrid tendency diagnosing

Figure 1. The schematic diagram of high-resolution data preprocessing (black arrows), and how DIFF-MP stochastically parameterizes moist physics. Blue and red arrows are the forward diffusion process and reverse diffusion process of DIFF-MP. Green arrows are conditional information flows during reverse process.

DIFF-MP is a hierarchical one-dimensional probabilistic diffusion model (PDM), with its structure shown in Figure A1. It draws inspiration from WaveGrad [45] for its network architecture and GAN-TTS [54] for its U-block structure. Feature-wise Linear Modulation (FiLM) [55] is employed to integrate information from the noise level $\sqrt{\bar{\alpha}_t}$, denoised latent $\boldsymbol{x}_t$, and conditional input $\boldsymbol{c}$. The noise level serves as an indicator, guiding DIFF-MP through the different stages of the denoising process. D-blocks and U-blocks are utilized to decode information from the denoised latent and conditional input layer by layer. The final output of DIFF-MP is the noise ϵ added to the denoised latent during the forward process. Based on hyperparameter tuning, DIFF-MP is configured with 128 filters and 6 layers.

The Adam optimization algorithm, combined with cyclical learning rates ranging from 1×10^{-4} to 1×10^{-3}, was used to train DIFF-MP [56,57]. The loss function was the mean squared error and the batch size was set to 8000. DIFF-MP was trained for five epochs, with the model weights saved at the end of the final epoch. The training was conducted using the Keras 3.0 Python package [58] on an Nvidia 4090 GPU. To thoroughly validate its performance, separate DIFF-MP models were trained for each resolution.

2.2.2. Inference Acceleration of DIFF-MP

The detailed training and sampling algorithms for DIFF-MP, including inference acceleration, are illustrated in Algorithms 1 and 2. Typically, PDMs are trained on a fixed noise schedule $\{\overline{\alpha}_t\}$. However, when fewer time steps are used, models follow a denoising path they were not trained on, leading to a degraded performance. To enhance DIFF-MP's adaptability to fewer denoising steps, it is directly conditioned on the noise level $\sqrt{\overline{\alpha}_t}$, a technique also adopted by Song and Ermon [59,60] in their score-matching framework. Moreover, we define a fixed noise schedule $\{\overline{\alpha}'_t\}$ with a total of T time steps. At each step during training, time t is sampled from a uniform distribution $U(\{1, 2, \ldots, T\})$ and $\overline{\alpha}_t$ is drawn from $U(\overline{\alpha}'_{t-1}, \overline{\alpha}'_t)$. This approach eliminates the need for a fixed series of noise levels, allowing DIFF-MP to be trained on an infinite range of noise levels, significantly improving its generalizability to various denoising schedules. Inference acceleration is achieved by employing different denoising schedules with fewer time steps. As shown in Algorithm 2, new schedules are interpolated from the noise level function $f(t)$ (see Figure S1). This algorithm is adapted from Chen et al. [45], with modifications made to the inference stage.

Algorithm 1. The training algorithm of DIFF-MP with inference acceleration.

```
Require: a fixed noise schedule $\{\overline{\alpha}'_t\}$
repeat
    $\boldsymbol{x}_0, \boldsymbol{c} \sim q(\boldsymbol{x}_0, \boldsymbol{c})$
    $t \sim U(\{1, 2, \ldots, T\})$
    $\overline{\alpha}_t \sim U(\overline{\alpha}'_{t-1}, \overline{\alpha}'_t)$
    $\boldsymbol{\varepsilon} \sim N(0, \mathbf{I})$
    take gradient descent step on
        $\nabla_\theta \left(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\left(\sqrt{\overline{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon},\ \boldsymbol{c}, \sqrt{\overline{\alpha}_t}\right)\right)^2$
until converged
```

Algorithm 2. The sampling algorithm of DIFF-MP with inference acceleration.

```
Require: denoising steps $T$
Require: noise level function $f(t)$
get denoising schedule $\{\overline{\alpha}_t\}$ from $f(t)$
$\boldsymbol{x}_T \sim N(0, \mathbf{I})$, $\boldsymbol{c} \sim q(\boldsymbol{x}_0, \boldsymbol{c})$
for $t = T, \ldots, 1$ do
    $\boldsymbol{z} \sim N(0, \mathbf{I})$if $t > 1$, else $\boldsymbol{z} = \mathbf{0}$
    $\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\overline{\alpha}_t}}\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t, \boldsymbol{c}, \sqrt{\overline{\alpha}_t}\right)\right) + \widetilde{\beta}_t\boldsymbol{z}$
end for
return $\boldsymbol{x}_0$
```

To demonstrate the adaptability of DIFF-MP with inference acceleration across different denoising steps, we compare it with DIFF-MPs trained on fixed time steps. The other DIFF-MPs are trained using fixed noise schedules, where α_t linearly decreases from 9.9999 to 9.94, with a total of n time steps. The values of n are set to 100, 50, 20, 10, 5, and 2. All models are trained for five epochs until convergence, with the model weights saved at the end of the final epoch.

The validation criteria include the mean squared error, Pearson correlation coefficient, coverage ratio, spread–skill correlation, standard deviation, kurtosis, and skewness. Since DIFF-MP is a stochastic parameterization scheme, validation must be conducted on an ensemble of outputs. For each conditional input, 32 different outputs are generated. The mean squared error and correlation coefficient assess the error of DIFF-MP. The coverage ratio measures the proportion of validation data that fall within the range of the DIFF-MP output ensemble, with a high coverage ratio indicating that the output ensemble effectively captures the validation data. The spread–skill correlation evaluates the relationship between DIFF-MP's prediction error and the spread of its outputs. It should be high, because larger errors should correspond to a greater spread. The standard deviation, kurtosis, and skewness capture the higher-order statistics of the DIFF-MP-generated samples, which should closely match those of the validation data. The calculations for these criteria are provided in the Supplementary Materials.

Figures 2 and 3 display the performances of the different DIFF-MPs on the validation data. These models were trained on data with a 120 km resolution, with similar results being observed for other resolutions. For the DIFF-MPs trained on fixed time steps, their performances degraded rapidly as the number of steps decreased, except for the coverage ratio (Figure 2). The increase in the coverage ratio was due to the sample spread becoming excessively large when the denoising steps were too few (Figure 2). In contrast, for the DIFF-MP trained with inference acceleration, the correlation coefficient, mean squared error, and spread–skill correlation remained relatively stable across different steps (Figure 3). The standard deviation, skewness, and kurtosis also deviated only slightly from the validation data (Figure 3). The DIFF-MP trained with inference acceleration significantly outperformed those trained on fixed steps. Training on a range of noise levels effectively mitigated overfitting to specific fixed steps and enhanced the model's generalization to different steps. Balancing inference acceleration with sample quality, DIFF-MP uses five denoising steps throughout the remainder of the study.

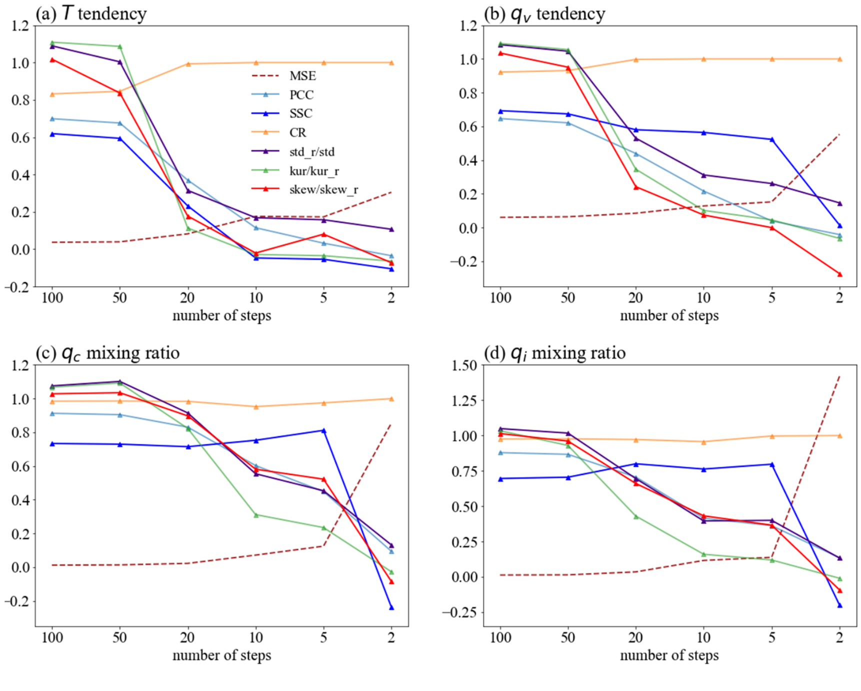

Figure 2. Validation data performance of different DIFF-MPs trained with fixed steps only. Results of $Tend_{T-sgs}$ (**a**), $Tend_{q_v-sgs}$ (**b**), q_c (**c**), and q_i (**d**) are shown. Validation criteria include mean squared error (MSE), Pearson correlation coefficient (PCC), spread–skill correlation (SSC), coverage ratio (CR), and ratio between the statistics of samples generated by DIFF-MP and validation data (std_r/std, kur/kur_r, and skew/skew_r). "std", "kur", and "skew" stand for standard deviation, kurtosis, and skewness of samples produced by DIFF-MP. "std_r", "kur_r", and "skew_r" are those statistics from validation data. Note that "std_r" is denominator in subplots. Different variables are normalized to the same scale. DIFF-MPs are trained on data of resolution 120 km.

