Appendix A Dynamical core of NeuralGCM

The dynamical core provides NeuralGCM with strong physics priors based on well understood and easy to simulate phenomena. In section A.1 we provide more details on spatial discretization of the atmospheric state in NeuralGCM. In section A.2 we summarize the governing equations of the dynamical core. In section A.3 we provide references to numerical implementations and rationale for our choices.

A.1 Discretization of the dynamical core

Our dynamical core uses a Gaussian grid and sigma coordinates [29] to discretize the computational domain. Gaussian grids enable fast and accurate transformations between the grid space representation and spherical harmonics basis. They result in equiangular longitude lines and unequal spacing latitudes defined by the Gaussian quadrature. Terrain-following sigma coordinates discretize the vertical direction by the fraction of the surface pressure, and thus correspond to non-stationary vertical height since surface pressure changes with time. Cell boundaries in sigma coordinates take values $\sigma \in [0, 1]$, with $\sigma = 0$ corresponding to the top of the atmosphere ($p = 0$ pressure boundary) and $\sigma = 1$ representing the earth's surface.

In this work we have trained a lineup of models that make forecasts at varying horizontal resolutions: 2.8°, 1.4°, and 0.7°, corresponding to truncated linear Gaussian grids TL63, TL127, TL255. The number in the grid name corresponds to the maximum total wavenumber of spherical harmonic that the grid can represent. These grids provide a framework for transforming data from grid space (nodal) to spherical harmonic representations with minimal loss of information. When solving model equations we use cubic truncation Gaussian grids T62, T125 and T253, that capture a similar number of spherical harmonics, while avoiding aliasing errors and minimizing the need to increase array dimensions above a multiple of 128, which is expensive on the Google TPU. See Table A.1 for resolution details. All models use 32 equidistant sigma levels for vertical discretization. We suspect that using higher vertical resolution with assimilation data from more levels could further improve the performance.

Grid name	Longitude nodes	Latitude nodes	Max total wavenumber
TL63	128	64	63
TL127	256	128	127
TL255	512	256	255
T62	190	95	62
T125	379	190	125
T254	766	383	254

Table A1 Spatial and spectral resolutions of horizontal grids used by NeuralGCM.

A.2 Primitive equations

The dynamical core of NeuralGCM solves the primitive equations, which represent a combination of (1) momentum equations, (2) the second law of thermodynamics,

(3) a thermodynamic equation of state (ideal gas), (4) continuity equation and (5) hydrostatic approximation. For solving the equations we use a divergence-vorticity representation of the horizontal winds, resulting in equations for the following seven prognostic variables: divergence δ, vorticity ζ, temperature T, logarithm of the surface pressure $\log p_s$, as well as 3 moisture species (specific humidity q, specific cloud ice q_{c_i} and specific liquid cloud water content q_{c_l}). To facilitate efficient time integration of our models we split temperature T into a uniform reference temperature on each sigma level $\bar{T}_\sigma$ and temperature deviations per level $T'_\sigma = T_\sigma - \bar{T}_\sigma$. The resulting equations are:

$$
\begin{aligned}
\frac{\partial \zeta}{\partial t} &= -\nabla \times \left((\zeta + f)\mathbf{k} \times \mathbf{u} + \dot{\sigma}\frac{\partial \mathbf{u}}{\partial \sigma} + RT'\nabla \log p_s \right) \\
\frac{\partial \delta}{\partial t} &= -\nabla \cdot \left((\zeta + f)\mathbf{k} \times \mathbf{u} + \dot{\sigma}\frac{\partial \mathbf{u}}{\partial \sigma} + RT'\nabla \log p_s \right) - \nabla^2 \left(\frac{||\mathbf{u}||^2}{2} + \Phi + R\bar{T}\log p_s \right) \\
\frac{\partial T}{\partial t} &= -\mathbf{u} \cdot \nabla T - \dot{\sigma}\frac{\partial T}{\partial \sigma} + \frac{\kappa T \omega}{p} = -\nabla \cdot \mathbf{u}T' + T'\delta - \dot{\sigma}\frac{\partial T}{\partial \sigma} + \frac{\kappa T \omega}{p} \\
\frac{\partial q_i}{\partial t} &= -\nabla \cdot \mathbf{u}q_i + q_i\delta - \dot{\sigma}\frac{\partial q_i}{\partial \sigma} \\
\frac{\partial \log p_s}{\partial t} &= -\frac{1}{p_s}\int_0^1 \nabla \cdot (\mathbf{u}p_s)\, d\sigma = -\int_0^1 (\delta + \mathbf{u} \cdot \nabla \log p_s)\, d\sigma
\end{aligned}
\tag{A1}
$$

