

Emulation to simulate low resolution atmospheric data

Vishwas Rao, Richard Archibald and Katherine Evans

October 4, 2012

1 Overview of CESM

The community earth system model(CESM) is a fully coupled, global climate model that provides computer simulations of the past, present and future climate states. The most recent version of the CESM is available to the community on the website [1]. The code, a reference manual and a user's guide for each component, input datasets and outputs from some basic models are freely available. A detailed overview of CCSM 4(an earlier version of CESM) is described in [2].

The CESM models the land, atmosphere, ice and the ocean. To characterize the uncertainties in the climate, solution to the governing PDEs (or stochastic PDEs) has to be obtained. One of the methods employed to solve the SPDEs are generalized polynomial chaos. This in turn requires us to run the simulations multiple times. Running simulations to determine the physical quantities consumes a lot of cpu time. In this paper we build an emulator which can be used to simulate the values of the physical quantities at unknown grid points within a certain error bound. As a proof of concept, we present the results obtained with a 2 degree atmospheric dataset [3].

2 Dataset

For the experiments, we have used the T42 dataset. The data arises out of the 42 - wave triangular truncation. This can very accurately treat the features and their horizontal derivatives down to approximately 950 KM. In

a T42 model the transform grid interval is approximately 300 KM at the equator. The T42 model is a 2.8 degree transform grid. These terms are almost always evaluated using parametrization techniques. The dataset is a grid of size 64 by 128. The data gives the time average of the temperature at each of the grid points. Community atmospheric model, version 3(CAM3) is the latest in a succession of atmospheric general circulation models that have been made widely available to the scientific community. The CAM3 incorporates a significant number of changes to the dynamical formulation, the treatment of cloud and precipitation processes, radiation processes and atmospheric aerosols. This is described in Collins et al. (2005b). The standard configuration of the CAM3 is based on an Eulerian spectral dynamical core, where the vertical discretization makes use of 26 levels treated using a second order finite-differences [3].

3 Stochastic collocation method

Consider the following Partial differential equation(PDE):

$$\mathcal{L}(u) = 0, \tag{1}$$

$$\mathcal{B}(u) = 0. \tag{2}$$

Suppose the above PDE has a variable parameter α then the solution equation can be rewritten as an SPDE as follows

$$\mathcal{L}(u; \alpha) = 0, \tag{3}$$

$$\mathcal{B}(u) = 0 \tag{4}$$

Where α follows a given distribution. The above system of equations model the climate and can be used to quantify the uncertainties associated with the climate.

The solution to the above system of equations is regarded as a random process and can be expanded by Wiener-Askey polynomial chaos as [4]

$$u(x, t; \alpha) = \sum_{i=0}^P u_i(x, t) \Phi_i(\zeta(\alpha)) \tag{5}$$

The approximation to the solution can be obtained in two ways, namely, Galerkin and Collocation method. Galerkin method is an intrusive method and collocation method is a non intrusive method. Here we use the stochastic collocation method described in [5].

3.1 Generalized Polynomial Chaos

The particular stochastic collocation method used for the study is known as Generalized polynomial chaos(gPC). We begin with the review of the basic ideas behind gPC. Let $D \subset \mathbb{R}^\ell$, $\ell = 1, 2, 3$, be a physical domain with boundary ∂D and coordinates $x = (x^{(1)}, \dots, x^{(\ell)})$ and let $T > 0$ be a real number. We consider the following general (scalar) stochastic partial differential equation

$$\begin{cases} u_t(x, t, y) = \mathcal{L}(u), & D \times (0, T] \times \mathbb{R}^d, \\ \mathcal{B}(u) = 0, & \partial D \times [0, T] \times \mathbb{R}^d, \\ u = u_0, & D \times \{t = 0\} \times \mathbb{R}^d, \end{cases} \quad (6)$$

where \mathcal{L} is a (nonlinear) differential operator, \mathcal{B} is the boundary condition operator, u_0 is the initial condition, and $y = (y^{(1)}, \dots, y^{(d)}) \in \mathbb{R}^d$, $d \geq 1$, are a set of independent random variables characterizing the random inputs to the governing equation from various sources, e.g., boundary condition, system parameters, etc. The solution is therefore a stochastic quantity,

$$u(t, x, Z) : \bar{D} \times [0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}. \quad (7)$$

