The Rationale for Developing Neural Network Emulations for CAM5 and CESM1/CAM5

J. Tribbia, P. Rasch, V. Krasnopolsky, and M. Fox-Rabinovitz

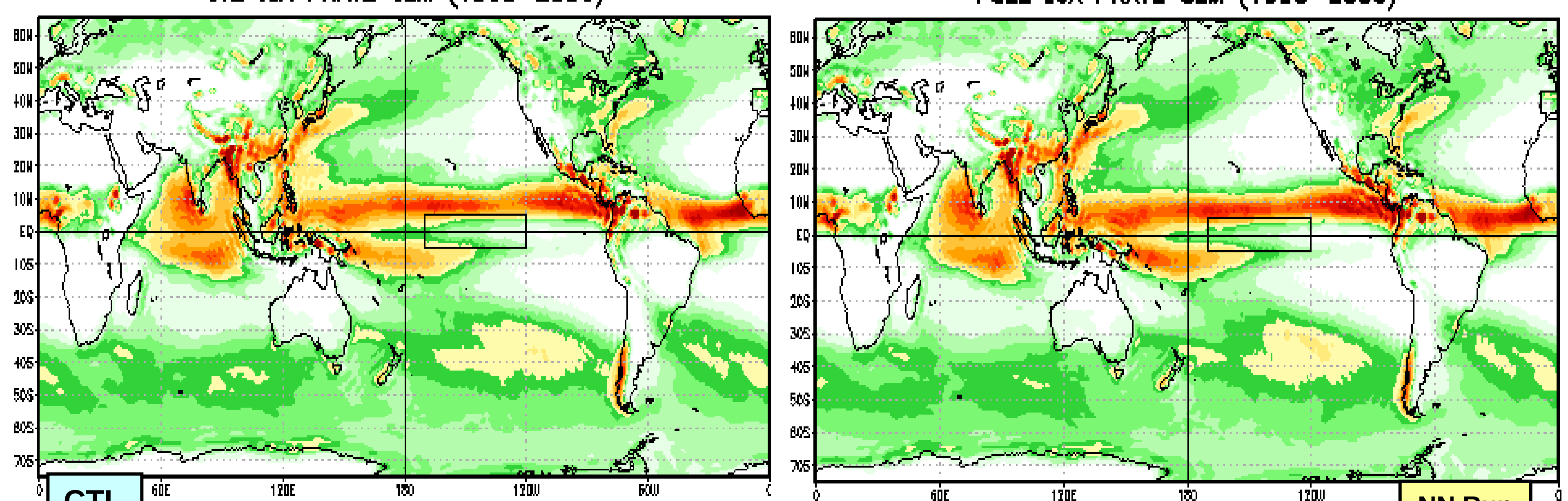
The 2010 CCSM Workshop, Breckenridge, CO, June 28-July 2, 2010