with horizontal velocity vector $\mathbf{u} = \nabla(\Delta^{-1}\delta) + \mathbf{k} \times \nabla(\Delta^{-1}\zeta)$, Coriolis parameter f, upward-directed unit vector parallel to the z-axis $\mathbf{k}$, ideal gas constant R, heat capacity at constant pressure C_p, $\kappa = \frac{R}{C_p}$, diagnosed vertical velocity in sigma coordinates $\dot{\sigma}$, diagnosed change in pressure of a fluid parcel $\omega \equiv \frac{dp}{dt}$, diagnosed geopotential Φ, diagnosed virtual temperature T_ν and each moisture species denoted as q_i.

Diagnostic quantities are computed as follows:

$$\dot{\sigma}_{k+\frac{1}{2}} = -\sigma_{k+\frac{1}{2}}\frac{\partial \log p_s}{\partial t} - \frac{1}{p_s}\int_0^{\sigma_{k+\frac{1}{2}}} \nabla \cdot (p_s\mathbf{u})\, d\sigma \tag{A2}$$

$$\frac{\omega_k}{p_s\sigma_k} = \mathbf{u}_k \cdot \nabla \log p_s - \frac{1}{\sigma_k}\int_0^{\sigma_k} (\delta + \mathbf{u} \cdot \nabla \log p_s)\, d\sigma \tag{A3}$$

$$\Phi_k = \Phi_s + R\int_{\log \sigma_k}^{0} T_\nu\, d\log\sigma \tag{A4}$$

$$T_\nu = T(1 + \left(\frac{R_{vap}}{R} - 1\right) q - q_{c_i} - q_{c_l}) \tag{A5}$$

where $\Phi_s = gz_s$ is the geopotential at the surface.

A.3 Numerics

Our choice of the numerical schemes for interpolation, integrals and diagnostics exactly follows Durran's book [30] §8.6, with the addition of moisture species (which are advected by the wind and only affect the dynamics through through their effect on the virtual temperature). We use semi-implicit time-integration scheme, where all right hand side terms are separated into groups that are treated either explicitly or implicitly. This avoids severe time step limitations due to fast moving gravity waves.

Our choice of dynamical core was also informed by our desire to run efficiently on machine learning accelerators, in particular Google TPUs [50]. TPUs have dedicated hardware for low-precision matrix-matrix multiplication, which conveniently is well suited for the bottleneck in spectral methods, which are forward and inverse spherical harmonic transformations. Accordingly, we use single-precision arithmetic throughout. We found that full single precision for spherical harmonic transformations was not required to obtain accurate results even on our largest grid sizes, and according use only three passes of bfloat16 matrix-multiplication rather than the six passes that would required for full single precision [51]. Our implementation supports parallelism across spatial dimensions (x, y, and z) for running on multiple accelerator cores, using XLA SPMD [52], with JAX's `shard_map` for parallelizing key bottlenecks including matrix-multiplications in spherical harmonic transforms [53].

