A learning-based method for efficient large-scale sensitivity analysis and tuning of single column atmosphere model (SCAM)

Jiaxu Guo[1,7], Juepeng Zheng[2], Haohuan Fu[3,7], Yidan Xu[6,8], Wei Xue[4,7], Lanning Wang[5,7], Lin Gan[4,7], Ping Gao[4,7], Wubing Wan[4,7], Xianwei Wu[1,7], Liang Hu[1], Gaochao Xu[1], and Xilong Che[1]

[1]College of Computer Science and Technology, Jilin University, Changchun, China
[2]School of Artificial Intelligence, Sun Yat-sen University, Zhuhai, China
[3]Department of Earth System Science, Ministry of Education Key Laboratory for Earth System Modeling, Tsinghua University, Beijing, China
[4]Department of Computer Science and Technology, Tsinghua University, Beijing, China
[5]College of Global Change and Earth System Science, Beijing Normal University, Beijing, China
[6]China Reinsurance (Group) Corporation, Beijing, China
[7]National Supercomputing Center in Wuxi, Wuxi, China
[8]School of Environment and Nature Resources, Renmin University of China, Beijing, China

Correspondence: Juepeng Zheng (zhengjp8@mail.sysu.edu.cn), Ping Gao (qdgaoping@gmail.com), Liang Hu (hul@jlu.edu.cn) and Xilong Che (chexilong@jlu.edu.cn)

Abstract. The Single Column Atmospheric Model (SCAM) is an essential tool for analyzing and improving the physics schemes of Community Atmosphere Model (CAM). Although it already largely reduces the compute cost from a complete CAM, the exponentially-growing parameter space makes a combined analysis or tuning of multiple parameters difficult. In this paper, we propose a hybrid framework that combines parallel execution and a learning-based surrogate model, to support large-scale sensitivity analysis (SA) and tuning of combinations of multiple parameters. We start with a workflow (with modifications to the original SCAM) to support the execution and assembly of a large number of sampling, sensitivity analysis, and tuning tasks. By reusing the sampling instances with the variation of 11 parameters, we train a learning-based surrogate model that achieves both accuracy and efficiency (with the computational cost reduced by several orders of magnitude). The improved balance between cost and accuracy enables us to integrate learning-based grid search into the traditional optimization methods to achieve better optimization results with fewer compute cycles. Using such a hybrid framework, we explore the joint sensitivity of multi-parameter combinations to multiple cases using a set of three parameters, identify the most sensitive three-parameter combination out of eleven, and perform a tuning process that reduces the error of precipitation by 6.4% to 24.4% in different cases.

1 Introduction

Earth System Models (ESMs) are important tools to help people recognize and understand the effects of global climate change. Community Earth System Model (CESM) is one of the most popular and widely used ESMs, which includes atmosphere, ocean, land, and other components (Bacmeister et al., 2014). Of these components, the Community Atmosphere Model (CAM) (Dennis et al., 2012), plays an important role as the atmospheric component of CESM. Most of the physics parts in CAM are

described as parameterization schemes with tunable parameters that are often derived from limited measurements or theoretical assumptions. However, since CAM needs to simulate all the grids, it takes a long time and a large amount of resources to run (Zhang et al., 2018). Thus, Single Column Atmospheric Model (SCAM) (Bogenschutz et al., 2013; Gettelman et al., 2019) has been developed as a cheaper and more efficient alternative model for the purpose of tuning physics parameters (Bogenschutz et al., 2020). And in order to tune the parameters, we often need to conduct a large number of simulated experiments. This will result in significant computational costs. Meanwhile, SCAM only needs to simulate one single column, and only one process is required for each run of one case to complete the simulation. As a result, SCAM becomes a natural tool for studying how the parameters would affect the uncertainty in the modeling results, and the use of SCAM for large-scale experiments is more practicable due to its advantage of lower requirements for computing resources.

Climate models are among some of the most complex models, for a model, we can abstract it as a function with numerous independent and dependent variables, and there exists uncertainty between them. In research, identifying independent variables that significantly affect the dependent variable can help to quickly understand the relationship between them. Sensitivity analysis (SA) is an important method used to achieve this purpose. (Saltelli et al., 2010). A rich set of numerical and statistical methods have been developed over the years to study the uncertainty in models in many different domains, ranging from natural sciences, to engineering, and risk management in finance and social sciences (Saltelli et al., 2008). SA of climate models generally involves two steps: generating representative samples with different values of parameters using a specific sampling method; and explore and identify the sensitivity metrics between the model output and the parameters to study. Typical approaches include: the Morris One-At-a-Time (MOAT) method that uses the Morris sampling scheme (Morris, 1991), which generates samples uniformly and has a good compute efficiency. To go with it, Morris SA can give the individual sensitivity of each parameter, including their interaction sensitivity. However, this is not intuitive enough if the user wants to know directly from a combined perspective which set of parameters has the most significant effect on the results. and the Sobol method that uses the probabilistic framework and adopts the decomposition of variance of the output to describe the sensitivities (Sobol, 1993). In contrast to the Morris method, the variance-based Sobol method generally requires a lot more samples to achieve a good coverage of the space (Sobol', 1967; Saltelli, 2002), but has the advantage of being capable to study the interaction effects between different parameters. Other similar ideas to achieve a good representation of the sample space with a quasi-random sequence include the quasi-Monte Carlo (QMC) (Caflisch, 1998) and the Latin hypercube (LHC) (McKay et al., 2000) sampling methods.

After we have determined the combination of parameters to be tuned, we can then tune them to improve the performance of the model. With a general goal to achieve modeling results as close to the observations as possible, we can apply different optimization methods, such as the Genetic algorithm (GE) (Mitchell, 1996), Differential Evolution (DE) (Storn and Price, 1997), Particle Swarm Optimization (PSO) (Kennedy and Eberhart, 2002), etc., to identify the most suitable set of parameters.

Continuous efforts have been put into the tunable parameters in climate models, especially for the their physics parameterization schemes (Yang et al., 2013; Guo et al., 2015; Pathak et al., 2022). Yang et al. (2013) analysed the sensitivity of nine parameters in the ZM deep convection scenario for CAM5 and used the Simulated Stochastic Approximation Annealing method to optimize the precipitation performance in different regions by zoning. Zou et al. (2014) conducted a sensitivity anal-

ysis for seven parameters in the MIT-Emanuel cumulus parameterization scheme in RegCM3. The precipitation optimization process for the CORDEX East Asia domain was carried out using the Multiple Very Fast Simulated Annealing method.

For all the stages mentioned above, the compute cost of running the model become a major constraining factor that stop us from exploring more samples and identifying more optimal solutions. People sometimes use surrogates (such as the generalized linear model (GLM) (Nelder and Wedderburn, 1972)) instead of the actual model to further reduce the compute cost. For example, the study of the sensitivity of simulated shallow cumulus and stratocumulus clouds to the tunable parameters of the subnormal uniform cloud layer (CLUBB) (Guo et al., 2015) investigated the sensitivity of 16 specific parameters, using the QMC sampling method and GLM as a surrogate, with experiments on three different cases (BOMEX, RICO, and DYCOMS-II RF01). A key problem to consider for the SA stage is to achieve a balance between the accuracy and the economy of compute (Saltelli et al., 2008). Another study (Pathak et al., 2022) used the single-column case ARM97 to explore 8 parameters related to the cloud processes, with Sobol as the sampling method, and spectral projection (SP) and basis pursuit denoising (BPDN) as the surrogate model. In an ideal case, a more thorough study of the parameters that can provide more concrete guidance for the parameter selection in CAM, would require a joint SA and tuning of different single-column cases, as well as combinational study of the most sensitive parameters. There are toolkits such as PSUADE (Gan et al., 2014), DAKOTA (Cravero et al., 2020) and STATA (Harada, 2012) that can implement SA and tuning. However, such a joint and combined exploration that involve multiple parameters and multiple cases would increase the space to explore in an exponential manner, and make the SA and tuning almost an impossible job.

In this paper, to facilitate researchers to better utilize SCAM, and to support a more efficient and convenient parameter tuning for the physical schemes in SCAM, we propose a learning-based method for efficient large-scale SA and tuning. We start with a scientific workflow (with modifications to the original SCAM) to support the execution and assembly of a large number of sampling, sensitivity analysis, and tuning tasks, which can support parallel execution of hundreds to thousands of parallel instances, and highly-efficient exploration of combinations of multiple parameters. In contrast to the packages mentioned above, from a method perspective, we add the comparison of the new SA methods in recent years, which haven't been fully supported by all the packages above. In terms of training surrogate models based on regression analysis, our proposed workflow uses more types of neural networks and supports the adaptive selection of the best performing network to train the surrogate model.

Therefore, in summary, we mainly make the following contributions. (1) We enable a scientific workflow (with modifications to the original SCAM) to support the configuration of parameters through the $namelist$, and execution and assembly of a large number of sampling, sensitivity analysis, and tuning tasks. (2) By reusing the sampling instances in the sampling stage, we innovatively introduce five different regression methods to train the surrogate model, and select the ResNet with the best results as the final test solution. The use of surrogate model is easier to obtain results and more tractable, which enables us to do sensitivity analysis of combinations of multiple parameters in a more efficient way. (3) We also integrate NN-based grid search into the traditional optimization methods. With a better capability to jump out of local optimums, we can achieve better optimization results with fewer compute cycles.

Table 1. List of single column atmosphere model cases tested.

Case	Full name	Lat	Lon	Date	Type
ARM95	ARM Southern Great Plains	36	-97	July 1995	Land convection
ARM97	ARM Southern Great Plains	36	-97	June 1997	Land convection
GATEIII	GATE Phase III	9	-24	August 1974	Tropical convection
TOGAII	Tropical W. Pacific Convection	-12	131	December 1992	Tropical convection
TWP06	Tropical Ocean Global Atmosphere	-2	154	January 2006	Tropical convection

Using our proposed learning-based surrogate models, we perform an extensive set of SA and tuning experiments for five cases of SCAM (both independently and jointly), targeting the precipitation performance. Besides SA analysis that provides sensitivity evaluation of each single parameter, we are also able to study the sensitivity of a combination of three, four or even five arbitrary parameters. At the tuning stage, our improved optimization scheme (targeting the same parameters) leads to 24.4% more accurate output of precipitation compared to control experiments, with a more than 50% saving in compute cost compared to using only the optimization algorithm. At the end of the paper, we also explore the relationship between several cases, and show the output distribution trend of the simulation results in the same parameter space with a 3D figure.This also suggests potential improvements for future location-based parameter tuning.

2 Enabling a workflow of SA and parameter tuning

2.1 Model description

This paper focuses on the single column model of the atmospheric model CAM5, i.e. SCAM5 (Bogenschutz et al., 2012), extracted from CESM version 1.2.2, one of the two versions that are efficiently supported on the Sunway TaihuLight Supercomputer (Fu et al., 2016). Our research of this paper is mainly based on five typical cases in SCAM5, as shown in Table 1. Among the five cases, two cases are located in the Southern Great Plains, which mainly study land convection. The other three cases are located in the tropics and mainly study tropical convection (Thompson et al., 1979; Webster and Lukas, 1992; May et al., 2008).