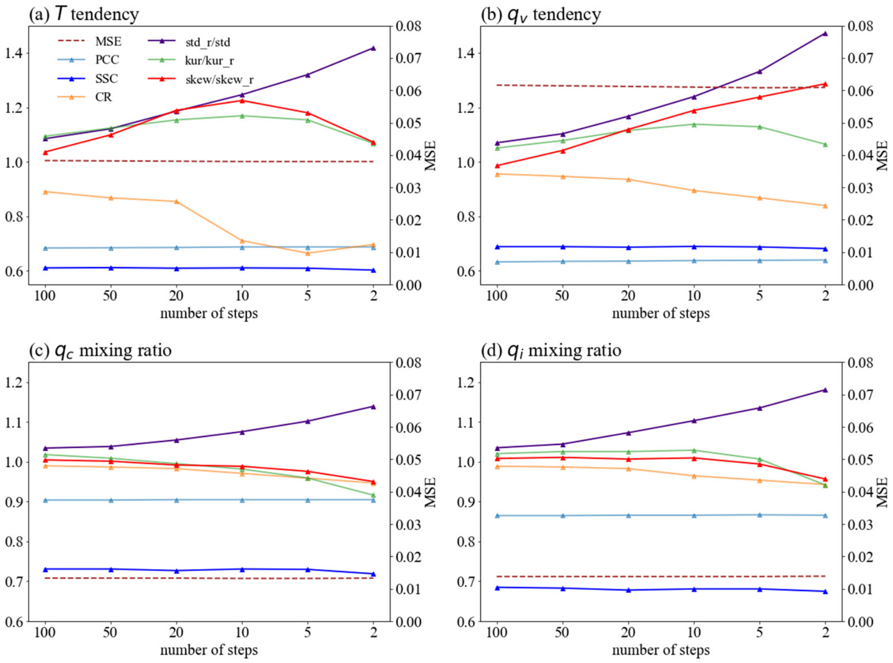

Figure 3. Same as Figure 2, but the validation data performance of DIFF-MP trained with inference acceleration when denoised on different time steps. Results of $Tend_{T-sgs}$ (**a**), $Tend_{q_v-sgs}$ (**b**), q_c (**c**), and q_i (**d**) are shown.

2.2.3. Classifier-Free Guidance of DIFF-MP

The standard deviation, skewness, and kurtosis of the samples generated by DIFF-MP deviate from the validation data for $Tend_{T-sgs}$ and $Tend_{q_v-sgs}$ when using five denoising steps (Figure 3a,b). The standard deviation is also lower for q_c and q_i (Figure 3c,d). These statistical deviations are due to the limited number of denoising steps used in inference acceleration. The unconditional DIFF-MP reproduces the statistics of the training data more accurately than the conditional DIFF-MP. This occurs because the conditional model tends to exploit shortcuts, directly linking the conditional input to the output, whereas the unconditional model does not have this option. A potential improvement could be achieved by combining the outputs of both the unconditional and conditional DIFF-MP models, enhancing the overall performance. This approach aligns with the philosophy of classifier-free guidance [46], where the statistics are restored to their original values by fusing the denoised latents of both models during the inference stage (Equation (8)).

During the training stage, DIFF-MP's conditional input $\boldsymbol{c}$ is replaced by the denoised latent $\boldsymbol{x}_t$ with a probability of 0.1, allowing an unconditional DIFF-MP to be trained simultaneously. The final output is a combination of the denoised latents from both the conditional and unconditional DIFF-MPs, blended using a mixing ratio ω. For each denoising step,

$$\widetilde{P}_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{c}) = (1+\omega)P_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{c}) - \omega P_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_t), \tag{8}$$

where P_θ is the original DIFF-MP and $\widetilde{P}_\theta$ is the combined DIFF-MP. The training and sampling algorithms of DIFF-MP using classifier-free guidance and inference acceleration are presented in Algorithms 3 and 4.

Figure 4 illustrates the effect of the mixing ratio ω on DIFF-MP's performance using validation data at a resolution of 120 km. As ω increases, kurtosis and skewness decrease significantly, while the standard deviation rises. Classifier-free guidance effectively aligns the statistics of the DIFF-MP-generated samples with those of the validation data. Meanwhile, the other four validation criteria remain relatively unchanged. The optimal ω for $Tend_{T-sgs}$ and $Tend_{q_v-sgs}$ is 0.5, whereas for q_c and q_i, no mixing yields better results. It is also observed that using different values of ω for different output variables does not cause interference during the inference stage. Similar validations for ω are conducted at other resolutions, and the optimal values for all four resolutions are provided in Table 2, which DIFF-MP follows in this study.

Algorithm 3. The training algorithm of DIFF-MP with classifier-free guidance and inference acceleration.

```
Require: a fixed noise schedule $\{\bar{\alpha}'_t\}$
Require: probability of unconditional training $p_{uncond}$
repeat
    $\boldsymbol{x}_0, \boldsymbol{c} \sim q(\boldsymbol{x}_0, \boldsymbol{c})$
    $t \sim U(\{1, 2, \ldots, T\})$
    $\bar{\alpha}_t \sim U(\bar{\alpha}'_{t-1}, \bar{\alpha}'_t)$
    $\boldsymbol{\varepsilon} \sim N(0, \mathbf{I})$
    $\boldsymbol{c} \leftarrow \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\varepsilon}$ with probability $p_{uncond}$
    take gradient descent step on
        $\nabla_\theta\left(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon},\ \boldsymbol{c}, \sqrt{\bar{\alpha}_t}\right)\right)^2$
until converged
```

Algorithm 4. The sampling algorithm of DIFF-MP with classifier-free guidance and inference acceleration.

```
Require: denoising steps $T$
Require: noise level function $f(t)$
Require: guidance strength $\omega$
get denoising schedule $\{\bar{\alpha}_t\}$ from $f(t)$
$\boldsymbol{x}_T \sim N(0, \mathbf{I})$, $\boldsymbol{c} \sim q(\boldsymbol{x}_0, \boldsymbol{c})$
for $t = T, \ldots, 1$ do
    $\boldsymbol{z} \sim N(0, \mathbf{I})$ if $t > 1$, else $\boldsymbol{z} = \mathbf{0}$
    $\widetilde{\boldsymbol{\epsilon}}_t = (1+\omega)\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t,\ c, \sqrt{\bar{\alpha}_t}\right) - \omega\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t, \boldsymbol{x}_t, \sqrt{\bar{\alpha}_t}\right)$
    $\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\widetilde{\boldsymbol{\epsilon}}_t\right) + \widetilde{\beta}_t\boldsymbol{z}$
end for
return $\boldsymbol{x}_0$
```

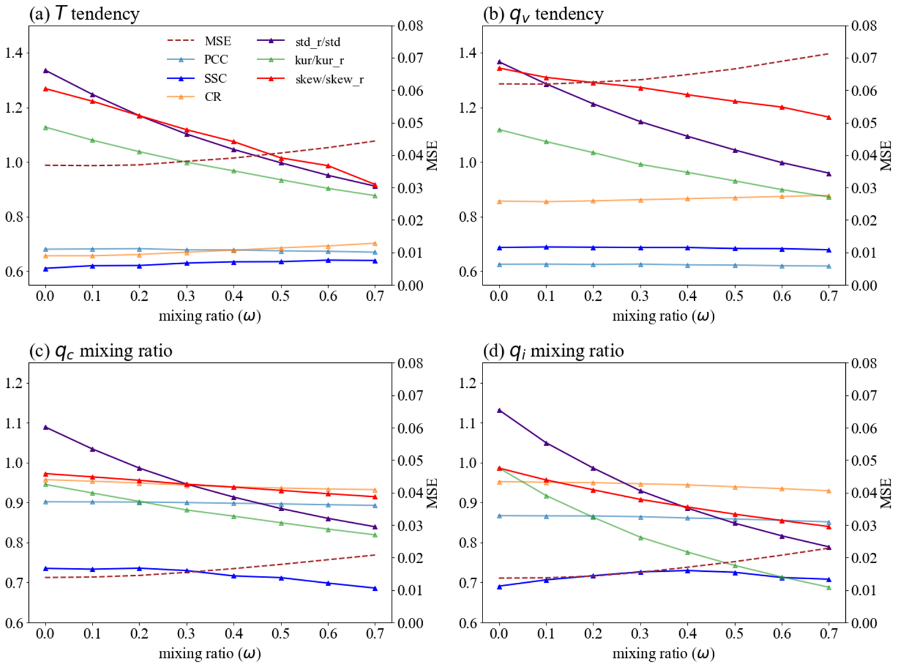

Figure 4. DIFF-MP performance on validation data for different mixing ratios (ω) at resolution of 120 km. Results of $Tend_{T-sgs}$ (**a**), $Tend_{q_v-sgs}$ (**b**), q_c (**c**), and q_i (**d**) are shown. The layout is similar to Figure 2. Note that y-axis of mean squared error is placed at the right-hand side of the subplots.

Table 2. The best mixing ratios (ω) for different output variables and different resolutions.