Appendix B Time integration

In NeuralGCM, the state of the atmosphere is advanced in time by integrating model equations that combine effects from the dynamical core and learned physics parameterizations. This is done iteratively using Implicit-Explicit integration scheme [33] described in B.1. Integration time step varies with resolution, as shown in Table B3. This results in iterative updates to the model state every 4-30 minutes, depending on model resolution. In contrast, data-driven methods commonly make predictions at 6-hour jumps [11, 13]. Throughout time integration, dynamical core tendencies are computed at every time step, while learned physics tendencies are only recomputed once every 60 minutes for our lowest resolution (2.8°) model and every 30 minutes for all others. This is done to avoid excessive backpropagation through the neural networks in learned physics. At higher resolutions it might be advantageous to include more frequent updates to learned physics tendencies to be able to account for short-time processes (rather than statistical effect that varies smoothly in time). Similar to traditional spectral GCMs we introduce spectral filters to improve numerical stability [56], which are described in B.2.

B.1 Time integration scheme

As is typical for atmospheric models, in NeuralGCM we use semi-implicit ODE solvers to the solve the primitive equations, by partitioning dynamical tendencies into "implicit" and "explicit" terms. "Implicit" tendencies include linear terms of Eq. A1 that give rise to the low amplitude, fast moving gravity waves. These terms are treated implicitly, allowing for longer stable time steps, while the rest of the terms are computed explicitly.

Rather than the traditional semi-implicit leapfrog method, we use implicit-explicit Runge-Kutta methods to avoid the complexity of keeping track of multiple time-steps and time-filtering required by the traditional semi-implicit leapfrog method.

Specifically, we use the semi-implicit Lorenz three cycle scheme (SIL3), which was developed specifically for global spectral weather models [33].

$$
\begin{array}{c|cccc}
1/3 & 1/3 & & & \\
2/3 & 1/6 & 1/2 & & \\
1 & 1/2 & -1/2 & 1 & \\
\hline
 & 1/2 & -1/2 & 1 & 0
\end{array}
\qquad
\begin{array}{c|cccc}
1/3 & 1/6 & 1/6 & & \\
1/3 & 0 & 1/3 & & \\
1 & 3/8 & 0 & 3/8 & 1/4 \\
\hline
 & 3/8 & 0 & 3/8 & 1/4
\end{array}
$$

Table B2 Butcher tableau for the IMEX SIL3 scheme.

B.2 Filtering

During time integration we use two exponential filters of different strengths ("hard" and "soft"). These filters correspond to hyper-diffusion, a standard component of spectral atmospheric models used to stabilize dynamics [57]. Each transform a scalar field x_{hml} in spherical harmonic representation as:

$$x_{hml} \to x_{hml} * e^{-a\left(\frac{k-c}{1-c}\right)^{2p}} \tag{B6}$$

with filter attenuation a, filter cutoff c, filter order p, and normalized total wavenumber $k \equiv \frac{l}{l_{max}}$.

Filter parameters used by different NeuralGCM models are summarized in Table B3, where filter attenuation is specified via attenuation time α and time step dt via $a = \frac{\alpha}{dt}$. Both hard and soft filters are applied to the model state at the end of each integration step. We additionally apply hard filter to the outputs of learned physics parameterizations to avoid injection of high frequency noise in each model step. The filtering strength sets the true length scale of the simulation, which is generally slightly larger than the grid spacing.

Model resolution	Time step [minutes]	Filter	Attenuation time [minutes]	Order	Cutoff
2.8°	12	hard	4	10	0.4
		soft	120	3	0.0
1.4°	6	hard	8	6	0.4
		soft	120	3	0.0
0.7°	3.75	hard	4	6	0.4
		soft	120	3	0.0

Table B3 Time step and filtering parameters of NeuralGCM models.

References

[1] Bauer, P., Thorpe, A. & Brunet, G. The quiet revolution of numerical weather prediction. *Nature* **525**, 47–55 (2015). URL http://dx.doi.org/10.1038/nature14956.

[2] Balaji, V. *et al.* Are general circulation models obsolete? *Proc. Natl. Acad. Sci. U. S. A.* **119**, e2202075119 (2022). URL http://dx.doi.org/10.1073/pnas.2202075119.

[3] Hourdin, F. *et al.* The art and science of climate model tuning. *Bull. Am. Meteorol. Soc.* **98**, 589–602 (2017). URL https://doi.org/10.1175/BAMS-D-15-00135.1.

