Stress-testing the coupled behavior of hybrid physics-machine learning climate simulations on an unseen, warmer climate

Jerry Lin
University of California, Irvine
jerryL9@uci.edu

Mohamed Aziz Bhouri
Columbia University
mb4957@columbia.edu

Tom Beucler
University of Lausanne
tom.beucler@unil.ch

Sungduk Yu
University of California, Irvine
sungduk@uci.edu

Michael Pritchard
University of California, Irvine and NVIDIA
mspritch@uci.edu

Abstract

Accurate and computationally-viable representations of clouds and turbulence are a long-standing challenge for climate model development. Traditional parameterizations that crudely but efficiently approximate these processes are a leading source of uncertainty in long-term projected warming and precipitation patterns. Machine Learning (ML)-based parameterizations have long been hailed as a promising alternative with the potential to yield higher accuracy at a fraction of the cost of more explicit simulations. However, these ML variants are often unpredictably unstable and inaccurate in *coupled* testing (i.e. in a downstream hybrid simulation task where they are dynamically interacting with the large-scale climate model). These issues are exacerbated in out-of-distribution climates. Certain design decisions such as "climate-invariant" feature transformation for moisture inputs, input vector expansion, and temporal history incorporation have been shown to improve coupled performance, but they may be insufficient for coupled out-of-distribution generalization. If feature selection and transformations can inoculate hybrid physics-ML climate models from non-physical, out-of-distribution extrapolation in a changing climate, there is far greater potential in extrapolating from observational data. Otherwise, training on multiple simulated climates becomes an inevitable necessity. While our results show generalization benefits from these design decisions, the obtained improvment does not sufficiently preclude the necessity of using multi-climate simulated training data.

1 Introduction and Motivation

Anthropogenic climate change is increasing the frequency and severity of climate extremes and natural disasters, requiring informed adaptation and mitigation measures from policymakers [1, 2, 3]. While domain scientists continue to achieve notable progress in improving our climate physics understanding, significant uncertainty in projected warming and precipitation patterns remains. Much of this uncertainty stems from the intractable computational expense of explicitly resolving subgrid processes like convection and radiation, making cheaper *conventional* parameterizations that approximate their effects necessary [1, 4]. Even if hardware advancements continue at the pace of Moore's Law, it would take decades to be able to run global climate simulations that resolve the turbulent eddies responsible for low cloud formation, a major source of uncertainty in projected warming [5].

Neural network parameterizations could be trained on more explicit simulations to emulate unresolved subgrid processes, enabling a higher fidelity representation using current-generation hardware [6, 7, 8, 9, 10, 11]. However, the task of parameterizing subgrid physics (in our case convection and

radiation) becomes stubbornly difficult when these neural network emulators are coupled to the large-scale climate model and integrated in time. Because coupled behavior is highly variable, large-scale coupled tests are necessary for drawing conclusions on surrogate model design decisions [12]. Based on proven coupled in-distribution benefits [12] and using large-scale coupled tests on an out-of-distribution climate, we rigorously test generalization improvement of the following three design decisions:

1. Using a relative humidity “climate-invariant" feature transformation for the moisture input [13].
2. Expanding the input vector to address potential omitted variable bias [12]
3. Incorporating memory effects (i.e. temporal history) in the input [14, 12]

Our results show that these design decisions improve generalization on an out-of-distribution climate relative to our baseline neural network configuration, but they are not sufficient to supplant multi-climate training, something that is argued necessary in previous works [15, 16].

2 Methods

2.1 Reference Climate Simulation

Our neural networks are trained on and validated against the Super-Parameterized Community Atmosphere Model v3 (SPCAM 3), which has served as a test-bed of for prototyping neural network emulators of subgrid convection in previous works [7, 17, 18, 13, 19]. In super-parameterization, a high-fidelity, 2D model of convection called a Cloud-Resolving Model (CRM) with 32 columns at a 4-km horizontal resolution is embedded inside each grid-cell of the host climate model [20, 21, 22]. For simplicity, we use a fixed season, prescribed sea surface temperatures, and a zonally-symmetric aquaplanet. The timestep is fixed to 30 minutes and 30 vertical levels are considered within each grid cell. Using SPCAM 3, we create two reference simulations, with one having prescribed sea surface temperatures that are 4K warmer. All hybrid-ML climate models are trained using the colder climate and coupled in both settings.