As shown in Table 2, the number of observations included in the IOP (Intensive Observation Periods, Gettelman et al. (2019)) file varies from case to case. In order to explore a joint parametric sensitivity analysis and tuning across all the five cases, we pick the intersection of the data owned by these cases, the total precipitation (PRECT), which is also one of the most important outputs of the model, as the main research subject. The parameters listed in Table 3 are the main study targets (Qian et al., 2015) in this paper and the ones tested in the workflow. The parameters are selected from the ZM deep convection scheme (Zhang, 1995), the UW shallow convection scheme (Park, 2014), and cloud fraction (Gettelman et al., 2008). In the

Table 2. Observed variables included in the IOP file of each case.

Variable	Description	ARM95	ARM97	GATEIII	TOGAII	TWP06
Prec	Precipitation rate	✓	✓	✓	✓	✓
totcld	Total cloud	✓	✓	-	-	✓
shflx	Surface sensible heat flux	✓	✓	-	✓	✓
lhflx	Surface latent heat flux	✓	✓	-	✓	✓
U	Eastward wind speed	✓	✓	✓	✓	✓
V	Northward wind speed	✓	✓	✓	✓	✓
Q	W.V. Mixing Ratio	✓	✓	✓	✓	✓
T	Temperature	✓	✓	✓	✓	✓
omega	vertical motion	✓	✓	✓	✓	✓
windsrf	Surface wind speed	✓	✓	-	✓	✓
REHUM	Relative humidity	-	-	✓	-	✓
CAPE	Convective available potential energy	-	-	✓	-	-

Table 3. List of parameters in the framework that can be tuned and applied to the experiment.

Abbr.	Name	Description	Low Range	Default	High Range	Category
pz1	c0_lnd	Deep convection precipitation efficiency over land	0.00295	0.0059	0.00885	ZM Deep convection
pz2	c0_ocn	Deep convection precipitation efficiency over ocean	0.0225	0.045	0.0675	ZM Deep convection
pz3	ke	Evaporation efficiency of precipitation	5e-7	1e-6	1.5e-6	ZM Deep convection
pz4	tau	Time scale for consumption rate deep CAPE	1800	3600	5400	ZM Deep convection
pz5	capelmt	Threshold value for CAPE	35	70	105	ZM Deep convection
pz6	alfa	Maximum cloud downdraft mass flux fraction	0.05	0.1	0.15	ZM Deep convection
pu1	rpen	Penetrative updraft entrainment efficiency	2.5	5.0	7.5	UW Shallow convection
pu2	kevp	Evaporative efficiency	1e-6	2e-6	3e-6	UW Shallow convection
pu3	rkm	Updraft lateral mixing efficiency	7	14	21	UW Shallow convection
pc1	rhminh	Threshold relative humidity for stratiform high clouds	0.7	0.8	0.9	Cloud fraction
pc2	rhminl	Threshold relative humidity for stratiform low clouds	0.7975	0.8975	0.9975	Cloud fraction

experiments in this paper, the lower and upper bounds of each parameter are 50% and 150% of the default value, respectively. For parameters with physical constraints, such as rhminh and rhminl, the values are ratios, so they are not more than 1.

In the original version of SCAM, only some of the parameters to be studied are tunable, while the rest are hard-coded in the model. To improve the flexibility of the model so that the 11 parameters we want to study are tunable, we have modified the source code of the model, to support the tuning and study of a wider range of parameters. The corresponding Fortran source code, as well as the XML documentation are also modified accordingly. In addition, the programs running on Sunway

TaihuLight needed to be recompiled due to the adoption of a different architecture. We recompiled using a compiler compatible with Sunway after making the above improvements, to enable execution of a larger number of concurrent instances on the Sunway supercomputer. With these upgrades, all parameters in Table 3 are supported for tuning.

2.2 The workflow of sampling, SA and parameter tuning

The submission of assignments and the collection of results are important issues when carrying out a large number of model experiments at the same time. Prior to conducting the experiments, the user is often presented with a broad set of boundaries of the parameters to be tuned, and the specific configuration of each experiment has to be decided in detail according to these ranges of values. After a large number of experiments have been completed, as the output of SCAM is stored in binary files in NetCDF format, the precipitation variables we want to study need to be extracted from a large number of output files in order to proceed to the next step. It is therefore necessary to provide the researcher with an automated experiment-diagnosis process. In general, which parameters to tune and how to tune them are questions that deserve our attention.

Based on the above needs, we have designed the SCAM parameter sampling, SA, tuning and analysis workflow. We integrate the collection and processing script for the post-sampling results. It supports a fully-automated parameter tuning and diagnostic analysis process, a large number of concurrent model tests, and the search for the best combination of parameter values for SCAM performance within a given parameter space. Also, with the help of the training surrogate model, more parameter fetches can be tested in less time. The simulation results of the real model will be used as validation. This will further accelerate the degree of automation of scientific workflows and thus accelerate the conduct of research in this area of the earth system models.

The overview of the whole scientific workflow is shown in Figure 1. In order to make full use of computing resources and complete the sampling process as soon as resources allow, the proposed method supports parallel sampling processes, as shown in Figure 2. Since SCAM is a single-process task and the computation time per execution is also short, it is feasible to execute a large batch of SCAM instances during the sampling stage.

Specifically for the application scenario in this paper, the execution process of the workflow is as follows.

1. Sampling and data collection (shown as Part I in Figure 1): In this part, our tool generates the sequence of samples to investigate in the sampling step. Our tool currently supports the Sobol sequence, which can later be used by the Sobol, Delta (Plischke et al., 2013), HDMR (Li et al., 2010), and RBD (Plischke et al., 2013) SA methods, and the Morris sampling sequence, which can later be used by the MOAT method (Morris, 1991). Users are suggested to adjust the size of the sequence according to the currently available computational resources. As the results of this step will be used as the training set for generating the surrogate model, users are encouraged to run a large batch when parallel resources are available, so as to improve the performance of the resulting surrogate model. The process of launching the parallel cases and collecting the results is handled by the SCAM launcher and collector.

2. Surrogate model training and sensitivity analysis (shown as Part II in Figure 1): Based on the sampling results from the Morris or the Sobol sequence, we integrate existing methods, such as MOAT, Sobol, Delta, HDMR, and RBD to achieve their individual evaluations of each single parameter's sensitivity, as well as a comparison result of these methods. We

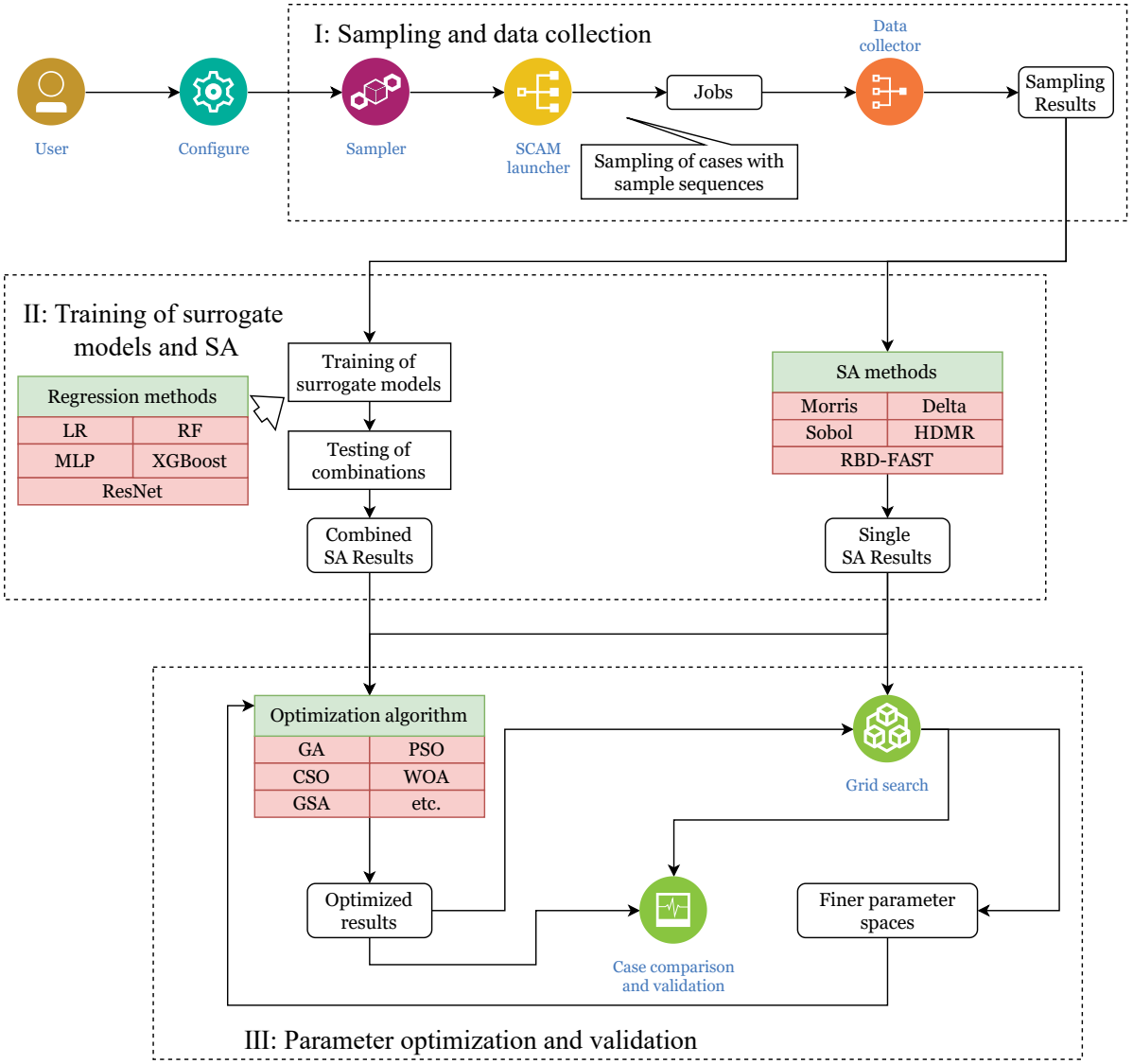

Figure 1. The overall workflow of the proposed method. Part I performs the sampling and the collection of results of parallel instances. Part II uses traditional SA methods to derive sensitivities of individual parameters, and at the same time, reuse the samples to derive learning-based surrogate models. Combining the surrogate models, we can then also perform joint sensitivity analysis of a set of parameters. Guided by the SA results from Part II, Part III performs parameter tuning, also with the surrogate models. SCAM launcher, the data collector and the jobs therein represent the batch execution of the SCAM algorithm, which is described in more detail in Figure 2.