[4] Bony, S. & Dufresne, J.-L. Marine boundary layer clouds at the heart of tropical cloud feedback uncertainties in climate models. *Geophysical Research Letters* **32** (2005).

[5] Webb, M. J., Lambert, F. H. & Gregory, J. M. Origins of differences in climate sensitivity, forcing and feedback in climate models. *Climate Dynamics* **40**, 677–707 (2013).

[6] Sherwood, S. C., Bony, S. & Dufresne, J.-L. Spread in model climate sensitivity traced to atmospheric convective mixing. *Nature* **505**, 37–42 (2014).

[7] Palmer, T. & Stevens, B. The scientific challenge of understanding and estimating climate change. *Proc. Natl. Acad. Sci. U. S. A.* **116**, 24390–24395 (2019). URL http://dx.doi.org/10.1073/pnas.1906691116.

[8] Fischer, E. M., Beyerle, U. & Knutti, R. Robust spatially aggregated projections of climate extremes. *Nature Climate Change* **3**, 1033–1038 (2013).

[9] Field, C. B. *Managing the risks of extreme events and disasters to advance climate change adaptation: special report of the intergovernmental panel on climate change* (Cambridge University Press, Cambridge, UK, 2012).

[10] Rasp, S. *et al.* WeatherBench 2: A benchmark for the next generation of data-driven global weather models (2023). Preprint at http://arxiv.org/abs/2308.15560.

[11] Keisler, R. Forecasting global weather with graph neural networks. *arXiv preprint arXiv:2202.07575* (2022).

[12] Bi, K. *et al.* Accurate medium-range global weather forecasting with 3d neural networks. *Nature* **619**, 533–538 (2023).

[13] Lam, R. *et al.* Learning skillful medium-range global weather forecasting. *Science* **382**, 1416–1421 (2023). URL https://www.science.org/doi/abs/10.1126/science.adi2336.

[14] Hersbach, H. *et al.* The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society* **146**, 1999–2049 (2020).

[15] Zhou, L. *et al.* Toward convective-scale prediction within the next generation global prediction system. *Bull. Am. Meteorol. Soc.* **100**, 1225–1243 (2019). URL https://journals.ametsoc.org/view/journals/bams/100/7/bams-d-17-0246.1.xml.

[16] Bonavita, M. On the limitations of data-driven weather forecasting models. *arXiv preprint arXiv:2309.08473* (2023).

[17] Weyn, J. A., Durran, D. R. & Caruana, R. Improving data-driven global weather prediction using deep convolutional neural networks on a cubed sphere. *Journal of Advances in Modeling Earth Systems* **12**, e2020MS002109 (2020).

[18] Watt-Meyer, O. *et al.* ACE: A fast, skillful learned global atmospheric model for climate prediction. *arXiv preprint arXiv:2310.02074* (2023).

[19] Bretherton, C. S. Old dog, new trick: Reservoir computing advances machine learning for climate modeling. *Geophysical Research Letters* **50**, e2023GL104174 (2023).

[20] Reichstein, M. *et al.* Deep learning and process understanding for data-driven earth system science. *Nature* **566**, 195–204 (2019). URL https://doi.org/10.1038/s41586-019-0912-1.

[21] Brenowitz, N. D. & Bretherton, C. S. Spatially extended tests of a neural network parametrization trained by coarse-graining. *Journal of Advances in Modeling Earth Systems* **11**, 2728–2744 (2019).

[22] Rasp, S., Pritchard, M. S. & Gentine, P. Deep learning to represent subgrid processes in climate models. *Proceedings of the National Academy of Sciences* **115**, 9684–9689 (2018).

[23] Yuval, J. & O'Gorman, P. A. Stable machine-learning parameterization of subgrid processes for climate modeling at a range of resolutions. *Nature communications* **11**, 3295 (2020).

[24] Kwa, A. *et al.* Machine-learned climate model corrections from a global storm-resolving model: Performance across the annual cycle. *Journal of Advances in Modeling Earth Systems* **15**, e2022MS003400 (2023).