Let us assume for all $i = 1, \dots, d$, the random variables $y^{(i)}$ are continuous with probability density functions(PDF) $\rho^i : \Gamma^{(i)} \rightarrow \mathbb{R}^+$, where $\Gamma^{(i)} \triangleq y^{(i)}(\Omega)$ is the image of $y^{(i)}$. Then the random vector $y = (y^{(1)}, \dots, y^{(d)})$ has a joint PDF

$$\rho(y) = \prod_{i=1}^d \rho^{(i)}(y^{(i)}). \quad (8)$$

In the following we will adopt the multi-index notation: Let $\mathbf{i} = (i_1, \dots, i_d) \in \mathbb{N}_0^d$ be a multi-index, with $|\mathbf{i}| = \sum_{k=1}^d i_k$, and $\mathbf{i} = \mathbf{j}$ iff. $i_k = j_k, \forall k = 1, \dots, d$. We also define an index set

$$\mathcal{J}_N \triangleq \{\mathbf{i} \in \mathbb{N}_0^d : |\mathbf{i}| \leq N\} \quad (9)$$

for a given integer $N \geq 0$. An N^{th} -order generalized polynomial chaos(gPC) expansion to the solution of (6) takes the form

$$u_N(t, x, y) = \sum_{i \in \mathcal{J}_N} \hat{u}_i(t, x) \Phi_i(y), \quad (10)$$

where, for the stochastic collocation method, the expansion coefficients are determined by solving,

$$u(t, x, y_k) = \sum_{i \in \mathcal{J}_N} \hat{u}_i(t, x) \Phi_i(y_k), \quad \forall k \in \mathcal{J}_N. \quad (11)$$

Thus, the stochastic collocation gPC approximation is determined by having knowledge of the solution of (6) for each independent realization, y_k , of the random inputs. Following classical approximation theory, the gPC expansion (10) converges when u is square integrable with respect to $\rho(y)$, that is for any fixed $(t, x) \in [0, T] \times \bar{D}$,

$$\|u - u_N\|_{L^2_\rho}^2 \triangleq \int (u - u_N)^2 \rho(y) dy \rightarrow 0, \quad N \rightarrow \infty \quad (12)$$

The convergence of the gPC approximation is directly related to the smoothness of the stochastic solution u . To solve for the coefficients, we are required to have the values of the physical quantity for different values of parameters. This in turn requires us to run the simulation multiple times. This requires significant compute power and time. Here we use multi-dimensional interpolation schemes to emulate the ensemble of simulation.

4 Interpolation

Below we illustrate the $2D$ lagrange interpolation. The same can be easily extended to n dimensions. $2D$ -Lagrange interpolation is based on $1D$ -Lagrange interpolation. The $2D$ Lagrange polynomials can be written as a product of $1D$ Lagrange polynomials.

$$L_{ij} = L_i(x)L_j(y) \quad 0 \leq i \leq n, \quad 0 \leq j \leq m$$

$$L_i(x) = \prod_{s=0, s \neq i}^n \frac{x - x_s}{x_i - x_s}, \quad L_j(y) = \prod_{s=0, s \neq j}^m \frac{y - y_s}{y_i - y_s}$$

So we have:

$$L_{ij}(x_r, y_s) = \begin{cases} 1 & \text{if } i = r \text{ and } j = s \\ 0 & \text{otherwise} \end{cases}$$

And then we have:

$$P(x, y) = \sum_{i=0}^n \sum_{j=0}^m f(x_i, y_j) L_{ij}(x, y)$$

Where $P(x, y)$ is a polynomial that interpolates $f(x, y)$ in the given data. The same concept can be extended to n -dimensions. Higher dimensional interpolation is useful when the physical quantity is treated as a multivariable function. We use the barycentric Lagrange interpolation [6].

5 Emulation and Analysis

To verify the concept, we performed an emulation with the available data. Here we consider the 2 degree dataset, which has been briefly described before in section 2. We consider a physical quantity to be a function of a parameter and time. Interpolation is performed to evaluate the physical quantity at an unknown timestep and parameter value. For eg., we use the temperature field at time steps t_1, t_2, t_4 and t_5 to determine the temperature field at time step t_3 and compare with the existing dataset. We also have performed multi-dimensional interpolation to evaluate the physical quantity at an unknown 'time' and a 'parameter value'. Interpolation and collection of statistics(mean) has been performed in two possible different orders, viz. collecting the statistics and interpolating and vice versa. Each have their own pros and cons. Collecting the statistics first and interpolating is much faster, but has very less information. The results are tabulated below. We have looked at different physical quantities like temperature, sea level pressure. These physical quantities are treated as a function of different parameters like low and high stable clouds, time scale for precipitation etc.