2.2 Neural Network Configurations

To assess the generalization benefits of our design decisions, we train and evaluate 330 neural networks for each design decision and each of the following configurations. The Specific Humidity (SH) Configuration is a baseline similar to previous work [17]. The Relative Humidity (RH) Configuration uses a “climate-invariant" relative humidity feature transformation for the moisture input variable. The Expanded Variables (EV) Configuration concatenates meridional wind, ozone mixing ratio, and cosine of zenith angle to the input variables of the RH configuration. Finally, the Previous Tendencies (PT) Configuration concatenates heating and moistening tendencies from one previous timestep to the input variables of the RH configuration. In order to leverage scalable off-the-shelf tools for coupling (in our case the Fortran Keras Bridges (FKB)), all neural networks are dense, feedforward neural networks. The input and output variables and hyperparameter search space used for sampling can be found in the Appendix.

3 Results

3.1 Uncoupled Results

The uncoupled error for all four configurations in both climates is given in Figure 1, showing higher values across the board in the warmer climate. As expected, the baseline SH configuration no longer clears the linear regression baseline when tested on the warmer climate, in line with smaller-scale uncoupled results from Beucler et al. (2021) [13]. Variance in uncoupled error for the SH (EV) configuration jumps by 2 (3) orders of magnitude in the warmer climate testing, and error from the EV configuration is higher compared to the RH configuration (a reversal from uncoupled results in the original climate), suggesting potential overfitting on some of the additional variables.

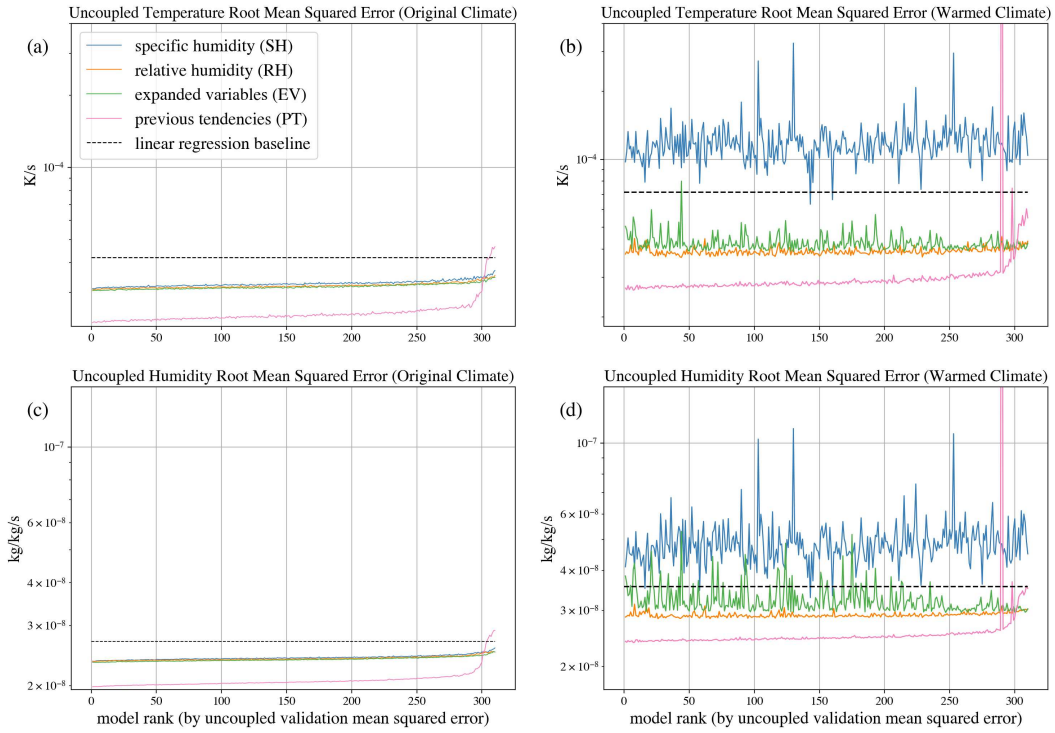

Figure 1: Uncoupled test error on in-distribution and out-of-distribution (warmer) climate for each configuration with models ranked by validation error.

