Skillful joint probabilistic weather forecasting from marginals

Ferran Alet[1*], Ilan Price[1*], Andrew El-Kadi[1*], Dominic Masters[1*], Stratis Markou[1], Tom R. Andersson[1], Jacklynn Stott[1], Remi Lam[1], Matthew Willson[1], Alvaro Sanchez-Gonzalez[1*] and Peter Battaglia[1]

[*]Equal contributions, [1]Google DeepMind

Machine learning (ML)-based weather models have rapidly risen to prominence due to their greater accuracy and speed than traditional forecasts based on numerical weather prediction (NWP), recently outperforming traditional ensembles in global probabilistic weather forecasting. This paper presents FGN, a simple, scalable and flexible modeling approach which significantly outperforms the current state-of-the-art models. FGN generates ensembles via learned model-perturbations with an ensemble of appropriately constrained models. It is trained directly to minimize the continuous rank probability score (CRPS) of per-location forecasts. It produces state-of-the-art ensemble forecasts as measured by a range of deterministic and probabilistic metrics, makes skillful ensemble tropical cyclone track predictions, and captures joint spatial structure despite being trained only on marginals.

1. Introduction

Accurate, reliable weather forecasts help protect and enrich human life, and provide tremendous value across the public and private sectors. In recent years, machine learning (ML)-based weather forecasting has made rapid advances, now rivaling and surpassing the top operational forecast models based on traditional, physics-based methods (Bi et al., 2022, 2023; Bodnar et al., 2025; Chen et al., 2023; Han et al., 2024; Keisler, 2022; Kurth et al., 2022; Lam et al., 2022, 2023; Li et al., 2023; Zhong et al., 2024). These advances have been driven by increasingly powerful ML methods which are driving much of the wider artificial intelligence (AI) revolution, as well as high-quality datasets of historical weather (ECMWF, 2019; Hersbach et al., 2020), and invaluable benchmarks that allow for measuring progress and comparing techniques (ECMWF, 2024; Rasp et al., 2020, 2023).

While most of the aforementioned ML weather models have been *deterministic*, typically modeling something close to the "average" weather forecast, the applied weather community has increasingly emphasized their probabilistic (ensemble) models, such the European Centre for Medium-Range Weather Forecasts (ECMWF)'s ENS (ECMWF, 2019) and NOAA's GEFS (Zhou et al., 2022), to capture not only the most likely future weather conditions, but the range of probable conditions that may unfold. The reason probabilistic forecasts are necessary is that the underlying physical dynamics are complex and non-linear, and our weather observations are only partial, so unobserved factors can have substantial influence on future weather conditions, and even the best weather models cannot make precise forecasts with perfect accuracy. Probabilistic weather models can inform critical decisions, by allowing decision-makers to plan for both the most likely scenario, as well as less probable scenarios, e.g. dangerous storms, which are important to anticipate and prepare for.

Advances in ML-based probabilistic weather forecasting are more nascent, with early attempts leveraging a range of generative modeling techniques including generative adversarial networks (Ravuri et al., 2021), flow-matching (Couairon et al., 2024), variational auto-encoders (Oskarsson et al., 2024; Zhong et al., 2024), as well as hybrid ML-physics models (Kochkov et al., 2023, 2024). ML-based global probabilistic forecasts have only recently begun to compete with operational systems (Kochkov et al., 2023; Lang et al., 2024). The first model to clearly outperform the top physics-based operational
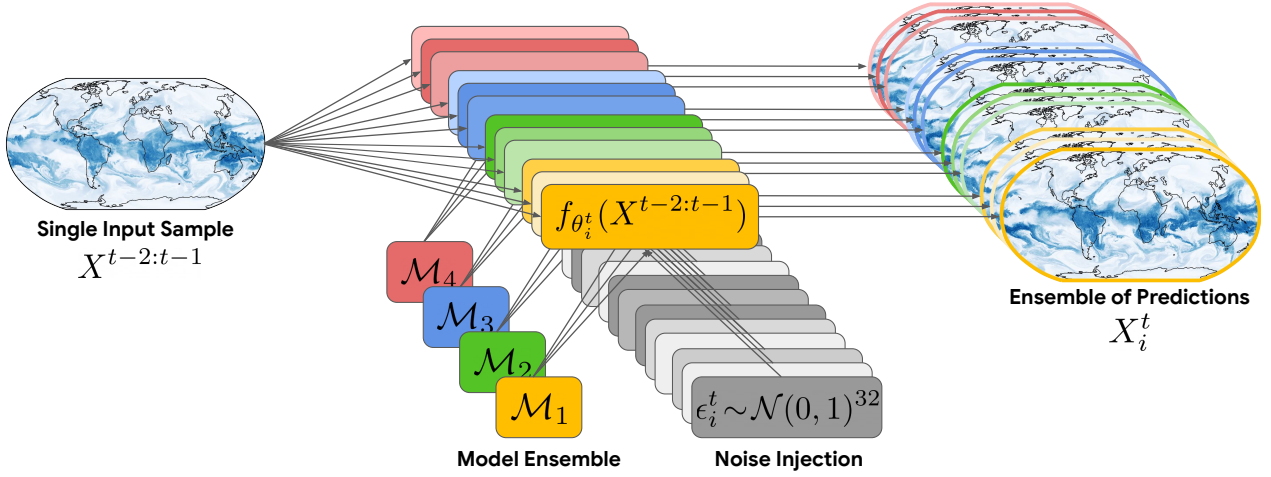

Figure 1 | An overview of the FGN generative process for producing a single step of a forecast ensemble from a single pair of input frames ($X^{t-2:t-1}$). Diversity is introduced at two levels, modeling aleatoric and epistemic uncertainty respectively. For a given model $\mathcal{M}_j$, aleatoric uncertainty is introduced at each step of a forecast trajectory by sampling a low-dimensional noise vector ϵ_i^t used for parameter-shared conditional normalization during the model forward pass. This can be interpreted as applying a perturbation to the neural network weights to obtain θ_i^t, and hence, as sampling the parameters of the neural network. To generate N ensemble members with aleatoric uncertainty, we simply condition on N different ϵ_i^t independently. The epistemic uncertainty is modeled by ensembling outputs of multiple models $\mathcal{M}_j$ (each with their own $\{\theta_j^*, \Delta_j\}$ parameters) trained independently, each of which generates a subset of ensemble members according to the procedure described above.

system, ECMWF's ENS, was GenCast (Price et al., 2023, 2025). While GenCast is also much more computationally efficient than traditional NWP, it is relatively slow compared to some other ML-based models, and does not provide a straightforwardly adaptable framework for weather models to move beyond dense gridded data.

Here we present FGN, a new combined architectural, training, and inference approach for probabilistic modeling of weather, which is faster, more flexible, and has greater performance than GenCast. We train FGN on ECMWF's ERA5 reanalysis and operational HRES initial conditions to optimize the Continuous Ranked Probability Score (CRPS), a popular top-line metric for probabilistic weather forecasts, similar to several related approaches which include or modify CRPS in their objective, including NeuralGCM (Kochkov et al., 2023), Graph-EFM (Oskarsson et al., 2024), FuXi-ENS (Zhong et al., 2024), and AIFS-CRPS (Lang et al., 2024). It models epistemic and aleatoric uncertainty with distinct mechanisms, model ensembles (Lakshminarayanan et al., 2017) for the former, and a method which is related to other work on stochastic functions (Alet et al., 2022; Huang and Belongie, 2017; Lang et al., 2024) for the latter. Our results show FGN comprehensively outperforms GenCast (and ENS) across a benchmark of deterministic and probabilistic metrics. Its forecast distributions generally have similar-to-better calibration than previous models, it predicts extremes as well or better, and it has better joint distribution structure in most cases we examined. FGN's tropical cyclone track forecasts are also significantly better ($p < 0.05$), for both mean trajectories and track probabilities.

2. Methods

2.1. Problem formulation

The ensemble weather forecasting problem we tackle in this work is to draw samples from the joint probability distribution, $p(X^{1:T}|X^{\leq 0})$, over T-step weather trajectories $X^{1:T}$ conditional on prior weather. In its fullest description, the ensemble forecasting problem requires conditioning directly on weather observations and measurements made during a time window that leads up to the initialization time $t = 0$. However, following standard practice in operational weather forecasting, we assume for now that the conditioning on observations is handled as part of a data assimilation process that we take as a given, generating the initial states $X^{\leq 0}$ on which our predictive distribution conditions.

When modeled as an n-th order Markov system, this distribution over whole trajectories factorizes into the product of distributions over each successive state, conditional on the prior n states. In this work we choose $n = 2$, and so we have

$$p(X^{1:T}|X^0, X^{-1}) = \prod_{t=1}^{T} p(X^t|X^{t-2:t-1}). \tag{1}$$

We thus train a model to sample from $p(X^t|X^{t-2:t-1})$, and then for each $t \in 1{:}T$ we sample X^t autoregressively conditional on the previous two steps $X^{t-2:t-1}$. Repeating this process multiple times generates an ensemble of trajectories.

In the results that follow, each X^t consists of 6 atmospheric variables at 13 vertical pressure levels and 6 surface variables (see Table A.1), at each point on a 0.25° latitude-longitude grid, at timestep t. We target 15 day forecasts with timesteps of 6 hours.

2.2. Functional generative networks: skillful joint distributions through functional variability

Sources of uncertainty are commonly distinguished as being either epistemic or aleatoric. Epistemic uncertainty is the uncertainty about the model due to finite training data and an imperfect learning algorithm. Aleatoric uncertainty is due to the inherent irreducible randomness in a process from the perspective of a model. We propose *functional generative networks* (FGNs), a simple, scalable approach which models both the epistemic and aleatoric uncertainty in the distribution of forecasts, by expressing uncertainty through functional variability. FGNs learn to model the joint distribution over the outputs while training on a loss that applies only to the marginal distributions, in a suitably constrained setting.

2.2.1. Modeling epistemic uncertainty with multi-model ensembling

One can account for epistemic uncertainty by drawing samples from the posterior predictive distribution,

$$p(X^{1:T}|X^{\leq 0}, \mathcal{D}) = \int p(X^{1:T}|X^{\leq 0}, \mathcal{M})\, p(\mathcal{M}|\mathcal{D})\mathrm{d}\mathcal{M} \tag{2}$$

which marginalizes over some posterior uncertainty about the model $\mathcal{M}$ conditional on our training dataset $\mathcal{D}$. We adopt a common approximation known as deep ensembles (Lakshminarayanan et al., 2017), in which we train multiple models $\mathcal{M}_1, \ldots, \mathcal{M}_J$ from independently-initialized parameters, and then later ensemble their predictions. Its connection to Bayesian inference has been studied elsewhere (Wild et al., 2023; Wilson and Izmailov, 2020), but here we apply it informally to approximate Bayesian inference.

In this work we train $J = 4$ models. To ensemble their predictions, we generate an equal number of ensemble member trajectories from each model, using the same model for all timesteps of a trajectory.

2.2.2. Modeling aleatoric uncertainty with learned functional perturbations

The weather states we model from ECMWF's ERA5 and HRES analysis do not fully capture the state of the weather system. It is common to use stochastic dynamics to model uncertainty due to unresolved phenomena, and from the perspective of the model this can be viewed as aleatoric uncertainty.

We incorporate this aleatoric uncertainty in our model by sampling functions. Concretely, we model different forecast samples X_i^t of $p\left(X^t | X^{t-2:t-1}\right)$ as coming from neural networks $f_{\theta_i^t}$ whose parameters θ_i^t are sampled from a learned distribution: $X_i^t = f_{\theta_i^t}(X^{t-2:t-1})$, where $\theta_i^t \sim \mathcal{P}_\mathcal{M}(\theta)$ independently for each ensemble member i and timestep t.

Alet (2022) justifies why modeling aleatoric uncertainty in parameter space can lead to more structured variations than modeling it in input or output space. This structure is due to the reuse of sampled parameters across spatial (or other relevant) dimensions in architectures with parameter-sharing layers. Similar insights have been leveraged in RL exploration (Fortunato et al., 2017), generative adversarial networks (Saatci and Wilson, 2017), and test-time adaptation (Alet et al., 2021; Bibas et al., 2019; Sun et al., 2020; Wang et al., 2020).

This idea is analogous to stochastic physics, where parameters inside a numerical weather prediction system are perturbed to obtain an ensemble, in addition to initial condition sampling (Ollinaho et al., 2017). ML weather models have naturally explored perturbing models in order to make them probabilistic (Kochkov et al., 2023; Lang et al., 2024), although these have generally used spatially-independent model perturbations.

2.2.3. Reparameterization trick and CRPS marginal training

To learn the distribution $\mathcal{P}_\mathcal{M}(\theta)$, we use the well-known reparameterization trick (see Kingma et al. 2013; Rezende et al. 2014; Titsias and Lázaro-Gredilla 2014, as well as Mohamed et al. 2020 for an instructive introduction): $\theta = \theta^* + \Delta \cdot \epsilon$, where Δ is a potentially-structured matrix used to map Gaussian noise ϵ to a perturbation in θ space. $\mathcal{M} := \{\theta^*, \Delta\}$ thus consists of θ^*, acting as the mean θ and describing a deterministic function, and a matrix Δ controlling the covariance. At inference time, we make the modeling assumption that we can compose both types of uncertainty by sampling one of multiple $\mathcal{M}_j$ trained independently, and then sampling ϵ_i^t to get $\theta_i^t = \theta_j^* + \Delta_j \cdot \epsilon_i^t$, as shown in Figure 1.

We use the Continuous Ranked Probability Score (CRPS) as our training objective, which is a strictly proper scoring rule for univariate distributions (Gneiting and Raftery, 2007). For a cumulative density function (CDF) F and target $y \in \mathbb{R}$,

$$\text{CRPS}(F, y) = \int_{-\infty}^{\infty} \left(F(z) - \mathbb{1}\left[z \geq y\right]\right)^2 dz. \tag{3}$$

For an empirical distribution coming from samples $x^{1:N}$ this is equal to: $\frac{1}{N} \sum_n |x^n - y| - \frac{1}{2N^2} \sum_{n,n'} |x^n - x^{n'}|$. However, this is a biased estimator of the CRPS of the underlying distribution F. For an unbiased CRPS estimate we can use the 'fair' CRPS estimator (Zamo and Naveau, 2018),

$$\text{fCRPS}(x^{1:N}, y) := \frac{1}{N} \sum_n |x^n - y| - \frac{1}{2N(N-1)} \sum_{n,n'} |x^n - x^{n'}|. \tag{4}$$

Whenever we refer to CRPS we mean this fair estimator, unless stated otherwise (e.g. in Section 4.1).

In practice during training we use $N = 2$, and average over all locations, variables, and levels,

$$\mathcal{L} := \frac{1}{|D|} \sum_d \frac{1}{G} \sum_i a_i \text{fCRPS}(x_{i,d}^{1:2}, y_{i,d}) \tag{5}$$

where d indexes into the dataset batch D, i indexes into the G variable-level-location tuples and a_i is the corresponding loss weighting which we take from Lam et al. (2022) and Price et al. (2023) (with the exception of the geopotential variables, for which we halved the loss weight to help tame overfitting as we scaled the model size).

As noted in the introduction, a number of other works have also included or modified the CRPS objective when training ML-based weather models (Kochkov et al., 2023; Lang et al., 2024; Oskarsson et al., 2024; Zhong et al., 2024). Our approach is also closely related to work on Scoring Rule minimization (Pacchiardi et al., 2024), DISCO Nets (Bouchacourt et al., 2016) and MMD nets (Gritsenko et al., 2020), where ensemble models are also trained with CRPS from noisy inputs. However, these latter works typically do not yield state-of-the-art joint distributions in high-dimensional problems. In this work, we show that modeling the noise in parameter space rather than the original inputs or outputs can be a simple, effective method to learn skillful joint distributions in a very high dimensional setting, given a suitably strong algorithmic inductive bias.

2.3. Model Architecture

Each of the constituent models in the FGN model-ensemble is implemented with a very similar neural network architecture to that of the denoiser in GenCast (Price et al., 2025), with a GNN encoder/decoder mapping from the lat/lon grid to a latent space defined on a spherical 6-times-refined icosahedral mesh, and a graph-transformer processor which operates on the nodes of this mesh.

There are important differences, however, between the FGN and GenCast architectures. FGN is a larger model, with ∼180m parameters per model seed, compared to GenCast's ∼57m in total. FGN has latent dimension 768 and 24 layers in its graph-transformer processor, compared with latent dimension of 512 and 16 layers in GenCast, and produces forecasts with a 6-hour timestep, whereas GenCast has a 12-hour timestep[1]. In Appendix A.4, we ablate the impact of these model ensembling and model size differences.

One key modeling choice was to use conditional normalization—which in GenCast are used to condition on the corresponding diffusion noise level σ—as the means of perturbing the model. To be precise, in FGN a global noise vector $z \sim \mathcal{N}(0, 1)^{32}$ is passed into all of the network's conditional layer-norm layers, such that sampling different vectors z for each ensemble member is what generates the variance across the ensemble.

This approach has two important features: 1) a low-dimensional noise source, which 2) is globally applied across all layers, and with learned conditional normalization parameters which are shared across the spatial dimensions of the model (i.e. mesh and grid node dimensions). Together these constrain the output distribution in a way that encourages the model to generate globally coherent variability despite being trained on a marginals-only loss. This is discussed in more detail in Section 6.

We also note that conditional normalization has had success in other contexts for semantically-coherent functional modulation, for example in diffusion to condition on noise levels, as noted above, or in generative models of images and audio to condition on style or voice (Chen et al., 2021; Karras et al., 2019). Lang et al. (2024) also use conditional normalization to inject stochasticity into their

---

[1]A number of other very minor differences in the encoder architecture are outlined in Appendix A.3.

weather model, but do so with location specific conditioning, which adds a high spatial frequency component to the noise injection.

2.4. Auto-regressive training

FGN is initially trained with a 'single-step' loss only. The final stages of training, however, involve autoregressive (AR) rollouts as part of each training step, where the model produces a number of forecast steps autoregressively, and the loss is averaged over all rollout steps, with gradients propagated back through the rollout. FGN is finetuned on rollouts up to 8 steps. The exact training stages and their hyperparameters are described in Appendix A.2. We find AR training to be helpful, but not essential, to performance—skillful joint forecast distributions are also achieved without it (see Appendix A.4).

3. Experiments

3.1. Training and evaluation data

We make use of two datasets: ERA5 and HRES-fc0 to train FGN (details in Appendix A.1). We use ERA5 for "pre-training", because it is a large corpus of historical weather reanalysis, from a single NWP that was used operationally in 2016, but is not available in real-time for operational settings. We use HRES-fc0 for "fine-tuning", because it is a smaller dataset that is comprised of analysis from NWPs used operationally from 2016 to the present, and is available in real-time in operational settings. While developing our model, we evaluated against HRES-fc0 as ground truth, using 2022 as the validation year. Once finished, we froze the architecture and training protocol and trained a model on data up to and including 2022. The resulting model was evaluated on HRES-fc0 from 2023 as the test year, yielding the results reported below.

3.2. Baseline: GenCast

We use GenCast (Price et al., 2023, 2025) as a strong baseline model, against which we compare FGN's performance. GenCast is the first probabilistic machine learning based weather prediction (MLWP) model to significantly outperform the top NWP system, ECMWF's ENS, at high resolution, and represents the state-of-the-art for probabilistic medium-range weather forecasting. GenCast is a diffusion model, and produces a single forecast trajectory by sampling one forecast timestep at a time (using an iterative denoising process), and rolling out step by step by feeding each output step back in as input to generate the subsequent step. It can generate an ensemble of stochastic trajectories by sampling single trajectories independently. It is trained explicitly to model the joint distribution over the weather at the next timestep, given the weather at the previous two timesteps. GenCast operates at 0.25° resolution, and generates ensembles of 15 day prediction with 12-hour timesteps.

While GenCast was trained and evaluated on ERA5 reanalysis data in Price et al. (2025), it has since been finetuned on operationally available HRES-fc0 data, as part of Google's WeatherNext family of operational models where it is called WeatherNext Gen. Here we focus our evaluations on the operational setting, and thus compare performance to the WeatherNext Gen forecasts, though we refer to the model as GenCast throughout this paper[2]. These forecasts are publicly available via

---

[2]Price et al. (2025) evaluated GenCast against ERA5 as ground truth, while ENS was evaluated against HRES-fc0. Appendix A.5.1 compares the WeatherNext Gen forecasts which we use as the baseline in this paper, against ENS for the full year of 2023, evaluating both models against HRES-fc0. The figures show that the results presented in Price et al. (2025) carry over to the operational setting.