	120 km	**60 km**	**30 km**	**15 km**
$Tend_{T-sgs}$	0.5	0.6	0.6	0.8
$Tend_{qv-sgs}$	0.5	0.6	0.6	1.0
q_c	0.0	0.0	0.0	0.3
q_i	0.0	0.0	0.0	0.0

3. Results

3.1. Baseline Models and Their Trainings

Conditional VAE (CVAE-MP) and conditional GAN (CGAN-MP) are selected as baseline models. VAE maps each sample from the training data to a known distribution in the latent space, enabling the random generation of similar data [24]. GAN consists of two competing networks, where the generator produces data that the discriminator cannot distinguish from real data [25]. Both models are widely used in generative learning and are suitable for the stochastic parameterization of moist physics. The variables in Table 1 are also used as conditional inputs for CVAE-MP and CGAN-MP. For a fair comparison between the generative models, DIFF-MP, CVAE-MP, and CGAN-MP are designed to have the same model size. The structures of CGAN-MP and CVAE-MP are shown in Figures S2 and S3, respectively.

The training settings for CVAE-MP and CGAN-MP follow those of DIFF-MP. The CGAN-MP output is the sum of a pretrained neural network and the generator, with the neural network providing deterministic output profiles and its weights being frozen during the CGAN-MP training. The Wasserstein GAN technique is applied to stabilize the CGAN-MP training [35]. CVAE-MP and CGAN-MP are trained for 2 and 22 epochs, respectively, until convergence. The best-performing models on the validation data are saved for further comparison with DIFF-MP on the testing data. The validation criterion is the average of the correlation coefficient, spread–skill correlation, and coverage ratio. Different models for CVAE-MP and CGAN-MP are also trained at different resolutions.

3.2. Performance Comparison between Models

The performances of CGAN-MP, CVAE-MP, and DIFF-MP on the testing data at different resolutions are shown in Figure 5. In terms of the mean squared error, DIFF-MP significantly outperforms the other models for q_c and q_i, and is nearly the best for $Tend_{T-sgs}$ and $Tend_{q_v-sgs}$. DIFF-MP consistently achieves the best performance for both the correlation coefficient and spread–skill correlation. It also surpasses the other two models in coverage ratio, except for $Tend_{T-sgs}$. Regarding the statistical properties of the generated samples, DIFF-MP aligns more closely with the testing data and demonstrates a more consistent performance across different resolutions. Overall, DIFF-MP is robustly superior to both CGAN-MP and CVAE-MP on the testing data.

It is noteworthy that the performances of the mean squared error and correlation coefficient decrease as the resolution increases. This is attributed to the fact that, as the grid spacing becomes larger, coarse graining involves more averaging over turbulent and cloud processes, making subgrid processes more predictable. A similar performance degradation has been observed in other studies as well [19,61].

The global distribution of the mean squared error by DIFF-MP and the error differences between the models at a 120 km resolution are shown in Figure 6. DIFF-MP's mean squared error is primarily concentrated in the midlatitudes, where extratropical cyclones are most active. Additionally, the mean squared errors for $Tend_{T-sgs}$ and $Tend_{q_v-sgs}$ are found along large-scale terrain and in tropical regions with active shallow convection (Figure 6a,d). Except for $Tend_{q_v-sgs}$, DIFF-MP consistently exhibits a lower mean squared error than the other two models globally. The red areas in the background highlight DIFF-MP's lower systematic error. Figure 7 presents the global distribution of whether the testing data are captured by the model ensemble at a 120 km resolution. DIFF-MP successfully covers nearly 90% of the testing data across all variables. The uncovered regions correspond to areas with a higher mean squared error, as seen in Figure 6. Notably, CVAE-MP fails to cover most of the $Tend_{T-sgs}$ testing data (Figure 7b), while CGAN-MP performs poorly on q_c and q_i (Figure 7i,l). Figures 6 and 7 further confirm DIFF-MP's robust performance and superiority over the other models.

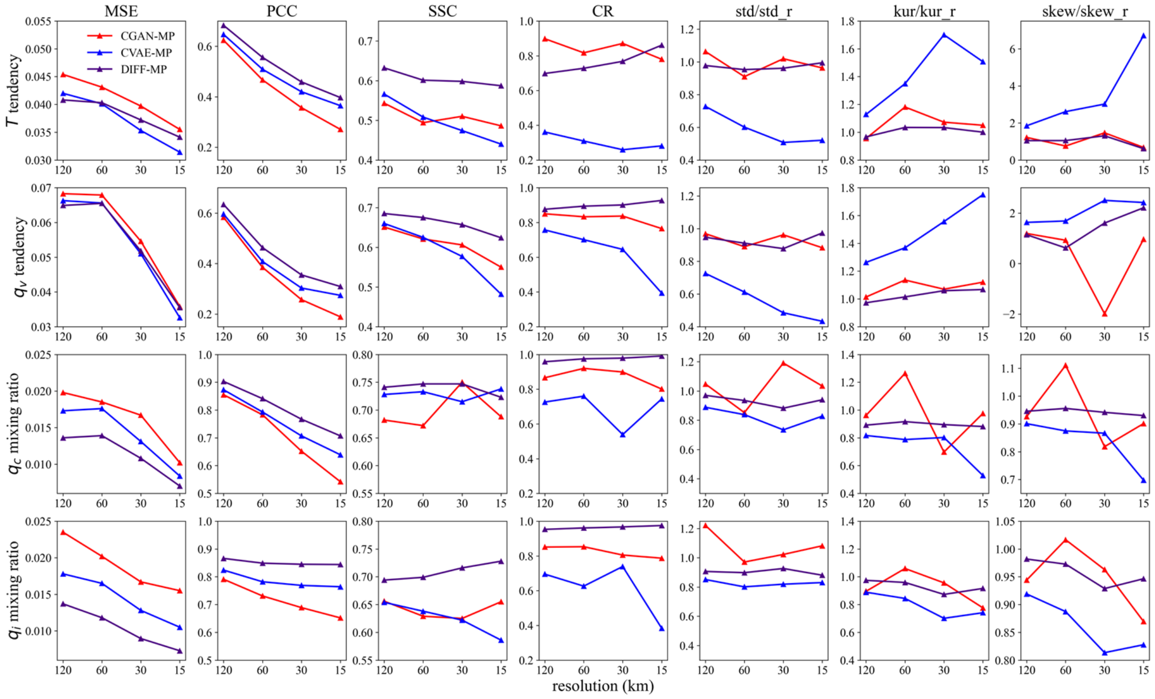

Figure 5. The testing data performance of CGAN-MP, CVAE-MP, and DIFF-MP on $Tend_{T-sgs}$, $Tend_{q_v-sgs}$, q_c, and q_i at different resolutions. Testing criteria are the same as in validation data in Figure 2. Data of different variables are normalized to the same scale.

Figure 8 displays the per-level distribution of the model-generated samples at a resolution of 120 km. The testing data distributions in Figure 8 are truncated at −3.0 and 3.0 due to the exclusion of abnormal extreme data. Two distinct signals of shallow and deep convection are evident in Figure 8a. $Tend_{T-sgs}$ is primarily positive at the upper levels of the boundary layer (around level seven) but negative at lower levels. This pattern results from the condensation of water vapor in shallow convection, where latent heat is released near the top of the boundary layer and collected at lower levels. In Figure 8e, $Tend_{q_v-sgs}$ is positive near the top of the boundary layer and negative below, indicating the vertical transport of water vapor by shallow convection. Shallow convection is also evident in the lower-level extremes of q_c and q_i (Figure 8i,m). Above the boundary layer, another heating peak for $Tend_{T-sgs}$ is observed between levels 10 and 15, corresponding to deep convection (Figure 8a). Deep convection is also reflected in the upper levels of q_i (Figure 8m).

All three generative models capture the signals of shallow and deep convection. However, their predictions for extreme data distributions differ. DIFF-MP nearly perfectly reproduces the testing data distribution, even reflecting the artificial cutoff at ±3.0. The only notable underestimation occurs for $Tend_{T-sgs}$ above level 15 (Figure 8b). CVAE-MP is overly conservative regarding extreme values, capturing only the probability density of large values (Figure 8d,h,l,p). In contrast, CGAN-MP is excessively aggressive, resulting in an overly broad probability distribution (Figure 8c,g,k,o), and it also predicts unrealistic negative values for q_c and q_i. Figure A2, which presents results from a 30 km resolution, aligns with the findings in Figure 8.