3.2 Coupled Results

3.2.1 Coupled Generalization on Unseen Climate

Figure 2 shows the coupled results for the warmer climate testing, highlighting potential success in coupled generalization to an unseen climate. However, only a fraction of the coupled simulations run for the entire year without prematurely terminating (4.5%, 13%, 23%, and 14% of simulations for the SH, RH, EV and PT configurations, respectively). While the coupled simulations with the lowest temperature and humidity RMSEs belong to the PT configuration, the best PT models for one climate are not necessarily the best for the other.

3.2.2 Coupled Generalization in Both Climates

Figure 3 shows a fairly clear relationship between coupled error in the warmer climate and the original one, with R^2 values of .82 and .61 for temperature and humidity, respectively. However, excluding the SH configuration, whose models under-performed linear regression in the warmer climate, drops these R^2 values to .4 and .25, respectively.

4 Discussion

As seen in Figure 3, there appears to be a weak relationship for coupled temperature error between in-distribution and out-of-distribution climates. Such a relationship is not only much weaker for moisture results, but the coupled moisture errors in the warmer climate are almost uniformly higher. This behavior indicates that generalizing on moistening in a coupled setting on an out-of-distribution climate deserves focus for future model development. It is possible that addressing this limitation in future model development will also tighten the relationship for temperature error between in-distribution and out-of-distribution climates.

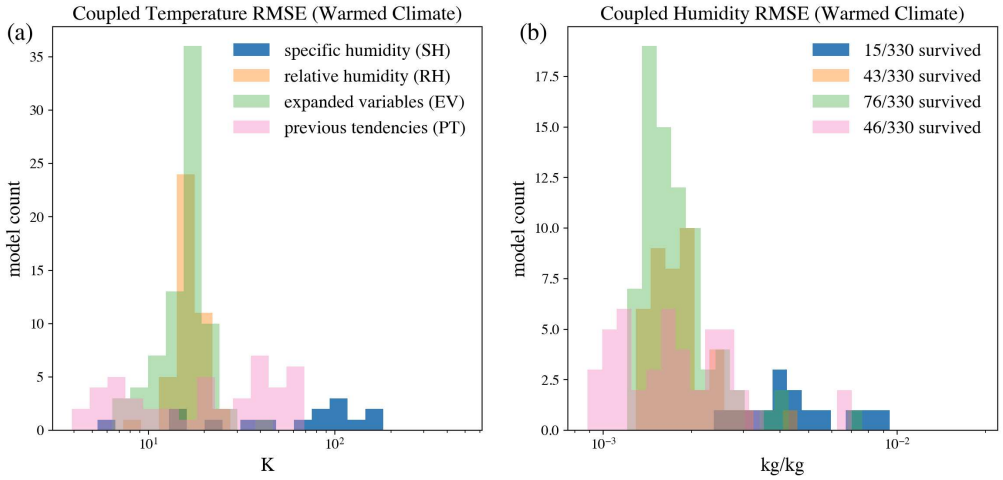

Figure 2: Histograms for coupled root mean squared error for temperature and humidity in the warmer (unseen) climate. Models corresponding to coupled simulations that prematurely terminated are excluded.