Earth Engine and BigQuery[3] where each year of data has been generated with a model finetuned on HRES-fc0 from 2016 up to the start of that year. As such, we make use of the 2022 and 2023 forecasts for validation and test years respectively.

3.3. Training and inference efficiency

Training FGN takes approximately 3 wall clock days, using a combined total of 490 TPUv5p and TPUv6e days of compute, for each of the four models in the model-ensemble (see Appendix A.2 for details).

For inference, generating a single 15-day forecast trajectory takes just under 1 minute on a single TPU v5p, and every ensemble member can be generated in parallel. We highlight that despite being a substantially larger model than GenCast, and generating twice as many forecast frames due to its 6-hour timestep, generating a 15-day forecast is still 8 times faster. This is due to the fact that FGN only requires a single forward pass through the network to generate a forecast, compared with the multiple forward passes needed for the iterative refinement process of a diffusion model.

4. Evaluation Methodology

To evaluate FGN, we follow a similar methodology as that used to evaluate GenCast in Price et al. (2025). One key difference is that in this paper all models are initialized with and evaluated against HRES-fc0, at all four initialization times of the day (00z, 06z, 12z, 18z), quantifying the performance of these ML-based weather models in realistic operational settings. We also evaluate ensembles of size 56 for both models (in the case of FGN, each model seed generates 14 members). We refer the reader to the appendix of Price et al. (2025) for a detailed description and motivation of the various metrics and their computation, including statistical significance testing.

4.1. Evaluating marginal forecast distribution

We first evaluate the predictive skill of marginal forecast distributions. Marginal distributions are pixel-wise, single-variable distributions most relevant to users interested in the weather at a given location and time. We assess these marginal distributions using the RMSE of the ensemble mean, and the CRPS of the ensemble, computed at each location and then globally averaged, for each variable and pressure level. We use the biased versions of these metrics because the FGN model ensemble violates some underlying assumptions of their bias-corrected variants[4].

Additionally, we assess the reliability and spread of these marginal distributions using spread-skill ratios, where well-calibrated forecasts will exhibit spread-skill ratios close to 1 (Fortin et al., 2014; Leutbecher, 2009). Subject to some mild, standard assumptions, spread-skill ratios less than 1 indicate under-dispersion (over-confidence) on average. Conversely, spread-skill ratios greater than 1 indicate over-dispersion (under-confidence) on average.

Finally, we quantify the value of a probabilistic model for decision-making relative to extreme weather scenarios using the Relative Economic Value (REV) (Richardson, 2000, 2006; Wilks, 2001). We use REV to quantify the benefit of using forecast distributions from a given model to decide whether to prepare (or not) for the event that a variable will exceed a given threshold. We evaluate REV for

---

[3]Note that another small difference between WeatherNext Gen and the one in Price et al. (2025) is that this version also predicts 100m wind. However, this has minimal effect on its skill on other variables and thus does not matter for the comparisons in this paper.

[4]Specifically, when we enforce that FGN generates an equal number of forecasts per model seed, this violates the independence assumption underlying the unbiased estimators.

different C/L ratios where C is the cost of preparing, and L is the loss incurred when encountering the event without preparation (but which could have been avoided if one had prepared). We compute REV curves for various thresholds corresponding to high and low percentiles of the climatological distribution (computed for each variable on a per-latitude, longitude, month-of-year and time-of-day basis, using 24 years of ERA5 from 1993 to 2016). REV is normalized relative to the value of a climatological forecast (REV=0) and a perfect forecast (REV=1).

4.2. Evaluating correlation structure of the joint distribution

Weather forecasts are used in many applications where the correlation between various parts of the predictions are important. For instance, one might be interested in the total amount of precipitation accumulated in an entire region, or in how several variables can be used to compute another quantity of interest. We assess the quality of the correlation structure of the joint forecast distributions via a number of standard proxies, each of which quantifies specific aspects of this joint distribution.

First, we assess the spatial correlations and dependencies in the forecast distribution by performing probabilistic evaluation on spatially pooled predicted fields. We follow the pooling protocol described in 'Spatially pooled CRPS evaluation' in Methods in Price et al. (2025), evaluating max-pooled and average-pooled CRPS for pool sizes ranging from 120 km to 3828 km. Note that when spatial dependencies exist, minimizing the CRPS of the marginal predictions is not sufficient to minimize the CRPS of the pooled predictions. Thus, the pooled CRPS provides a useful metric to assess the spatial structure of forecasts' joint distributions.

Second, we evaluate the cross-variable dependencies in the forecasts by computing the CRPS for two derived quantities of interest: 10m wind speed (derived from the 10u and 10v predicted fields), as well as the difference in geopotential at pressure levels 300hPa and 500hPa, which is a critical quantity for tropical cyclones predictions as well as weather more generally. Skill in forecasting these derived quantities relies on correctly modeling the dependencies of their constituent predicted fields.

Finally, we report the spherical harmonic power spectra of the forecasts and compare it to those of the ground truth HRES-fc0. Power spectra are a standard, albeit imperfect, measure of physical plausibility. They are an important tool to diagnose blurring, a common but "unphysical" way deterministic ML-based weather models achieve lower RMSE.

4.3. Evaluating tropical cyclone tracks

We evaluate the predictive skill of the models at the task of cyclone track prediction using two metrics: the position error of the ensemble mean trajectories, as well as the REV of forecast track probabilities, using the protocol from Price et al. (2025).

Position error is calculated relative to named cyclones from the International Best Track Archive for Climate Stewardship (IBTrACS, Knapp et al. 2010) which we treat as ground truth. Forecast cyclone tracks are produced by applying the Tempest Extremes tracker (Ullrich et al., 2021) to the forecasts of each model, using hyper-parameters detailed in Methods and Supplementary Information of Price et al. (2025). Forecast tracks are then paired with ground truth tracks if their initial position is within 100 km at lead time zero. We consider the mean position prediction to "exist" while 50% or more ensemble members continue to forecast a cyclone (as measured by the continued existence of a Tempest Extremes track for that member). We then average the position error of ensemble mean cyclone positions over the intersection of named cyclone and lead time pairs for which all models provide a track, as is standard practice.

Position error is a 'deterministic' evaluation in the sense that it does not evaluate any aspect of

the forecast distribution beyond its mean. For a probabilistic evaluation, we report the REV of track probability predictions i.e., the predicted probability that a cyclone center will pass through a given place at a given time. Specifically, we compute 1° resolution track probability heatmaps, and the predicted probability in each 1° cell of the heatmap is the fraction of ensemble members forecasting a cyclone center within that cell.

5. Results

5.1. Results on marginal predictive skill and calibration

Figure 2a shows that FGN achieves better (lower) CRPS than GenCast on 99.9% of the targets, with improvements of up to 18%, and an average of 6.5%. The results are shown in the form of a scorecard, where blue cells indicate those variable-level-leadtime combinations where FGN achieved better CRPS, while red would indicate that GenCast performed better. Hatched regions indicate where the difference in performance was not statistically significant ($p \geq 0.05$). We see similar results on ensemble mean RMSE, with FGN outperforming GenCast in 100% of variable-level-leadtime combinations, with improvements of up to 18% and an average of 5.8%, as shown in the scorecard in Figure A.11 in Appendix A.5.2. These results indicate FGN has greater forecast skill GenCast in its marginal forecast distributions[5].

Figure 2b-f shows that FGN is also generally very well calibrated, with spread-skill scores close to 1 at all lead times. GenCast exhibits good but slightly poorer calibration over the first 5 - 7 days lead time, relative to FGN[6].

FGN also exhibits as good or better REV for predictions of extreme weather. Figure 2 shows REV for predictions of 2m temperature (2g) and 10m wind speed (2h) exceeding the 99.99th percentile, with FGN matching GenCast on the former, and outperforming GenCast on the latter. Results on more variables and thresholds are shown in Figures A.14 to A.17.

5.2. Results on structure of the joint distribution

FGN achieves average improvements of 8.7% and 7.5% over GenCast on average-pooled CRPS and max-pooled CRPS respectively across different spatial scales. Scorecards for 1 and 7 day lead times are shown in Figure 3a and b. The observed general trend of these results are maintained across other lead times, with the average improvement decreasing as lead time increases towards 15 days. In total, across all level, variable, and pool-size combinations across all lead times, FGN achieves better average-pooled and max-pooled CRPS in 99.9% and 99% of cases respectively. These results show that FGN succeeds in capturing certain spatial dependencies in the joint distribution.

Figure 3c and d shows that FGN is also on average 4.8% more skillful than GenCast when evaluating 10m wind speed predictions derived from its raw '10u' and '10v' predictions, and 7.8% better on forecasts of z300 − z500, derived from its z300 and z500 predicted fields. Improvements range from 10.4% and 15.6% respectively at short lead times, to 1 − 2% at longer lead times, highlighting that FGN succeeds in modeling these inter-variable dependencies.

From the spherical harmonic power spectra shown in Figure 3, FGN's predictions generally match well with the ground truth. In particular, FGN avoids the loss of power at high frequencies

---

[5]Note, due to our lack of confidence in the quality of ground-truth precipitation data, we exclude precipitation results from our main figures, and refer readers to Figures A.12 and A.13.

[6]We note that the calibration of this operational GenCast model, while still good overall, is slightly worse than that of the ERA5 version of the model presented in Price et al. (2025).
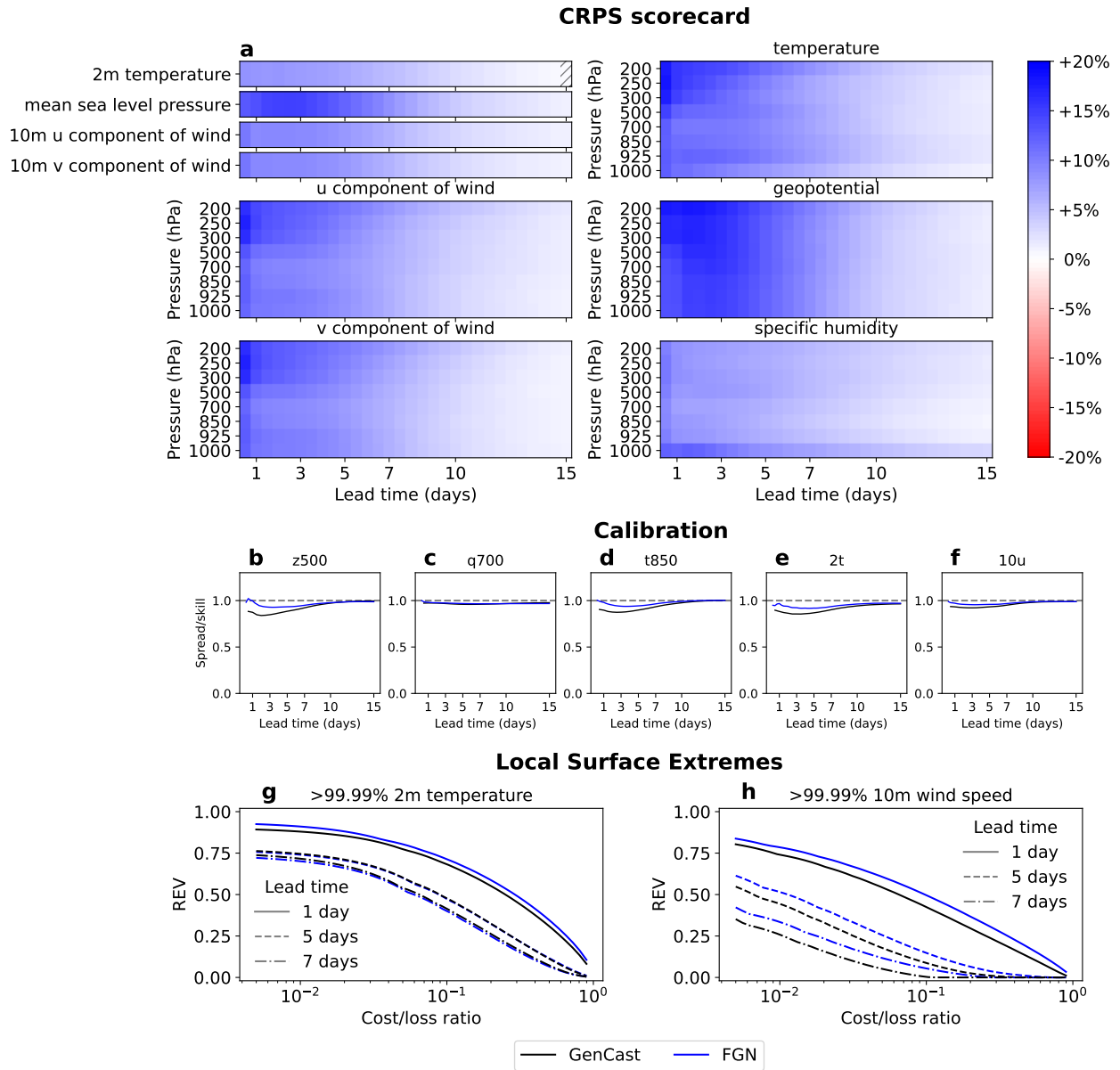
Figure 2 | **FGN produces more skillful marginal forecasts than GenCast while preserving good calibration**. (a) A scorecard comparing CRPS achieved by FGN and GenCast, where blue cells indicate FGN outperforms GenCast, red denotes where GenCast is better, and hatched regions indicate where the difference in performance was not statistically significant. FGN is significantly better ($p < 0.05$) in 99.9% of cases, with an average improvement of 6.5%. (b-f) Spread skill plots for a number of variables across all lead times. FGN maintains a spread skill ratio very close to 1 across all lead times, indicating well calibrated ensemble spread. (g, h) REV for forecasts of extreme high (>99.99th percentile) 2m temperature and 10 wind speed, at lead times of 1, 5, and 7 days. FGN achieves better REV than GenCast in the case of 10m wind speed, and matches performance on 2m temperature.
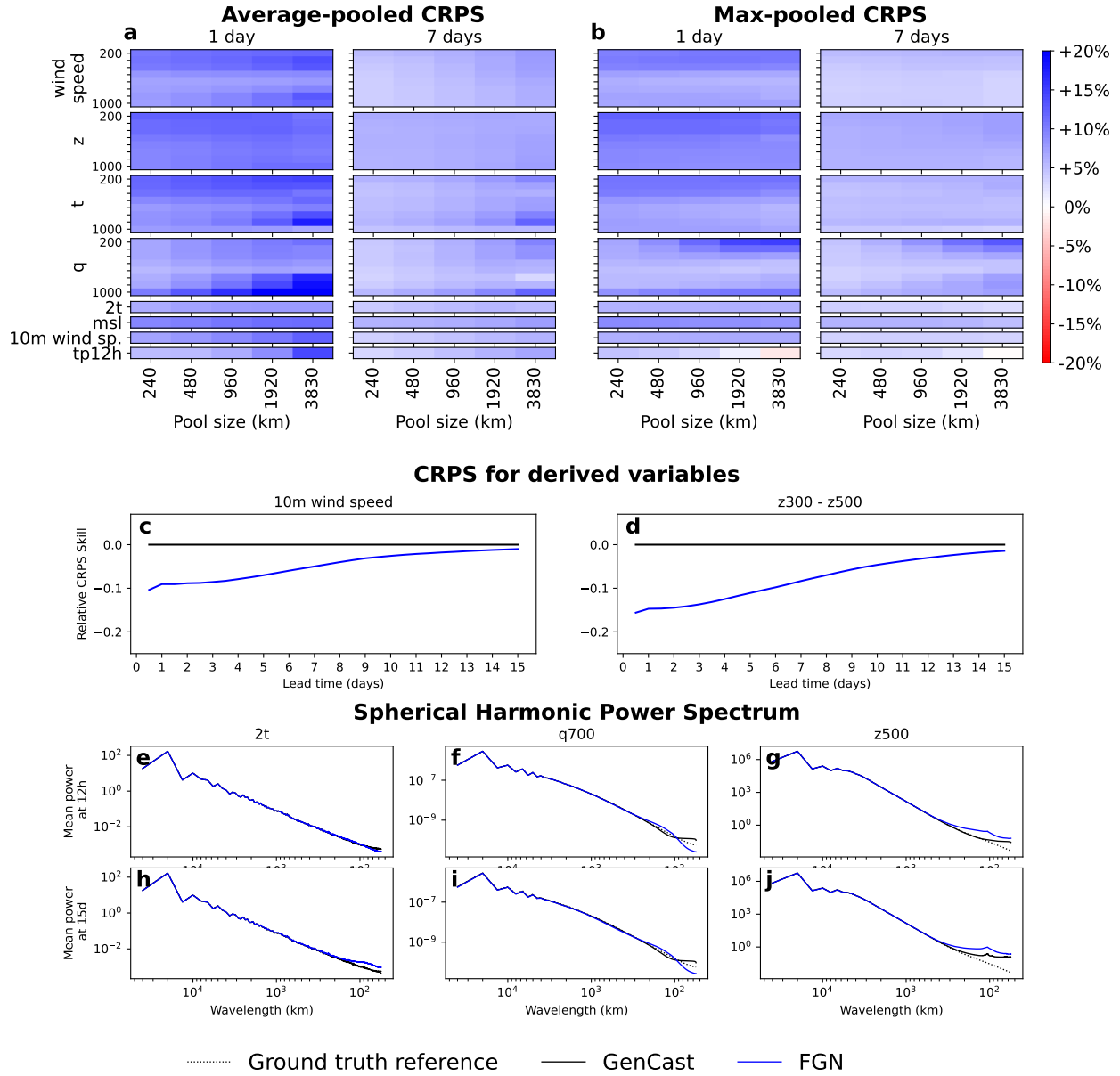
Figure 3 | FGN produces skillful joint forecast distributions. (a, b) CRPS scorecards evaluating average-pooled (respectively max-pooled) fields, for different pool sizes, at 1 and 7 day lead times. FGN achieves better CRPS in average-pooled evaluations in 99.9% (respectively 99%) of all pool-size, variable, leadtime, level combinations, evidencing that FGN skillfully models spatial correlations in the forecast distribution. (c, d) Evaluations of skill on two quantities of interest derived from predicted fields. FGN obtains 10.4% and 15.6% better CRPS on 10m wind speed and $z300 - z500$ at short lead times respectively, with this improvement decreasing as lead time increases. This shows FGN is able to capture across-variable dependencies despite not being directly trained on them. (e - j) Spectra of FGN compared to GenCast, and the ground truth. In some variables (e.g. z500), FGN has a spike at the mesh frequency and some additional high frequency content, more so than GenCast, but orders of magnitude smaller than the dominant powers in the signal.
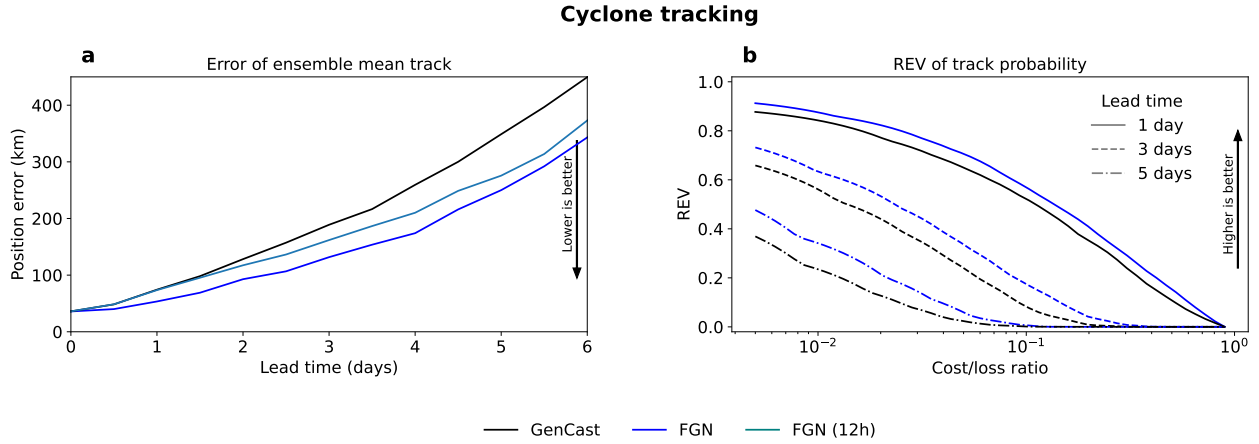
Figure 4 | **FGN achieves state-of-the-art cyclone track prediction.** (a) Position error of the ensemble mean track. FGN achieves up to a 24h improvement in position error over GenCast. Some of this improvement is due to the TempestExtremes cyclone tracker working better on 6-hour timesteps than 12-hour timesteps. However, as shown in the plot, a 12h-step version of FGN still achieves better position error beyond 2 day lead times. (b) REV of track probability predictions. FGN exhibits better REV than GenCast across lead times of up to 5 days, across all cost/loss ratios at which either model is better than climatology.

characterizing blurry predictions. This holds out to 15 day lead times, as shown in Figures 3e,h,f,i which plot the spectra of 2m temperature and specific humidity at 700hPa. In certain fields, in particular smoother fields like geopotential at 500hPa (Figures 3g,j), FGN's forecasts contain some additional high frequency signal and in certain cases minor artifacts around a wavelength of 100km, consistent with the artifacts explained in Section 6. Similar phenomena were observed in a few variables in GraphCast (Lam et al., 2022), but are more common with FGN. There is also some minor build up in high frequency signal observed in other variables, like 2m temperature, at longer lead times, as is visible in Figure 3h. Spectrum plots for additional variables at more lead times are shown in Figure A.18 and A.19 in Appendix A.5.2.