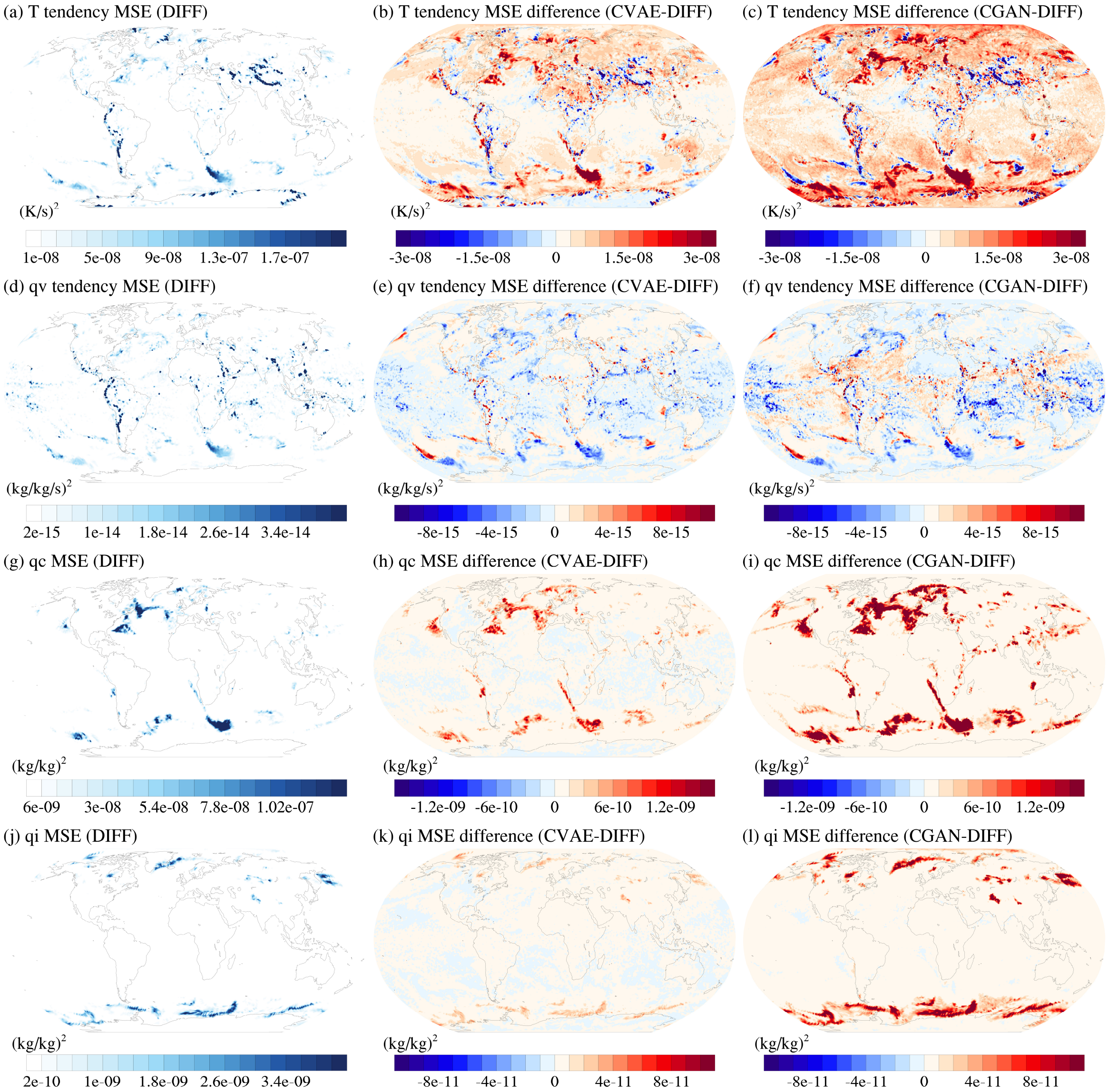
(a) T tendency MSE (DIFF)
(b) T tendency MSE difference (CVAE-DIFF)
(c) T tendency MSE difference (CGAN-DIFF)
(K/s)²
1e-08 5e-08 9e-08 1.3e-07 1.7e-07
-3e-08 -1.5e-08 0 1.5e-08 3e-08
(d) qv tendency MSE (DIFF)
(e) qv tendency MSE difference (CVAE-DIFF)
(f) qv tendency MSE difference (CGAN-DIFF)
(kg/kg/s)²
2e-15 1e-14 1.8e-14 2.6e-14 3.4e-14
-8e-15 -4e-15 0 4e-15 8e-15
(g) qc MSE (DIFF)
(h) qc MSE difference (CVAE-DIFF)
(i) qc MSE difference (CGAN-DIFF)
(kg/kg)²
6e-09 3e-08 5.4e-08 7.8e-08 1.02e-07
-1.2e-09 -6e-10 0 6e-10 1.2e-09
(j) qi MSE (DIFF)
(k) qi MSE difference (CVAE-DIFF)
(l) qi MSE difference (CGAN-DIFF)
(kg/kg)²
2e-10 1e-09 1.8e-09 2.6e-09 3.4e-09
-8e-11 -4e-11 0 4e-11 8e-11

Figure 6. The global distribution of mean squared error by DIFF-MP (**a**,**d**,**g**,**j**), and mean-squared-error difference between DIFF-MP and CVAE-MP (**b**,**e**,**h**,**k**), DIFF-MP, and CGAN-MP (**c**,**f**,**i**,**l**). The time is 20 April 2005, UTC 00:00. The testing data are at about 400 m height under 120 km resolution.

Figure 9 compares the ensemble output profiles from different models with the corresponding profiles from testing data at a 120 km resolution. All models' ensembles capture the overall vertical variability of the testing data profiles. However, for $Tend_{T-sgs}$ and $Tend_{q_v-sgs}$, the ensembles from CGAN-MP and CVAE-MP are centered around zero and deviate from the testing data, particularly between levels 10 and 15 (Figure 9a,b,d,e). In contrast, DIFF-MP's ensemble closely follows the vertical variability of the testing data (Figure 9c,f). For q_c and q_i, CGAN-MP's ensemble extends to higher values (2.0–2.5), as seen in Figure 9g,h, consistent with the excessive extreme values predicted by CGAN-MP in Figure 8. In comparison, the value ranges of CVAE-MP's and DIFF-MP's ensembles better align with the testing data. The results at a 30 km resolution are similar to those in Figure 9 and are presented in Figure A3.

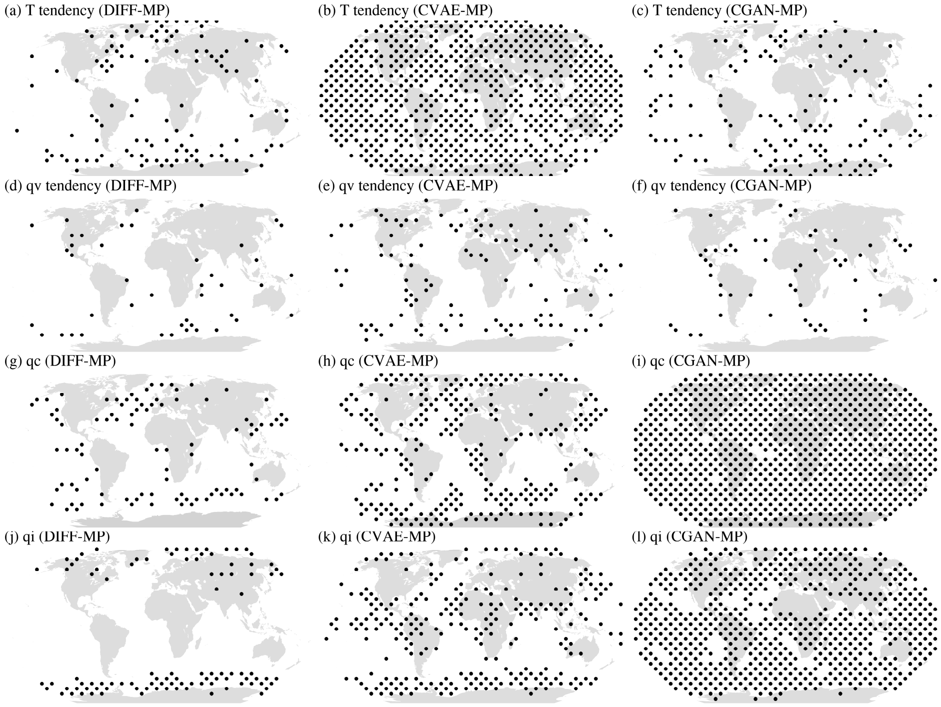

Figure 7. The global distribution of whether the testing data are covered by the model ensemble. Results of DIFF-MP (**a**,**d**,**g**,**j**), CVAE-MP (**b**,**e**,**h**,**k**), and CGAN-MP (**c**,**f**,**i**,**l**) are shown. The locations where the testing data are NOT covered are labeled as dots. The time is 20 April 2005, UTC 00:00. The testing data are at about 400 m height under 120 km resolution.

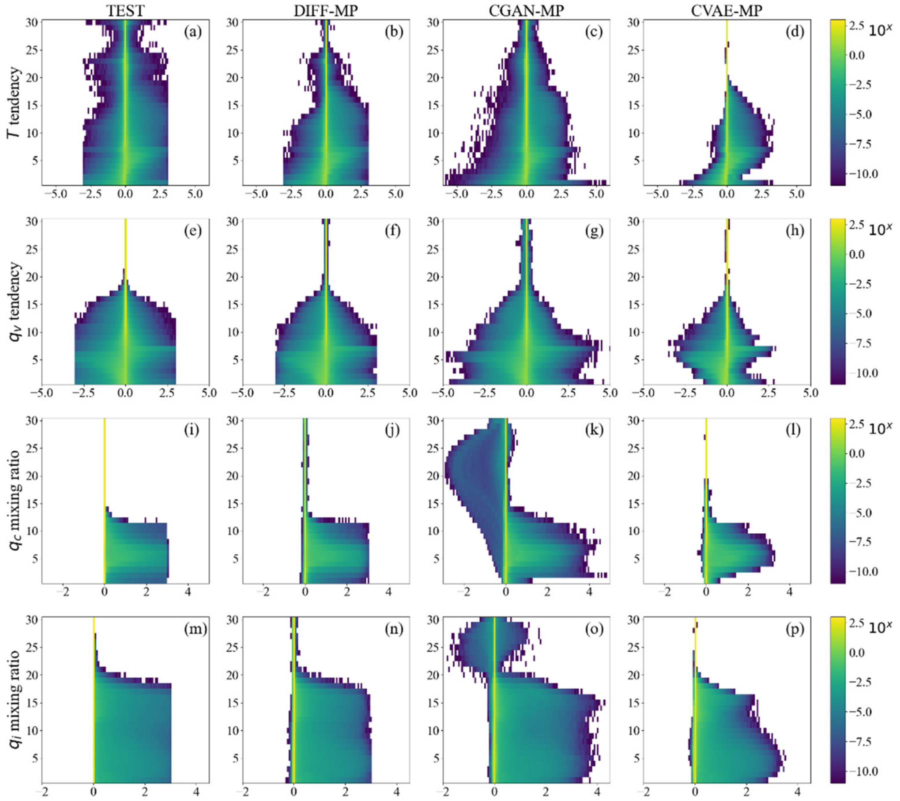

Figure 8. Per-level data distribution of testing data and model-generated samples. Results of $Tend_{T-sgs}$ (**a**–**d**), $Tend_{q_v-sgs}$ (**e**–**h**), q_c (**i**–**l**), and q_i (**m**–**p**) are presented. Different variables are all normalized to the same scale for comparison. The resolution is 120 km.