[25] Arcomano, T., Szunyogh, I., Wikner, A., Hunt, B. R. & Ott, E. A hybrid atmospheric model incorporating machine learning can capture dynamical processes not captured by its physics-based component. *Geophysical Research Letters* **50**, e2022GL102649 (2023).

[26] Han, Y., Zhang, G. J. & Wang, Y. An ensemble of neural networks for moist physics processes, its generalizability and stable integration. *Journal of Advances in Modeling Earth Systems* **15**, e2022MS003508 (2023).

[27] Gelbrecht, M., White, A., Bathiany, S. & Boers, N. Differentiable programming for earth system modeling. *Geoscientific Model Development* **16**, 3123–3135 (2023). URL https://gmd.copernicus.org/articles/16/3123/2023/.

[28] Bradbury, J. *et al.* JAX: composable transformations of Python+NumPy programs (2018). URL http://github.com/google/jax.

[29] Bourke, W. A Multi-Level Spectral Model. I. Formulation and Hemispheric Integrations. *Mon. Weather Rev.* **102**, 687–701 (1974). URL https://journals.ametsoc.org/view/journals/mwre/102/10/1520-0493_1974_102_0687_amlsmi_2_0_co_2.xml.

[30] Durran, D. R. *Numerical methods for fluid dynamics: With applications to geophysics* Second edn, Vol. 32 (Springer, New York, 2010).

[31] Wang, P., Yuval, J. & O'Gorman, P. A. Non-local parameterization of atmospheric subgrid processes with neural networks. *Journal of Advances in Modeling Earth Systems* **14**, e2022MS002984 (2022).

[32] Daley, R. Normal mode initialization. *Reviews of Geophysics* **19**, 450–468 (1981).

[33] Whitaker, J. S. & Kar, S. K. Implicit–explicit runge–kutta methods for fast–slow wave problems. *Monthly weather review* **141**, 3426–3434 (2013).

[34] Gilleland, E., Ahijevych, D., Brown, B. G., Casati, B. & Ebert, E. E. Intercomparison of spatial forecast verification methods. *Weather and forecasting* **24**, 1416–1430 (2009).

[35] Gneiting, T. & Raftery, A. E. Strictly Proper Scoring Rules, Prediction, and Estimation. *J. Am. Stat. Assoc.* **102**, 359–378 (2007). URL https://doi.org/10.1198/016214506000001437.

[36] Rasp, S. & Lerch, S. Neural networks for postprocessing ensemble weather forecasts. *Monthly Weather Review* **146**, 3885–3900 (2018).

[37] Pacchiardi, L., Adewoyin, R., Dueben, P. & Dutta, R. Probabilistic forecasting with generative networks via scoring rule minimization. *arXiv preprint arXiv:2112.08217* (2021).

[38] Fortin, V., Abaza, M., Anctil, F. & Turcotte, R. Why should ensemble spread match the RMSE of the ensemble mean? *J. Hydrometeorol.* **15**, 1708–1713 (2014). URL http://journals.ametsoc.org/doi/10.1175/JHM-D-14-0008.1.

[39] Holton, J. R. *An introduction to dynamic meteorology* Fifth edn (Elsevier Academic Press, Waltham, MA, USA, 2004).

[40] Cheng, K.-Y. *et al.* Impact of warmer sea surface temperature on the global pattern of intense convection: insights from a global storm resolving model.

Geophysical Research Letters **49**, e2022GL099796 (2022).

[41] Stevens, B. *et al.* Dyamond: the dynamics of the atmospheric general circulation modeled on non-hydrostatic domains. *Progress in Earth and Planetary Science* **6**, 1–17 (2019).

[42] Ullrich, P. A. *et al.* Tempestextremes v2. 1: A community framework for feature detection, tracking, and analysis in large datasets. *Geoscientific Model Development* **14**, 5023–5048 (2021).

[43] Haimberger, L., Tavolato, C. & Sperka, S. Toward elimination of the warm bias in historic radiosonde temperature records—some new results from a comprehensive intercomparison of upper-air data. *Journal of Climate* **21**, 4587–4606 (2008).