5.3. Results on tropical cyclone tracks

The ensemble mean track of FGN is significantly more accurate than GenCast's ($p < 0.05$), achieving on average approximately a 24 hour advantage in track accuracy from 3 - 5.5 day lead times. In other words, GenCast's accuracy at 2 days ahead is roughly achieved by FGN 3 days ahead (Figure 4a). Part of this improvement may be due FGN predicting at 6-hour timesteps, which results in fewer 'tracker jumps' and spurious tracker errors when applying the Tempest Extremes tracks to the forecasts to estimate the tracks. However, most of the improvement is due to FGN's overall greater forecast accuracy. To establish this, we include a 12-hour step version of FGN in Figure 4a to ablate the impact of the model timestep. This 12-hour step FGN shows higher average error relative to the 6-hour step FGN, but still lower error than GenCast from 2-day lead times onwards, with the gap increasing as lead time increases.

FGN also produces better track probability forecasts than GenCast, exhibiting better REV (Figure 4b) across almost all cost/loss ratios at which either model is better than climatology. The REV improvement is especially pronounced for smaller cost/loss ratios and longer lead times, which are most relevant for cyclone preparation decision making. This suggests that FGN's cyclone forecasts could offer significant additional value in operational cyclone-related decision-making settings.

6. Discussion

Our results show FGN offers substantial improvements over previous ML-based probabilistic weather models, and sets a new state-of-the-art in ensemble forecasting. It outperforms GenCast with both the skill of its marginal forecasts, including on extreme weather, as well as its skill in forecasting quantities dependent on the dependency structure of the joint forecast distribution, notably including tropical cyclone tracks. It uses an ensemble of independently trained models to capture epistemic uncertainty, each of which captures aleatoric uncertainty through learned functional variability, which can be understood as perturbations applied to each models parameters when generating a forecast. It generates forecasts more efficiently than GenCast, and can easily accommodate auto-regressive training, which is a powerful method for optimizing models over longer lead times.

It is straightforward to extend FGN to broader tasks than modeling dense, gridded analysis data. As a preliminary demonstration, we recently built an experimental extension of FGN, which is trained to directly predict cyclones. Our internal evaluations indicate its performance is substantially better on predicting cyclone tracks, intensities, and sizes than that reported in the results above, which used the TempestExtremes tracker to extract tracks from FGN forecasts. Its tropical cyclone forecasts are available on our website (<https://deepmind.google.com/science/weatherlab>).

Perhaps the most fascinating feature of our results is how well FGN captures the covariance of the joint spatial distribution of the forecasts. Optimizing CRPS of the forecast marginal distributions, alone, does not guarantee good joint distributions (Pic et al., 2025). This is trivially shown—a collection of independent models, one model per location, predicting the weather at each location independently, cannot capture spatial correlations, even if each model minimizes CRPS on its own forecast targets. Nonetheless, as we show in Appendix A.4, even the most simple, stripped down version of FGN—which is smaller (512 latent dimension and 16 layers), has only a single model seed, and no autoregressive training meaning a truly marginals-only training objective—still outperforms the prior state-of-the-art almost across the board.

The reason FGN captures joint structure is likely a result of how stochasticity is injected and transformed by the model's architecture. The source of all stochasticity in the full 87 million-dimensional output distribution is a single 32-dimensional vector passed into the network's conditional normalization layers. This means that for a given input, the function output has only 32 degrees of freedom, and the impact of traversing these degrees of freedom on the network's computations is shared across its mesh and spatial dimensions. Notice also that this low effective dimensionality is independent of the resolution of the output grid, or indeed whether the output is a fixed and regular grid at all—the same will apply across any combination of dense and/or sparse predictions. While this does not provide guarantees for getting good joint forecast distributions by optimizing only the CRPS of its marginals, it does provide some intuition. We speculate that under such heavy distributional constraints and with the inductive biases of FGN's architecture, the easiest way for the model to jointly optimize the CRPS of all marginals is to try to model their inter-dependencies as well.

We also observed several weaknesses, which future work should address. One was that there can be subtle artifacts in the forecast states, which appear as visible 'honeycomb' patterns corresponding to the processor's mesh structure. These artifacts tend to appear in higher frequency variables which were allocated smaller loss weights, specifically low pressure levels of specific humidity (see Figure 5), though these appear as only very minor artifacts in the spectra of these fields. While forecasts of some smoother variables, like z500, exhibit more notable artifacts or 'kinks' in their spectra, these do not manifest as visual artifacts at all, because high frequencies at which these 'kinks' are present, contribute so little to the overall signal. The emergence of these artifacts may in fact be due to a slight *over-constraining* of the output covariance structure which could enforce spurious

local correlations. This would suggest that perhaps a slightly higher-dimensional noise input (and therefore higher-dimensional output manifold) might enable the forecasts to avoid these artifacts while still encouraging the learning of good joint structure.
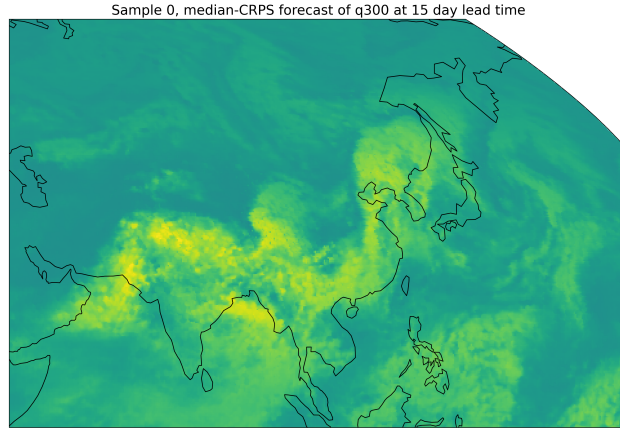
Sample 0, median-CRPS forecast of q300 at 15 day lead time

Figure 5 | Visualization of a q300 forecast at 15 day lead time, zoomed in to highlight the 'honeycomb' artifacts visible in this variable.

Another minor issue we observed was the potential for poor model seeds, which can result in some unstable forecast rollouts. Specifically, at one point during model validation, we found that a particular training seed produced a number of unstable rollouts, which was detected by examining the averaged spectra of the validation year (2022) forecasts. We removed this seed and retrained that particular model with a different seed, confirming that this model behaved as expected again using our validation set. This highlights the importance of careful evaluation of trained models before deployment.

Overall, our work shows that FGN provides a general-purpose approach for modeling the different sources of uncertainty in weather forecasts, and capturing the joint spatial structure of forecast distributions. The FGN framework is easy to work with and extend, and represents a key step forward in applying powerful statistical learning methods to one of the most complex and important learned simulation problems.

7. Acknowledgments

References

F. Alet. *Learning inside the prediction function*. PhD thesis, Massachusetts Institute of Technology, 2022.

F. Alet, M. Bauza, K. Kawaguchi, N. G. Kuru, T. Lozano-Pérez, and L. Kaelbling. Tailoring: encoding inductive biases by optimizing unsupervised objectives at prediction time. *Advances in Neural Information Processing Systems*, 34:29206–29217, 2021.

F. Alet, C. Gehring, T. Lozano-Pérez, K. Kawaguchi, J. B. Tenenbaum, and L. P. Kaelbling. Functional risk minimization. *preprint*, 2022.

K. Bi, L. Xie, H. Zhang, X. Chen, X. Gu, and Q. Tian. Pangu-Weather: A 3D high-resolution model for fast and accurate global weather forecast. *arXiv preprint arXiv:2211.02556*, 2022.

K. Bi, L. Xie, H. Zhang, X. Chen, X. Gu, and Q. Tian. Accurate medium-range global weather forecasting with 3d neural networks. *Nature*, 619(7970):533–538, 2023.

K. Bibas, Y. Fogel, and M. Feder. Deep pnml: Predictive normalized maximum likelihood for deep neural networks. *arXiv preprint arXiv:1904.12286*, 2019.

C. Bodnar, W. P. Bruinsma, A. Lucic, M. Stanley, A. Allen, J. Brandstetter, P. Garvan, M. Riechert, J. A. Weyn, H. Dong, J. K. Gupta, K. Thambiratnam, A. T. Archibald, C.-C. Wu, E. Heider, M. Welling, R. E. Turner, and P. Perdikaris. A foundation model for the earth system. *Nature*, may 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09005-y. URL https://doi.org/10.1038/s41586-025-09005-y.

D. Bouchacourt, P. K. Mudigonda, and S. Nowozin. Disco nets: Dissimilarity coefficients networks. *Advances in Neural Information Processing Systems*, 29, 2016.

K. Chen, T. Han, J. Gong, L. Bai, F. Ling, J.-J. Luo, X. Chen, L. Ma, T. Zhang, R. Su, et al. FengWu: Pushing the skillful global medium-range weather forecast beyond 10 days lead. *arXiv preprint arXiv:2304.02948*, 2023.

M. Chen, X. Tan, B. Li, Y. Liu, T. Qin, S. Zhao, and T.-Y. Liu. Adaspeech: Adaptive text to speech for custom voice, 2021.

G. Couairon, R. Singh, A. Charantonis, C. Lessig, and C. Monteleoni. ArchesWeather & ArchesWeatherGen: a deterministic and generative model for efficient ml weather forecasting. *arXiv preprint arXiv:2412.12971*, 2024.

ECMWF. *IFS Documentation CY46R1 - Part V: Ensemble Prediction System*. 2019. doi: 10.21957/38yug0cev. URL https://www.ecmwf.int/node/19309.

ECMWF. *IFS Documentation CY48R1 - Part V: Ensemble Prediction System*. Number 5. ECMWF, 06/2023 2023. doi: 10.21957/e529074162.

ECMWF. IFS Cycle 49r1 HRES Scorecards. https://sites.ecmwf.int/ifs/scorecards/scorecards-49r1HRES.html, 2024. Accessed on June 9, 2025.

V. Fortin, M. Abaza, F. Anctil, and R. Turcotte. Why should ensemble spread match the rmse of the ensemble mean? *Journal of Hydrometeorology*, 15(4):1708–1713, 2014.

M. Fortunato, M. G. Azar, B. Piot, J. Menick, I. Osband, A. Graves, V. Mnih, R. Munos, D. Hassabis, O. Pietquin, et al. Noisy networks for exploration. *arXiv preprint arXiv:1706.10295*, 2017.

T. Gneiting and A. E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, pages 359–378, 2007.

A. Gritsenko, T. Salimans, R. van den Berg, J. Snoek, and N. Kalchbrenner. A spectral energy distance for parallel speech synthesis. *Advances in Neural Information Processing Systems*, 33:13062–13072, 2020.

T. Haiden, M. J. Rodwell, D. S. Richardson, A. Okagaki, T. Robinson, and T. Hewson. Intercomparison of global model precipitation forecast skill in 2010/11 using the seeps score. *Monthly Weather Review*, 140(8):2720–2733, 2012.

T. Han, S. Guo, F. Ling, K. Chen, J. Gong, J. Luo, J. Gu, K. Dai, W. Ouyang, and L. Bai. FengWu-GHR: Learning the kilometer-scale medium-range global weather forecasting. *arXiv preprint arXiv:2402.00059*, 2024.

H. Hersbach, B. Bell, P. Berrisford, S. Hirahara, A. Horányi, J. Muñoz-Sabater, J. Nicolas, C. Peubey, R. Radu, D. Schepers, et al. The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, 146(730):1999–2049, 2020.

X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017.

T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

R. Keisler. Forecasting global weather with graph neural networks. *arXiv preprint arXiv:2202.07575*, 2022.

D. P. Kingma, M. Welling, et al. Auto-encoding variational bayes, 2013.

K. R. Knapp, M. C. Kruk, D. H. Levinson, H. J. Diamond, and C. J. Neumann. The international best track archive for climate stewardship (IBTrACS): Unifying tropical cyclone data. *Bull. Am. Meteorol. Soc.*, 91(3):363–376, Mar. 2010.

D. Kochkov, J. Yuval, I. Langmore, P. Norgaard, J. Smith, G. Mooers, J. Lottes, S. Rasp, P. Düben, M. Klöwer, et al. Neural general circulation models. *arXiv preprint arXiv:2311.07222*, 2023.

D. Kochkov, J. Yuval, I. Langmore, P. Norgaard, J. Smith, G. Mooers, M. Klöwer, J. Lottes, S. Rasp, P. Düben, et al. Neural general circulation models for weather and climate. *Nature*, 632(8027): 1060–1066, 2024.

T. Kurth, S. Subramanian, P. Harrington, J. Pathak, M. Mardani, D. Hall, A. Miele, K. Kashinath, and A. Anandkumar. FourCastNet: Accelerating global high-resolution weather forecasting using adaptive fourier neural operators. *arXiv preprint arXiv:2208.05419*, 2022.

B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

R. Lam, A. Sanchez-Gonzalez, M. Willson, P. Wirnsberger, M. Fortunato, A. Pritzel, S. Ravuri, T. Ewalds, F. Alet, Z. Eaton-Rosen, et al. GraphCast: Learning skillful medium-range global weather forecasting. *arXiv preprint arXiv:2212.12794*, 2022.

R. Lam, A. Sanchez-Gonzalez, M. Willson, P. Wirnsberger, M. Fortunato, F. Alet, S. Ravuri, T. Ewalds, Z. Eaton-Rosen, W. Hu, et al. Learning skillful medium-range global weather forecasting. *Science*, page eadi2336, 2023.

S. Lang, M. Alexe, M. C. Clare, C. Roberts, R. Adewoyin, Z. B. Bouallègue, M. Chantry, J. Dramsch, P. D. Dueben, S. Hahner, et al. Aifs-crps: Ensemble forecasting using a model trained with a loss function based on the continuous ranked probability score. *arXiv preprint arXiv:2412.15832*, 2024.

M. Leutbecher. Diagnosis of ensemble forecasting systems. In *Seminar on diagnosis of forecasting and data assimilation systems*, pages 235–266, 2009.

H. Li, L. Chen, X. Zhong, F. Zhang, Y. Cheng, Y. Xu, and Y. Qi. Fuxi: A cascade machine learning forecasting system for 15-day global weather forecast. *npj Climate and Atmospheric Science*, 6(190), 2023.

I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018.

S. Mohamed, M. Rosca, M. Figurnov, and A. Mnih. Monte carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(132):1–62, 2020.

R. North, M. Trueman, M. Mittermaier, and M. J. Rodwell. An assessment of the seeps and sedi metrics for the verification of 6 h forecast precipitation accumulations. *Meteorological Applications*, 20(2):164–175, 2013.

P. Ollinaho, S.-J. Lock, M. Leutbecher, P. Bechtold, A. Beljaars, A. Bozzo, R. M. Forbes, T. Haiden, R. J. Hogan, and I. Sandu. Towards process-level representation of model uncertainties: stochastically perturbed parametrizations in the ecmwf ensemble. *Quarterly Journal of the Royal Meteorological Society*, 143(702):408–422, 2017.

J. Oskarsson, T. Landelius, M. Deisenroth, and F. Lindsten. Probabilistic weather forecasting with hierarchical graph neural networks. *Advances in Neural Information Processing Systems*, 37:41577–41648, 2024.

L. Pacchiardi, R. A. Adewoyin, P. Dueben, and R. Dutta. Probabilistic forecasting with generative networks via scoring rule minimization. *Journal of Machine Learning Research*, 25(45):1–64, 2024.

R. Pic, C. Dombry, P. Naveau, and M. Taillardat. Proper scoring rules for multivariate probabilistic forecasts based on aggregation and transformation. *Advances in Statistical Climatology, Meteorology and Oceanography*, 11(1):23–58, 2025.

I. Price, A. Sanchez-Gonzalez, F. Alet, T. R. Andersson, A. El-Kadi, D. Masters, T. Ewalds, J. Stott, S. Mohamed, P. Battaglia, et al. GenCast: Diffusion-based ensemble forecasting for medium-range weather. *arXiv preprint arXiv:2312.15796*, 2023.

I. Price, A. Sanchez-Gonzalez, F. Alet, T. R. Andersson, A. El-Kadi, D. Masters, T. Ewalds, J. Stott, S. Mohamed, P. Battaglia, et al. Probabilistic weather forecasting with machine learning. *Nature*, 637(8044):84–90, 2025.

S. Rasp, P. D. Dueben, S. Scher, J. A. Weyn, S. Mouatadid, and N. Thuerey. WeatherBench: a benchmark data set for data-driven weather forecasting. *Journal of Advances in Modeling Earth Systems*, 12(11):e2020MS002203, 2020.

S. Rasp, S. Hoyer, A. Merose, I. Langmore, P. Battaglia, T. Russel, A. Sanchez-Gonzalez, V. Yang, R. Carver, S. Agrawal, et al. WeatherBench 2: A benchmark for the next generation of data-driven global weather models. *arXiv preprint arXiv:2308.15560*, 2023.

S. Ravuri, K. Lenc, M. Willson, D. Kangin, R. Lam, P. Mirowski, M. Fitzsimons, M. Athanassiadou, S. Kashem, S. Madge, et al. Skilful precipitation nowcasting using deep generative models of radar. *Nature*, 597(7878):672–677, 2021.

D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In E. P. Xing and T. Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1278–1286, Bejing, China, 22–24 Jun 2014. PMLR. URL https://proceedings.mlr.press/v32/rezende14.html.

D. S. Richardson. Skill and relative economic value of the ECMWF ensemble prediction system. *Quarterly Journal of the Royal Meteorological Society*, 126(563):649–667, 2000.

D. S. Richardson. *Predictability and economic value*, page 628–644. Cambridge University Press, 2006.

M. J. Rodwell, D. S. Richardson, T. D. Hewson, and T. Haiden. A new equitable score suitable for verifying precipitation in numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 136(650):1344–1363, 2010.

I. Russell and S. Kertész. Metview, 2017. URL https://www.ecmwf.int/node/18136.

Y. Saatci and A. G. Wilson. Bayesian GAN. *Advances in neural information processing systems*, 30, 2017.

Y. Sun, X. Wang, Z. Liu, J. Miller, A. Efros, and M. Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pages 9229–9248. PMLR, 2020.

R. Swinbank, M. Kyouda, P. Buchanan, L. Froude, T. Hamill, T. Hewson, J. Keller, M. Matsueda, J. Methven, F. Pappenberger, M. Scheuerer, H. Titley, L. Wilson, and M. Yamaguchi. The tigge project and its achievements. *Bulletin of the American Meteorological Society*, 97:49–67, 02 2016. doi: 10.1175/BAMS-D-13-00191.1.

M. Titsias and M. Lázaro-Gredilla. Doubly stochastic variational bayes for non-conjugate inference. In *International conference on machine learning*, pages 1971–1979. PMLR, 2014.

P. A. Ullrich, C. M. Zarzycki, E. E. McClenny, M. C. Pinheiro, A. M. Stansfield, and K. A. Reed. TempestExtremes v2.1: a community framework for feature detection, tracking, and analysis in large datasets. *Geosci. Model Dev.*, 14(8):5023–5048, Aug. 2021.

D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

V. D. Wild, S. Ghalebikesabi, D. Sejdinovic, and J. Knoblauch. A rigorous link between deep ensembles and (variational) bayesian methods. *Advances in Neural Information Processing Systems*, 36:39782–39811, 2023.

D. Wilks. A skill score based on economic value for probability forecasts. *Meteorological Applications*, 8(2):209–219, 2001.