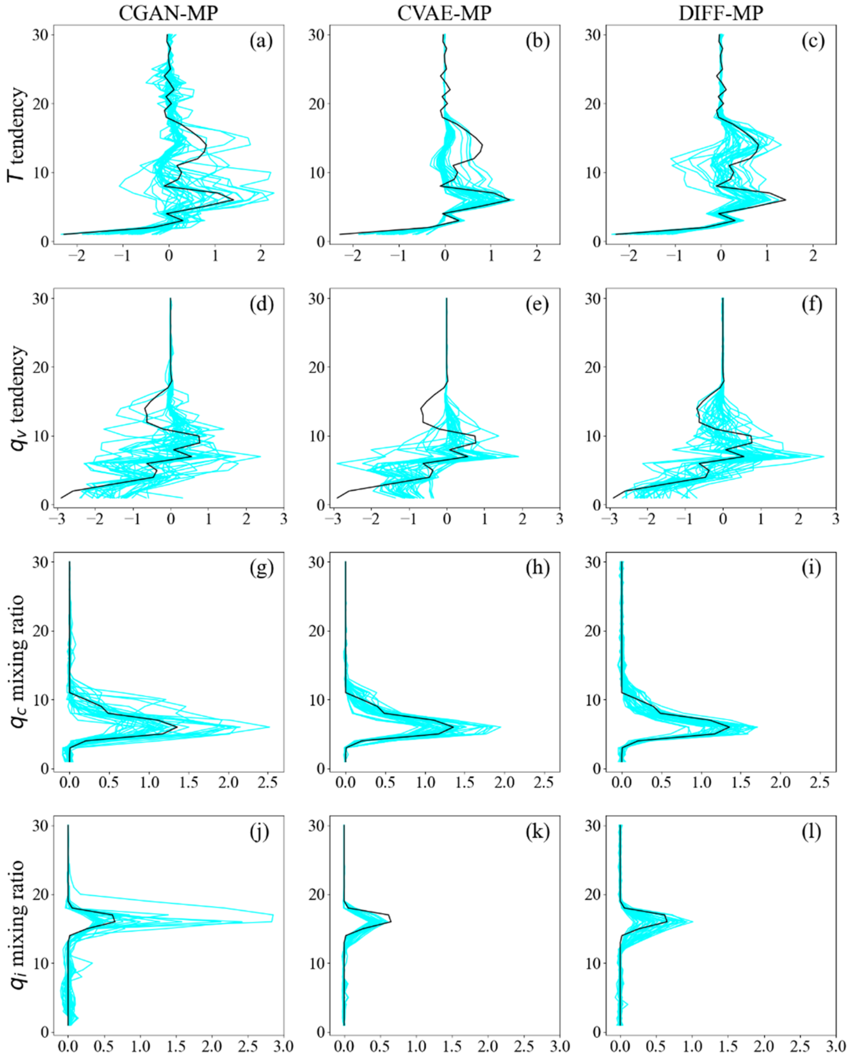

Figure 9. Vertical profiles of an ensemble of 32 samples from different models (blue) and the corresponding profiles (black) in testing data. Profiles of $Tend_{T-sgs}$ (**a**–**c**), $Tend_{q_v-sgs}$ (**d**–**f**), q_c (**g**–**i**), and q_i (**j**–**l**) are presented. They are all normalized to the same scale for comparison. The resolution is 120 km.

DIFF-MP is more effective than CGAN-MP and CVAE-MP at reproducing the data distributions for subgrid moistening, heating, and cloud processes. Additionally, DIFF-MP generates a more accurate ensemble that better encompasses the output profiles from the testing data compared to the other two models. Overall, DIFF-MP demonstrates a superior performance in the stochastic parameterization of moist physics.

3.3. Interpretability of DIFF-MP

Testing the interpretability of DIFF-MP is crucial for ensuring its physical robustness. Figure 10 illustrates the effect of boundary-layer stratification changes on DIFF-MP's outputs. Initially, the boundary layer exhibits unstable stratification. When the T profile is neutralized, both $Tend_{T-sgs}$ and $Tend_{q_v-sgs}$ approach zero (Figure 10c,d), and q_i at higher levels decreases, indicating the cessation of shallow and deep convection (Figure 10f). Meanwhile, q_c and q_i increase within the boundary layer (Figure 10e,f) as q_v accumulates due to suppressed shallow convection, leading to excessive condensation. After q_v is neutralized, $Tend_{q_v-sgs}$ becomes positive and q_c disappears in the boundary layer (Figure 10d,e). This is because q_v is not saturated after the neutralization, causing shallow clouds to evaporate and DIFF-MP to restore q_v in the boundary layer. Figure 11 shows the impact of surface flux.

As the surface flux decreases, $Tend_{T-sgs}$, $Tend_{q_v-sgs}$, and q_i at higher levels significantly decrease (Figure 11a,b,d) due to the reduction in deep convection caused by a lower surface flux. q_c in the boundary layer initially increases and then decreases as the surface flux is further reduced. Similar to Figure 10e, q_c accumulates when convection is suppressed, but diminishes when convection becomes severely constrained. Figures 10 and 11 confirm that DIFF-MP's response to input variation is physically reasonable, supporting its future implementation in the GRIST model.

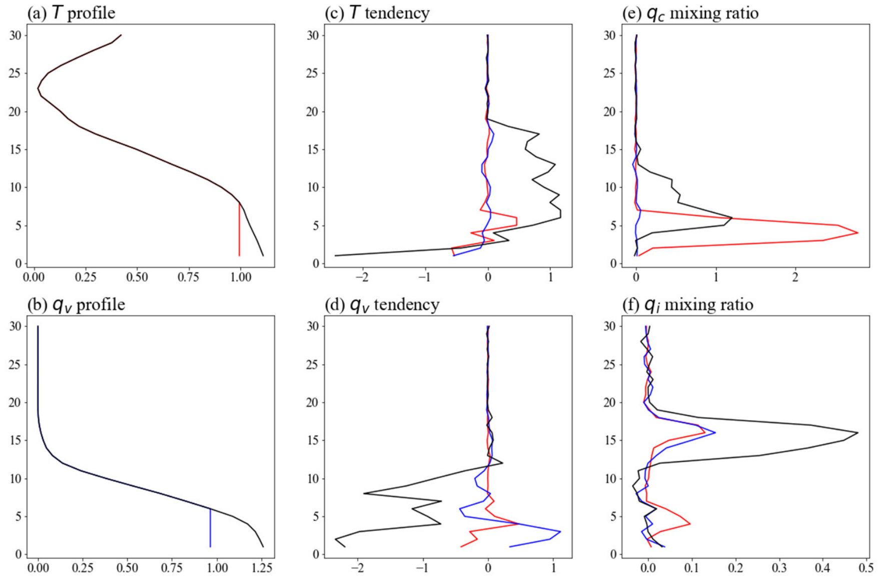

Figure 10. Interpretability experiment of DIFF-MP showing how output profiles change with stratification change in boundary layer. Subplots (**a**,**b**) show the way how T and q_v profiles change. The corresponding output profiles of $Tend_{T-sgs}$ (**c**), $Tend_{q_v-sgs}$ (**d**), q_c (**e**), and q_i (**f**) due to the change in T and q_v are colored as red and blue. The original input and output profiles are black lines. They are all normalized to the same scale for comparison.

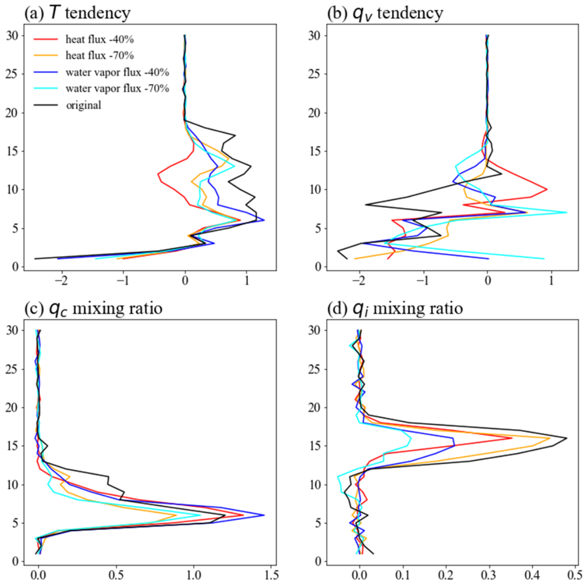

Figure 11. Interpretability experiment of DIFF-MP showing how output profiles change with surface heat flux and water vapor flux reduction. The vertical profiles produced by DIFF-MP after heat flux and water vapor flux are reduced for 40% (red and blue lines) or 70% (green and pink lines) are presented. Profiles of $Tend_{T-sgs}$ (**a**), $Tend_{q_v-sgs}$ (**b**), q_c (**c**), and q_i (**d**) are shown. The original input and output profiles are black lines. They are all normalized to the same scale for comparison.