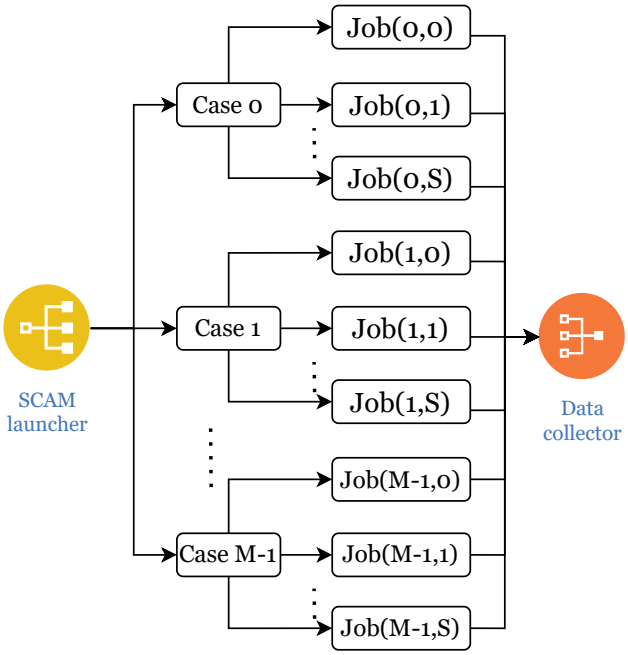

Figure 2. Detailed parallelism schematic when running a large number of instances of multiple cases simultaneously. In this process the required SCAM tasks are launched simultaneously by the launcher and the results of their runs are collected by the data collector.

also use the sampling results of the Saltelli sequence and the Morris sequence, to train regression including neural network (NN) based surrogate models. In this section, different regression methods are used to compare their fits and the best fitting method to train the final surrogate model. With the efficiency to project a result in seconds rather than minutes, we can apply it for evaluation of sensitivities of a combination of multiple parameters.

3. Parameter tuning and validation (shown as Part III in Figure 1): With the NN-based surrogate model to cover expanded search space with less time, we also propose an optimization method that combines grid searches by the surrogate model, which achieves better results with less compute time. The results of parameter tuning are then validated through running of real SCAM models. In addition, at the very end, we perform a comparison on the optimization results between the joint optimization across five cases and the independent optimization of the five cases. Results demonstrate the different and the correlation of different cases, and the potential of performing grid-specific tuning in the future.

3 Methodology

3.1 Sampling of SCAM

As an important preprocedure, the sampling provides the basis for analysis of SA. It will generate a sequence of changing inputs and parameters to observe the corresponding change in the output. The different mathematical approach that we take to perform sampling would certainly affect the features that can be captured from the system.

In our proposed workflow, we integrate both Morris and Saltelli for the sampling step in our tuning workflow, as both of them are still used in many climate model related SA studies (Pathak et al., 2022). The Morris sampling drives the MOAT SA module afterwards, while the Saltelli sampling drives four different SA modules (Sobol, Delta, HDMR, and RBD-FAST) shown in Table 4. The number of samples generated by these two sampling methods follows the following two equations:

$$S_{Morris} = N_{Morris} \times (D+1) \tag{1}$$

$$S_{Saltelli} = N_{Saltelli} \times (2D+2) \tag{2}$$

Here, S is the number of samples generated by the two methods, N is the coefficient, and D is the dimension of the problem to be solved, that is, the number of parameters. Since $D = 11$ in this paper, the number of samples generated should be a multiple of 12 for Morris. For Saltelli, the sample number needs to be a multiple of 24. Taking into account the computational resources we have, we decide another S=768, where N_{Morris} is 64 and $N_{Saltelli}$ is 32.

After sampling, we will conduct a preliminary analysis of the sampled results to find out the proportion distribution in which the output results are better than the control trials under different values of each parameter. RMSE (Root mean square error) will be used to measure the error between the output and the observed values and is defined as follows.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{3}$$

Where $\hat{y}_i$ is the output at the ith time step of the current sample, while y_i is the observation at the corresponding time step.

3.2 Training a learning-based surrogate model

Surrogate models are an important tool to speed up our large-scale parametric experiments. It can replace the running process of the original model, thus saving computing resources. The essence of training surrogate models for SCAM is a regression analysis problem. In this paper, we introduce the following regression methods to generate surrogate models. These include Linear regression (LR,Yan et al. (2015)), but also ensemble learning methods such as Random forest (RF, Breiman (2001)) and eXtreme Gradient Boosting (XGBoost, XingFen et al. (2018)). Meanwhile, we also incorporate methods that use neural networks, such as Multilayer Perceptron (MLP, Tang et al. (2016)) and Residual Network (ResNet, He et al. (2016); Shi et al. (2022)). Compared to other networks, ResNet has the advantage of using less pooling. Meanwhile, most neural networks have

Table 4. SA methods integrated in the workflow and used as a cross-reference to the proposed method.

Name of method	Abbr.	Reference
Morris sensitivity analysis	Morris	Morris (1991)
Delta moment-independent measure	Delta	Plischke et al. (2013)
Sobol' sensitivity analysis	Sobol	Sobol (1993)
High-dimensional model representation	HDMR	Li et al. (2010)
Random balance designs fourier amplitude sensitivity test	RBD-FAST	Goffart et al. (2015)

fully connected layers before the output layer, so they have the disadvantage of losing part of the information of the input data as it passes through these layers. Therefore, ResNet has the advantage of retaining the complete information of the input data. The reason is that it does not have a fully connected layer other than the output layer. ResNet18 with a depth of 18 layers is used in our proposed workflow.

In order to combine the various regression analysis methods described above, in this paper we design an adaptive scheme to determine the method that best captures the non-linear characteristics of the original model, and thus obtain the most appropriate surrogate model. To keep an acceptable level of accuracy of surrogate models, we choose to train respective models for each different SCAM case (the underlying assumption is that the model should learn the different patterns in different SCAM case locations). As the Saltelli sequence of samples have a good representation of the entire parameter space, we expect the model to perform generally well in different parameter combinations. In addition, for the hyper-parameters in the process of training the surrogate model, ablation experiments will also be performed to determine the most appropriate hyper-parameters, leading to better training.

3.3 Sensitivity analysis for a single parameter and combinations of parameters (enabled by the NN-based surrogate model)

The SA methods, similar to the climate model itself, have their corresponding uncertainties. The SA methods provide a best estimation of each parameter's sensitivity, according to their respective analytical principles. Therefore, each different method might have its advantages and disadvantages in different ranges of the parameter values.

As a result, in our workflow shown in Figure 1, we choose to integrate multiple SA methods, including the ones that can be built on the Sobol sequence, such as Delta, HDMR, and RBD-FAST, and the Morrison method, which is still used often for climate models, due to its efficiency advantage. The integration of multiple methods enables us to evaluate the uncertainty of different SA methods. As the sensitivity values calculated by the different methods are of different orders of magnitude, these sensitivities have been normalised for comparison purposes.

The parameters may interact with each other, and the effects of multiple parameters on the simulation results may be superimposed. Therefore, tuning multiple parameters generally have a more significant effect than tuning a single parameter.

With the help of commonly used sensitivity analysis methods mentioned above, it is easy to obtain the sensitivity of individual parameters. However, in cases where we need to identify and tune a set of parameters in a combined way, both the SA and the tuning task would involve a significantly improved level of compute resources.

Here, we take an example of analyzing a combination of M different parameters, and adopt a grid-based sampling approach to explore the sensitivity. Assuming that we divide the possible value ranges of each parameter into L levels, to cover a complete grid with possible changes of all M parameters, we need to explore L^M different combinations. Thus, in the sampling stage, we would need to apply the above L^M parameter values to SCAM to obtain the same number of simulation results, and find the result with the largest difference from the outputs from default value and its corresponding parameter value combination. In this scenario, letting N be the number of simulations required to carry out a set of tests, we have:

$$N_{MPP} = C \times \binom{D}{p} \times L^p \tag{4}$$

where C represents the number of cases to be tuned by user, D represents the total number of parameters used for study, p represents the number of parameters we perturb in each test, and L represents the number of levels we cover within the value range of each parameter.

Combining Equation (4), when we do a combined study to identify and tune a most sensitive set of three different parameters, if $L = 10$, the number of tasks to run is already at the level of multiple thousand. If we expand the case to four or five parameters, the number of tasks would grow rapidly to tens of millions of runs. Even for the SCAM model, although the computational cost of a single run is not that large, such a combined cost becomes impractical.

An obvious benefit of using a surrogate model for training is that it is very fast to compute. Since the model has been trained, the time taken to output the surrogate results is much shorter. Therefore, we can try as many combinations of parameters as possible with less computational resources. By applying the trained surrogate model, we will test the maximum fluctuation in the output of each case when the number of parameters adjusted at the same time is different. This way, we can determine the number of parameters in the combination, taking into account the tuning effect and the amount of computation.

Here we look at which combinations of three parameters lead to the most significance in output while taking into account the computational overhead (using PRECT as an example). The utilization of the learning-based surrogate model, on the other hand, provides a more feasible solution to this problem. Since the surrogate model has a very short running time and almost does not need to consider the problem of job queuing time, it makes it possible to complete a large scale parameter experiment in a short time.

3.4 Parameter tuning enhanced with NN-based surrogate model and grid search

After the most sensitive sets of parameters have been found, the stage of optimization will begin.

In the optimization stage, on the basis of combining the existing optimization algorithms (such as GA, PSO, WOA, etc.), we propose an enhanced optimization method based on grid search. By using grid search, the search scope can be further narrowed before and during the call to the existing optimization algorithm, so as to improve the efficiency of searching for better solutions. The process is as follows.

1. Determine the overall parameter space range for conducting the grid search based on the initial setup of the experiments.

2. The simulation output values corresponding to these grids are calculated in the surrogate model. Then the best performing points are selected and the new finer-grained search space is determined based on the aggregation of these points.

3. The optimization algorithm is carried out in the newly defined space and depending on the available computational resources it is decided whether SCAM is invoked to compute in each iteration round. Meanwhile, the final results will also be substituted into SCAM for verification.

4. If the same optimization result is obtained in three consecutive iterations, or less than a certain threshold of improvement compared to the last, the grid search operation of the first two steps is performed again, thus further reducing the search space, as can be seen from Algorithm 1. In this step, a new parameter space will be constructed with the position of the current best point at the centre and the distance of each parameter from the last best point as the radius.

5. After the refinement of the parameter search space, if the result remains the same, the optimization process can be considered to be over and SCAM can be run again using the derived parameters to verify the optimization results.

Algorithm 1 Optimization process combined with grid search.

if $Y_i - Y_{i-1} < \epsilon$ **and** $Y_{i-1} - Y_{i-2} < \epsilon$ **then**
  for $j = 1$ **to** $j = p$ **do**
    if $x_{j,i} - |x_{j,i} - x_{j,i-3}| > X_{j,min}$ **then**
      $X^*_{j,min} = x_{j,i} - |x_{j,i} - x_{j,i-3}|$
    else
      $X^*_{j,min} = X_{j,min}$
    end if
    if $x_{j,i} + |x_{j,i} - x_{j,i-3}| < X_{j,max}$ **then**
      $X^*_{j,max} = x_{j,i} + |x_{j,i} - x_{j,i-3}|$
    else
      $X^*_{j,max} = X_{j,max}$
    end if
  end for
end if
Grid seach.
return $x_{j,i+1}$ **and** Y_{i+1}

In Algorithm 1, Y_i is the value of the function in the ith iteration, ϵ is the threshold at which the results converge, p is the total number of parameters to be tuned, $x_{j,i}$ is the coordinates of x_j in the ith iteration, X_j denotes the maximum and minimum values initially set for x_j, and X^*_j denotes the maximum and minimum values of x_j in the upcoming grid search. Note that if

one of these coordinates is outside the initial parameter space, the original parameter space boundary will be used as the new boundary.