A. G. Wilson and P. Izmailov. Bayesian deep learning and a probabilistic perspective of generalization. *Advances in neural information processing systems*, 33:4697–4708, 2020.

M. Zamo and P. Naveau. Estimation of the continuous ranked probability score with limited information and applications to ensemble weather forecasts. *Mathematical Geosciences*, 50(2):209–234, 2018.

X. Zhong, L. Chen, H. Li, J. Liu, X. Fan, J. Feng, K. Dai, J.-J. Luo, J. Wu, and B. Lu. FuXi-ENS: A machine learning model for medium-range ensemble weather forecasting. *arXiv preprint arXiv:2405.05925*, 2024.

X. Zhou, Y. Zhu, D. Hou, B. Fu, W. Li, H. Guan, E. Sinsky, W. Kolczynski, X. Xue, Y. Luo, et al. The development of the NCEP global ensemble forecast system version 12. *Weather and Forecasting*, 37 (6):1069–1084, 2022.

Appendix

A.1. Data

The following descriptions of ERA5 and HRES-fc0 datasets are largely adapted from Lam et al. (2022) and Price et al. (2023). We include some additional details around preprocessing of HRES-fc0 for the purposes of finetuning FGN.

A.1.1. ERA5

For developing FGN, we built our "pre-training" dataset from a subset of ECMWF's ERA5 (Hersbach et al., 2020)[7] archive, which is a large corpus of data that represents the global weather from 1959 to the present, at 1 hour increments, for hundreds of static, surface, and atmospheric variables. The ERA5 archive is based on *reanalysis*, which uses ECMWF's HRES model (cycle 42r1) that was operational for most of 2016, within ECMWF's 4D-Var data assimilation system. ERA5 assimilates 12-hour windows of observations, from 21z-09z and 09z-21z, as well as previous forecasts, into a dense representation of the weather's state, for each historical date and time. The ERA5 archive consists of a deterministic reanalysis (ERA5) computed at 0.281 25° native latitude/longitude resolution.

Our ERA5 dataset contains a subset of available variables in ECMWF's ERA5 archive (Table A.1), on 13 pressure levels[8] corresponding to the levels of the WeatherBench (Rasp et al., 2020) benchmark: 50, 100, 150, 200, 250, 300, 400, 500, 600, 700, 850, 925, 1000 hPa. The range of years included was 1979-01-01 to 2018-01-15, which were downsampled to 6-hour time intervals (corresponding to 00z, 06z, 12z and 18z each day). The downsampling is performed by subsampling, except for the total precipitation, which is accumulated for the 6 hours leading up to the corresponding downsampled time.

Variables with NaNs. ERA5 sea surface temperature (SST) data contains NaNs over land by default. As preprocessing of the SST training and evaluation data, values over land are replaced with the minimum sea surface temperature seen globally in a subset of ERA5.

A.1.2. HRES-fc0

For finetuning and evaluation, we make use of HRES-fc0, which contains the initial (zeroeth) step of the ECMWF HRES forecast at each initialization time (00z, 06z, 12z, 18z) of the day. The HRES-fc0 data is similar to ERA5, but for a given forecast time, it is assimilated using the latest ECMWF NWP model, and only with observations from ±3 hours.

Variables with corresponding NaNs in ERA5. Unlike ERA5, HRES-fc0 does not make use of NaNs to represent land within the sea surface temperature (SST) data. Rather, we found a placeholder value was used. To ensure a consistent representation between the two datasets in training, for all locations with a NaN in the original ERA5 dataset, we imputed the corresponding location in HRES-fc0 with the minimum sea surface temperature as well.

---

[7]See ERA5 documentation: `https://confluence.ecmwf.int/display/CKB/ERA5`.

[8]We follow common practice of using pressure as our vertical coordinate, instead of altitude. A "pressure level" is a field of altitudes with equal pressure. E.g., "pressure level 500 hPa" corresponds to the field of altitudes for which the pressure is 500 hPa. The relationship between pressure and altitude is determined by the geopotential variable.

Type	Variable name	Short name	ECMWF Parameter ID	Role (accumulation period, if applicable)
Atmospheric	Geopotential	z	129	Input/Predicted
Atmospheric	Specific humidity	q	133	Input/Predicted
Atmospheric	Temperature	t	130	Input/Predicted
Atmospheric	U component of wind	u	131	Input/Predicted
Atmospheric	V component of wind	v	132	Input/Predicted
Atmospheric	Vertical velocity	w	135	Input/Predicted
Single	2 metre temperature	2t	167	Input/Predicted
Single	10 metre u wind component	10u	165	Input/Predicted
Single	10 metre v wind component	10v	166	Input/Predicted
Single	Mean sea level pressure	msl	151	Input/Predicted
Single	Sea Surface Temperature	sst	34	Input/Predicted
Single	Total precipitation	tp	228	Predicted (6h)
Static	Geopotential at surface	z	129	Input
Static	Land-sea mask	lsm	172	Input
Static	Latitude	n/a	n/a	Input
Static	Longitude	n/a	n/a	Input
Clock	Local time of day	n/a	n/a	Input
Clock	Elapsed year progress	n/a	n/a	Input

Table A.1 | **ECMWF variables used in our datasets.** The "Type" column indicates whether the variable represents a *static* property, a time-varying *single*-level property (e.g., surface variables are included), or a time-varying *atmospheric* property. The "Variable name" and "Short name" columns are ECMWF's labels. The "ECMWF Parameter ID" column is a ECMWF's numeric label, and can be used to construct the URL for ECMWF's description of the variable, by appending it as suffix to the following prefix, replacing "ID" with the numeric code: `https://apps.ecmwf.int/codes/grib/param-db/?id=ID`. The "Role" column indicates whether the variable is something our model takes as input and predicts, or only uses as input context (the double horizontal line separates predicted from input-only variables, to make the partitioning more visible).

Total precipitation. As an accumulated variable, total precipitation (*tp*) takes a value of 0 in HRES-fc0, inhibiting its use in a setting where the model is initialized on the dataset. To account for this, FGN is trained to only output *tp*, not taking it as input. Further, while ERA5 accumulates *tp* over the interval leading up to the initialization time, using HRES *tp* would require accumulating the variable forwards in time (since the interval prior is another forecast initialization's trajectory). To mitigate this discrepancy, we continue to use ERA5's *tp* for both finetuning and evaluation purposes.

A.2. Model Training

The FGN results reported in this paper were generated by a model trained in the following multi-stage process:

> **Stage 1:** 400000 steps with a 1° input and ground truth ERA5 dataset at a 12-hour temporal resolution.
> **Stage 2:** 100000 steps with a 1° input and ground truth ERA5 dataset at a 6-hour temporal resolution.
> **Stage 3:** 32000 steps with a 0.25° input and ground truth ERA5 dataset at a 6-hour temporal resolution.
> **Stage 4:** 18000 autoregressive (AR) steps with a 0.25° ground truth HRES-fc0 dataset at a 6-hour temporal resolution. Including 8000 steps at 1AR, 4 000 steps at 2AR and 1000 steps at 3-8AR each.

In all cases, the model is trained with a batch size of 64. A full list of hyper-parameters used for the stages are detailed in Table A.2.

We subsample the 6-hourly data with a stride of two to generate the 12-hourly dataset. Accumulated variables, such as total precipitation, are accumulated over the longer interval accordingly. Similarly, we downscale ERA5 from 0.25° to 1° by subsampling the spatial dimensions with a stride of four.

When training on 1° data, the graph-transformer processor operates on a 5-times refined icosahedral mesh. This is further refined a sixth time in the stages training on 0.25° data, where the architecture is updated to input and output the higher resolution data, but using the same model weights. To preserve the scale of the incoming signal at the start of Stage 3, we account for the increase in the number of number of messages being received by each mesh node and divide the sum of the message vectors by 4 in the Encoder GNN.

A.3. Architecture

FGN uses a very similar architecture to that of GenCast, with some hyperparameter changes and some minor architectural differences.

Conditioning on input physical states. GenCast is a diffusion model, requiring multiple forward passes of the denoiser to generate a single forecast frame in a process of iterative refinement. As such, GenCast conditions on the two prior weather states, $X^{t-1,t-2}$, as well as a noisy version of the targets Z_σ, all concatenated along the channel dimension. FGN generates a forecast in a single forward pass, and thus, similar to GraphCast (Lam et al., 2022) it simply conditions on the two prior weather states, $X^{t-1,t-2}$, without taking any noisy targets as inputs.

Shared encoder for conditional layer-norm layers. GenCast uses conditional layer-norm layers to condition on the noise level. First, the noise level is encoded into a vector of sine–cosine Fourier

	Stage 1	Stage 2	Stage 3	Stage 4
Training dataset	ERA5	ERA5	ERA5	HRES-fc0
Time-step size	12 hr	6 hr	6 hr	6 hr
Spatial resolution	1°	1°	0.25°	0.25°
Number of AR steps	1	1	1	1-8
Batch size	64			
Num training samples	2			
Optimizer	AdamW (Loshchilov and Hutter, 2018)			
Weight decay	0.1			
LR decay schedule	Cosine			
Peak LR	8e-4	8e-5	8e-5	8e-5, 8e-6, 8e-7 thereafter
Linear warm-up steps	1000	1000	1000	800 1AR, 400 2AR, 100 thereafter
Total train steps	400 000	100 000	32 000	8 000 1AR, 4 000 2AR, 1 000 thereafter

Table A.2 | Model training hyperparameters.

features, then it is passed through a small 2-layer MLP to obtain 16-dimensional noise-level encodings before ultimately being used to condition the various layer-norm layers in the architecture. FGN also uses conditional layer-norm layers but to condition on the noise vector of 32 elements. In this case, the noise vector is directly encoded with a single matrix multiplication to produce another vector of 32 elements, and is then passed as conditioning to the various layer-norm layers. Initially this encoder was added as an implementation detail, but we later found that it seemed to help with optimization during training.

Capacity. GenCast contained MLPs with hidden and output sizes of 512, 4 attention heads, and 16 transformer layers. FGN has a higher capacity with hidden and output sizes of 768, 6 attention heads, and 24 transformer layers. The only exception to this is the capacity of the MLPs that embed the edge features for the grid-to-mesh encoder graph and the mesh-to-grid decoder graph. For GenCast, this had hidden and output sizes of 512, while in FGN these were reduced to 32. As there are $O(10^5)$ edges, but only 4 edge features to embed, we found this had no detrimental effect on performance while yielding significant savings in compute and memory.

Global features. GenCast (and GraphCast) condition on the sine and cosine of the of year progress to tell the model which time of the year is being modeled. They broadcast this feature, which is constant in space, across the entire grid, also providing it as input to the encoder in addition to other grid features. In FGN, we additionally broadcast this feature across the entire mesh, concatenating it with the rest of the mesh input features.

Encoder from the grid to mesh. GenCast (and GraphCast) use a graph neural network (GNN) to translate features between the grid and mesh structures by building a bipartite graph with edges pointing from the grid nodes to the mesh nodes. In these, the "edge model" (also often referred as "message function") is a function that conditions on both the features of the sender grid nodes, and the receiver mesh nodes, via gather operations, for every edge of the graph. For FGN, we found that we could remove this conditioning on the receiver mesh nodes in the message function at no performance cost and significant compute and memory savings.

A.4. Ablations

Figure A.1 compares GenCast to a version of FGN which has the same timestep (12 hours), same number of model seeds (1) and same model capacity (16-layers, and latent dimension of 512) as

GenCast—and which has only been trained on a single step loss. This stripped-down version of FGN still outperforms GenCast on more than 90% of targets. Figure A.2 shows an FGN model of the same capacity and timestep but which has also undergone autoregressive training up to 8 rollout steps. The results further improve, with this version of FGN outperforming GenCast on CRPS on 99% of targets and with average improvement over GenCast increasing from 2.7% to 4%.

Figure A.3 reproduces Figure 3 but comparing GenCast to a version of FGN which has the same timestep (12 hours), same number of model seeds (1), same model capacity (16-layers, and latent dimension of 512) as GenCast, and without auto-regressive training. The results show that even when controlling for model capacity, model ensemble size, and number of rollout steps during training, FGN generally outperforms GenCast, successfully capturing spatial and inter-variable correlations.

Figure A.4 compares REV of the track probability predictions of GenCast to three models: the two smaller, 12-hour step, single seed versions of FGN described in the previous paragraph, as well as a 12-hour step version of the full size, 4-model-seed FGN model. The single-step trained FGN version of the same size as GenCast (green curve) performs as well as or better than GenCast. Adding autoregressive training (orange curve) further improves REV, and increasing the model size and using a model ensemble (blue) performs best.

Overall these results show that the simple CRPS-based training setup described in Section 2.2.3—without model ensembling, model scaling, or autoregressive training—is sufficient to match or exceed SOTA forecast skill, but that the addition of these elements further boosts performance.

One drawback of the single-step trained FGN is that, unlike in the 8AR step trained models, we observed some (rare) instances of unstable rollouts which correspond to forecasts with highly non-physical, artifact dominated forecasts states at late lead times. As an ad hoc investigation into this phenomena we visually inspected outlier forecasts as measured by any of the per-forecast mean absolute error, bias or L1 distance between the predictions and targets power spectra, for each variable for each level. This investigation suggested that the prevalence of this behavior was on the order of 0.1% at 15-day lead-times (and smaller at shorter lead-times). Crucially we did not find any instances of this behavior for the autoregressively trained models (the ones evaluated in the main paper), which may suggest that the autoregressive training enables the models to be more robust and stable over rollouts.

A.5. Supplementary Results and Figures

A.5.1. Comparisons with ENS

Price et al. (2025) established GenCast as the new state-of-the-art in probabilistic global weather forecasts, outperforming the prior state-of-the-art, ECMWF's ENS. As detailed in Section 3.2, this work makes use of a version of GenCast finetuned on HRES-fc0, whose operationally-initialized forecasts we compare to those of ENS over the year 2023. Figure A.5 shows this GenCast model outperforms ENS on more than 96% of targets, mirroring the results found in Price et al. (2025), and confirming that GenCast is indeed the appropriate state-of-the-art baseline in the operational setting.

In these comparisons, we make use of the unbiased CRPS estimator to enable fair comparison, despite the fact that the GenCast ensemble is 56 members while ENS is 50 members. Moreover, as in Price et al. (2025), we compare the models on 00z and 12z initializations as this is the data made available in the public TIGGE archive (Swinbank et al., 2016).

We also note that on the 27th of June 2023, the resolution of ENS changed from 0.2° to 0.1°, as part of the upgrade to Cycle 48r1 from Cycle 47r3 (ECMWF, 2023). To account for this, all data was
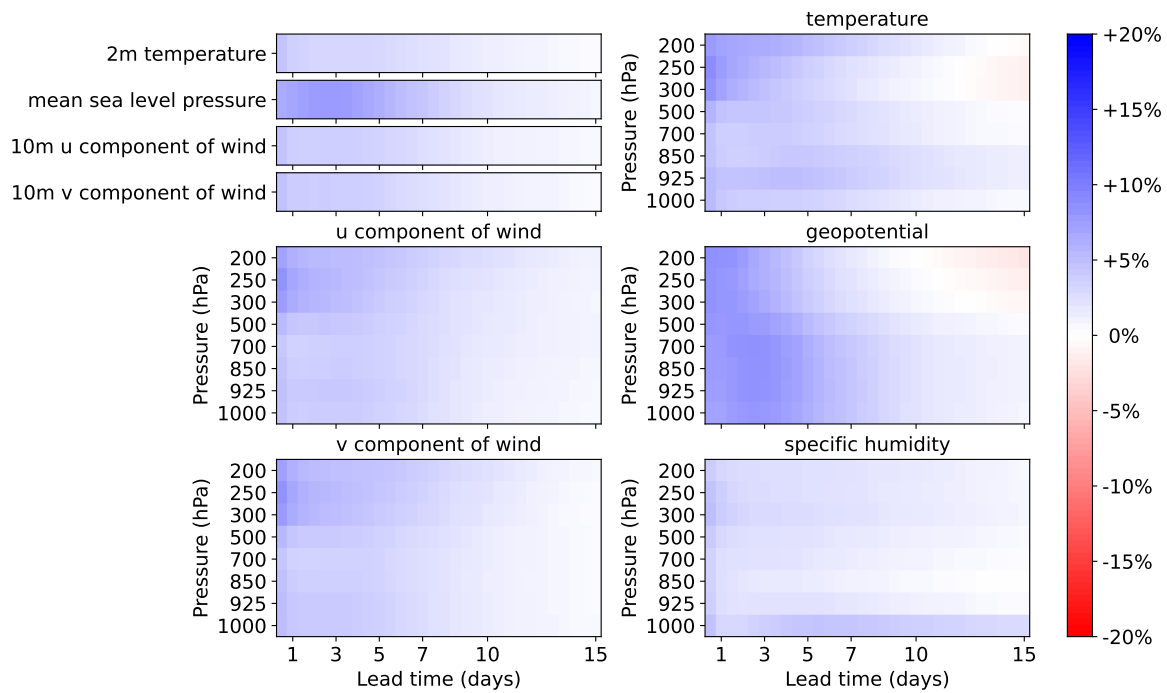
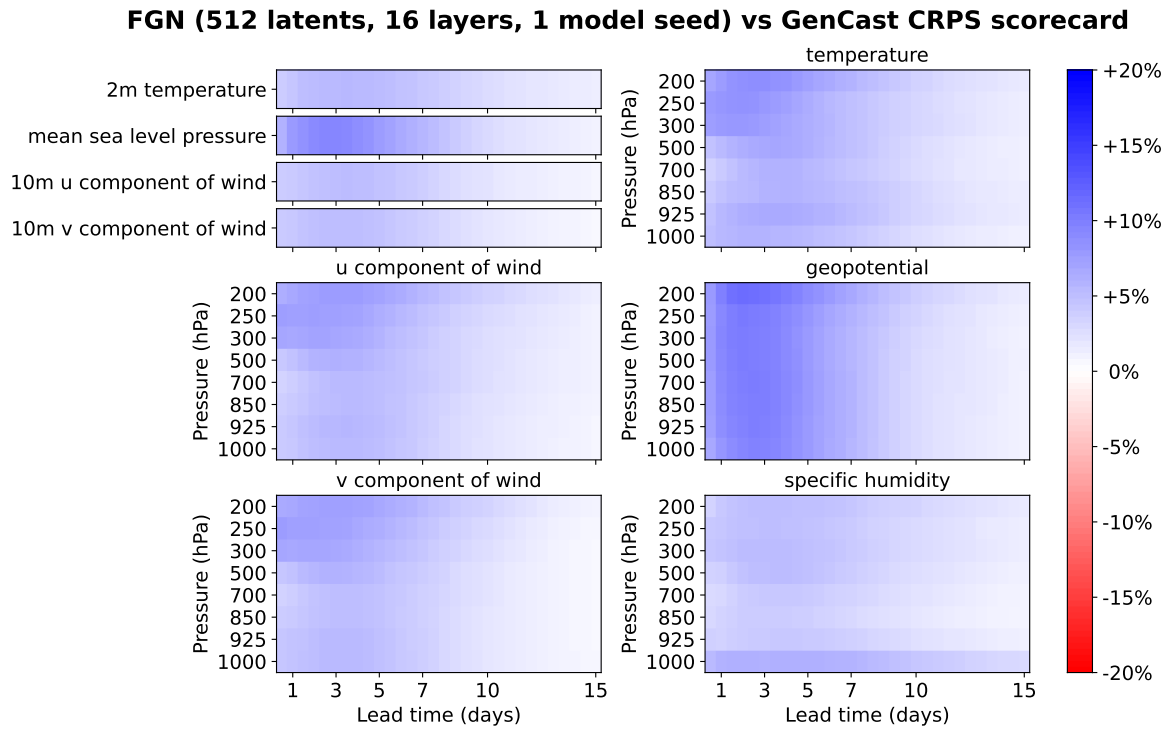**FGN (512 latents, 16 layers, 1 model seed, 1AR only) vs GenCast CRPS scorecard**

Figure A.1 | CRPS scorecard comparing GenCast with a version FGN which has the same timestep (12 hours), same number of model seeds (1) and same model capacity (16-layers, and latent dimension of 512)—and which has also not undergone any autoregressive training. FGN still matches or outperforms GenCast on globally averaged CRPS on 91% of targets, with average improvement of 2.7% and maximum improvement of 9%, achieved on geopotential at shorter lead times.