Background – 1

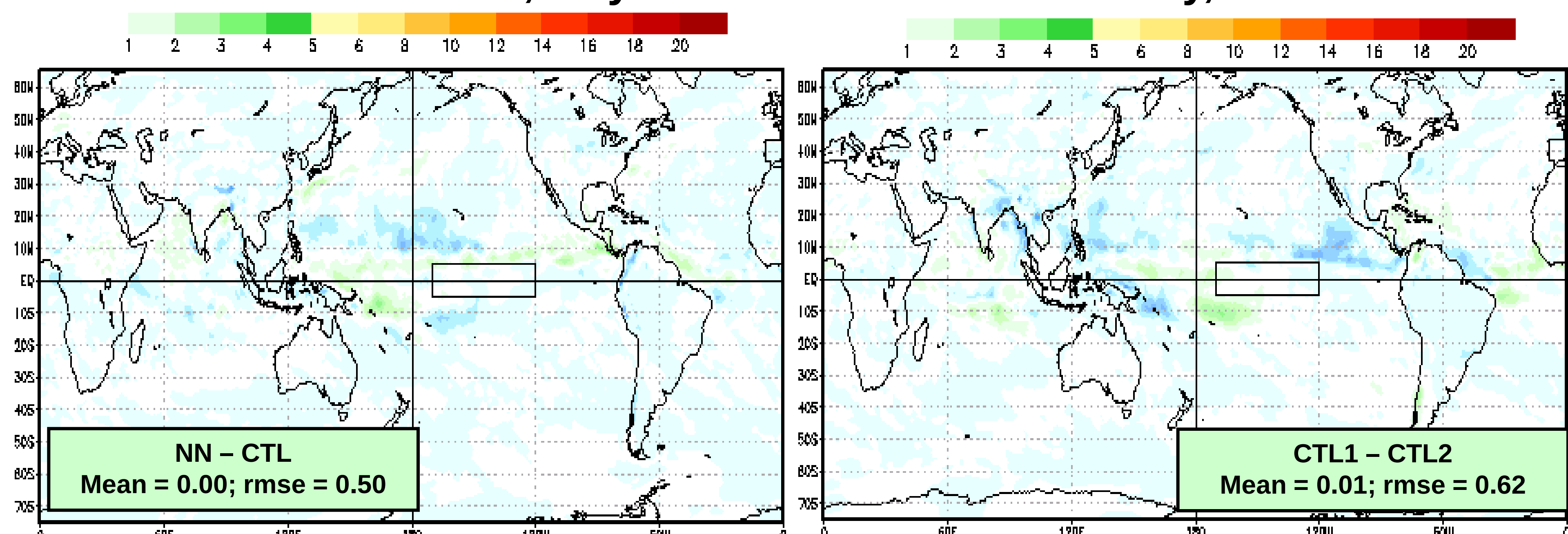
- Any parameterization of model physics can be reproduced (emulated) to a very high accuracy using Neural Network (NN)
- A "representative" training dataset consisting of "inputs" (the "intent in" variables) and outputs (the "intent out") variables that appear in a parameterization should be given to train NN.
- "Representative" set spans the range of variation that one expects the parameterization to be subjected to. It can be created using model simulated data.
- We call such a NN emulation – a "**parameterization emulator**"

Background – 2

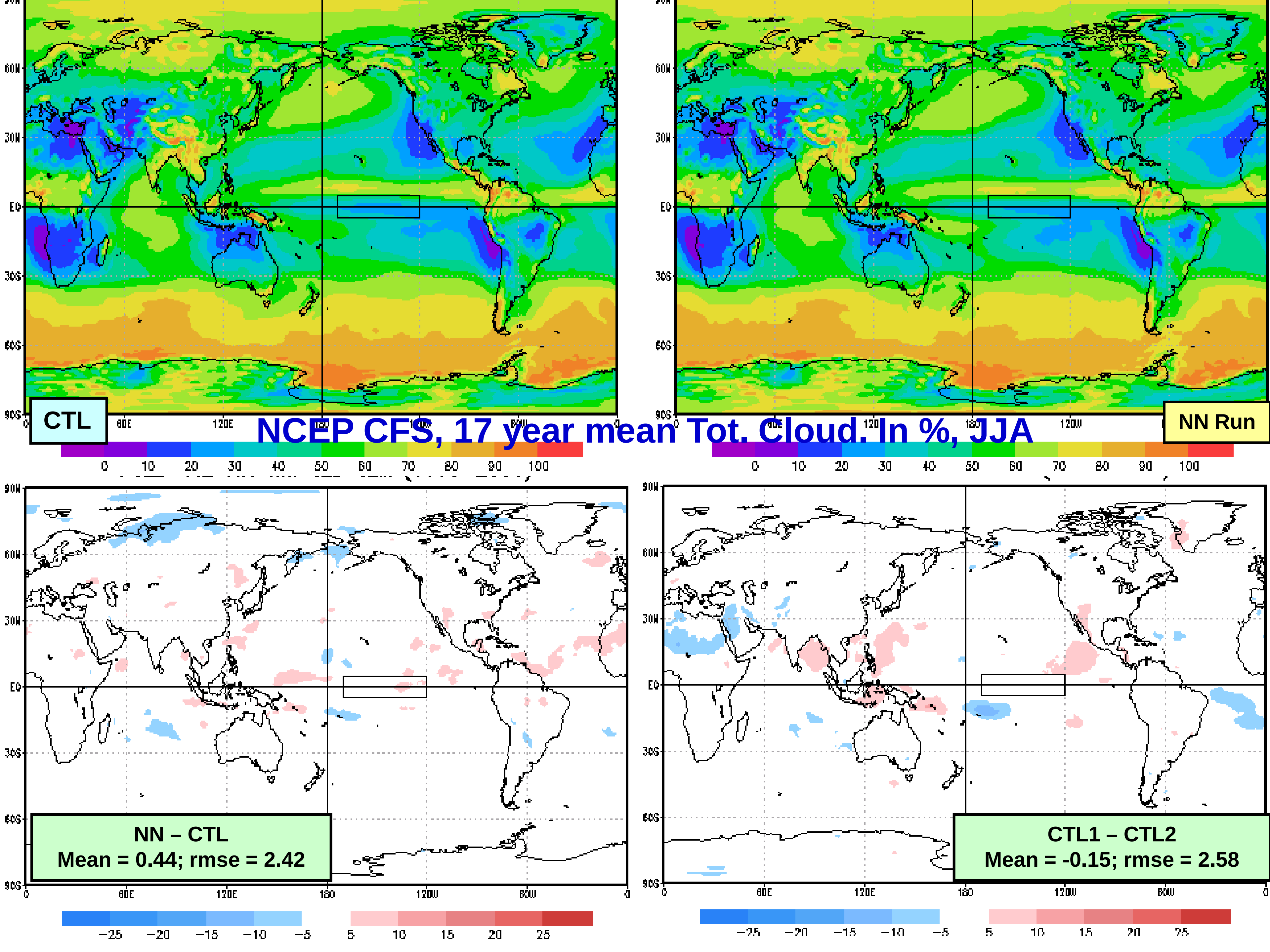
- **“A posteriori" corrections to the NN emulation to guarantee, for example, that the heating rates and fluxes are internally consistent, can be introduced.**
- **Parameterization emulator is carefully tested:**
  - **on independent (not used for training) set of simulated data**
  - **in parallel runs:** (1) Control run (model with the original parameterization) and (2) NN run (model with the NN emulation)
- **Several methods have been developed to control the accuracy of the NN emulation and correct it during the model run.**
- **NN emulation can be adjusted to climate changes**
- **This method has been applied to develop:**
  - **NN emulation of CAMRT long- and short wave radiation**
  - **NN emulation of RRTM long- and short wave radiation in NCEP CFS**
  - **NN based convection parameterization using CRM data**