4. Conclusions and Discussions

This study introduces a stochastic moist physics parameterization scheme, DIFF-MP, based on a one-dimensional PDM. DIFF-MP is trained on a range of noise levels, enhancing its generalizability to large denoising steps and achieving a 20-fold acceleration without significant performance degradation. Classifier-free guidance is employed to minimize the statistical deviations between the DIFF-MP-generated samples and the validation data.

DIFF-MP's capability to stochastically parameterize the subgrid contributions of moist physics is compared with that of CVAE-MP and CGAN-MP on the testing data. DIFF-MP consistently outperforms the other models in terms of the prediction error, spread–skill correlation, coverage ratio, and reproduction of subgrid contribution statistics, including standard deviation, kurtosis, and skewness. Its performance remains consistent across four different resolutions, with improvements in prediction error reaching up to 40% compared to the other models.

DIFF-MP's prediction error and the testing data not captured by its predicted ensemble are concentrated along large-scale terrain and midlatitude regions, where extratropical cyclones are most active. Globally, DIFF-MP exhibits lower prediction errors than the other models. In terms of coverage ratio, DIFF-MP includes nearly 90% of the testing data within its predicted ensemble. Furthermore, DIFF-MP's predicted ensemble profiles demonstrate a more reasonable vertical variability and value ranges than those produced by the other models.

DIFF-MP nearly perfectly reproduces the per-level distributions of different variables. CGAN-MP tends to predict excessive extreme values, while CVAE-MP predicts too few. When unstable stratification in the boundary layer is neutralized or surface flux is reduced, deep convection is significantly suppressed, while low clouds accumulate in the boundary layer due to constrained shallow convection. The interpretability experiments confirm that DIFF-MP's predictions are physically consistent.

This study focuses solely on parameterizing moist physics as a proof of concept for PDMs in stochastic parameterization. DIFF-MP can be extended to include additional physical processes such as boundary-layer turbulence, longwave radiation, and shortwave radiation to create a unified parameterization for all physical processes. Future work will involve implementing DIFF-MP into the GRIST model and investigating its impact on the numerical simulation of the Madden–Julian Oscillation, the intertropical convergence zone, and climate mean and variability.

The integration of Python-based machine learning models into Fortran-based Earth system models remains challenging. Previous efforts often hard-coded Python models into Fortran through self-developed tools, which can be cumbersome and time-consuming [11,16,17]. It is crucial for Earth system model development teams to create official tools to streamline the implementation process. One potential solution is to modularize Earth system models, wrapping them with Python interfaces that allow for easy implementation and support for heterogeneous computing using both GPUs and CPUs [62]. This approach would enable machine learning models to run on GPUs while numerical integration is performed on CPUs, optimizing efficiency. Additionally, with advances in large language models, it may become feasible to translate Fortran-based Earth system models into Python, enabling the entire system to run on GPUs, thereby eliminating the implementation bottleneck and accelerating the development of machine-learning-based parameterization schemes [63,64].

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/atmos15101219/s1, Figure S1: The noise level function $f(t)$; Figure S2: The structure of CGAN-MP; Figure S3: The structure of CVAE-MP; Figure S4: Training and sampling algorithms of PDM. Equations (S5)–(S16): Mathematical derivation on subgrid contributions of moist physics. Equations (S17)–(S33): Mathematical derivation on probabilistic diffusion model. Equations (S34)–(S44): Mathematical derivation of validation criteria.

Author Contributions: Conceptualization, L.W.; methodology, L.W.; software, Y.W.; validation, L.W.; formal analysis, L.W.; writing—original draft preparation, L.W.; writing—review and editing, L.W.,

X.H., H.W. and R.Z.; visualization, L.W.; project administration, L.W.; funding acquisition, L.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by China Meteorological Service Association (Grant No. CMSA2023MC010), Natural Science Foundation of Chongqing, China (Grant No. CSTB2024NSCQ-MSX0641), and National Natural Science Foundation of China (Grant No. 42405152).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Restrictions apply to the availability of these data. Data were obtained from PIESAT Information Technology Co., Ltd. and are available from Y.W. with the permission of the company.

Acknowledgments: L.W. appreciates the manuscript revision by Baoxiang Pan from Institute of Atmospheric Physics, Chinese Academy of Sciences, and Dazhi Xi from Department of Earth Sciences, the University of Hong Kong.

Conflicts of Interest: The authors declare no conflicts of interest. Yiming Wang is employee of 2035 Future Laboratory, PIESAT Information Technology Co., Ltd. The paper reflects the views of the scientists, and not the company.

Appendix A

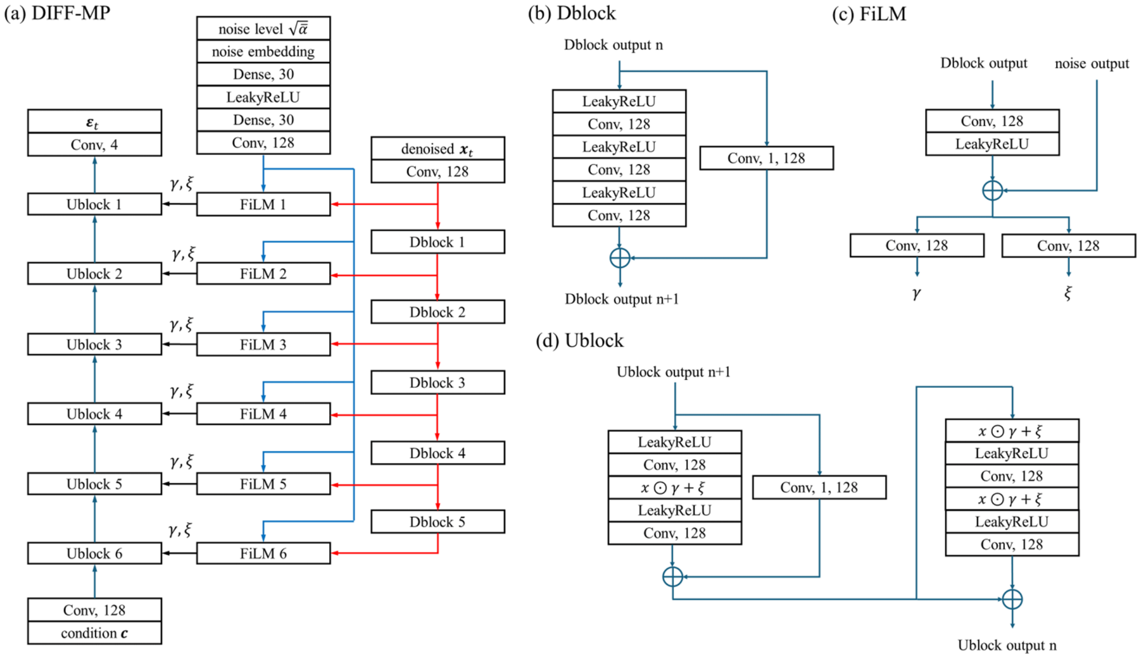

Figure A1. The structure of DIFF-MP (**a**). Detail structures of Dblock (**b**), FiLM (**c**), and UBlock (**d**) modules are also depicted. "Conv, 128" is one-dimensional convolution module with kernel size 3 and 128 filters. "Conv, 1, 128" has kernel size 1 and 128 filters. "Dense, 30" is fully connected layer of 30 neurons. "Noise embedding" adopts the sinusoidal positional embedding of Vaswani et al. [65] with minor modifications. "⊙" is element-wise multiplication.

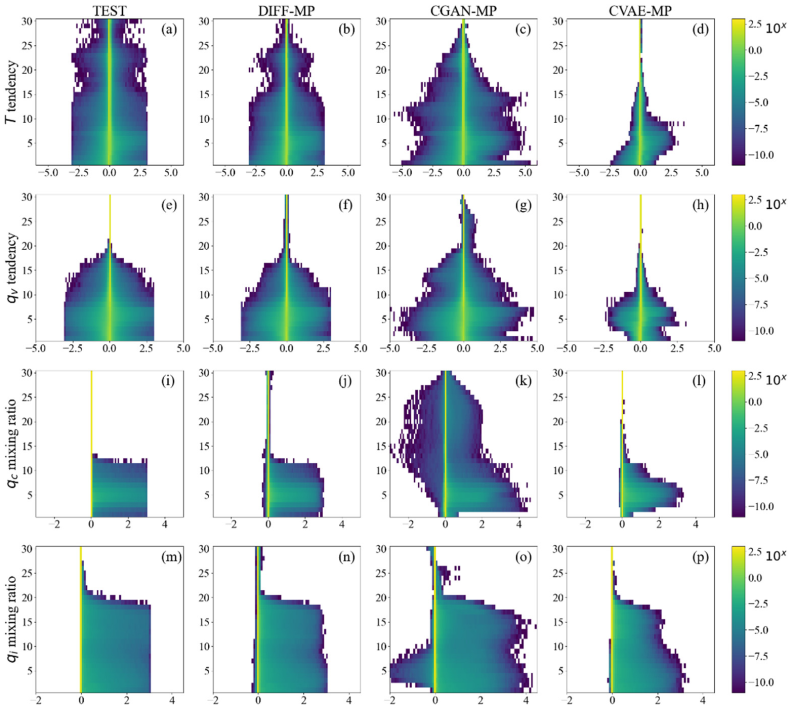

Figure A2. Figure layout is the same as Figure 8, but for resolution of 30 km. Results of $Tend_{T-sgs}$ (**a**–**d**), $Tend_{q_v-sgs}$ (**e**–**h**), q_c (**i**–**l**), and q_i (**m**–**p**) are presented.