Figure A.2 | CRPS scorecard comparing GenCast with a version of FGN which has the same timestep (12 hours), same number of model seeds (1) and same model capacity (16-layers, and latent dimension of 512)—but which has still undergone autoregressive training up to 8 rollout steps. This version of FGN matches or outperforms GenCast on globally averaged CRPS on 99.8% of targets, with average improvement of 4% and maximum improvement of 11.5%, achieved on geopotential at shorter lead times.
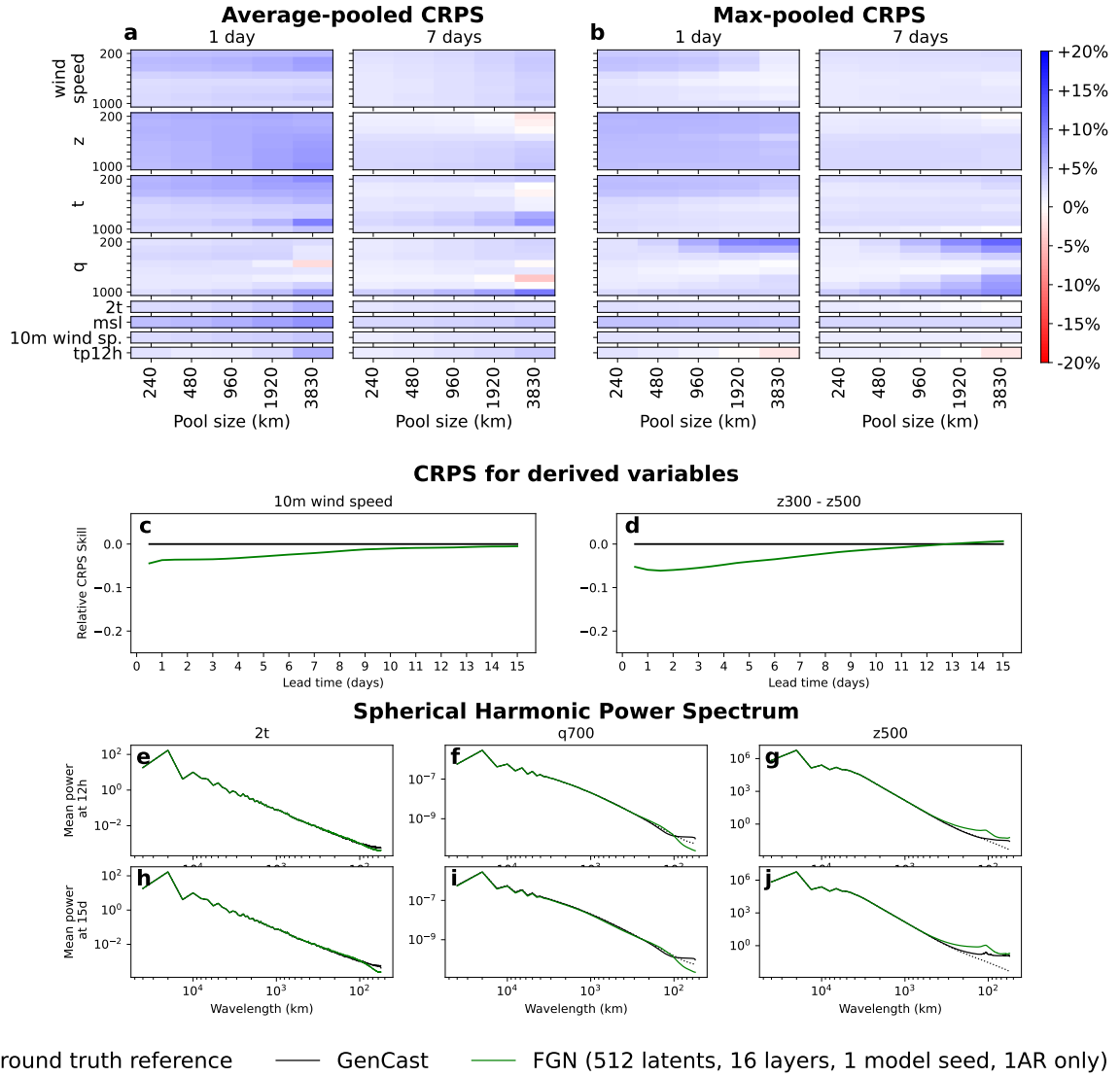
Figure A.3 | Evaluations of spatial and inter-variable structure in the joint forecast distributions, comparing GenCast with a version FGN which has the same timestep (12 hours), same number of model seeds (1) and same model capacity (16-layers, and latent dimension of 512)—and which has also not undergone any autoregressive training. (a, b) CRPS scorecards evaluating average-pool and max-pool fields respectively, for different pool sizes, at 1 and 7 day lead times. (c, d) Evaluations of skill on two quantities of interest derived from predicted fields, 10m wind speed and $z300 - z500$. (e - j) Power spectra for a number of variables.
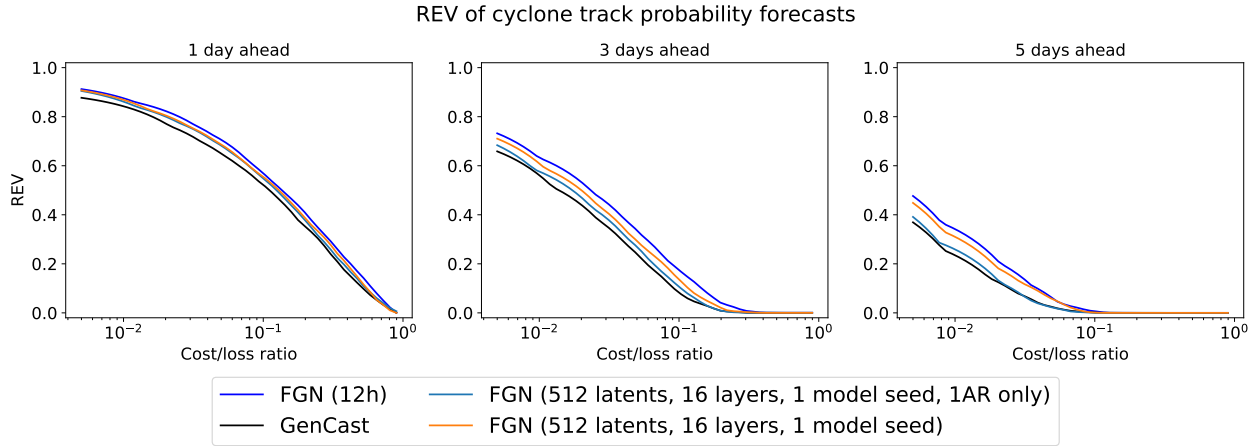
Figure A.4 | REV of cyclone track probability forecasts, comparing the impact of autoregressive training, model size, and model ensembling. The single-step trained FGN version of the same depth and width as GenCast (green curve) performs as well as or better than GenCast. Adding autoregressive training (orange curve) further improves REV, and increasing the model size and using a model ensemble (blue) performs best.

regridded to 0.25° using MetView (Russell and Kertész, 2017) defaults (conservative downsampling). Figures A.6 and A.7 present evaluations on dates in 2023 before and after this cycle change respectively, showing that while ENS improves in some variables after the upgrade, GenCast still outperforms ENS on 95% of targets.

For completeness, Figure A.8 compares FGN to ENS, showing very large improvements (note the 30% colorbar vlim), averaging more than 10% across all lead times, ranging up to close to 30%. Similarly, we report performance before and after the cycle change in Figures A.9 and A.10, where FGN outperforms ENS on more than 99% of targets in both cases. In this comparison, technically neither the fair nor biased CRPS estimators or exactly appropriate - the FGN ensemble being evaluated has 56 members while ENS has 50, but FGN's ensemble also violates the independence assumption of the fair CRPS estimator when we enforce that each model seed generates an equal number of forecasts. Figures A.8 to A.10 report results using the fair estimator, which we found gives almost indistinguishable results from the biased setting, and so we are confident that these are representative.

A.5.2. Extended Results

Figure A.11 shows the ensemble mean RMSE scorecards comparing FGN to GenCast.

We evaluate precipitation forecasting skill using RMSE and CRPS as well as Stable Equitable Error in Probability Space (SEEPS), which is often used as a specialized metric for precipitation evaluation (Haiden et al., 2012; North et al., 2013; Rodwell et al., 2010), using the same methodology as in Lam et al. (2023) and Price et al. (2025). Because SEEPS is a metric for deterministic categorical forecasts (dry, light-rain, and heavy-rain) we evaluated the ensembles computing the category of the ensemble mean. Results for 12-hour and 24-hour accumulated precipitation are shown in Figure A.12 and Figure A.13 respectively.

Figures A.14 to A.16, and A.17 show REV results on prediction of extreme high 10m wind speed, high 2m temperature, low 2m temperature, and low mean sea level pressure, respectively, for different extreme thresholds, across different cost/loss ratios, for different lead times. FGN achieves significantly higher REV than GenCast ($p < 0.05$), with a few exceptions at longer lead times for certain cost-loss
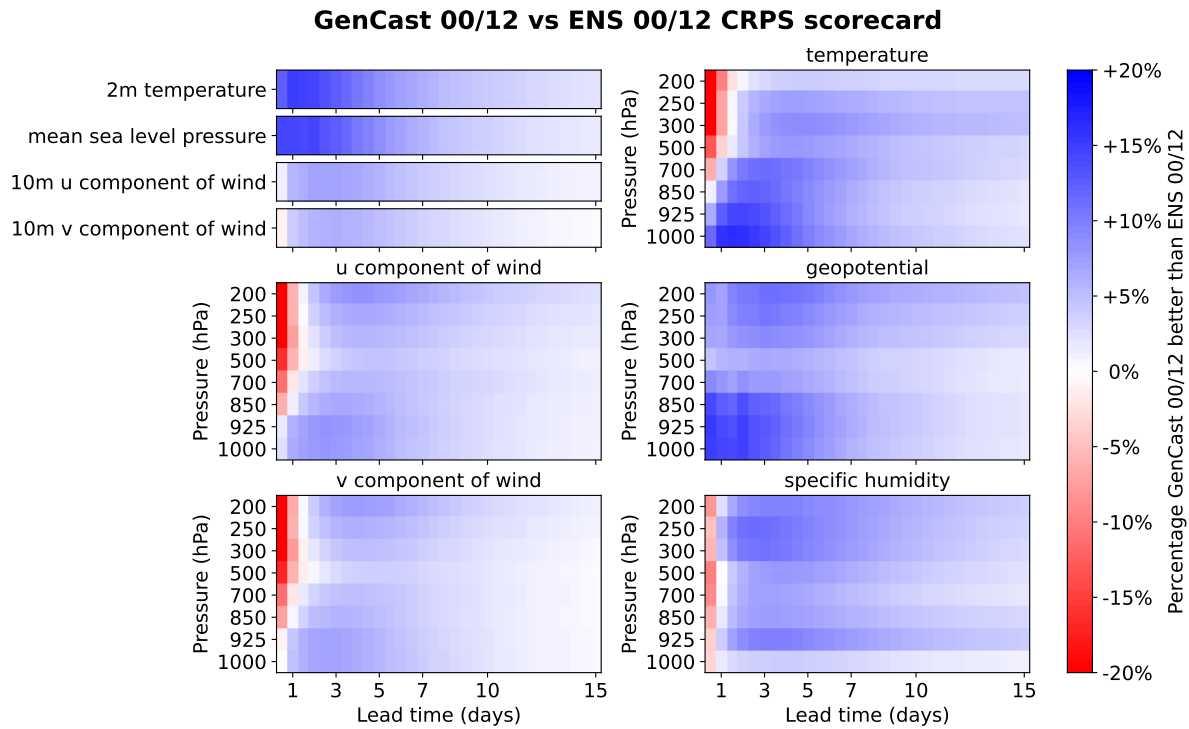
Figure A.5 | CRPS scorecard comparing the currently operational version of GenCast (used as the baseline in this work) to ENS on all 00/12 initialization times in 2023, against HRES-fc0 ground truth. GenCast outperforms ENS on 96.5% of targets, with an average improvement of 4.7% and a maximum improvement of 16.6%. This confirms that the results established in Price et al. (2023) on ERA5 data generally carry over to the operational setting.
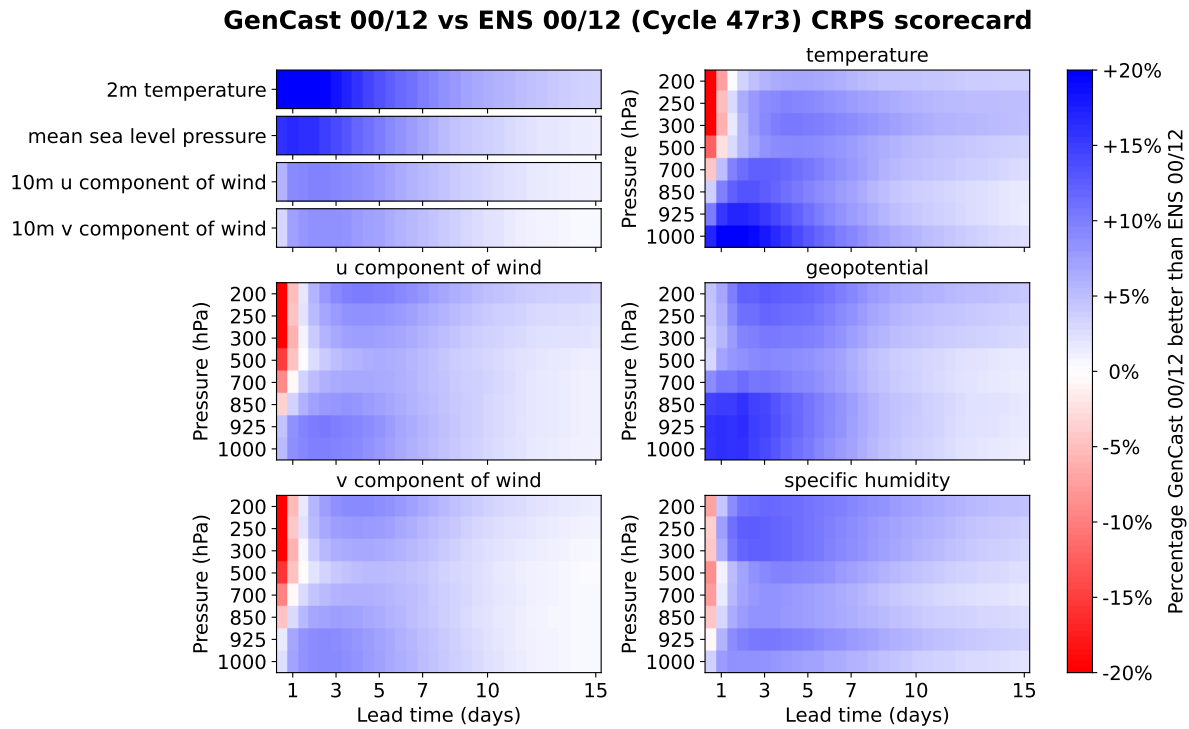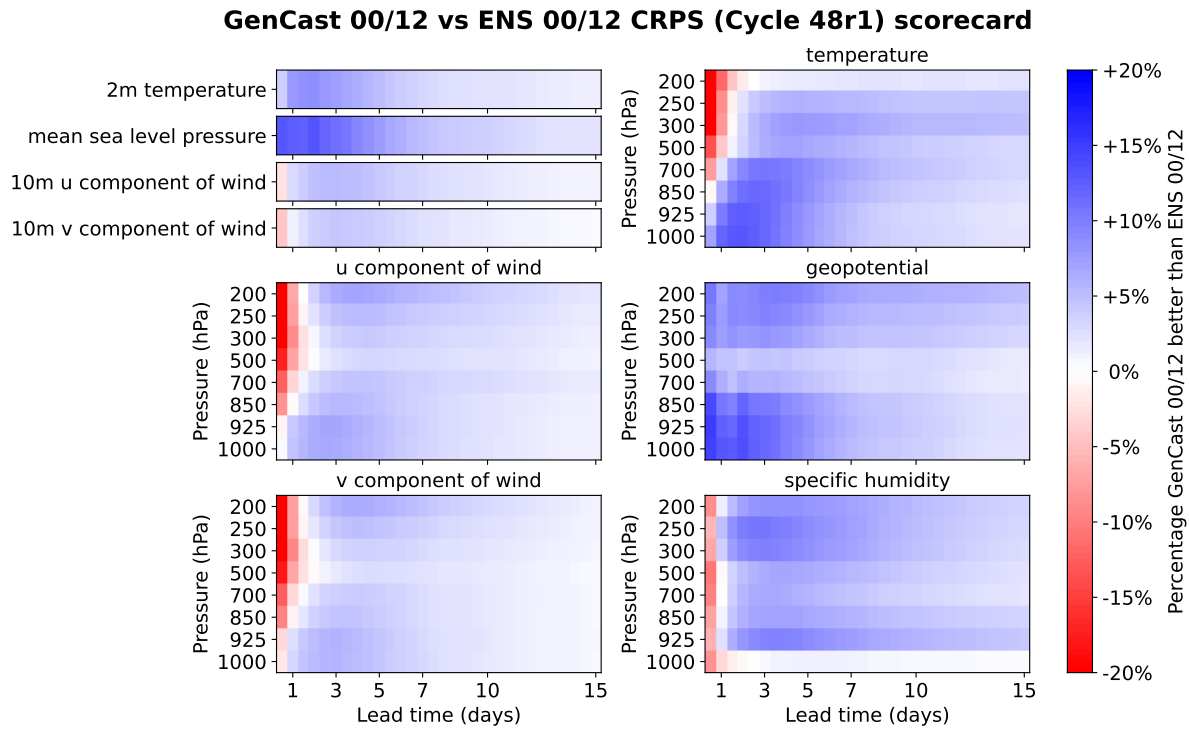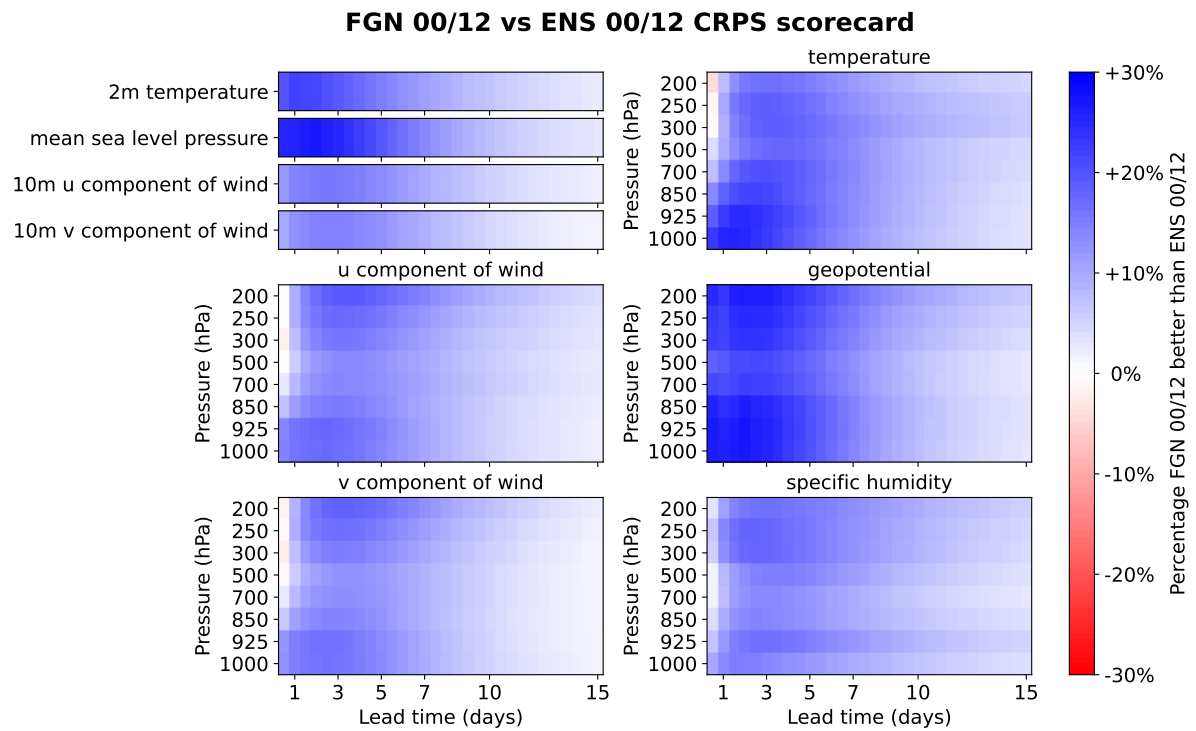
Figure A.6 | CRPS scorecard comparing the currently operational version of GenCast (used as the baseline in this work) to ENS on 00/12 initialization times before the 27th of June 2023, against HRES-fc0 ground truth. On these initializations, both models natively operate at resolutions (close to) 0.25°. GenCast outperforms ENS on 97.0% of targets, with an average improvement of 5.6% and a maximum improvement of 23.2%.