3.5 Case correlation analysis based on Pearson correlation coefficient

Following the end of the tuning process, we further perform a comparison study among different cases, to derive valuable insights that would potentially lead to a physics module design that can accommodate the different features in different locations. In our proposed method, the Pearson correlation coefficient method is used to measure the similarity between two cases (Schober et al., 2018). This coefficient is defined as the quotient of the covariance and standard deviation between two variables. By analyzing the similarity between the 'optimal' parameters of each case, these cases can be clustered. In this process, we can explore whether for similar types of case, they may have similar responses to various parameters. This will also help us choose better parameter combinations when analyzing other cases, and even regional and global models, so as to achieve better simulations. The implementation is shown below.

1. Search the sensitive parameter sets for each output variable in each case.
2. For each case, the occurrence times of each parameter in the sensitive parameter combination are accumulated to obtain a variable of M dimension, where M is the number of parameters. For example, for a combination of three parameters, the vector can be expressed as $(Value1, Value2, Value3)$.
3. The correlation between vectors of each case can be computed by the Pearson correlation coefficient method.

By the above procedure, we can obtain the similarity of the set of sensitive parameters among the cases. The correlation between different cases can also be analyzed for the case of finding the optimal values in the same parameter space. After each case has found its optimal parameter values, this set of parameter values can also be represented by vectors. The same approach can be used to analyze the correlation between these vectors.

4 Experimental results

4.1 Sampling of the SCAM cases

As mentioned earlier, the sampling scheme, which determines the set of points to represent the entire parameter space, is of essential importance for the following sensitivity analysis and parameter tuning steps. Considering the compatibility between the sampling methods and the SA methods, our platform includes both the Morris (driving the MOAT SA method) and the Sobol sampling scheme (driving the Sobol, Delta, HDMR, and RBD-FAST SA methods). We use a total of 7,680 samples, with 1,536 samples for each of the five SCAM case. Of these, half (i.e. 768 samples) are used for MOAT, while the remaining half are used for the Sobol sequence. In this part of the experiment, we use the job parallelism mechanism mentioned earlier in the text to execute these sample cases.

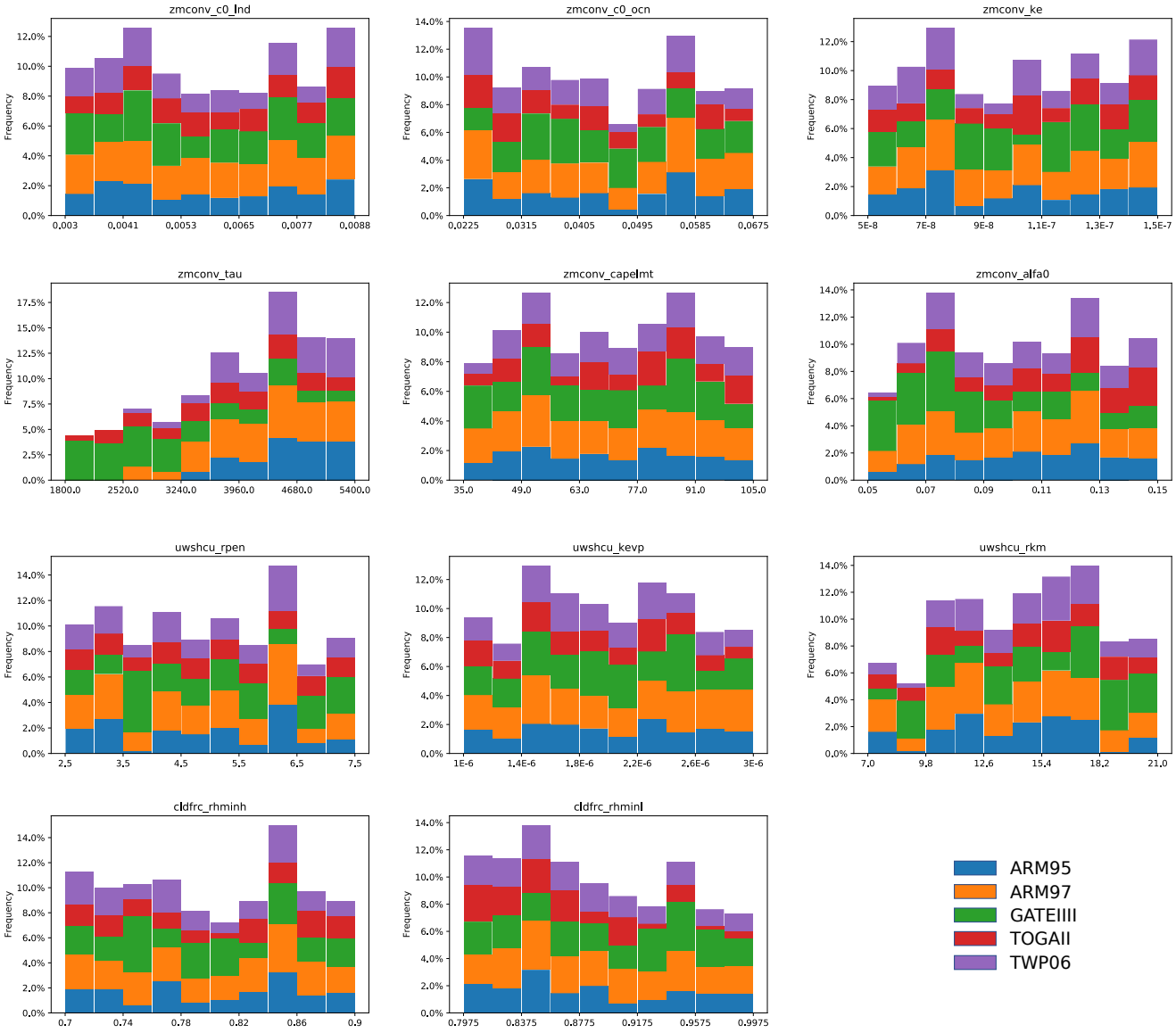

Figure 3. The proportion of sampled results where the output is better than the control experiments (i.e. experiments using default values). The x-axis shows the range of values for each parameter.

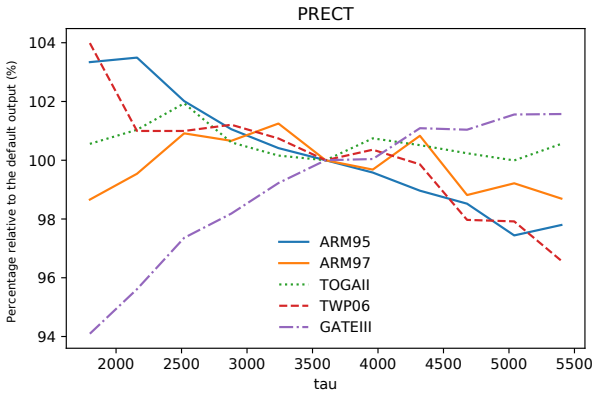

Figure 4. Magnitude of output variation of the parameter tau to precipitation related output indicators in five cases. The solid line shows the land convection case and the dashed line shows the tropical convection case. The single parameter perturbation method is used for tuning.

In our sampling, SA, and tuning study, we focus on the total precipitation output (PRECT). Figure 3 reflects the proportion of PRECT that outperforms the control experiments when each parameter is tuned from low to high in the range of values taken. From the results, we can see that for different cases there are differences in their response to parameter changes. Of particular interest is the distribution of pz4(tau). We can clearly see that for GATEIII, when the value of tau is small, there are more good outputs; However, for the other cases, it is that when the value of tau is large, there are more outputs that perform well. It can also be seen from the proportion that tuning tau can lead to more good outputs. This is also in line with the results of our later experiments.

To further verify the conclusions here, we performed a single-parameter perturbation test on tau while keeping the other parameters as default values. Figure 4 shows the change of PRECT for different pz4 (tau) values in different cases. Even for the same parameter and the same parameter space, the trend of their effects on PRECT varies, and even in opposite directions. For example, increasing tau tends to increase total precipitation in GATEIII, while in the other cases it brings the opposite result. Different from other cases, this case is a ocean case and is located in the Atlantic Ocean.

4.2 Training learning-based models for parameter tuning

After obtaining the sampling results, we can train the surrogate model for each of the five cases using the method presented in Section 3.2. The samples generated by the two sampling methods are combined to form the dataset on which we train our surrogate model. In other words, each case has 1536 samples, and all cases have a total of 7680 samples to participate in the training. We split the training and test set in an 8:2 ratio and use RMSE as the loss function during training. For the hyper-parameters in training, we also conduct ablation experiments to achieve better training results. The final hyper-parameters used to train the neural network are shown in Table 5.

We trained surrogate models using five regression methods and used RMSE as a loss function to measure the training error. A comparison of the various methods is shown in Table 6. It can be seen that ResNet has the best performance on the five

Table 5. Hyper-parameters including learning rate and batch size from ablation experiments.

Case	Learning rate	Batch size
MLP	0.01	32
ResNet	0.01	32

cases, and its error on the test set is lower than that of the other methods. Therefore, we will use the surrogate model trained by ResNet for the following experiments.

Table 6. Comparison of errors during training using various methods. RMSE is used as the loss function. The errors are from test sets.

Case	LR	RF	MLP	XGBoost	ResNet
ARM95	0.235	0.197	0.751	0.184	**0.038**
ARM97	0.188	0.158	0.555	0.136	**0.045**
GATEIII	0.646	0.432	1.335	0.538	**0.137**
TOGAII	0.179	0.112	0.223	0.118	**0.041**
TWP06	0.344	0.220	0.594	0.220	**0.040**

4.3 Single-parameter sensitivity analysis across different cases

After sampling, five SA methods listed in Table 4 are used to compute the sensitivity of these parameters to the PRECT output. In addition, for the surrogate models we trained, we also adopted the single-parameter perturbation method to test their sensitivities. Heat maps are used to characterize the sensitivities of each parameter. As can be seen from Figure 5, there are differences in the results obtained from different SA methods.

From the results, it can be seen that the response of ARM95 and ARM97 to each parameter is basically the same, only a few quantitative differences exist in parameters such as pz6(alfa), pu2(kevp) and so on. Moreover, both cases are sensitive to pz4(tau), which significantly outweighs the other parameters. For GATEIII, we can see that pz4(tau) similarly has a significant influence on it, but pz6(alfa) and pu3(krm) also have a large influence. In addition, the two SA methods of Morris and Sobol simultaneously show that pc2(rhminl) also has a non-small effect. For TOGAII, it is interesting to find that tau and rhminl have similarly significant effects, but it is worth noting that the differences between the individual parameters are not as wide as in the other cases. However, tau still has a far greater influence than other parameters on TWP06, which is similar to ARM95/97. But the difference is that pz2(c0_ocn) affects it a little bit more. This is also consistent with its position. To sum up, tau has a very significant significance for all cases.