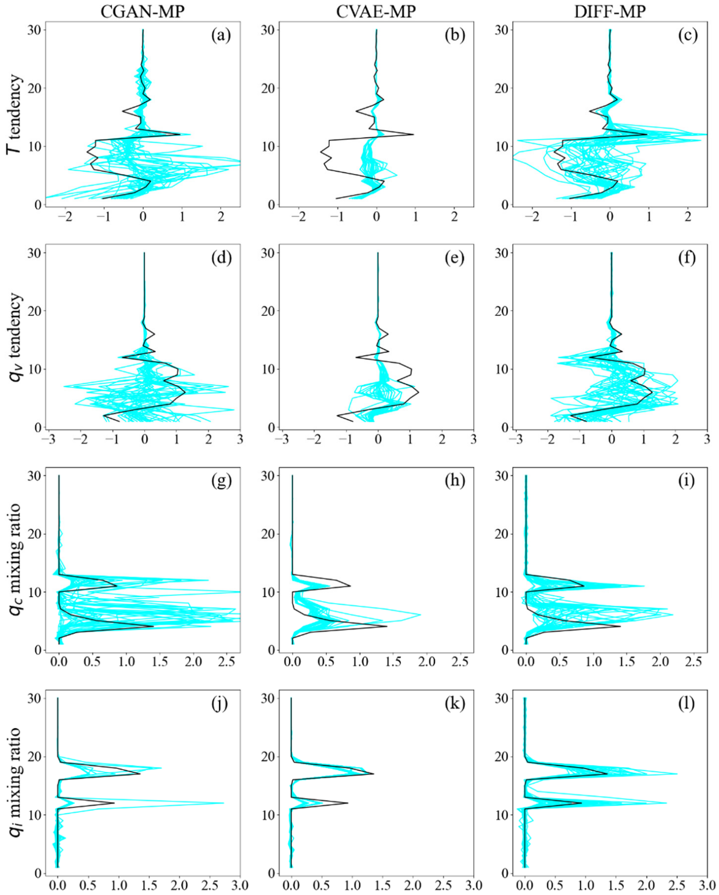

Figure A3. Figure layout is the same as Figure 9, but for resolution of 30 km. Profiles of $Tend_{T-sgs}$ (**a**–**c**), $Tend_{q_v-sgs}$ (**d**–**f**), q_c (**g**–**i**), and q_i (**j**–**l**) are presented.

References

1. Daleu, C.L.; Plant, R.S.; Woolnough, S.J.; Sessions, S.; Herman, M.J.; Sobel, A.; Wang, S.; Kim, D.; Cheng, A.; Bellon, G.; et al. Intercomparison of methods of coupling between convection and large-scale circulation: 1. Comparison over uniform surface conditions. *J. Adv. Model. Earth Syst.* **2015**, *7*, 1576–1601. [CrossRef] [PubMed]
2. Daleu, C.L.; Plant, R.S.; Woolnough, S.J.; Sessions, S.; Herman, M.J.; Sobel, A.; Wang, S.; Kim, D.; Cheng, A.; Bellon, G.; et al. Intercomparison of methods of coupling between convection and large-scale circulation: 2. Comparison over nonuniform surface conditions. *J. Adv. Model. Earth Syst.* **2016**, *8*, 387–405. [CrossRef]
3. Arnold, N.P.; Branson, M.; Burt, M.A.; Abbot, D.S.; Kuang, Z.; Randall, D.A.; Tziperman, E. Effects of explicit atmospheric convection at high CO_2. *Proc. Natl. Acad. Sci. USA* **2014**, *111*, 10943–10948. [CrossRef]
4. Hohenegger, C.; Stevens, B. Coupled radiative convective equilibrium simulations with explicit and parameterized convection. *J. Adv. Model. Earth Syst.* **2016**, *8*, 1468–1482. [CrossRef]
5. Bony, S.; Stevens, B.; Frierson, D.M.W.; Jakob, C.; Kageyama, M.; Pincus, R.; Shepherd, T.G.; Sherwood, S.C.; Pier Siebesma, A.; Sobel, A.H.; et al. Clouds, circulation, and climate sensitivity. *Nat. Geosci.* **2015**, *8*, 261–268. [CrossRef]
6. Coppin, D.; Bony, S. Physical mechanisms controlling the initiation of convective self-aggregation in a general circulation model. *J. Adv. Model. Earth Syst.* **2015**, *7*, 2060–2078. [CrossRef]
7. Nie, J.; Shaevitz, D.A.; Sobel, A.H. Forcings and feedbacks on convection in the 2010 Pakistan flood: Modeling extreme precipitation with interactive large-scale ascent. *J. Adv. Model. Earth Syst.* **2016**, *8*, 1055–1072. [CrossRef]
8. Brenowitz, N.D.; Bretherton, C.S. Prognostic validation of a neural network unified physics parameterization. *Geophys. Res. Lett.* **2018**, *45*, 6289–6298. [CrossRef]
9. Gentine, P.; Pritchard, M.; Rasp, S.; Reinaudi, G.; Yacalis, G. Could machine learning break the convection parameterization deadlock? *Geophys. Res. Lett.* **2018**, *45*, 5742–5751. [CrossRef]
10. Beucler, T.; Gentine, P.; Yuval, J.; Gupta, A.; Peng, L.; Lin, J.; Yu, S.; Rasp, S.; Ahmed, F.; O'Gorman, P.A.; et al. Climate-invariant machine learning. *Sci. Adv.* **2024**, *10*, eadj7250. [CrossRef]
11. Brenowitz, N.D.; Bretherton, C.S. Spatially extended tests of a neural network parametrization trained by coarse-graining. *J. Adv. Model. Earth Syst.* **2019**, *11*, 2728–2744. [CrossRef]
12. Han, Y.; Zhang, G.J.; Huang, X.; Wang, Y. A moist physics parameterization based on deep learning. *J. Adv. Model. Earth Syst.* **2020**, *12*, e2020MS002076. [CrossRef]
13. Han, Y.; Zhang, G.J.; Wang, Y. An ensemble of neural networks for moist physics processes, its generalizability and stable integration. *J. Adv. Model. Earth Syst.* **2023**, *15*, e2022MS003508. [CrossRef]
14. Lin, J.; Yu, S.; Peng, L.; Beucler, T.; Wong-Toi, E.; Hu, Z.; Gentine, P.; Geleta, M.; Pritchard, M. Sampling Hybrid Climate Simulation at Scale to Reliably Improve Machine Learning Parameterization. *arXiv* **2024**, arXiv:2309.16177.
15. Mooers, G.; Pritchard, M.; Beucler, T.; Ott, J.; Yacalis, G.; Baldi, P.; Gentine, P. Assessing the potential of deep learning for emulating cloud superparameterization in climate models with real-geography boundary conditions. *J. Adv. Model. Earth Syst.* **2021**, *13*, e2020MS002385. [CrossRef]
16. Rasp, S.; Pritchard, M.S.; Gentine, P. Deep learning to represent subgrid processes in climate models. *Proc. Natl. Acad. Sci. USA* **2018**, *115*, 9684–9689. [CrossRef]
17. Wang, X.; Han, Y.; Xue, W.; Yang, G.; Zhang, G.J. Stable climate simulations using a realistic general circulation model with neural network parameterizations for atmospheric moist physics and radiation processes. *Geosci. Model. Dev.* **2022**, *15*, 3923–3940. [CrossRef]
18. Watt-Meyer, O.; Brenowitz, N.D.; Clark, S.K.; Henn, B.; Kwa, A.; McGibbon, J.; Perkins, W.A.; Harris, L.; Bretherton, C.S. Neural network parameterization of subgrid-scale physics from a realistic geography global storm-resolving simulation. *J. Adv. Model. Earth Syst.* **2024**, *16*, e2023MS003668. [CrossRef]
19. Yuval, J.; O'Gorman, P.A. Stable machine-learning parameterization of subgrid processes for climate modeling at a range of resolutions. *Nat. Commun.* **2020**, *11*, 3295. [CrossRef]
20. Yuval, J.; O'Gorman, P.A.; Hill, C.N. Use of neural networks for stable, accurate and physically consistent parameterization of subgrid atmospheric processes with good performance at reduced precision. *Geophys. Res. Lett.* **2021**, *48*, e2020GL091363. [CrossRef]
21. Buizza, R.; Miller, M.; Palmer, T.N. Stochastic representation of model uncertainties in the ECMWF ensemble prediction system. *Q. J. R. Meteorol. Soc.* **1999**, *125*, 2887–2908. [CrossRef]
22. Christensen, H.M.; Berner, J.; Coleman, D.; Palmer, T.N. Stochastic parametrisation and the El Niño-Southern oscillation. *J. Clim.* **2017**, *30*, 17–38. [CrossRef]
23. Weisheimer, A.; Corti, S.; Palmer, T. Addressing model error through atmospheric stochastic physical parametrizations: Impact on the coupled ECMWF seasonal forecasting system. *Phil. Trans. R. Soc. A* **2014**, *372*, 20130290. [CrossRef]
24. Kingma, D.; Welling, M. Auto-encoding variational Bayes. In Proceedings of the International Conference on Learning Representations, Banff, AB, Canada, 14–16 April 2014.
25. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montréal, QC, Canada, 8–13 December 2014.