Figure A.7 | CRPS scorecard comparing the currently operational version of GenCast (used as the baseline in this work) to ENS on 00/12 initialization times after the 27th of June 2023, against HRES-fc0 ground truth. On these dates, ENS natively operates at a resolution 0.1° and is conservatively downsampled to 0.25° for this evaluation. GenCast outperforms ENS on 95.0% of targets, with an average improvement of 3.7% and a maximum improvement of 15.2%.

Figure A.8 | CRPS scorecard comparing FGN to ENS, on all 00/12 initialization times in 2023, against HRES-fc0 ground truth (note the vlim of 30%). FGN achieves better CRPS on 99.3% of targets, with an average improvement of 10.8% and a maximum improvement of 27.7%. Notably, while GenCast suffered compared to ENS at short lead times, FGN is strong across all lead times.
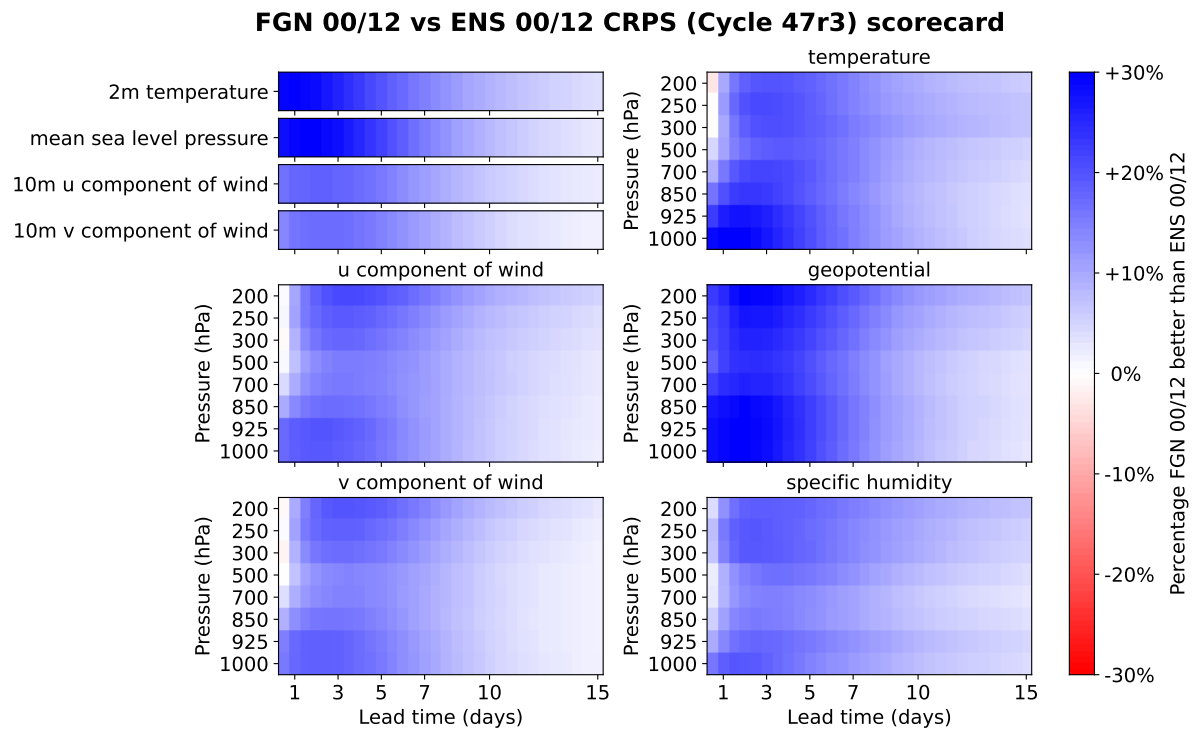
Figure A.9 | CRPS scorecard comparing FGN to ENS, on 00/12 initialization times before the 27th of June 2023, against HRES-fc0 ground truth (note the vlim of 30%). On these initializations, both models natively operate at resolutions (close to) 0.25°. FGN achieves better CRPS on 99.5% of targets, with an average improvement of 12.0% and a maximum improvement of 30.6%.
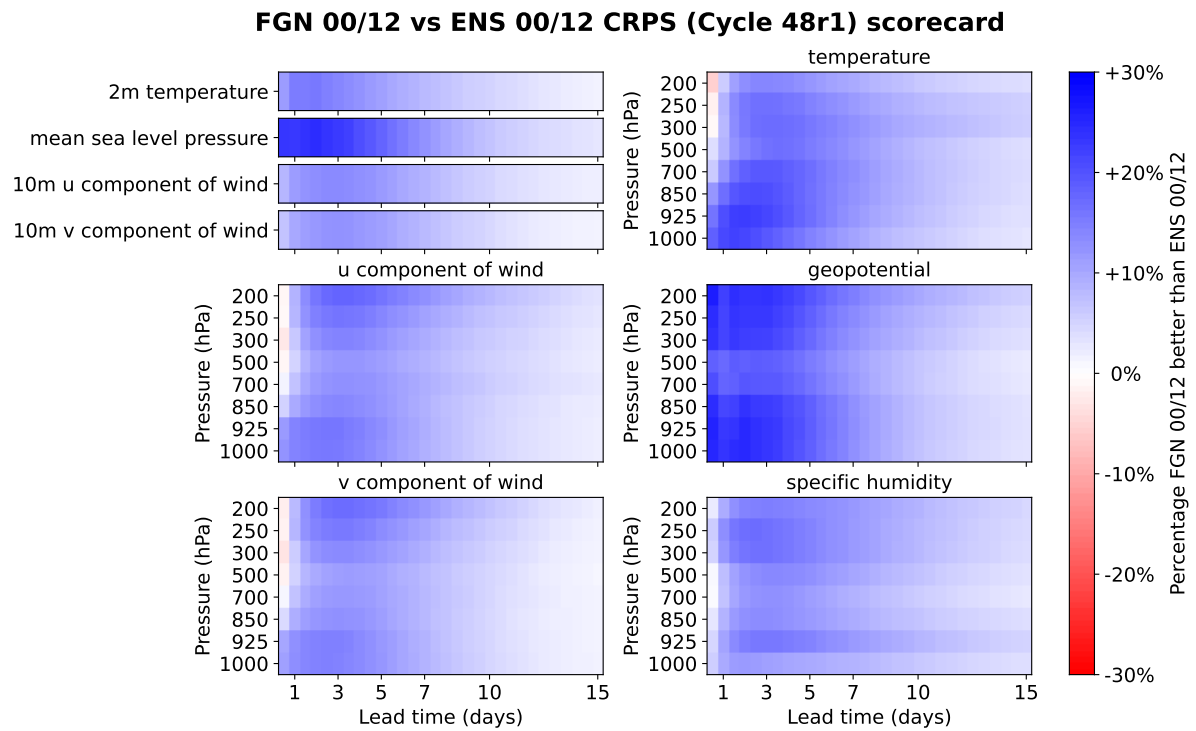
Figure A.10 | CRPS scorecard comparing ENS to FGN on 00/12 initialization times, after the 27th of June 2023, against HRES-fc0 ground truth (note the vlim of 30%). On these initializations, ENS natively operates at a resolution 0.1° and is conservatively downsampled to 0.25° for comparison. FGN achieves better CRPS on 99.2% of targets, with an average improvement of 9.7% and a maximum improvement of 26.5%.

FGN vs GenCast ensemble mean RMSE scorecard
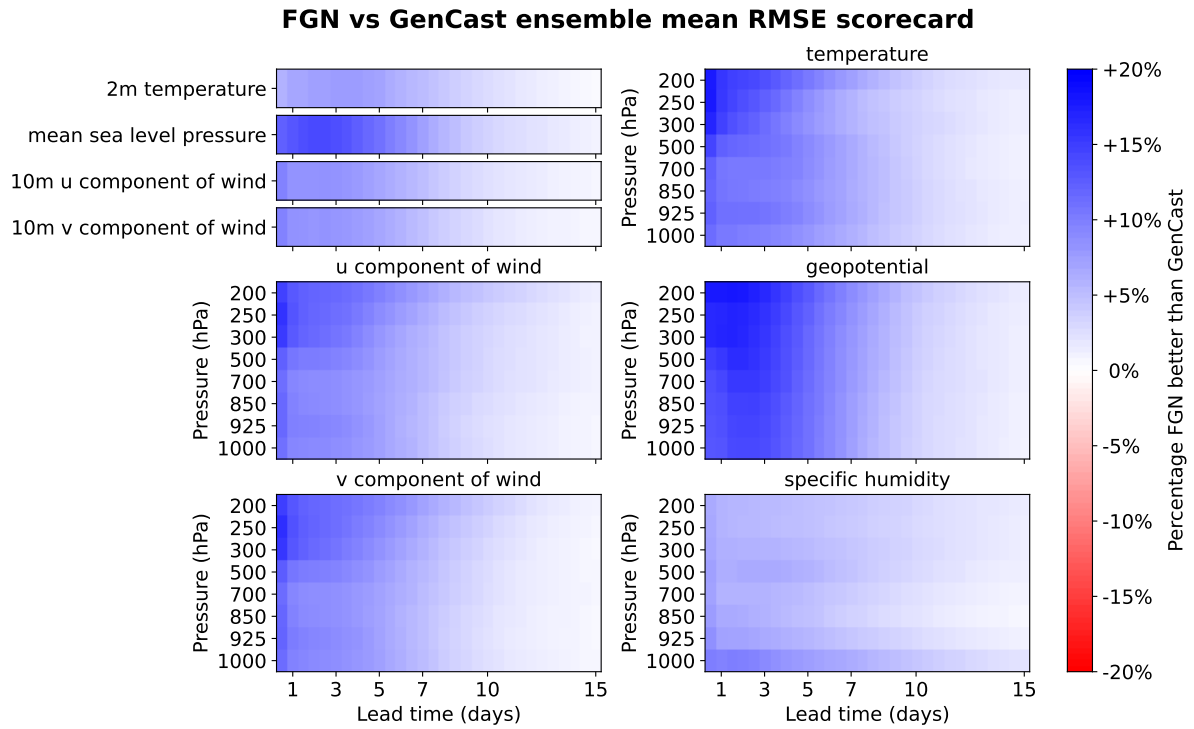
Figure A.11 | Ensemble mean RMSE scorecard comparing FGN to GenCast, where blue cells indicate FGN outperforms GenCast, red denotes where GenCast is better. FGN is significantly better ($p < 0.05$) in 100% of cases, with improvements of up to 18% and an average of 5.8%.
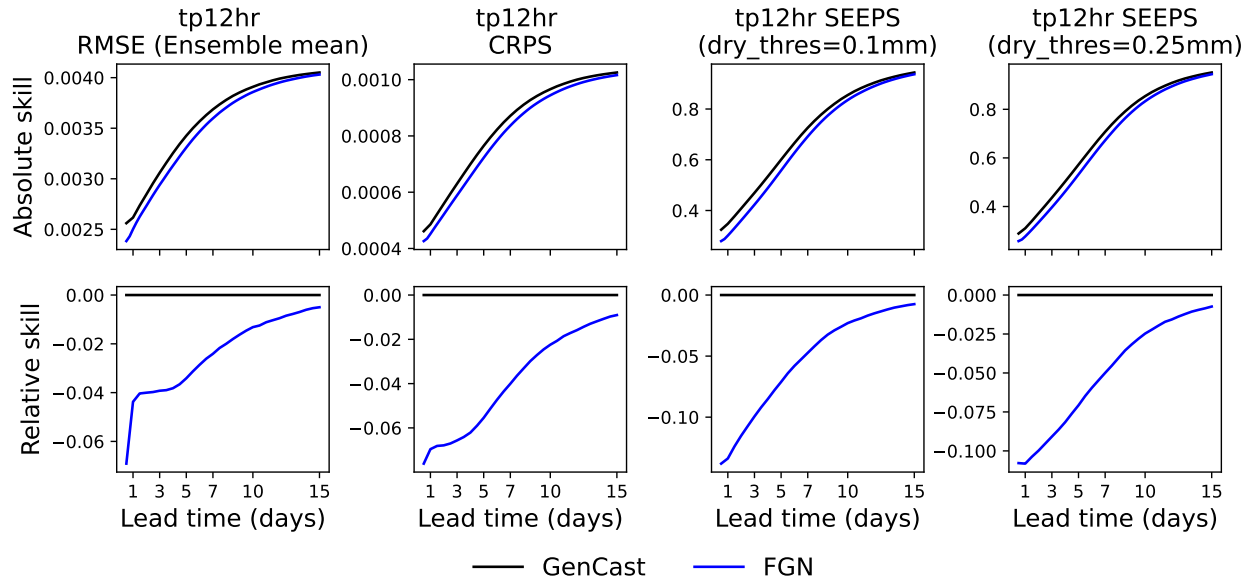
Figure A.12 | Results on 12h accumulated precipitation against ERA5 show FGN outperforming GenCast when comparing Ensemble-Mean RMSE, CRPS and SEEPS (with dry thresholds of 0.1 and 0.25) for the predicted total accumulated precipitation over 12 hours. All metrics were calculated globally and over the full test period, except for SEEPS which excludes very dry regions according to the criteria in Lam et al. (2023). SEEPS is evaluated on the ensemble mean prediction. Absolute and relative plots are shown in the top and bottom rows respectively.
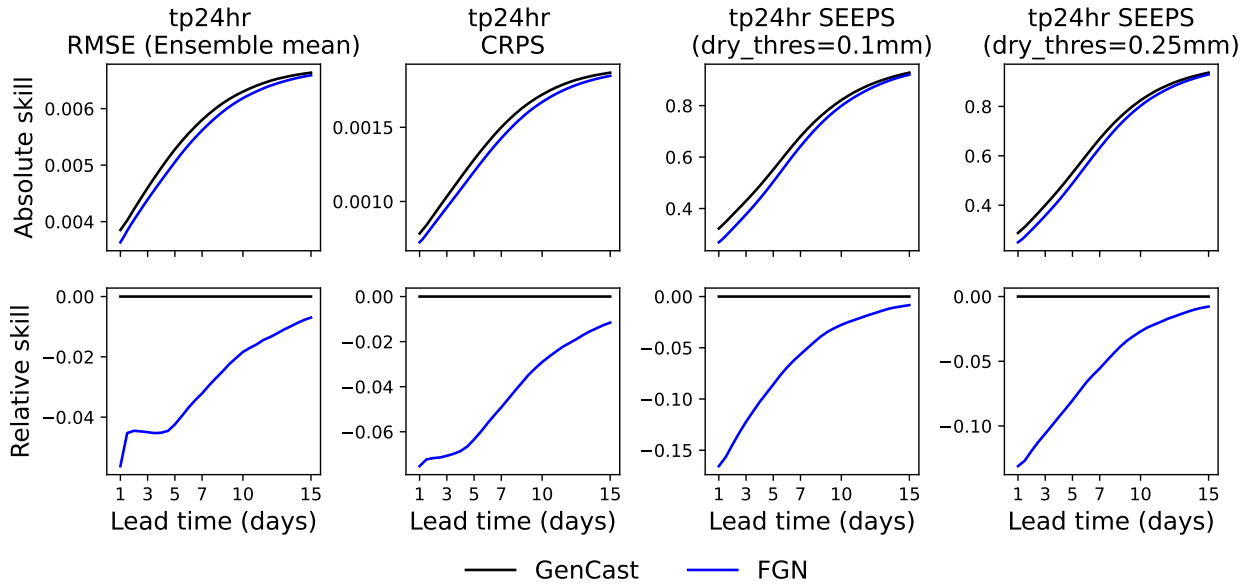
Figure A.13 | Similar to Figure A.12, results on 24h accumulated precipitation against ERA5 also show FGN outperforming GenCast.

and lead time combinations for high 2m temperature.