Physical Quantity	CMFTAU	RHMINL	ZMCONVKE
Temperature	7.3386e-4	7.6201e-4	7.9195e-4
SWCF	0.011593	0.010047	0.014185
LWCF	0.011004	0.013628	0.012268
CLDTOT	0.008997	0.011015	0.009601
Q	0.005506	0.006094	0.005246
U	0.080686	0.095737	0.086530

Table 1: **Statistics + Interpolation**

Physical Quantity	CMFTAU	RHMINL	ZMCONVKE
Temperature	7.3389e-4	7.6208e-4	7.9191e-4
SWCF	0.011600	0.010040	0.014182
LWCF	0.011003	0.013627	0.012267
CLDTOT	0.008990	0.011018	0.009609
Q	0.005508	0.006096	0.005245
U	0.080685	0.095740	0.086528

Table 2: **Interpolation + Statistics**

From the tables it is clear that the order of interpolation and collecting the statistics does not make much of a difference to the relative errors. Hence either of them can be used to generate the ensembles based on the needs. It is also interesting to note that some of the physical quantities (like U(Zonal velocity)) exhibit large error percentages. One possible explanation for it is that the zonal winds have a large deviation between successive years. This makes the interpolation inaccurate, where as temperature fields are much more steadier, hence we obtain better results on interpolation. We also looked at how lower degree interpolation schemes perform for temperature and velocity. It turns out that in case of temperature, lower degree interpolation schemes perform fairly well, but in case of velocity, the higher degree interpolation performs much better. The lower degree interpolations perform fairly well in cases where we are interested only in statistics. But if the metric is the overall relative norm, then the 3rd degree interpolation performs the best.

Phy. Qty.	0 th Order	1 st Order	2 nd Order	3 rd Order
U	5.0958e+03	5.8761e+03	4.9994e+03	48.8494
Temp	0.5327	0.4558	0.5579	0.4742

Table 3: **Norm of the error in interpolation**

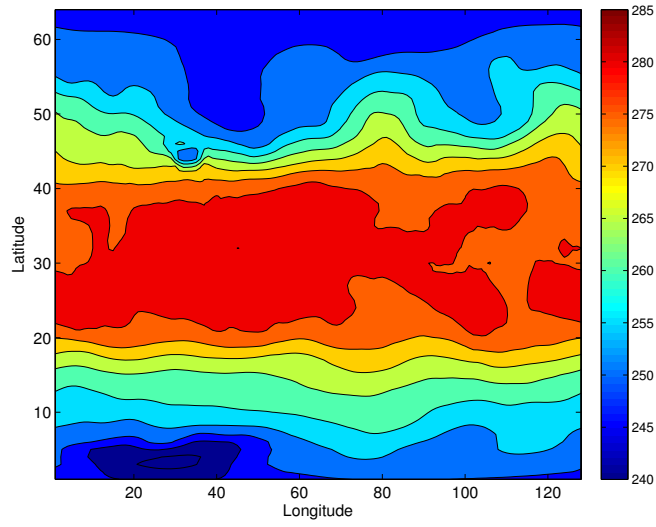


Figure 1: Temperature field (Simulated)

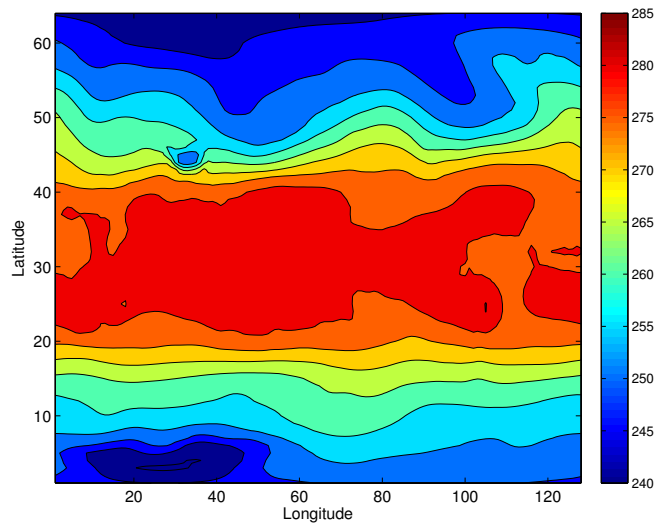


Figure 2: Temperature field (Interpolated)