NCEP CFS, 17 year mean PRATE in mm/day, JJA



The contour intervals for the PRATE fields are 1 mm/day for the 0 – 6 mm/day range and 2 mm/day for the 6 mm/day and higher; for the PRATE differences the contour intervals are 1 mm/day


NCEP CFS, 17 year mean Tot. Cloud. In %, JJA
CTL
NN Run
0 10 20 30 40 50 60 70 80 90 100
NN – CTL
Mean = 0.44; rmse = 2.42
CTL1 – CTL2
Mean = -0.15; rmse = 2.58
−25 −20 −15 −10 −5
5 10 15 20 25

Pros and Cons

- Pros
  - one can develop an NN "emulation" of a parameterization that is indistinguishable from the original parameterization, but is **one to two orders of magnitude faster** than the original, which can be used to:
    - Speed up the model integration (~25% for NCEP CFS T126L64)
    - Increase the frequency of radiation calculations (e.g., calculate it every time step)
    - Use ensemble of NNs (perturbed and stochastic physics)
    - Increase model resolution, etc
- Cons
  - Is not physically transparent as the original parameterization is
  - Should be retrained after major changes in the model (e.g., after a change of vertical resolution).

Conclusions

- Several NN emulations of model physics have been successfully developed:
  - **NN emulation of CAMRT long- and short wave radiations (Krasnopolsky et al., MWR, 2005 and 2008)**
  - **NN emulation of RRTM long- and short wave radiations in NCEP CFS (Krasnopolsky et al., MWR, 2010)**
  - **NN based convection parameterization using CRM data**
- The approach is carefully tested and ready for "production" (including NN ensembles)
- Why would you not adopt such a strategy for CAM5 and CESM1/CAM5?

Additional Slides

FAQ

- **Q:** *Why ECMWF uses NeuroFlux for 4D-Var only but not for NWP?*
- **A:** Because NeuroFlux has many limitations related to its design (Morcrette et al., 2008) and also (Krasnopolsky et al. 2005):
  - both accuracy and rapidity could not be kept at once at higher vertical resolution (60 and more layers)
  - The accuracy in the lowest and uppermost atmospheric layers is not satisfactory due to the increased non-linearity there.
- **Our NN emulation approach is different and free of the above limitations** so that it has been successfully applied to:
  - LWR and SWR for CAMRT and NCEP CFS RRTM
  - Currently it is being applied to convection
  - Works with high vertical resolution
  - Is significantly more accurate and faster
  - Provides a better Jacobian
  - Allows using NN ensembles as stochastic physics to reduced uncertainties