We also perform a comparison of sensitivity analysis results between different SA methods. To demonstrate the applicability of using our learning-based surrogate model in the SA process, we also show the SA results by using our trained learning-based surrogate models. For comparison across different methods, the most sensitive parameter obtained by each method is

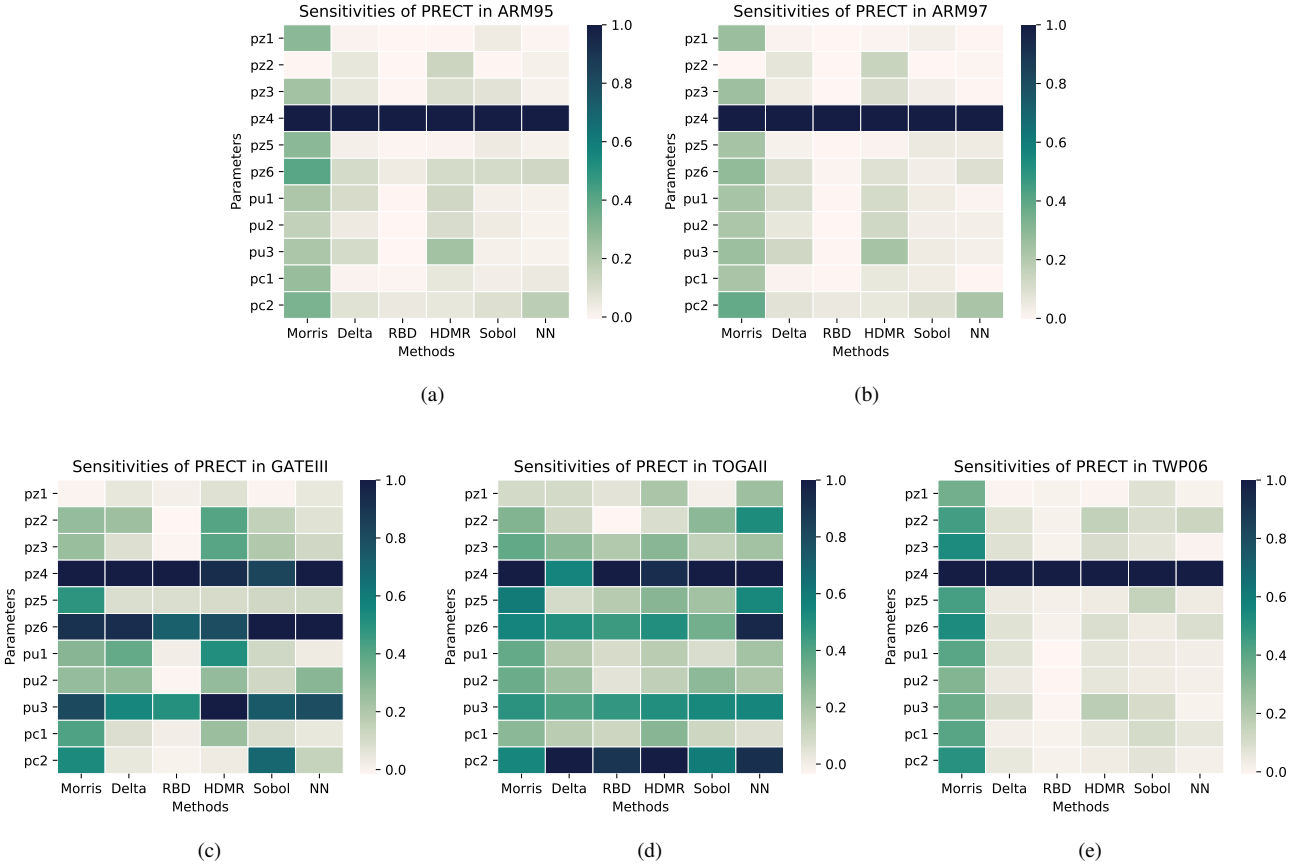

Figure 5. The comparison of the sensitivity of each parameter to PRECT comes from different analysis methods (including NN) in five cases. Sensitivity results were normalised. (a)ARM95, (b)ARM97, (c)GATEIII, (d)TOGAII, (e)TWP06. The qualitative and quantitative similarities and differences in the sensitivity of each case to each parameter are reflected.

tau, but there are differences in the sensitivity of other parameters to some extent. For example, Delta and HDMR yield slightly greater sensitivity for the other parameters. In addition, Morris concluded that the sensitivity of these parameters is significantly greater than that of the other methods, that is, the gap between tau and the other parameters is smaller. This may be related to the sampling method, as Morris needs to use a different sequence of samples.

4.4 Joint multi parameter sensitivity analysis using learning-based models

As mentioned earlier, by using the learning-based surrogate models, we now have the capability to explore the sensitivity of multi-parameter combinations. How many parameters is it reasonable to tune at the same time? Figure 6 illustrates the maximum PRECT output fluctuation that can be brought by different parameter combination sizes. As the number of parameters to be tuned simultaneously increases, the fluctuations that can be brought about are also greater. However, the computation

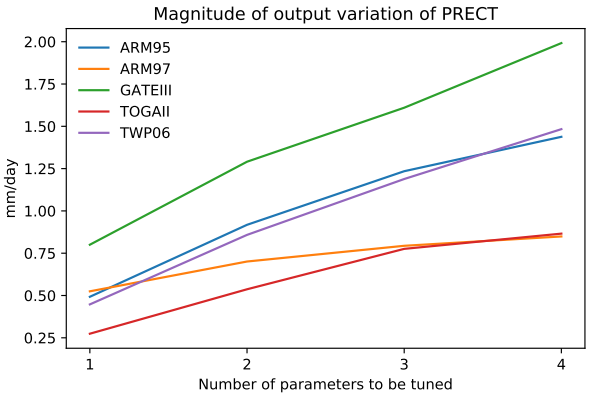

Figure 6. The maximum fluctuation in the output of each case when perturbing one to four parameters. The Y-axis represents the largest gap between the maximum and minimum values of precipitation output that can occur during the simulation period.

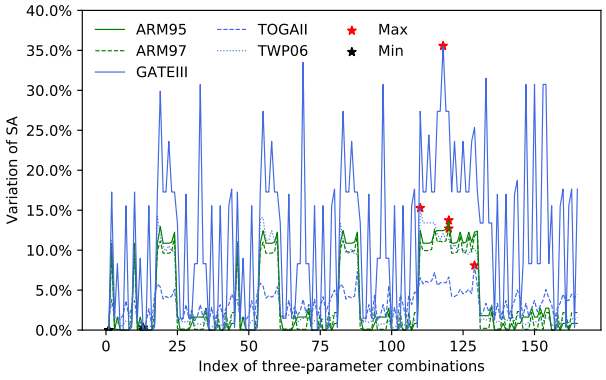

Figure 7. The variation of SA among different three-parameter combinations in five different cases. The same index indicates the same combination of parameters.

amount increases exponentially during the test. Therefore, although the effect of tuning four parameters is better than tuning three parameters, considering the tuning effect and the computational overhead, we decided to use the experiment of tuning three parameters as a demonstration. In addition, we also note that for the ARM97 and TOGAII, their precipitation response for four parameters is even smaller than the response of the GATEIII for tuning one parameter. An important reason for this is that these two cases themselves have smaller precipitation values than GATEIII, whereas the Figure 6 uses the absolute values of precipitation.

As our study investigates 11 parameters related to PRECT, there are a total of 165 three-parameter combinations. It would be difficult to test all these combinations using the original SCAM model, due to the high computational overhead. However, with the help of the surrogate model, we can instead accomplish these tests in less than a minute.

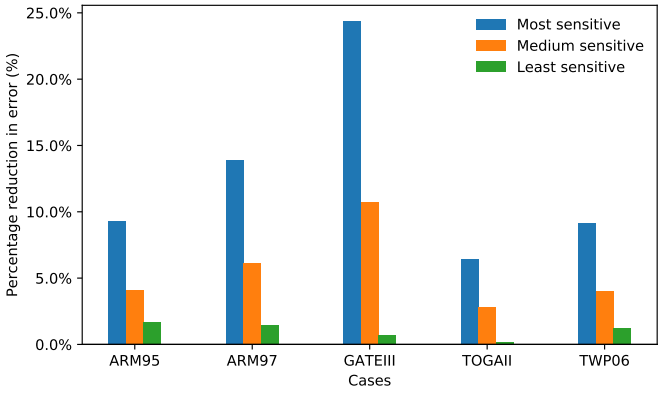

Figure 8. Comparison of different combinations of parameters under the same optimization algorithm after SA filtering and sorting. Three representative parameter combinations are selected for each case. It can be seen that a more sensitive parameter combination can lead to a better tuning effect.

For each case, the magnitude of the change in output has been evaluated for all possible combinations of the three parameters in turn, with the help of the surrogate model trained. This is shown in Figure 7. It can be seen from this that there are significant differences in the magnitude of output variation that can be brought about by different combinations of parameters. The maximum variation relative to the default test is up to 35%. The comparison with the previous tests also fully illustrates that the combined tuning effect of the three parameters is more significant.

In order to show the impact of different parameter combinations on the optimization results, the first, middle and last ranked parameter combinations are selected and combined with the same algorithm to carry out the optimization. In the scenario where the three parameters are tuned, there are 165 combinations of parameters that can be tuned, so the combination ranked 83rd is chosen as the one in the middle of the ranking. These combinations are listed in Table 7. Whale optimization algorithm(WOA, Mirjalili and Lewis (2016)) has been chosen as the method in this stage. Set the number of whales in the optimization algorithm to 32 and the maximum number of loop iterations to 16.

RMSE is also selected as the metric of optimization effect. As can be seen from Figure 8, the higher ranked parameter combinations do have smaller errors for the same optimization conditions. Especially for the case TOGAII, an inappropriate combination of parameters can hardly lead to better results, which is therefore a good example of the need to choose the right combination of parameters as the object to be optimized. Next, we will explore further the sensitive parameter combinations.

4.5 Joint optimization for SCAM cases combined with grid search and learning-based models

To validate a more efficient search method, we applied typical optimization algorithms and a grid search that combines them to find the optimum.

In order to verify the effectiveness of the method, we first choose Sphere function (Karaboga and Basturk, 2008) as an example to carry out an optimization test. The trend of the meaning average error during the test is shown in the Figure 9. It

Table 7. The most, middle and last sensitive parameter combinations selected for comparison to evaluate the differences.

Rank	ARM95	ARM97	GATEIII	TOGAII	TWP06
Most	pz4, pz6, pc2	pz4, pz6, pc2	pz4, pz6, pu3	pz4, pu3, pc2	pz4, pz5, pz6
Middle	pz2, pz5, pc2	pz5, pz6, pu1	pz5, pz6, pc2	pz5, pu6, pu3	pz5, pu6, pu3
Last	pz1, pz2, pz3	pz1, pz2, pz3	pz1, pz2, pz3	pz1, pz3, pu1	pz1, pz3, pu1

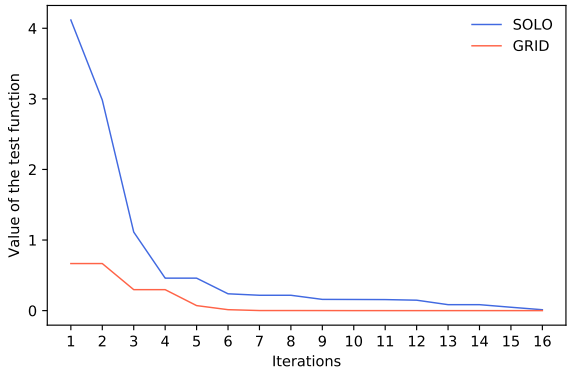

Figure 9. Take the sphere test function as an example of the optimization process after combining it with grid search. SOLO means using the optimization algorithm alone, GRID means combined with grid search. The latter leads to faster convergence.

can be seen that the introduction of grid search not only helps us to understand the distribution pattern of the parameters, but also helps to improve the performance of optimization process. The optimization process can converge earlier, while obtaining better tuning results. The parameter space used for optimization will be reduced to the range identified in the grid search process above and more fine-grained optimization will be carried out.