We provide additional spectral results to supplement those the main paper, showing 10 representative variables: z500, z850, q700, q925, t850, t300, u850, 2t, 10u and msl (Figure A.18 and Figure A.19).
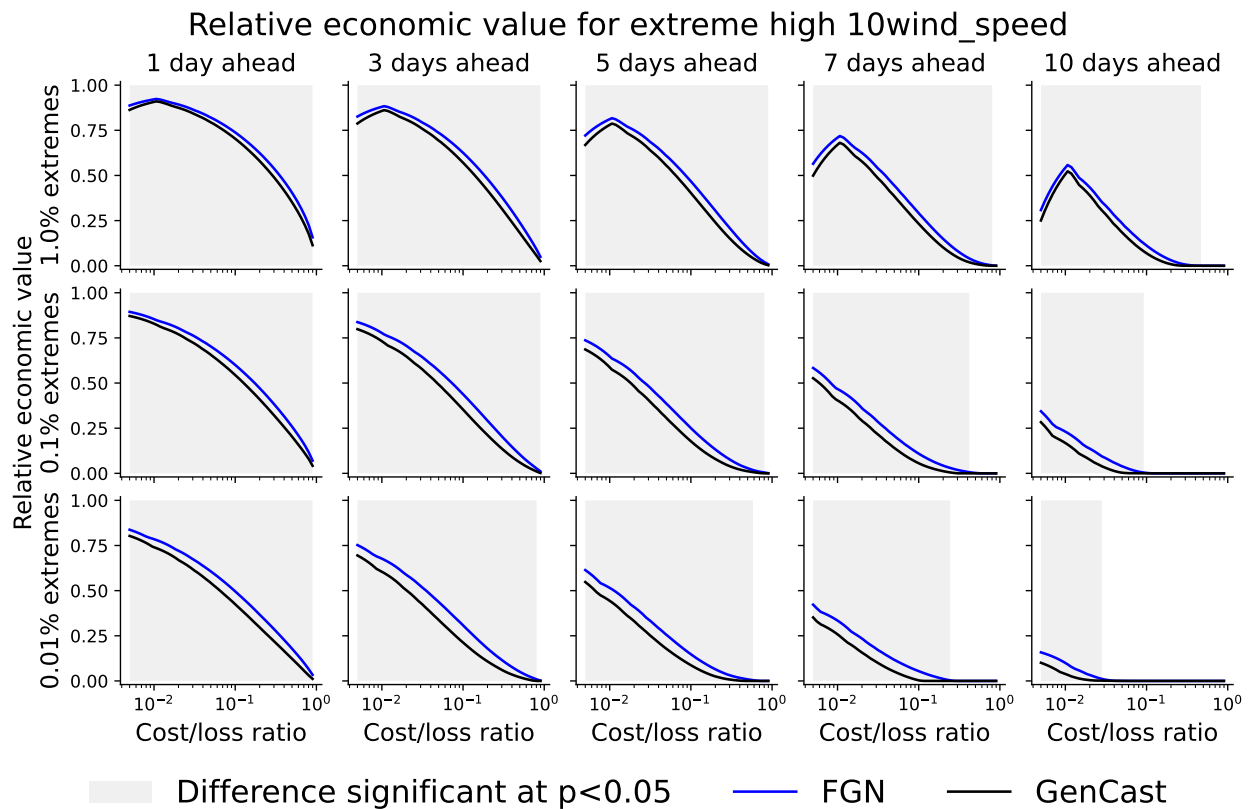
Figure A.14 | REV of predictions of extreme high 10m wind speed, across multiple cost/loss ratios, exceedance thresholds, and lead times. Cost/loss regions in which the difference between FGN and GenCast is statistically significant ($p < 0.05$) are shaded in gray.
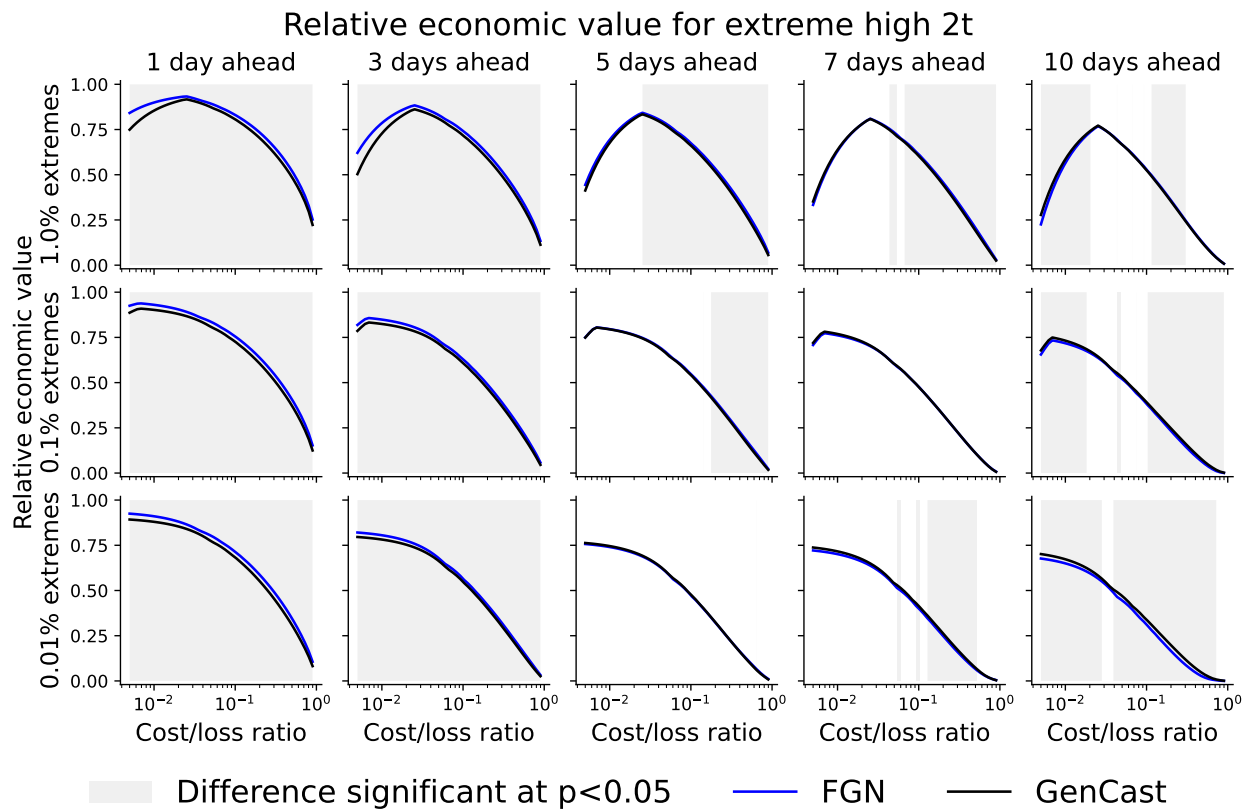
Figure A.15 | REV of predictions of extreme high 2m temperature, across multiple cost/loss ratios, exceedance thresholds, and lead times. Cost/loss regions in which the difference between FGN and GenCast is statistically significant ($p < 0.05$) are shaded in gray.
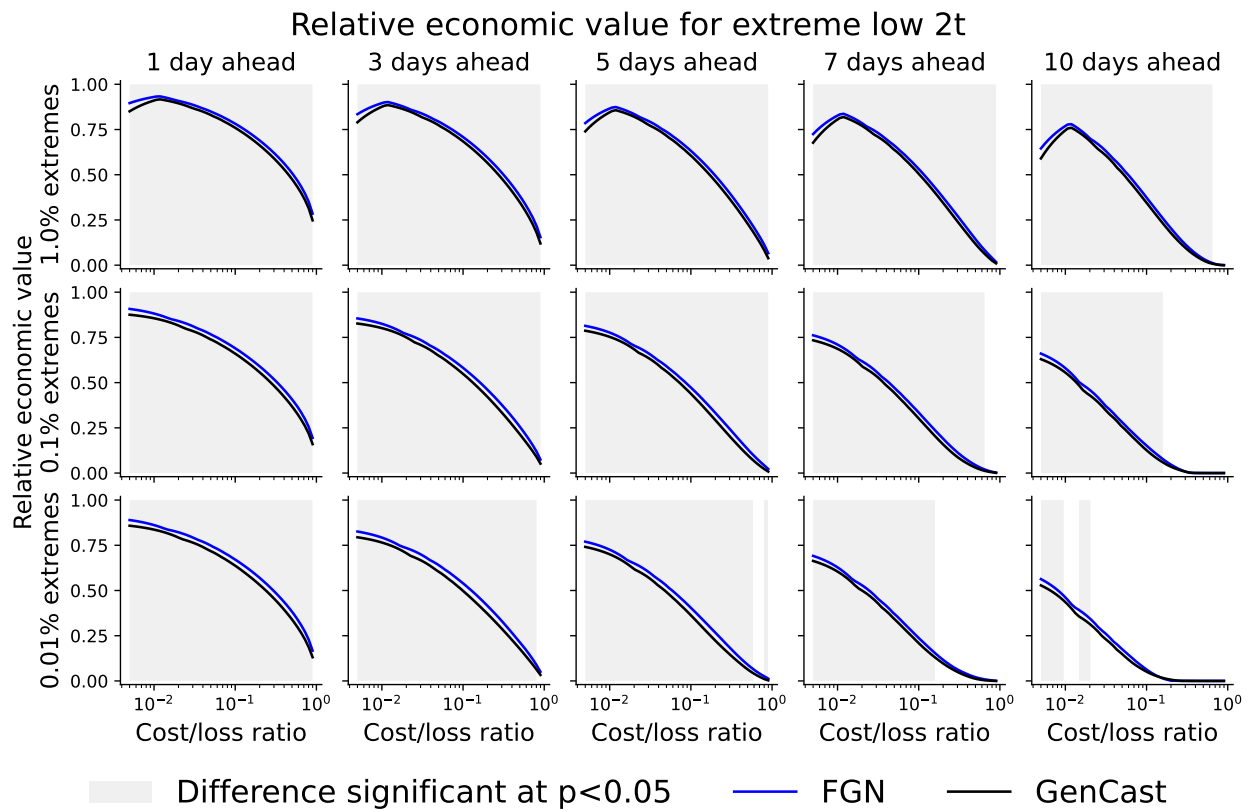
Figure A.16 | REV of predictions of extreme low 2m temperature, across multiple cost/loss ratios, exceedance thresholds, and lead times. Cost/loss regions in which the difference between FGN and GenCast is statistically significant ($p < 0.05$) are shaded in gray.
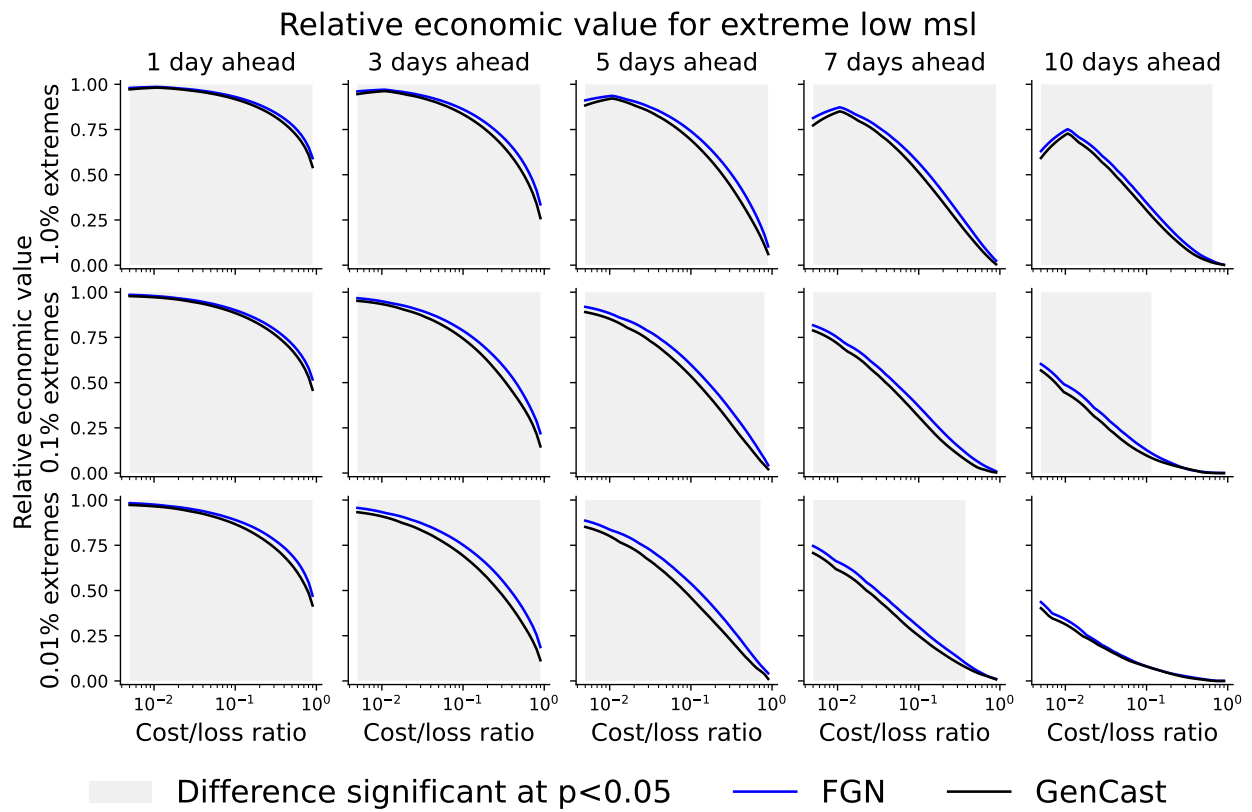
Figure A.17 | REV of predictions of extreme low mean sea level pressure, across multiple cost/loss ratios, exceedance thresholds, and lead times. Cost/loss regions in which the difference between FGN and GenCast is statistically significant ($p < 0.05$) are shaded in gray.
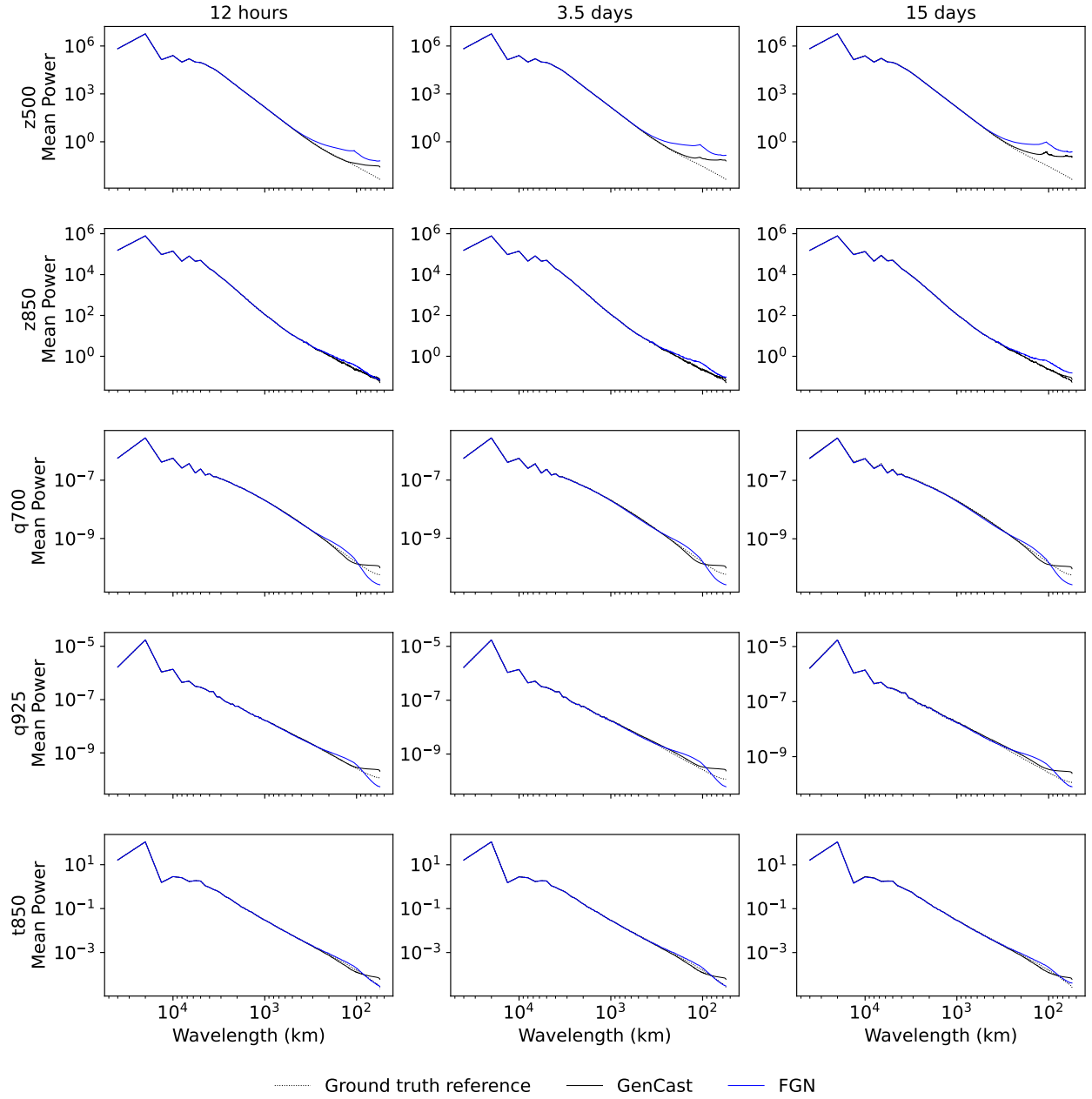
Figure A.18 | Power spectrum plots for z500, z850, q700, q925 and t850.
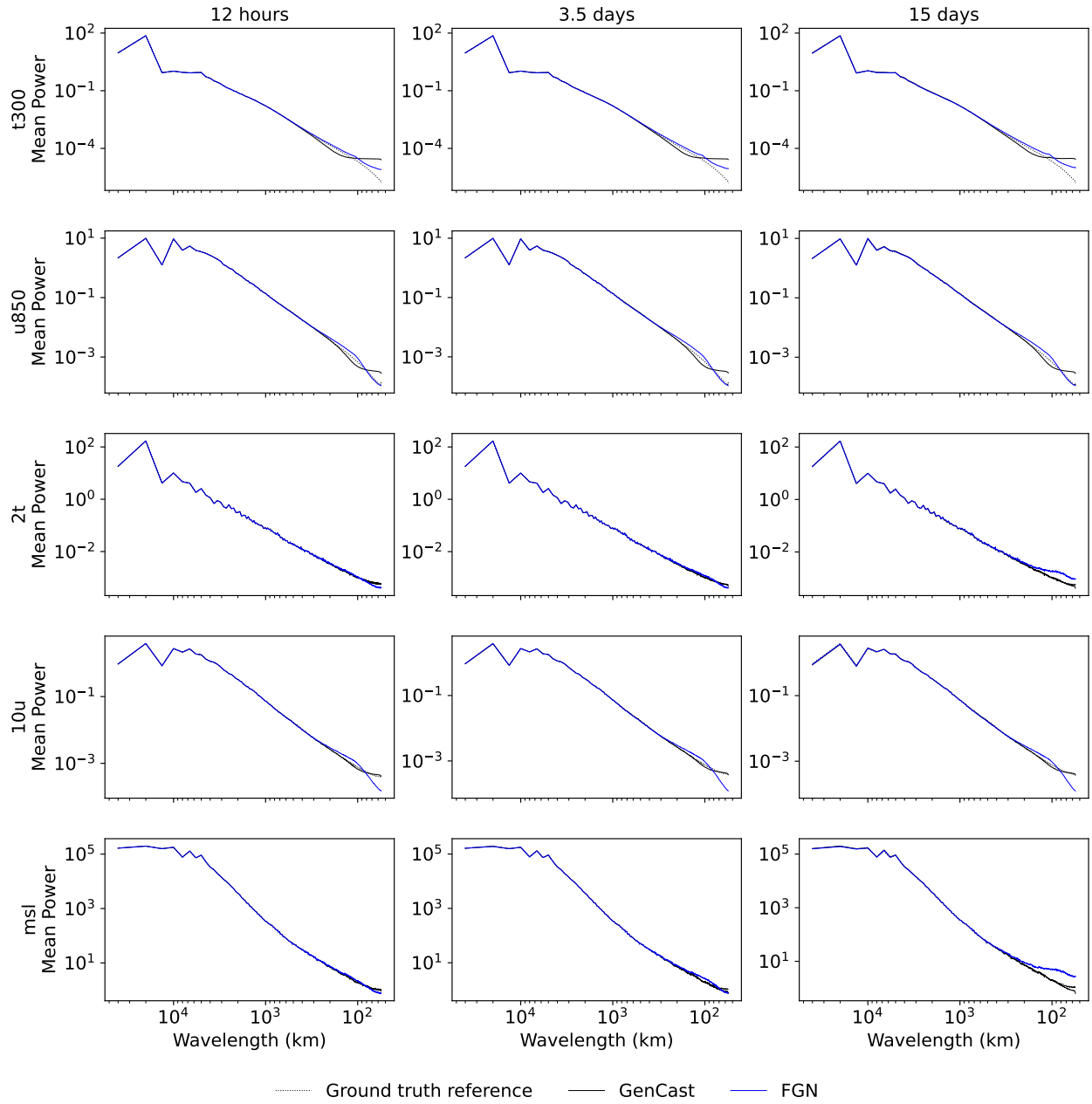
Figure A.19 | Power spectrum plots for t300, u850, 2t, 10u and msl.