Evaluation of NN emulation

- **Validation on independent set of simulated data:**
  - Accuracy of approximation
  - Speed up (code by code comparison)
- **Validation in parallel runs:**
  - (1) Control run (model with the original parameterization) and (2) NN run (model with NN emulation)
  - Differences are evaluated and are comparable with:
    - Observation errors
    - Uncertainties of reanalysis
    - Model “internal variability”

Radiation – a computational bottleneck

- **Cost of the radiation is a problem for GCMs, NWP and other models**:
  - ECMWF calculates radiation on a coarse grid and then interpolates horizontally to a fine grid
  - Canadian operational model calculates radiation at reduced vertical resolution and then interpolates vertically
  - NCAR, NCEP and UKMO calculate radiation less frequently then other model components

Background

- Any parameterization of model physics is a relationship or **MAPPING** (continuous or almost continuous) between two vectors: a vector of input parameters, *X*, and a vector of output parameters, *Y*,

$$Y = F(X); \quad X \in \Re^n \; and \; Y \in \Re^m$$

- NN is a generic approximation for **any** continuous or almost continuous mapping given by a set of its input/output records:

$$\text{SET} = \{X_i, Y_i\}_{i = 1, \ldots, N}$$

Fast and Accurate Neural Network Radiation

- **Fast NN emulations of LWR and SWR**:
  - Are very accurate; the changes they introduce in the model results are of the order of the model “internal variability”
  - Reduce significantly (**one to two orders of magnitude**) the computation cost of radiation
  - Improve the load balance
- **NN radiation is very flexible, the improved computational performance can be used to:**
  - Speed up the model integration (~25% for NCEP CFS T126L64)
  - Increase the frequency of radiation calculations (e.g., calculate it every time step)
  - Use ensemble of NNs (perturbed and stochastic physics)
  - Increase model resolution, etc

Bulk Approximation Statistics (all errors are in K/Day)

Statistics Types	Statistics	LWR			SWR	
		NCAR CAMRT (L = 26)	NCEP CFS (L = 64)		NCAR CAM (L = 26)	NCEP CFS (L = 64)
			RRTMG	RRTMF		RRTMG
Total (3D) Error Statistics (K/day)	**Bias**	$3. \cdot 10^{-4}$	$2.\cdot 10^{-3}$	$7. \cdot 10^{-4}$	$-4. \cdot 10^{-3}$	$5. \cdot 10^{-3}$
	RMSE	0.34	0.49	0.42	0.19	0.20
Bottom Layer (2D) Error Statistics	**Bias**	$-2.\ 10^{-3}$	$-1.\cdot 10^{-2}$	$6. \cdot 10^{-3}$	$-5. \cdot 10^{-3}$	$9. \cdot 10^{-3}$
	RMSE	0.86	0.64	0.67	0.43	0.22
Top Layer Error (2D) Statistics	**Bias**	$-1. \cdot 10^{-3}$	$-9.\cdot 10^{-3}$	$2. \cdot 10^{-3}$	$2. \cdot 10^{-3}$	$1.3 \cdot 10^{-2}$
	RMSE	0.06	0.18	0.09	0.17	0.21
Speedup, η	***Times***	**150**	**16 (20)**	**21**	**20**	**60 (90)**

Neural Network

Continuous Input to Output Mapping

Y = F$_{NN}$(X)

$$y_q = a_{q0} + \sum_{j=1}^{k} a_{qj} \cdot t_j$$

Neuron ➡ $$t_j = \tanh(b_j + \sum_{i=1}^{n} \Omega_{ji} \cdot x_i)$$

Major Advantages of NNs:

- NNs are *generic,* very *accurate and convenient* mathematical (statistical) models which are able to emulate numerical model components, which are complicated nonlinear input/output relationships (continuous or almost continuous mappings).
- NNs are ***robust*** with respect to random noise and fault-tolerant.
- NNs are *analytically differentiable* (training, error and sensitivity analyses): almost free Jacobian!
- NNs emulations are **accurate and fast but there is**

NO FREE LUNCH!

  – Training is a complicated and time consuming nonlinear optimization procedure; *however, training should be done only once for a particular application!*

- NNs are well-suited for parallel and vector processing

Development of NN Emulations of Model Physics Parameterizations

Learning from Data