Table 8. Optimized output RMSE comparison between surrogate model and SCAM.

Cases	ARM95	ARM97	GATEIII	TOGAII	TWP06
Surrogate	11.051	8.694	5.902	4.001	8.488
SCAM	10.942	8.479	5.776	3.979	8.443
Error	0.99%	2.53%	2.17%	0.54%	0.54%

Consequently, experiments combining the grid search and optimization algorithms will be performed. As above, WOA is still used as the optimization algorithm for this stage. To verify the correctness of the surrogate model, we also compare its

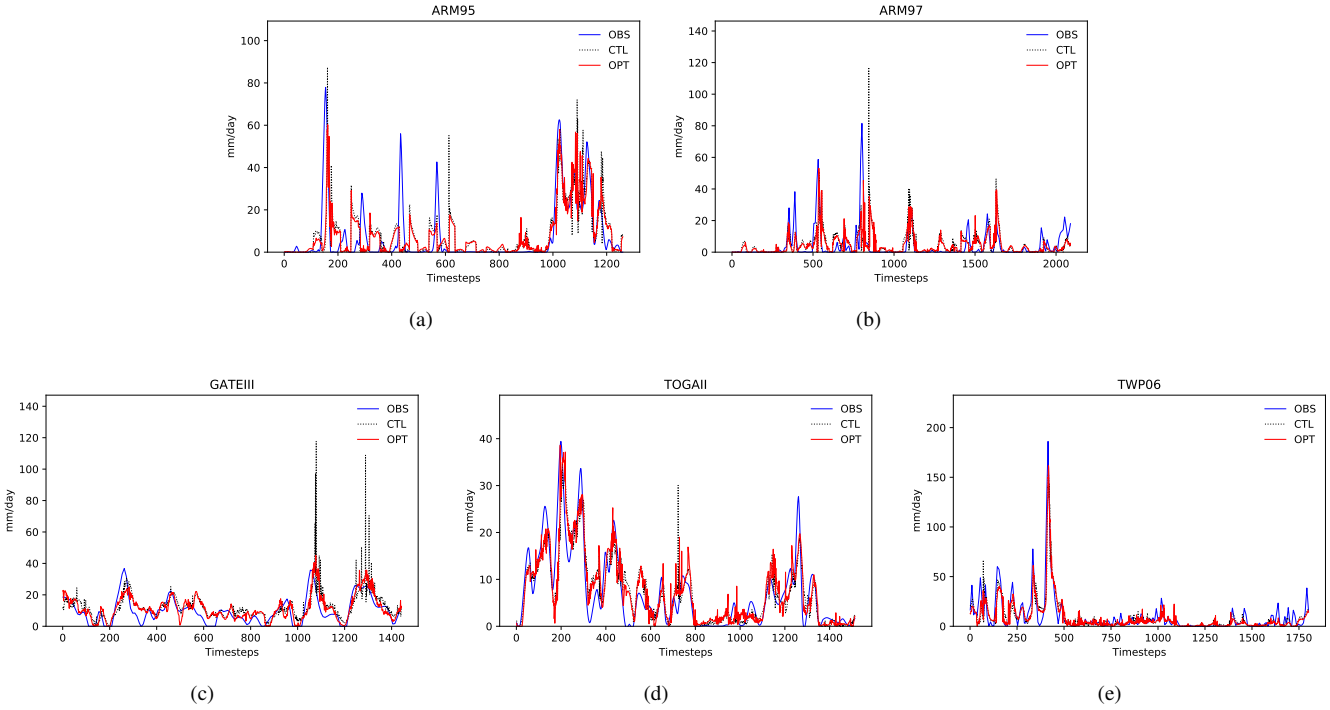

Figure 10. Model simulation output before and after tuning versus observed values. Where OBS indicates the observed value, CTL indicates the output before tuning and OPT indicates the output after tuning. (a)ARM95, (b)ARM97, (c)GATEIII, (d)TOGAII, (e)TWP06. It can be seen that the optimized output of each case is closer to the observation.

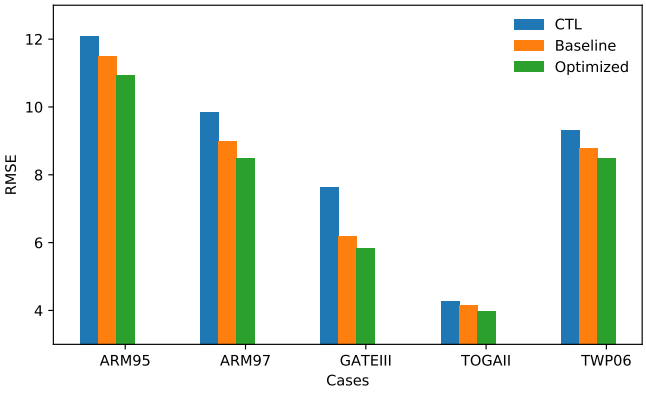

Figure 11. Comparison of RMSE in different experiments. CTL refers to the control experiment using the default value. Baseline refers to an experiment using only one optimization method. Optimized refers to the optimization experiment that combines the surrogate model and grid search.

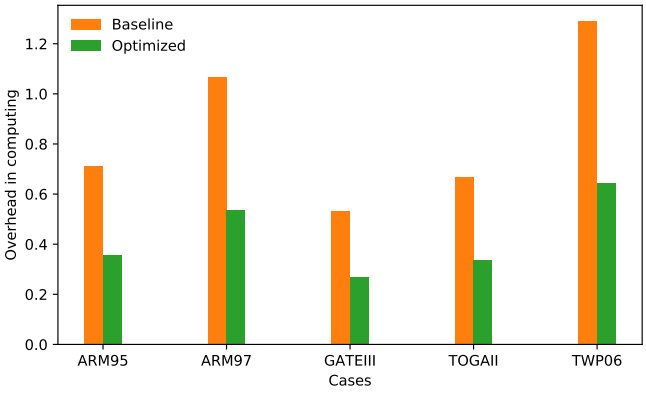

Figure 12. Comparison of computational overhead with without using the surrogate model and grid search. The Y-axis is the total computational hours.

output with SCAM. The pair of their RMSE performance is shown in Table 8. It can be seen that the surrogate model for these cases has better predictive ability. Compared with SCAM, the maximum error is also no more than 3%.

For the effectiveness of the parameter tuning, the output after tuning is compared with the output of the control experiment (i.e. before tuning) and the observed data after the individual cases had been tuned, as shown in Figure 10. Here, the experiments with the best optimization results are chosen for comparison. The tuning of the SCAM parameters is quite productive on the time scale. It is easy to see that in the control experiment there are several spikes where the simulated output is significantly higher than the observed values, as is the case in the first four cases. This is to say that these time steps, where the output is significantly larger than the observation in the control trial, are reduced after optimization, making it closer to the observation. Although still below the observed level at about 1300 steps of TOGAII, improvement is also reflected. However, for TWP06, there are cases where the default output is significantly smaller than the observed value. The performance of this case also improved after optimization. This demonstrates the significance of the parameter tuning provided by the workflow for model.

A cross-sectional comparison of the effects of these sets of experiments is shown in Figure 11. Meanwhile, the computational overheads of the various strategies are compared, as shown in Figure 12. The computational overhead here includes the time in the job management system when the batch is queued waiting to be allocated computational resources. In contrast to using the optimization algorithm alone, the grid search combined with the optimization algorithm can achieve better results on these SCAM cases. The use of NN trained surrogate models for parameter tuning can further save computational resource overhead and, in terms of results, can meet or exceed traditional optimization methods in most cases.

Therefore, we can find that it is possible to achieve a win-win situation in terms of computational resources and computational efficiency by training a surrogate model of SCAM based on NN. The model can get an enhancement in performance from 6.4%-24.4% in precipitation output. Thus, using the proposed method, the main computational overhead comes from sampling and training. The computational overhead can be saved by more than 50% compared to the case where the above

experiments are all run using the full SCAM. In particular, the proposed method demonstrates its effectiveness and usability for situations such as large-scale grid testing, which is almost impossible to accomplish using the full SCAM.

These results also show that the method used in the workflow outperforms previous methods in most cases. Furthermore, the methods in the workflow test a wide range of combinations of values in the parameter space. Thus using the workflow provides a more complete picture of the parametric characteristics of the different cases in SCAM than optimization algorithms that only provide results but no more information about the spatial distribution of the parameters. Doing a finer-grained grid search in the vicinity of the optimal value point is also an approach that is worth testing in the future. From the experimental results, it can be seen that utilizing the surrogate model through the sampling with the optimization algorithm not only saves resources, but also improves the optimization effect, and meanwhile improves the robustness of the optimization method.

4.6 A deeper exploration of the relationships between the cases

The training of surrogate models makes it possible to conduct larger scale experiments in a shorter period of time. For the most sensitive combinations of parameters in each of the cases obtained based on the NN method, we are able to explore the distribution pattern of the results using a grid search. These experiments are carried out on the surrogate model for reasons of improving experimental efficiency.

Grid search can also be performed to determine the possible aggregation range of the better-valued solution. To make it easy to compare the same and different cases, we specify the same parameter space for each case. Based on the combined ranking of sensitive parameters under each case, we choose pz4(tau), pz6(alfa) and pc2(rhminl) as the parameter space common to all cases.The results are shown in Figure 13. The value of each parameter is also divided into 11 levels within its upper and lower bounds. The parameter space remains the same as originally set at the beginning of this paper.

Table 9. When tuning the same sensitive parameter combination, the value of each parameter after optimization.

Parameter	ARM95	ARM97	GATEIII	TOGAII	TWP06	Multi(a)	Multi(b)	Multi(c)	Multi(d)
pz4	5118	5343	1800	3000	5400	5343	3800	4600	5199
pz6	0.1297	0.1234	0.0722	0.0833	0.1389	0.1361	0.0611	0.1371	0.0796
pc2	0.8006	0.8006	0.9308	0.8419	0.8863	0.8006	0.8197	0.8006	0.8131

As can be seen, after the parameter space has been replaced, the better-valued solutions for each case show a clear trend towards aggregation, and although the distribution of TOGAII is slightly scattered, it can still be grouped into a cluster. The same parameter space is more conducive to cross-sectional comparisons. It is easy to see that the two land convection cases are closer, which matches our expectations since the two cases are themselves co-located. They have relatively high tau and alfa, and relatively lowest rhminl. For the three tropical convective cases, their distributions have their own characteristics. TWP06 has the highest tau and alfa, while TOGAII is in the middle for all parameters. The parametric response distributions