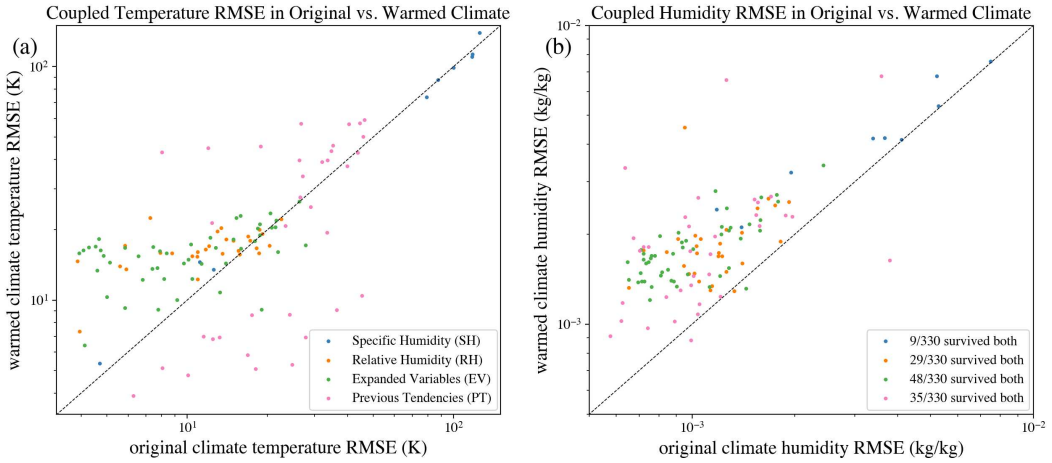

Figure 3: Scatterplots of coupled error for hybrid physics-ML simulations that did not prematurely terminate in either climate. Dashed line is a 1-to-1 line that intersects the origin.

5 Conclusion

Coupled out-of-distribution generalization might still be possible without multi-climate training when using more sophisticated network architectures, physics-informed neural networks (PINNs), pruned feature selection, and additional "climate-invariant" feature transformations (e.g. for temperature and latent heat flux) [23, 24, 13, 12]. It is also worth noting that enforcing conservation laws for our task is not possible without additional inputs and outputs. Nevertheless, our results point to the necessity of stress-testing out-of-distribution in a coupled setting. The combined task of generalizing out-of-distribution and remaining stable and accurate when coupled is a demonstrably more difficult challenge that requires further collaboration between domain scientists and machine learning experts.

Acknowledgments and Disclosure of Funding

We would like to thank Justus Will and Ritwik Gupta for helpful feedback on this manuscript. High-performance computing was facilitated by Bridges2 at the Pittsburgh Supercomputing Center through allocation ATM190002 from the Advanced Cyberinfrastructure Coordination Ecosystem:

Services & Support (ACCESS) program, which is supported by NSF grants #2138259, #2138286, #2138307, #2137603, and #2138296. J.L., S.Y., and M.S.P acknowledge support from the DOE (DE-SC0023368) and NSF (AGS-1912134). M.A.B acknowledges National Science Foundation funding from an AGS-PRF Fellowship Award (AGS2218197). Finally, we would like to thank Xiaojuan Liu for donating unused CPU hours on Bridges2.

References

[1] IPCC, *Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. IPCC, 2021.

[2] M. G. Donat, A. J. Pitman, and S. I. Seneviratne, "Regional warming of hot extremes accelerated by surface energy fluxes," *Geophys. Res. Lett.*, vol. 44, pp. 7011–7019, July 2017.

[3] L. R. Vargas Zeppetello, A. E. Raftery, and D. S. Battisti, "Probabilistic projections of increased heat stress driven by climate change," *Communications Earth & Environment*, vol. 3, pp. 1–7, Aug. 2022.

[4] B. Tian and X. Dong, "The double-ITCZ bias in CMIP3, CMIP5, and CMIP6 models based on annual mean precipitation," *Geophys. Res. Lett.*, vol. 47, p. e2020GL087232, Apr. 2020.

[5] T. Schneider, J. Teixeira, C. S. Bretherton, F. Brient, K. G. Pressel, C. Schär, and A. Pier Siebesma, "Climate goals and computing the future of clouds," *Nat. Clim. Chang.*, vol. 7, pp. 3–5, Jan. 2017.