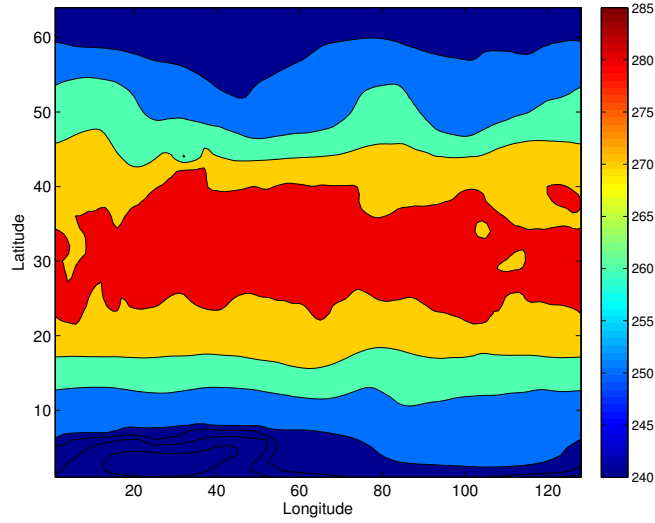


Figure 3: Temperature field (2D Simulated)

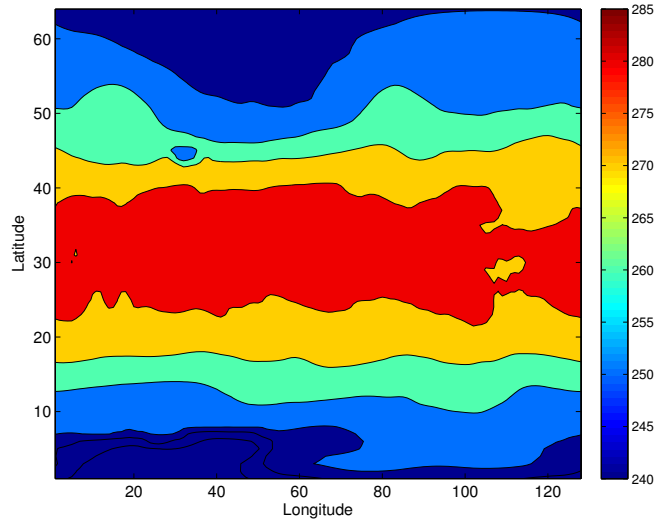


Figure 4: Temperature field (2D Interpolated)

From the figures it is clear that the interpolated temperature field is fairly close to the simulated field. We used different degrees of interpolation to check the accuracy of the interpolation. It turns out that interpolation odd degrees perform better because of the symmetry. The interpolation was performed for the parameter CMFTAU. Linear interpolation marginally outperforms the cubic interpolation.

Physical Quantity	1^{st}	3^{rd}
Temperature	0.455815	0.4742052
Velocity	0.0199419	0.0249795

Table 4: The comparison of error norms for different degrees of interpolation

References

- [1] www.cesm.ucar.edu (2012).
- [2] P. Gent, G. Danabasoglu, L. Donner, The community climate system model version 4, *Journal of Climate* 24 (2011) 4973–4991.
- [3] W. Collins, P. Rasch, B. Boville, Description of the NCAR Community Atmosphere Model (CAM 4.0) (2010).
- [4] D. Xiu, G. Karniadakis, The wiener–askey polynomial chaos for stochastic differential equations, *SIAM Journal on Scientific Computing* 24 (2) (2002) 619–644.
- [5] R. Archibald, M. Chakoumakos, T. Zhuang, Characterizing the elements of earth’s radiative budget: Applying uncertainty quantification to the cesm, *Procedia Computer Science* 9 (0) (2012) 1014 – 1020, proceedings of the International Conference on Computational Science, ICCS 2012.
- [6] J. Berrut, L. Trefethen, Barycentric lagrange interpolation, *SIAM Review* 46 (3) (2004) 501–517.