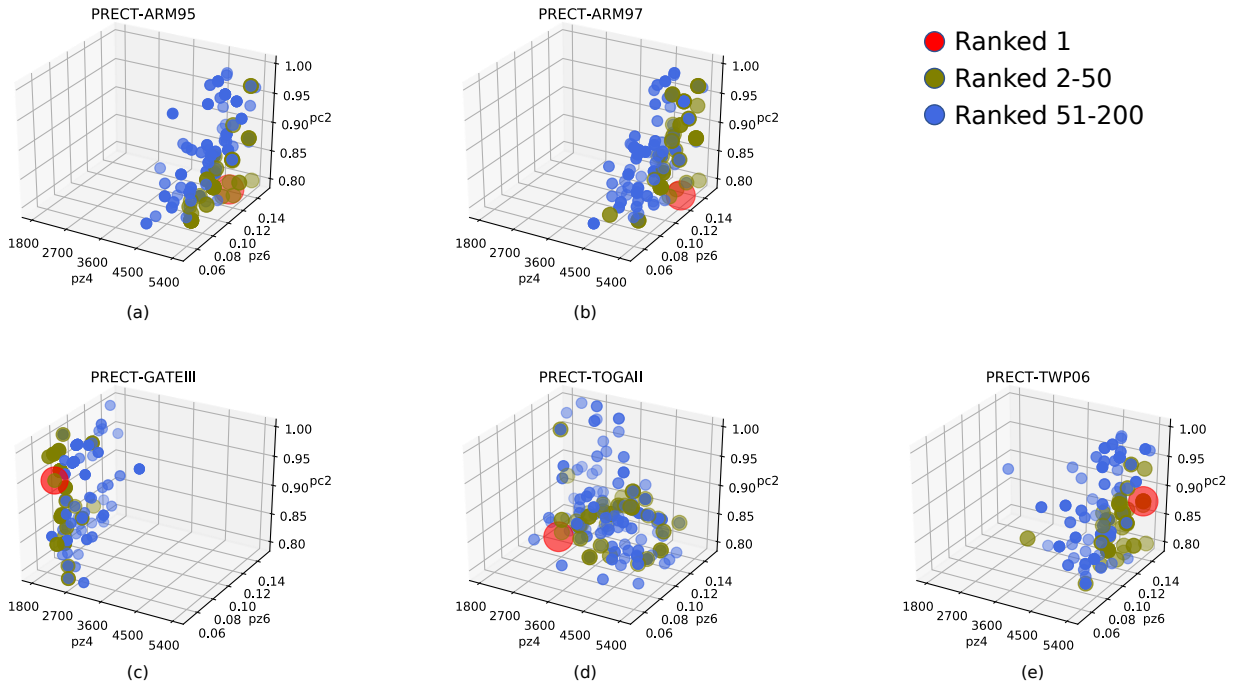

Figure 13. The distribution of better performing parameter solutions for each case in the same 3D parameter space. (a)ARM95, (b)ARM97, (c)GATEIII, (d)TOGAII, (e)TWP06. The points closest to the observed data are shown in red, those ranked 2-50 are shown in olive, and those ranked 51-200 are shown in blue.

of GATEIII are much more different. Its better performance relies on lower tau and alfa, and higher rhminl. As a site that is far away from all other cases, this also coincides with the previous results.

From the results, it can be seen that the distribution of the better values are different for different cases within the same parameter space. A typical example is the parameter pz4 (tau). This reflects the fact that it may be useful and necessary to adopt different parameter configurations for different cases or regions.

Now that we have discovered the pattern that the aggregation range of the more optimal solution for each case by applying a full-space grid search in the same parameter space. From the experimental results, it can be seen that the two cases focusing on land convection are most similar to each other, and two of three cases focusing on tropical convection are also more similar to each other, except for GATEIII.

Subsequently, we tried to combine several cases together for multi-objective grid search experiment. Based on the above results, we classify the combination of cases into the following four scenarios: (a) two land convection cases, (b) three tropical convection cases, (c) two western tropical Pacific cases, and (d) all five cases. The results are shown in Figure 14. It can be seen that the distributions in scenario (a) is very close to the individual cases it contain. This is also in line with our expectations,

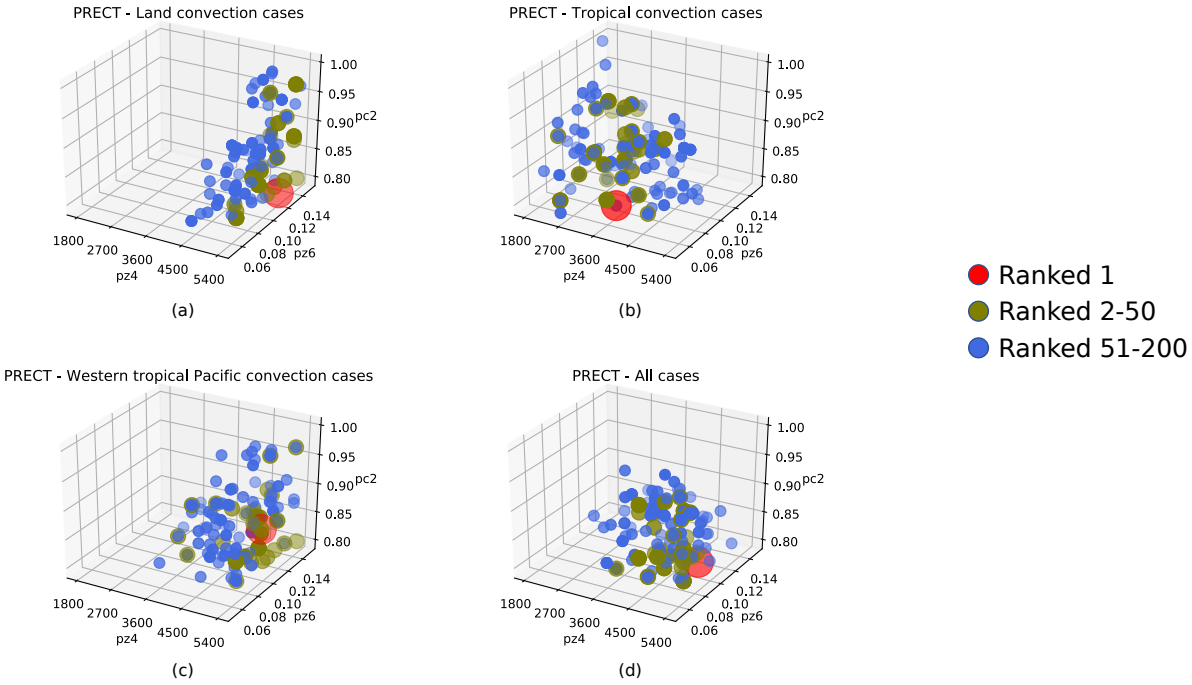

Figure 14. The distribution of better performing parameter solutions for multi-objective scenarios in the same 3D parameter space. Where (a) is the scenario of optimizing two land convection cases with the same set of parameters, (b) is the scenario of optimizing three tropical convection cases with the same set of parameters, (c) is the scenario of optimizing two western tropical Pacific cases with the same set of parameters, and (d) is the scenario of optimizing all five cases with the same set of parameters. The points closest to the observed data are shown in red, those ranked 2-50 are shown in olive, and those ranked 51-200 are shown in blue.

since the two cases themselves are located in the same location. Due to the inclusion of GATEIII in (b), its result is between GATEIII, TOGAII and TWP06. Considering that TOGAII and TWP06 are closer to each other, we designed (c) group of trials. It is relatively closer to the results in TWP06. The results in (d) are closer to those in (a) and (c). At the same time, it also exhibits greater clustering. The specific values are shown in Table 9. This can provide a basis for designing regional parameter combinations and values later.

Pearson correlation coefficient method is used to compare the best set of values for each case. The obtained similarity is shown in Figure 15. The average p-value in this experiment is less than 0.05, so it is relatively reliable. From this figure, we can see that the parameter values taken between the two cases of land convection are positively correlated in the same parameter space. The two of three cases of tropical convection, TOGAII and TWP06, are also positively correlated with each other. The respective distributions of the above four cases are also positively correlated. The more special one is GATEIII, which is negatively correlated with the remaining four cases. These results are also well matched to those obtained in the previous

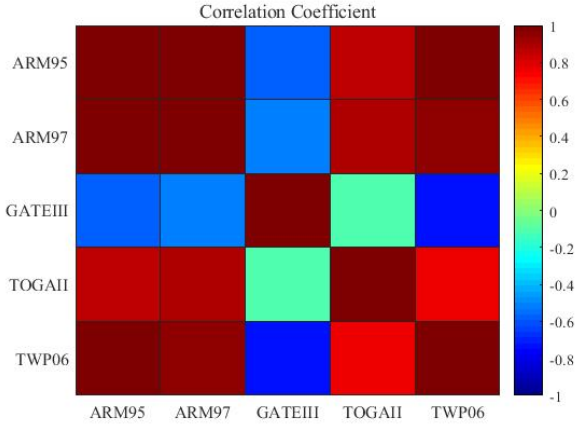

Figure 15. Correlation of the optimal solutions of the cases in the same parameter space. The similarity difference between different cases can be seen.

experiments. It is also clear from the results above that SCAM cases in similar locations and of the same type are more relevant when it comes to parameterization.

5 Conclusions

In this paper, a learning-based integrated method for SCAM parameter tuning on the HPC is proposed which enables a fully automated diagnostic analysis process from sensitive tests to parameter optimization and case comparison. The workflow makes it possible to run SCAM parameters on a large scale, thus allowing more trials on parametric scenario studies to be carried out in a shorter period of time. An integration of several sensitivity methods approach is used for sensitivity analysis of parameters, with just once sampling, the different SA methods can be invoked for analysis and their results combined.

With the enhancement of artificial intelligence and machine learning techniques, the role played by neural networks in regression analysis has become increasingly evident. In our proposed experimental workflow, multiple regression methods including NNs incorporating sampling techniques are likewise used in the parametric analysis process of SCAM. A precursor to sensitivity analysis: the results of the sampling are used to train an NN-based surrogate model, which validates the accuracy of both the sensitivity analysis and improves the parameter tuning process by the surrogate model. In these stages, a grid search strategy for parameter space based on multi-parameter perturbations is used. With the computational capabilities of the HPC, the method can search the most suitable parameter values within less iterations. The combination of grid search and optimization algorithms can also improves the performance of the optimization algorithm in model parameter tuning. In addition, the application of neural network-trained surrogate models can also saves computational resources, which is beneficial in achieving the goal of green computing.

To verify the completeness and validity of the proposed workflow, multi-group experiments based on five typical SCAM cases is implemented on the workflow. The sensitivity of the above parameters to typical output variables related to precipitation is analyzed. Experiments based on the proposed workflow have shown that there are differences in the sensitivities of the parameters with respect to the different cases and different output variables. This includes both the differences between the different convection types of cases and the differences in the effects of the deep and shallow convective precipitation parameterization schemes on the respective precipitation. To summarise, determining the appropriate values for each of the SCAM cases located at different locations facilitates is also meaningful for model development. This also provides a heuristic for future research on similar parametric schemes on other models.

Code and data availability. Open source code for this article is available at https://doi.org/10.6084/m9.figshare.21407109.v6 (Guo, 2023) along with the data used. The above is subject to the MIT License Agreement.

Author contributions. The manuscript was written by all authors. JG and HF proposed the main idea and method. JZ provides important guidance on neural networks. YX, WX and LW have provided important advice. JZ, PG and WW assisted in revising the manuscript. LH, GX and XC provided assistance and support with the experiments. XW assisted in debugging the program. LG provides help with access to computing resources.

Competing interests. The authors declare that they have no conflict of interest.

Acknowledgements. This research was supported in part by the National Key Research and Development Plan of China (Grant No. 2020YFB0204800), National Natural Science Foundation of China (Grant No. T2125006, U1839206), Jiangsu Innovation Capacity Building Program (Project No. BM2022028), Project of Jilin Province Development and Reform Commission No. 2019FGWTZC001 and the Science and Technology Development Plan of Jilin Province of China under Grant 20220101115JC.