[6] N. D. Brenowitz and C. S. Bretherton, "Prognostic validation of a neural network unified physics parameterization," *Geophysical Research Letters*, vol. 45, no. 12, pp. 6289–6298, 2018.

[7] P. Gentine, M. Pritchard, S. Rasp, G. Reinaudi, and G. Yacalis, "Could machine learning break the convection parameterization deadlock?," *Geophys. Res. Lett.*, vol. 45, pp. 5742–5751, June 2018.

[8] P. A. O'Gorman and J. G. Dwyer, "Using machine learning to parameterize moist convection: Potential for modeling of climate, climate change, and extreme events," *Journal of Advances in Modeling Earth Systems*, vol. 10, no. 10, pp. 2548–2563, 2018.

[9] G. Mooers, M. Pritchard, T. Beucler, J. Ott, G. Yacalis, P. Baldi, and P. Gentine, "Assessing the potential of deep learning for emulating cloud superparameterization in climate models with real-geography boundary conditions," *J. Adv. Model. Earth Syst.*, vol. 13, May 2021.

[10] X. Wang, Y. Han, W. Xue, G. Yang, and G. J. Zhang, "Stable climate simulations using a realistic general circulation model with neural network parameterizations for atmospheric moist physics and radiation processes," *Geosci. Model Dev.*, vol. 15, pp. 3923–3940, May 2022.

[11] S. Yu, W. M. Hannah, L. Peng, M. A. Bhouri, R. Gupta, J. Lin, B. Lütjens, J. C. Will, T. Beucler, B. E. Harrop, B. R. Hillman, A. M. Jenney, S. L. Ferretti, N. Liu, A. Anandkumar, N. D. Brenowitz, V. Eyring, P. Gentine, S. Mandt, J. Pathak, C. Vondrick, R. Yu, L. Zanna, R. P. Abernathey, F. Ahmed, D. C. Bader, P. Baldi, E. A. Barnes, G. Behrens, C. S. Bretherton, J. J. M. Busecke, P. M. Caldwell, W. Chuang, Y. Han, Y. Huang, F. Iglesias-Suarez, S. Jantre, K. Kashinath, M. Khairoutdinov, T. Kurth, N. J. Lutsko, P.-L. Ma, G. Mooers, J. David Neelin, D. A. Randall, S. Shamekh, A. Subramaniam, M. A. Taylor, N. M. Urban, J. Yuval, G. J. Zhang, T. Zheng, and M. S. Pritchard, "ClimSim: An open large-scale dataset for training high-resolution physics emulators in hybrid multi-scale climate simulators," *arxiv*, June 2023. arxiv:2306.08754.

[12] J. Lin, S. Yu, T. Beucler, P. Gentine, D. Walling, and M. Pritchard, "Systematic sampling and validation of machine Learning-Parameterizations in climate models," *arXiv*, Sept. 2023.

[13] T. Beucler, M. Pritchard, J. Yuval, A. Gupta, L. Peng, S. Rasp, F. Ahmed, P. A. O'Gorman, J. David Neelin, N. J. Lutsko, and P. Gentine, "Climate-Invariant machine learning," *arxiv*, Dec. 2021. arxiv:2112.08440.

[14] Y. Han, G. J. Zhang, X. Huang, and Y. Wang, "A moist physics parameterization based on deep learning," *J. Adv. Model. Earth Syst.*, vol. 12, Sept. 2020.

[15] S. K. Clark, N. D. Brenowitz, B. Henn, A. Kwa, J. McGibbon, W. A. Perkins, O. Watt-Meyer, C. S. Bretherton, and L. M. Harris, "Correcting a 200 km resolution climate model in multiple climates by machine learning from 25 km resolution simulations," *J. Adv. Model. Earth Syst.*, vol. 14, Sept. 2022.

[16] M. A. Bhouri, L. Peng, M. S. Pritchard, and P. Gentine, "Multi-fidelity climate model parameterization for better generalization and extrapolation," *arxiv*, Sept. 2023. arxiv:2309.10231.