26. Alcala, J.; Timofeyev, I. Subgrid-scale parametrization of unresolved scales in forced Burgers equation using generative adversarial networks (GAN). *Theor. Comp. Fluid. Dyn.* **2021**, *35*, 875–894. [CrossRef]
27. Bhouri, M.A.; Gentine, P. History-Based, Bayesian, Closure for Stochastic Parameterization: Application to Lorenz' 96. *arXiv* **2022**, arXiv:2210.14488.
28. Crommelin, D.; Edeling, W. Resampling with neural networks for stochastic parameterization in multiscale systems. *Phys. D Nonlinear Phenom.* **2021**, *422*, 132894. [CrossRef]
29. Gagne, D.J.; Christensen, H.; Subramanian, A.; Monahan, A.H. Machine learning for stochastic parameterization: Generative adversarial networks in the lorenz' 96 model. *J. Adv. Model. Earth Syst.* **2020**, *12*, e2019MS001896. [CrossRef]
30. Nadiga, B.T.; Sun, X.; Nash, C. Stochastic parameterization of column physics using generative adversarial networks. *Environ. Data Sci.* **2022**, *1*, e22. [CrossRef]
31. Parthipan, R.; Christensen, H.M.; Hosking, J.S.; Wischik, D.J. Using probabilistic machine learning to better model temporal patterns in parameterizations: A case study with the Lorenz 96 model. *Geosci. Model. Dev.* **2023**, *16*, 4501–4519. [CrossRef]
32. Perezhogin, P.; Zanna, L.; Fernandez-Granda, C. Generative data-driven approaches for stochastic subgrid parameterizations in an idealized ocean model. *J. Adv. Model. Earth Syst.* **2023**, *15*, e2023MS003681. [CrossRef]
33. Bao, J.; Chen, D.; Wen, F.; Li, H.; Hua, G. CVAE-GAN: Fine-grained image generation through asymmetric training. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
34. Ichikawa, Y.; Hukushima, K. Learning Dynamics in Linear VAE: Posterior Collapse Threshold, Superfluous Latent Space Pitfalls, and Speedup with KL Annealing. In Proceedings of the 27th International Conference on Artificial Intelligence and Statistics, PMLR, València, Spain, 2–4 May 2024.
35. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In Proceedings of the 34th International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017.
36. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
37. Huang, H.; Li, Z.; He, R.; Sun, Z.; Tan, T. IntroVAE: Introspective variational autoencoders for photographic image synthesis. In Proceedings of the 32nd International Conference on Neural Information Processing Systems, Montréal, QC, Canada, 2–8 December 2018.
38. Kim, T.; Cha, M.; Kim, H.; Lee, J.K.; Kim, J. Learning to discover cross-domain relations with generative adversarial networks. In Proceedings of the 34th International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017.
39. Ho, J.; Jain, A.; Abbeel, P. Denoising diffusion probabilistic models. In Proceedings of the 34th Annual Conference on Neural Information Processing Systems, Online, 6–12 December 2020.
40. Esser, P.; Kulal, S.; Blattmann, A.; Entezari, R.; Müller, J.; Saini, H.; Levi, Y.; Lorenz, D.; Sauer, A.; Boesel, F.; et al. Scaling rectified flow transformers for high-resolution image synthesis. In Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria, 21–27 July 2024.
41. Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv* **2022**, arXiv:2204.06125.
42. Luo, C. Understanding diffusion models: A unified perspective. *arXiv* **2022**, arXiv:2208.11970.
43. Nichol, A.Q.; Dhariwal, P. Improved denoising diffusion probabilistic models. In Proceedings of the 38th International Conference on Machine Learning, PMLR, Online, 18–24 July 2021.
44. Dhariwal, P.; Nichol, A. Diffusion models beat gans on image synthesis. In Proceedings of the 35th Annual Conference on Neural Information Processing Systems, Online, 6–14 December 2021.
45. Chen, N.; Zhang, Y.; Zen, H.; Weiss, R.J.; Norouzi, M.; Chan, W. Wavegrad: Estimating gradients for waveform generation. *arXiv* **2020**, arXiv:2009.00713.
46. Ho, J.; Salimans, T. Classifier-free diffusion guidance. *arXiv* **2022**, arXiv:2207.12598.
47. Zhang, Y.; Li, J.; Yu, R.; Zhang, S.; Liu, Z.; Huang, J.; Zhou, Y. A layer-averaged nonhydrostatic dynamical framework on an unstructured mesh for global and regional atmospheric modeling: Model description, baseline evaluation, and sensitivity exploration. *J. Adv. Model. Earth Syst.* **2019**, *11*, 1685–1714. [CrossRef]
48. Zhang, Y.; Li, J.; Yu, R.; Liu, Z.; Zhou, Y.; Li, X.; Huang, X. A multiscale dynamical model in a dry-mass coordinate for weather and climate modeling: Moist dynamics and its coupling to physics. *Mon. Weather Rev.* **2020**, *148*, 2671–2699. [CrossRef]
49. Heikes, R.; Randall, D.A. Numerical integration of the shallow-water equations on a twisted icosahedral grid. Part II. A detailed description of the grid and an analysis of numerical accuracy. *Mon. Weather Rev.* **1995**, *123*, 1881–1887. [CrossRef]
50. Hersbach, H.; Bell, B.; Berrisford, P.; Hirahara, S.; Horányi, A.; Muñoz-Sabater, J.; Nicolas, J.; Peubey, C.; Radu, R.; Schepers, D.; et al. The ERA5 global reanalysis. *Q. J. R. Meteorol. Soc.* **2020**, *146*, 1999–2049. [CrossRef]
51. Hong, S.Y.; Noh, Y.; Dudhia, J. A new vertical diffusion package with an explicit treatment of entrainment processes. *Mon. Weather Rev.* **2006**, *134*, 2318–2341. [CrossRef]
52. Hong, S.Y.; Lim, J.O.J. The WRF single-moment 6-class microphysics scheme (WSM6). *Asia-Pac. J. Atmos. Sci.* **2006**, *42*, 129–151.
53. Iacono, M.J.; Delamere, J.S.; Mlawer, E.J.; Shephard, M.W.; Clough, S.A.; Collins, W.D. Radiative forcing by long-lived greenhouse gases: Calculations with the AER radiative transfer models. *J. Geophys. Res. Atmos.* **2008**, *113*, D13103. [CrossRef]
54. Bińkowski, M.; Donahue, J.; Dieleman, S.; Clark, A.; Elsen, E.; Casagrande, N.; Cubo, L.C.; Simonyan, K. High fidelity speech synthesis with adversarial networks. *arXiv* **2019**, arXiv:1909.11646.

55. Park, T.; Liu, M.Y.; Wang, T.C.; Zhu, J.Y. Semantic image synthesis with spatially-adaptive normalization. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019.
56. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
57. Smith, L.N. Cyclical learning rates for training neural networks. In Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, USA, 24–31 March 2017.
58. Keras. Available online: https://keras.io (accessed on 10 September 2024).
59. Song, Y.; Ermon, S. Generative modeling by estimating gradients of the data distribution. In Proceedings of the 33rd Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019.
60. Song, Y.; Ermon, S. Improved techniques for training score-based generative models. In Proceedings of the 34th Annual Conference on Neural Information Processing Systems, Online, 6–12 December 2020.
61. Wang, L.-Y.; Tan, Z.-M. Deep learning parameterization of the tropical cyclone boundary layer. *J. Adv. Model. Earth Syst.* **2023**, *15*, e2022MS003034. [CrossRef]
62. McGibbon, J.; Brenowitz, N.D.; Cheeseman, M.; Clark, S.K.; Dahm, J.P.; Davis, E.C.; Elbert, O.D.; George, R.C.; Harris, L.M.; Henn, B.; et al. fv3gfs-wrapper: A Python wrapper of the FV3GFS atmospheric model. *Geosci. Model. Dev.* **2021**, *14*, 4401–4409. [CrossRef]
63. Pietrini, R.; Paolanti, M.; Frontoni, E. Bridging Eras: Transforming Fortran legacies into Python with the power of large language models. In Proceedings of the 2024 IEEE 3rd International Conference on Computing and Machine Intelligence, Mount Pleasant, MI, USA, 16–17 March 2024.
64. Zhou, A.; Hawkins, L.; Gentine, P. Proof-of-concept: Using ChatGPT to Translate and Modernize an Earth System Model from Fortran to Python/JAX. *arXiv* **2024**, arXiv:2405.00018.
65. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.