References

Bacmeister, J. T., Wehner, M. F., Neale, R. B., Gettelman, A., Hannay, C., Lauritzen, P. H., Caron, J. M., and Truesdale, J. E.: Exploratory high-resolution climate simulations using the community atmosphere model (CAM), Journal of Climate, 27, 3073 – 3099, https://doi.org/10.1175/JCLI-D-13-00387.1, cited by: 163; All Open Access, Green Open Access, 2014.

Bogenschutz, P. A., Gettelman, A., Morrison, H., Larson, V. E., Schanen, D. P., Meyer, N. R., and Craig, C.: Unified parameterization of the planetary boundary layer and shallow convection with a higher-order turbulence closure in the Community Atmosphere Model: Single-column experiments, Geoscientific Model Development, 5, 1407–1423, https://doi.org/10.5194/gmd-5-1407-2012, 2012.

Bogenschutz, P. A., Gettelman, A., Morrison, H., Larson, V. E., Craig, C., and Schanen, D. P.: Higher-Order Turbulence Closure and Its Impact on Climate Simulations in the Community Atmosphere Model, Journal of Climate, 26, 9655–9676, 2013.

Bogenschutz, P. A., Tang, S., Caldwell, P. M., Xie, S., Lin, W., and Chen, Y. S.: The E3SM version 1 single-column model, Geoscientific Model Development, 13, 4443–4458, https://doi.org/10.5194/gmd-13-4443-2020, 2020.

Breiman, L.: Random Forests, Machine Learning, 2001.

Caflisch, R. E.: Monte carlo and quasi-monte carlo methods, Acta numerica, 7, 1–49, 1998.

Cravero, C., De Domenico, D., and Ottonello, A.: Uncertainty quantification approach on numerical simulation for supersonic jets performance, Algorithms, 13, https://doi.org/10.3390/A13050130, 2020.

Dennis, J. M., Edwards, J., Evans, K. J., Guba, O., Lauritzen, P. H., Mirin, A. A., St-Cyr, A., Taylor, M. A., and Worley, P. H.: CAM-SE: A scalable spectral element dynamical core for the Community Atmosphere Model, International Journal of High Performance Computing Applications, 26, 74 – 89, https://doi.org/10.1177/1094342011428142, cited by: 258, 2012.

Fu, H., Liao, J., Yang, J., Wang, L., Song, Z., Huang, X., Yang, C., Xue, W., Liu, F., Qiao, F., Zhao, W., Yin, X., Hou, C., Zhang, C., Ge, W., Zhang, J., Wang, Y., Zhou, C., and Yang, G.: The Sunway TaihuLight supercomputer: system and applications, Science China Information Sciences, 59, https://doi.org/10.1007/s11432-016-5588-7, 2016.

Gan, Y., Duan, Q., Wei, G., Tong, C., Sun, Y., Wei, C., Ye, A., Miao, C., and Di, Z.: A comprehensive evaluation of various sensitivity analysis methods: A case study with a hydrological model, Environmental Modelling & Software, 51, 269–285, 2014.

Gettelman, A., Morrison, H., and Ghan, S. J.: A New Two-Moment Bulk Stratiform Cloud Microphysics Scheme in the Community Atmosphere Model, Version 3 (CAM3). Part II: Single-Column and Global Results, Journal of Climate, 21, 3660–3679, 2008.

Gettelman, A., Truesdale, J. E., Bacmeister, J. T., Caldwell, P. M., Neale, R. B., Bogenschutz, P. A., and Simpson, I. R.: The Single Column Atmosphere Model Version 6 (SCAM6): Not a Scam but a Tool for Model Evaluation and Development, Journal of Advances in Modeling Earth Systems, 11, 1381–1401, https://doi.org/10.1029/2018MS001578, 2019.

Goffart, J., Rabouille, M., and Mendes, N.: Uncertainty and sensitivity analysis applied to hygrothermal simulation of a brick building in a hot and humid climate, Journal of Building Performance Simulation, 10, 37–57, 2015.

Guo, J.: Learning-based_SCAM_Tuner, https://doi.org/10.6084/m9.figshare.21407109.v6, 2023.

Guo, Z., Wang, M., Qian, Y., Larson, V. E., Ghan, S., Ovchinnikov, M., Bogenschutz, P. A., Zhao, C., Lin, G., and Zhou, T.: A sensitivity analysis of cloud properties to CLUBB parameters in the Single Column Community Atmosphere Model (SCAM5), Journal of Advances in Modeling Earth Systems, 6, 829–858, 2015.

Harada, M.: GSA: Stata module to perform generalized sensitivity analysis, Statistical Software Components, 2012.

He, K., Zhang, X., Ren, S., and Sun, J.: Deep Residual Learning for Image Recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

Karaboga, D. and Basturk, B.: On the performance of artificial bee colony (ABC) algorithm, Applied Soft Computing, 8, 687–697, https://doi.org/https://doi.org/10.1016/j.asoc.2007.05.007, 2008.

Kennedy, J. and Eberhart, R.: Particle swarm optimization, in: Neural Networks, 1995. Proceedings., IEEE International Conference on, vol. 4, pp. 1942–1948, 2002.

Li, G., Rabitz, H., Yelvington, P. E., Oluwole, O. O., Bacon, F., Kolb, C. E., and Schoendorf, J.: Global Sensitivity Analysis for Systems with Independent and/or Correlated Inputs, Journal of Physical Chemistry A, 2, 7587–7589, 2010.

May, P. T., Mather, J. H., Vaughan, G., Bower, K. N., Jakob, C., Mcfarquhar, G. M., and Mace, G. G.: The tropical warm pool international cloud experiment, Bulletin of the American Meteorological Society, 89, 629–+, 2008.

McKay, M. D., Beckman, R. J., and Conover, W. J.: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics, 42, 55–61, 2000.

Mirjalili, S. and Lewis, A.: The Whale Optimization Algorithm, Advances in Engineering Software, 95, 51–67, https://doi.org/10.1016/j.advengsoft.2016.01.008, 2016.

Mitchell, M.: An Introduction to Genetic Algorithms, 1996.

Morris, M. D.: Factorial sampling plans for preliminary computational experiments, Technometrics, 33, 161–174, https://doi.org/10.1080/00401706.1991.10484804, 1991.

Nelder, J. A. and Wedderburn, R. W.: Generalized linear models, Journal of the Royal Statistical Society: Series A (General), 135, 370–384, 1972.

Park, S.: A Unified Convection Scheme (UNICON). Part I: Formulation, Journal of the Atmospheric Sciences, 71, 3902–3930, 2014.

Pathak, R., Dasari, H. P., El Mohtar, S., Subramanian, A. C., Sahany, S., Mishra, S. K., Knio, O., and Hoteit, I.: Uncertainty Quantification and Bayesian Inference of Cloud Parameterization in the NCAR Single Column Community Atmosphere Model (SCAM6), New techniques for improving climate models, predictions and projections, 2022.

Plischke, E., Borgonovo, E., and Smith, C. L.: Global sensitivity measures from given data, European Journal of Operational Research, 226, 536–550, 2013.

Qian, Y., Yan, H., Hou, Z., Johannesson, G., Klein, S., Lucas, D., Neale, R., Rasch, P., Swiler, L., Tannahill, J., Wang, H., Wang, M., and Zhao, C.: Parametric sensitivity analysis of precipitation at global and local scales in the Community Atmosphere Model CAM5, Journal of Advances in Modeling Earth Systems, 7, 382–411, https://doi.org/https://doi.org/10.1002/2014MS000354, 2015.

Saltelli, A.: Making best use of model evaluations to compute sensitivity indices, Computer physics communications, 145, 280–297, 2002.

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S.: Global sensitivity analysis: the primer, John Wiley & Sons, 2008.

Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., and Tarantola, S.: Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index, Computer Physics Communications, 181, 259–270, 2010.

Schober, P., Boer, C., and Schwarte, L. A.: Correlation Coefficients: Appropriate Use and Interpretation., Anesthesia & Analgesia, 126, 1763–1768, 2018.

Shi, L., Copot, C., and Vanlanduit, S.: Evaluating Dropout Placements in Bayesian Regression Resnet, Journal of Artificial Intelligence and Soft Computing Research, 12, 61–73, https://doi.org/10.2478/jaiscr-2022-0005, 2022.

Sobol', I. M.: On the distribution of points in a cube and the approximate evaluation of integrals, Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki, 7, 784–802, 1967.

Sobol, I. M.: Sensitivity analysis for non-linear mathematical models, Mathematical modelling and computational experiment, 1, 407–414, 1993.

Storn, R. and Price, K.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, Journal of global optimization, 11, 341–359, 1997.

Tang, J., Deng, C., and Huang, G.-B.: Extreme Learning Machine for Multilayer Perceptron, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, 27, 809–821, https://doi.org/10.1109/TNNLS.2015.2424995, 2016.

Thompson, R. M., Payne, S. W., Recker, E., and Reed, R. J.: Structure and Properties of Synoptic-Scale Wave Disturbances in the Intertropical Convergence Zone of the Eastern Atlantic, J. Atmos., 36, 53–72, 1979.

Webster, P. J. and Lukas, R.: TOGA COARE: the coupled ocean-atmosphere response experiment, Bulletin of the American Meteorological Society, 73, 1377–1416, 1992.

XingFen, W., Xiangbin, Y., and Yangchun, M.: Research on User Consumption Behavior Prediction Based on Improved XGBoost Algorithm, in: 2018 IEEE International Conference on Big Data (Big Data), pp. 4169–4175, IEEE, Seattle, WA, USA, https://doi.org/10.1109/BigData.2018.8622235, 2018.

Yan, J., Chen, J., and Xu, J.: Analysis of Renting Office Information Based on Univariate Linear Regression Model, in: INTERNATIONAL CONFERENCE ON ELECTRICAL AND CONTROL ENGINEERING (ICECE 2015), pp. 780–784, Adv Sci Res Ctr, international Conference on Electrical and Control Engineering (ICECE), Guilin, PEOPLES R CHINA, APR 18-19, 2015, 2015.

Yang, B., Qian, Y., Lin, G., Leung, L. R., Rasch, P. J., Zhang, G. J., Mcfarlane, S. A., Zhao, C., Zhang, Y., and Wang, H.: Uncertainty quantification and parameter tuning in the CAM5 Zhang-McFarlane convection scheme and impact of improved convection on the global circulation and climate, Journal of Geophysical Research Atmospheres, 118, 395–415, 2013.

Zhang, G. J.: Sensitivity of climate simulation to the parameterization of cumulus convection in the Canadian Climate Centre general circulation model, Atmos. Ocean, 33, 1995.

Zhang, T., Zhang, M., Lin, W., Lin, Y., Xue, W., Yu, H., He, J., Xin, X., Ma, H. Y., Xie, S., and Zheng, W.: Automatic tuning of the Community Atmospheric Model (CAM5) by using short-term hindcasts with an improved downhill simplex optimization method, Geoscientific Model Development, 11, 5189–5201, https://doi.org/10.5194/gmd-11-5189-2018, 2018.

Zou, L., Qian, Y., Zhou, T., and Yang, B.: Parameter Tuning and Calibration of RegCM3 with MIT–Emanuel Cumulus Parameterization Scheme over CORDEX East Asia Domain, Journal of Climate, 27, 7687 – 7701, https://doi.org/https://doi.org/10.1175/JCLI-D-14-00229.1, 2014.