[17] S. Rasp, M. S. Pritchard, and P. Gentine, "Deep learning to represent subgrid processes in climate models," *Proc. Natl. Acad. Sci. USA*, vol. 115, no. 39, pp. 9684–9689, 2018.

[18] J. Ott, M. Pritchard, N. Best, E. Linstead, M. Curcic, P. Baldi, and M. E. Acacio Sanchez, "A Fortran-Keras deep learning bridge for scientific computing," *Sci. Program.*, vol. 2020, Jan. 2020.

[19] G. Behrens, T. Beucler, P. Gentine, F. Iglesias-Suarez, M. Pritchard, and V. Eyring, "Non-Linear dimensionality reduction with a variational encoder decoder to understand convective processes in climate models," *J Adv Model Earth Syst*, vol. 14, p. e2022MS003130, Aug. 2022.

[20] M. Khairoutdinov, D. Randall, and C. DeMott, "Simulations of the atmospheric general circulation using a Cloud-Resolving model as a superparameterization of physical processes," *J. Atmos. Sci.*, vol. 62, pp. 2136–2154, July 2005.

[21] M. S. Pritchard, C. S. Bretherton, and C. A. DeMott, "Restricting 32-128 km horizontal scales hardly affects the MJO in the superparameterized community atmosphere model v.3.0 but the number of cloud-resolving grid columns constrains vertical mixing," *J. Adv. Model. Earth Syst.*, vol. 6, pp. 723–739, Sept. 2014.

[22] T. R. Jones, D. A. Randall, and M. D. Branson, "Multiple-instance superparameterization: 1. concept, and predictability of precipitation," *J. Adv. Model. Earth Syst.*, vol. 11, pp. 3497–3520, Nov. 2019.

[23] T. Beucler, M. Pritchard, S. Rasp, J. Ott, P. Baldi, and P. Gentine, "Enforcing analytic constraints in neural networks emulating physical systems," *Physical Review Letters*, vol. 126, no. 9, p. 098302, 2021.

[24] F. Iglesias-Suarez, P. Gentine, B. Solino-Fernandez, T. Beucler, M. Pritchard, J. Runge, and V. Eyring, "Causally-informed deep learning to improve climate models and projections," *arxiv*, Apr. 2023. arxiv:2304.12952.

5.1 Input and output variables

Table 1 depicts the input variables for the neural network configurations. All neural network configurations other than the baseline specific humidity (SH) configuration use relative humidity (%) for moisture. Variables marked with a single asterisk * are exclusive to the expanded variables (EV) configuration, and variables marked with a double asterisk ** are exclusive to the previous tendencies configuration. Input variables are normalized by subtracting the mean and dividing by the standard deviation.

All neural network configurations share output variables shown in 2. Output variables are multiplied by 1004 and 2.5e6 for heating and moistening tendencies, respectively, to put them in similar orders of magnitude.

Table 1: Input variables

Input variable	Unit	
Temperature	K	30
Humidity	kg/kg or %	30
Surface pressure	Pa	1
Incoming solar radiation	W/m^2	1
Sensible heat flux	W/m^2	1
Latent heat flux	W/m^2	1
Meridional wind*	m/s	30
Ozone mixing ratio*	m^3/m^3	30
Cosine of zenith angle*		1
$(t-1)$ Heating tendency**	K/s	30
$(t-1)$ Moistening tendency**	$kg/kg/s$	30

Table 2: Output variables

Input variable	Unit	Vertical levels
Heating tendency	K/s	30
Moistening tendency	$kg/kg/s$	30

5.2 Hyperparameter search space

All neural networks uniformly randomly subsample the search space depicted in Table 3. For the learning rate range, the entire interval is log transformed such that different orders of magnitude are sampled at similar rates.

Table 3: Hyperparameter search space

Hyperparameter	Range
Hidden layers	[[4, 11]]
Nodes per layer	[[128, 512]]
Batch normalization	{On, Off}
Dropout	[0.0, 0.25]
Optimizer	{RMSprop, Adam, RAdam, QHAdam}
Leaky ReLu slope	[0.0, 0.4]
Learning rate	[1e-5, 1e-2]