[44] Eyring, V. *et al.* Overview of the coupled model intercomparison project phase 6 (cmip6) experimental design and organization. *Geoscientific Model Development* **9**, 1937–1958 (2016).

[45] Mitchell, D. M., Lo, Y. E., Seviour, W. J., Haimberger, L. & Polvani, L. M. The vertical profile of recent tropical temperature trends: Persistent model biases in the context of internal variability. *Environmental Research Letters* **15**, 1040b4 (2020).

[46] Ruiz, J. J., Pulido, M. & Miyoshi, T. Estimating model parameters with ensemble-based data assimilation: A review. *Journal of the Meteorological Society of Japan. Ser. II* **91**, 79–99 (2013).

[47] Schneider, T., Lan, S., Stuart, A. & Teixeira, J. Earth system modeling 2.0: A blueprint for models that learn from observations and targeted high-resolution simulations. *Geophysical Research Letters* **44**, 12–396 (2017).

[48] Schneider, T., Leung, L. R. & Wills, R. C. J. Opinion: Optimizing climate models with process-knowledge, resolution, and AI. *EGUsphere [preprint]* (2024). URL https://egusphere.copernicus.org/preprints/2024/egusphere-2024-20/.

[49] Sutskever, I., Vinyals, O. & Le, Q. V. Sequence to sequence learning with neural networks. *Advances in neural information processing systems* **27** (2014).

[50] Jouppi, N. *et al.* *Tpu v4: An optically reconfigurable supercomputer for machine learning with hardware support for embeddings*, ISCA '23 (Association for Computing Machinery, New York, NY, USA, 2023). URL https://doi.org/10.1145/3579371.3589350.

[51] Henry, G., Tang, P. T. P. & Heinecke, A. *Leveraging the bfloat16 artificial intelligence datatype for higher-precision computations* (IEEE, Kyoto, Japan, 2019). URL https://ieeexplore.ieee.org/document/8877427/.

[52] Xu, Y. *et al.* GSPMD: General and scalable parallelization for ML computation graphs (2021). URL http://arxiv.org/abs/2105.04663.

[53] Douglas, S., Vikram, S., Bradbury, J., Zhang, Q. & Johnson, M. shmap (shard_map) for simple per-device code (2023). URL https://jax.readthedocs.io/en/latest/jep/14273-shard-map.html. Accessed: 2023-10-21.

[54] Palmer, T. N. *et al.* Stochastic parametrization and model uncertainty (2009). URL https://www.ecmwf.int/sites/default/files/elibrary/2009/11577-stochastic-parametrization-and-model-uncertainty.pdf.

[55] Battaglia, P. W. *et al.* Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261* (2018).

[56] Jablonowski, C. & Williamson, D. L. The pros and cons of diffusion, filters and fixers in atmospheric general circulation models. *Numerical techniques for global atmospheric models* 381–493 (2011).

[57] Jablonowski, C. & Williamson, D. L. in *The pros and cons of diffusion, filters and fixers in atmospheric general circulation models* (eds Lauritzen, P., Jablonowski, C., Taylor, M. & Nair, R.) *Numerical Techniques for Global Atmospheric Models* 381–493 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2011). URL https://doi.org/10.1007/978-3-642-11640-7_13.

[58] Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[59] May, R. M. *et al.* Metpy: A meteorological python library for data analysis and visualization. *Bulletin of the American Meteorological Society* **103**, E2273 – E2284 (2022). URL https://journals.ametsoc.org/view/journals/bams/103/10/BAMS-D-21-0125.1.xml.

[60] Roberts, M. J. *et al.* Impact of model resolution on tropical cyclone simulation using the highresmip–primavera multimodel ensemble. *Journal of Climate* **33**, 2557–2583 (2020).

[61] Vallis, G. K., Zurita-Gotor, P., Cairns, C. & Kidston, J. Response of the large-scale structure of the atmosphere to global warming. *Quarterly Journal of the Royal Meteorological Society* **141**, 1479–1501 (2015).

[62] Beucler, T. *et al.* Climate-invariant machine learning. *Science Advances* **10**, eadj7250 (2024).