A.6. Forecast visualizations

We provide forecast visualizations for 10 representative variables: 2t (Figure A.20), 10u (Figure A.21, msl Figure A.22), q700 (Figure A.23), q925 (Figure A.24), t300 (Figure A.25), t850 (Figure A.26), u850 (Figure A.27), z500 (Figure A.28) and z850 (Figure A.29). For each variable-lead time combination we choose the 2023 initialization time with median CRPS error. This means that each figure show snapshots from a different forecast trajectory at each lead time. In each case we plot the first ensemble member.
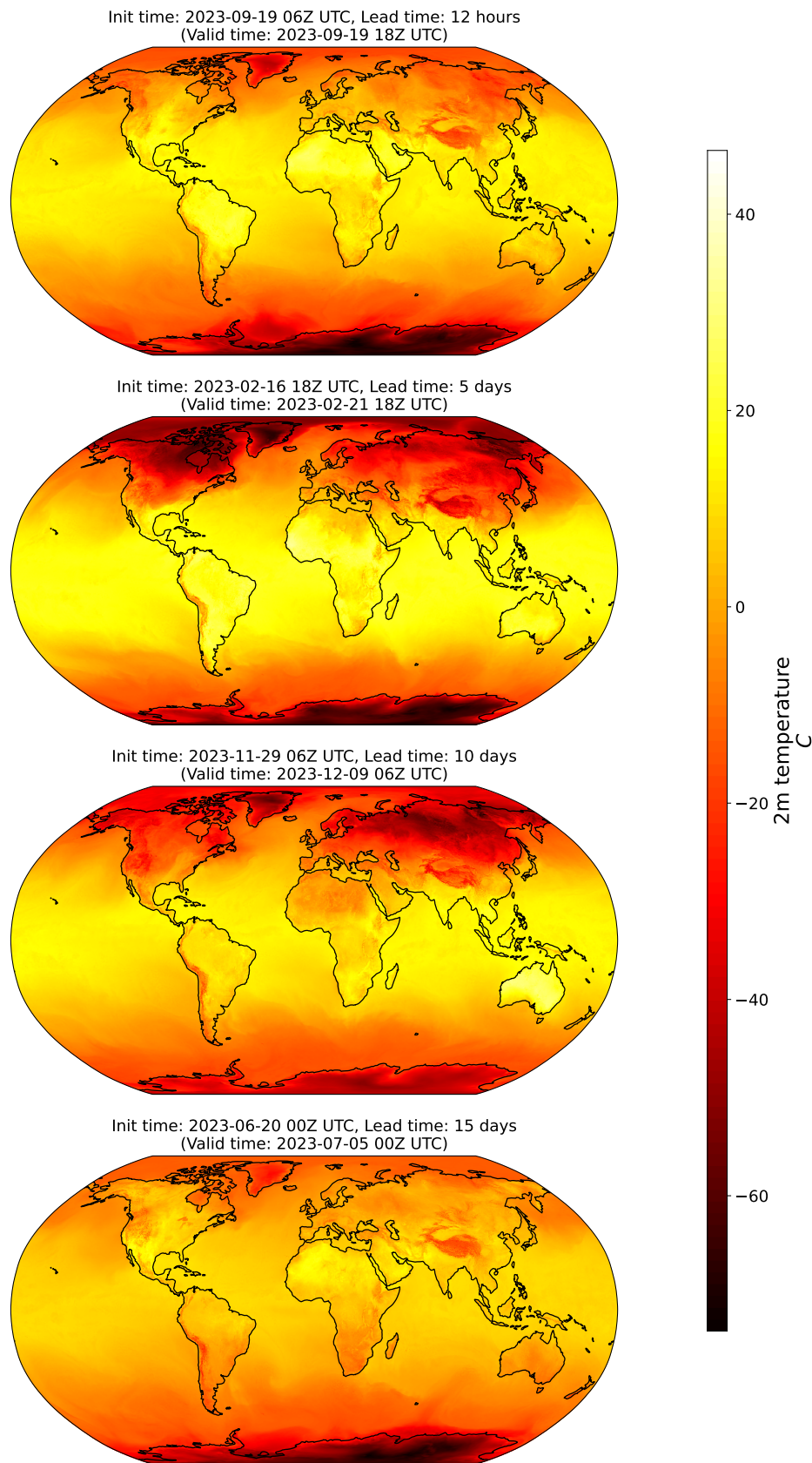
Figure A.20 | Visualization of 2m temperature.
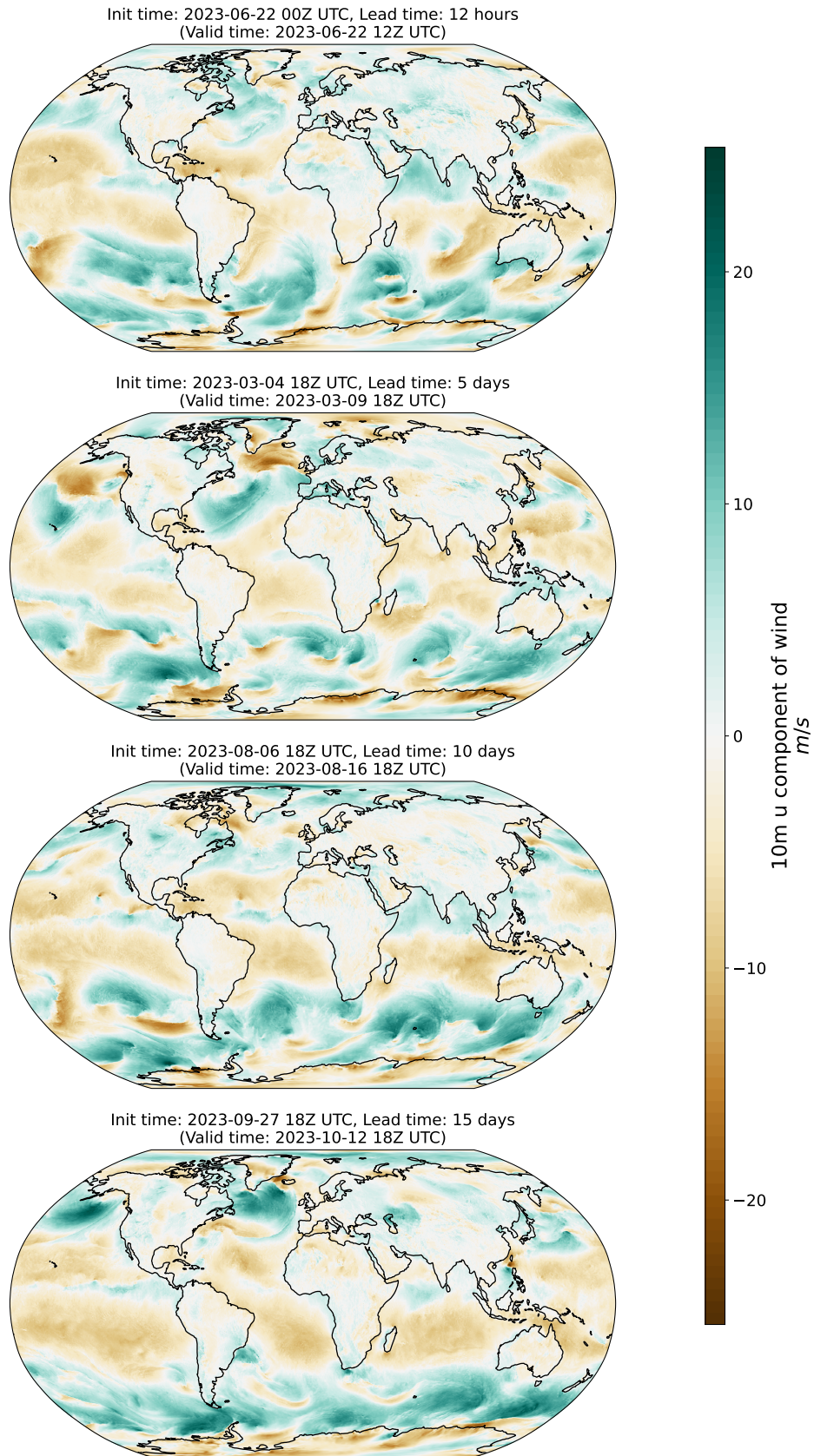
Figure A.21 | Visualization of 10 m u component of wind.
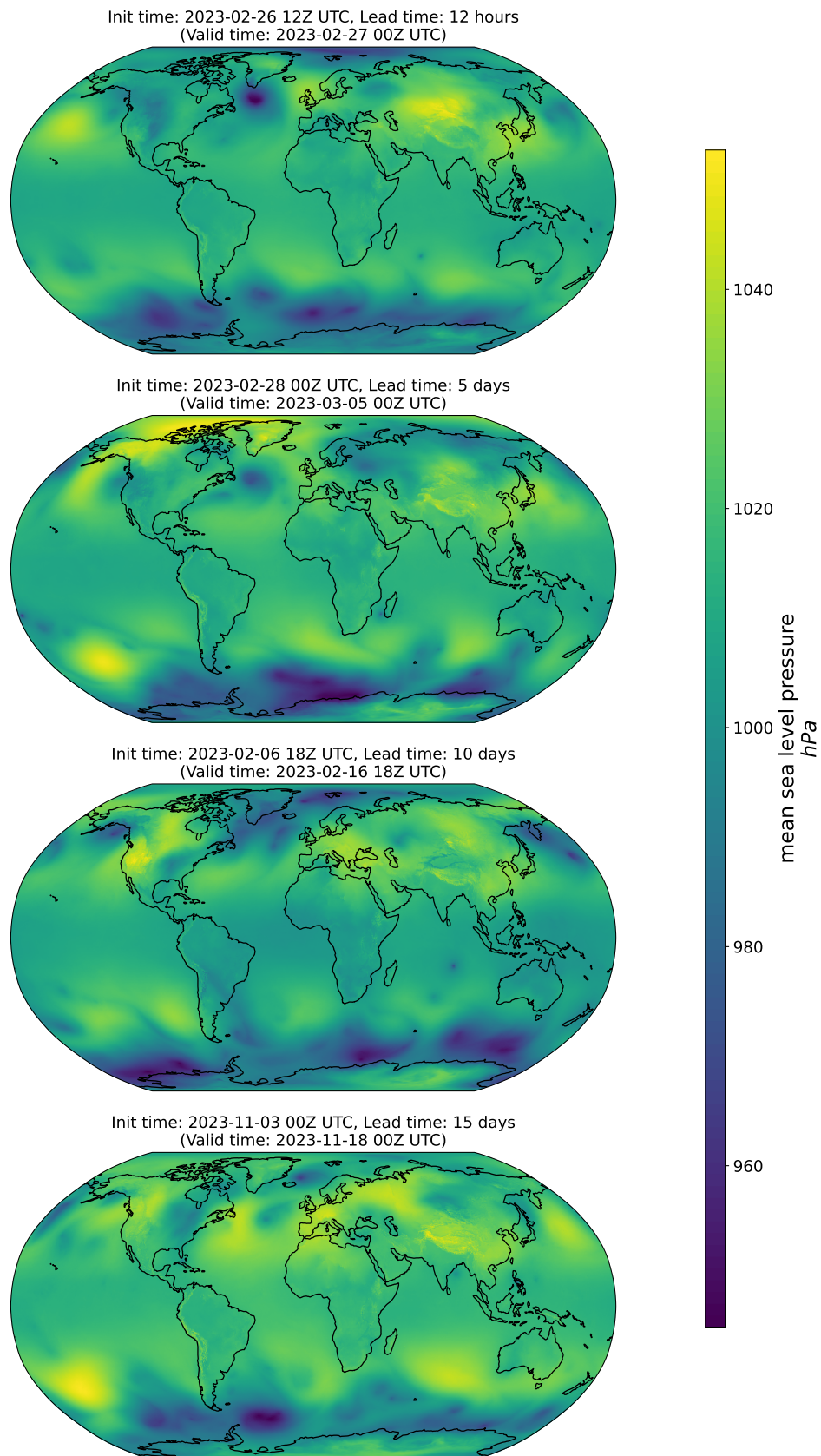
Figure A.22 | Visualization of mean sea level pressure.
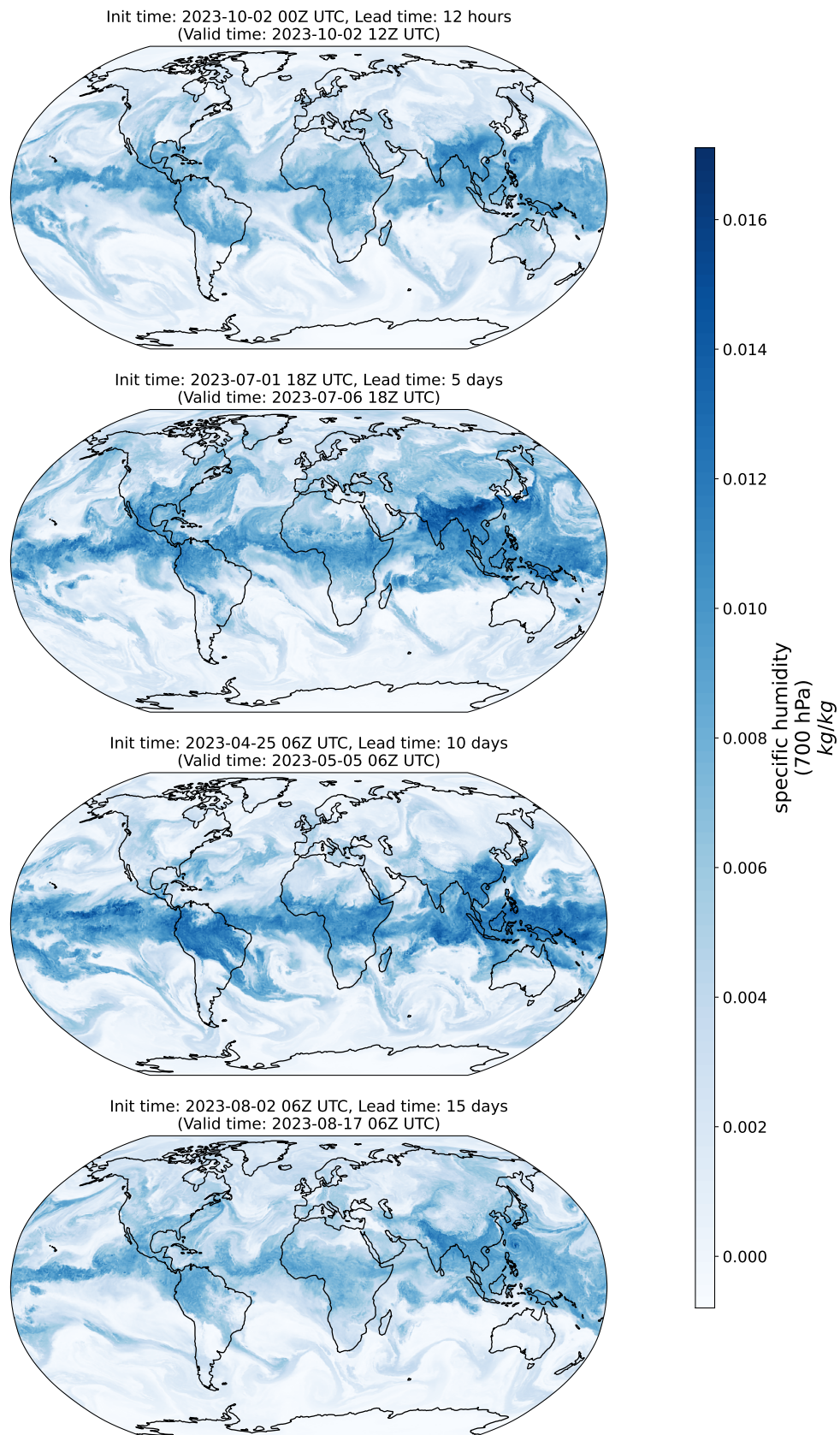
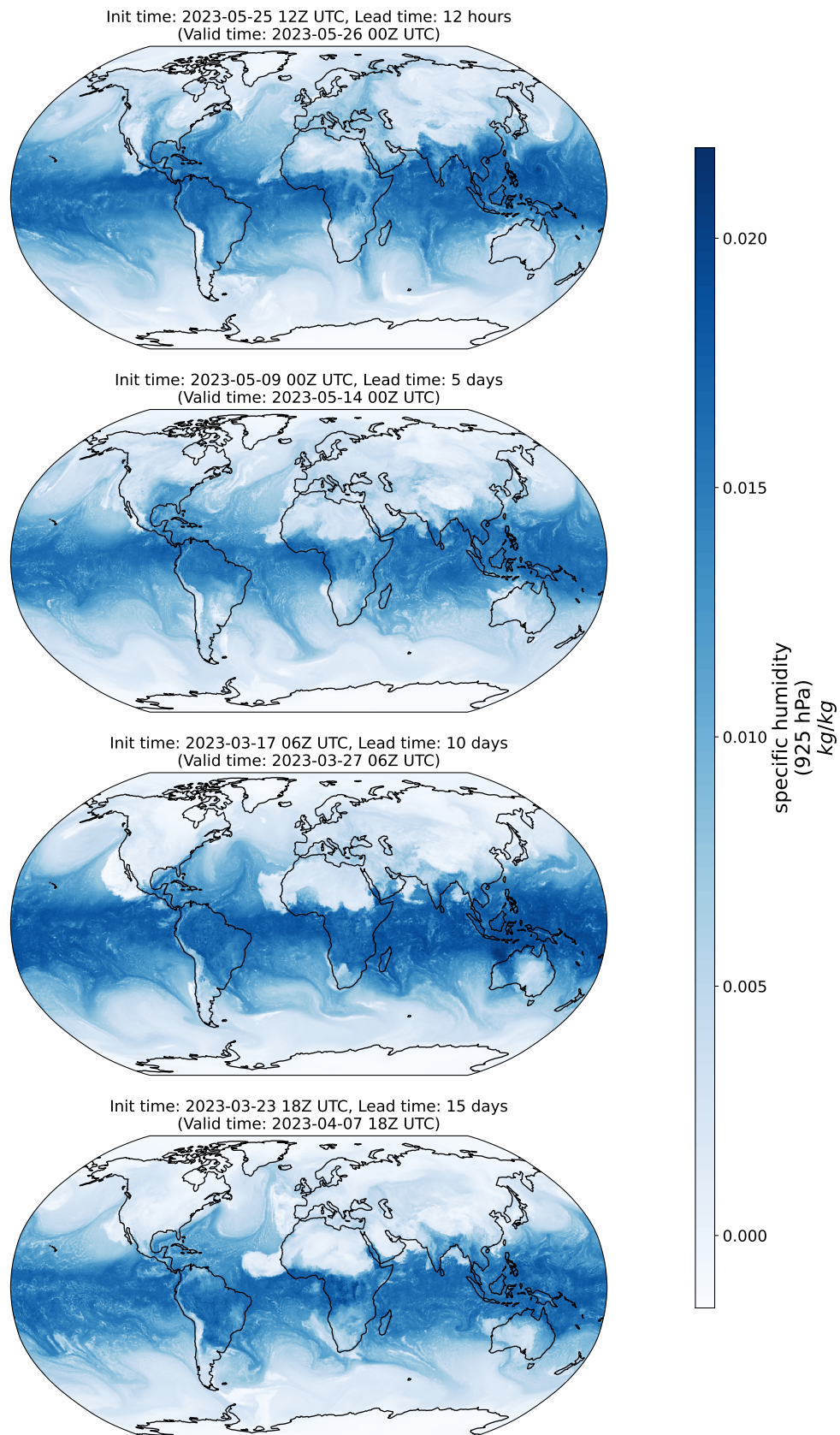Figure A.23 | Visualization of specific humidity at 700 hPa.

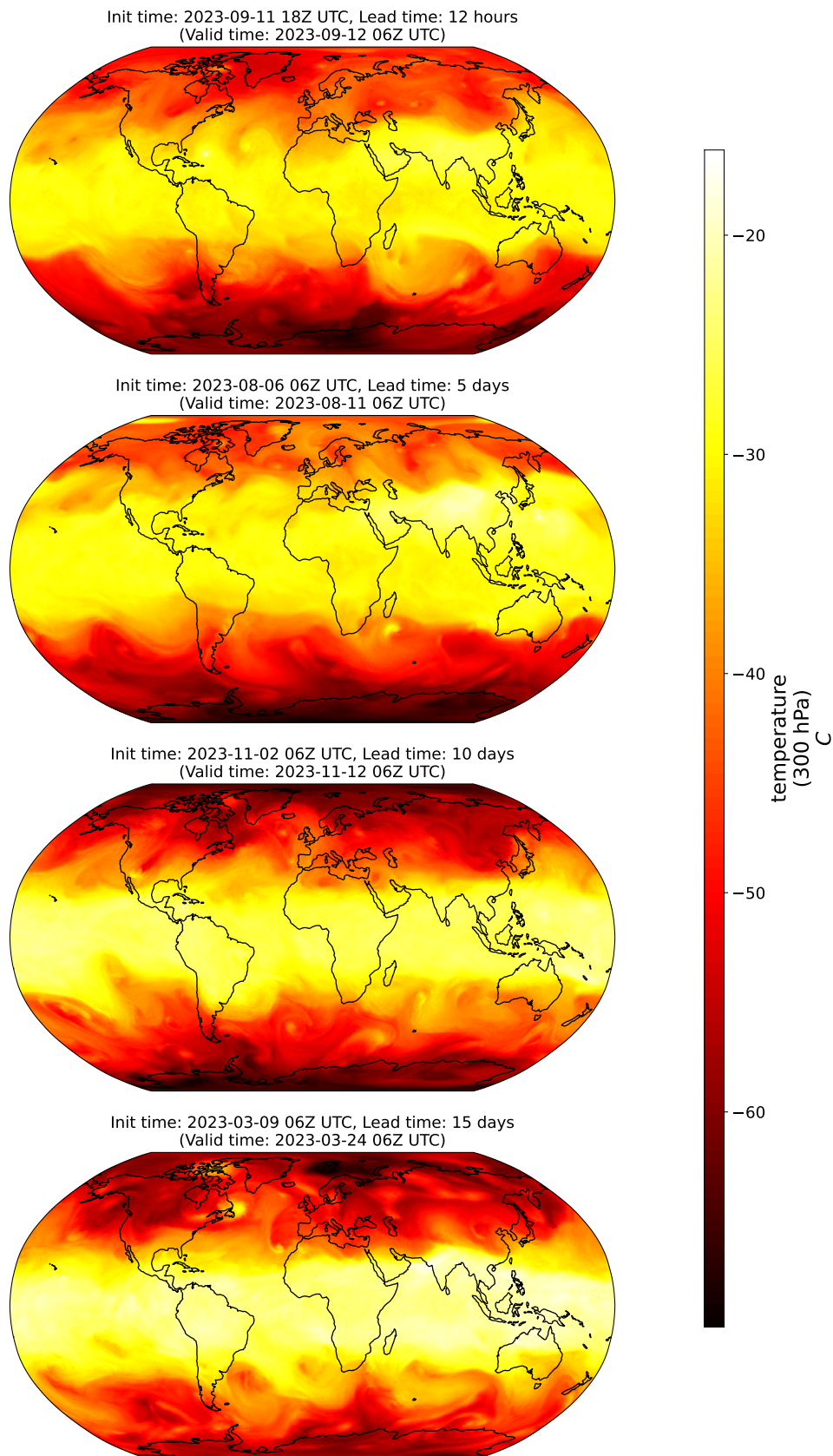Figure A.24 | Visualization of specific humidity at 925 hPa.

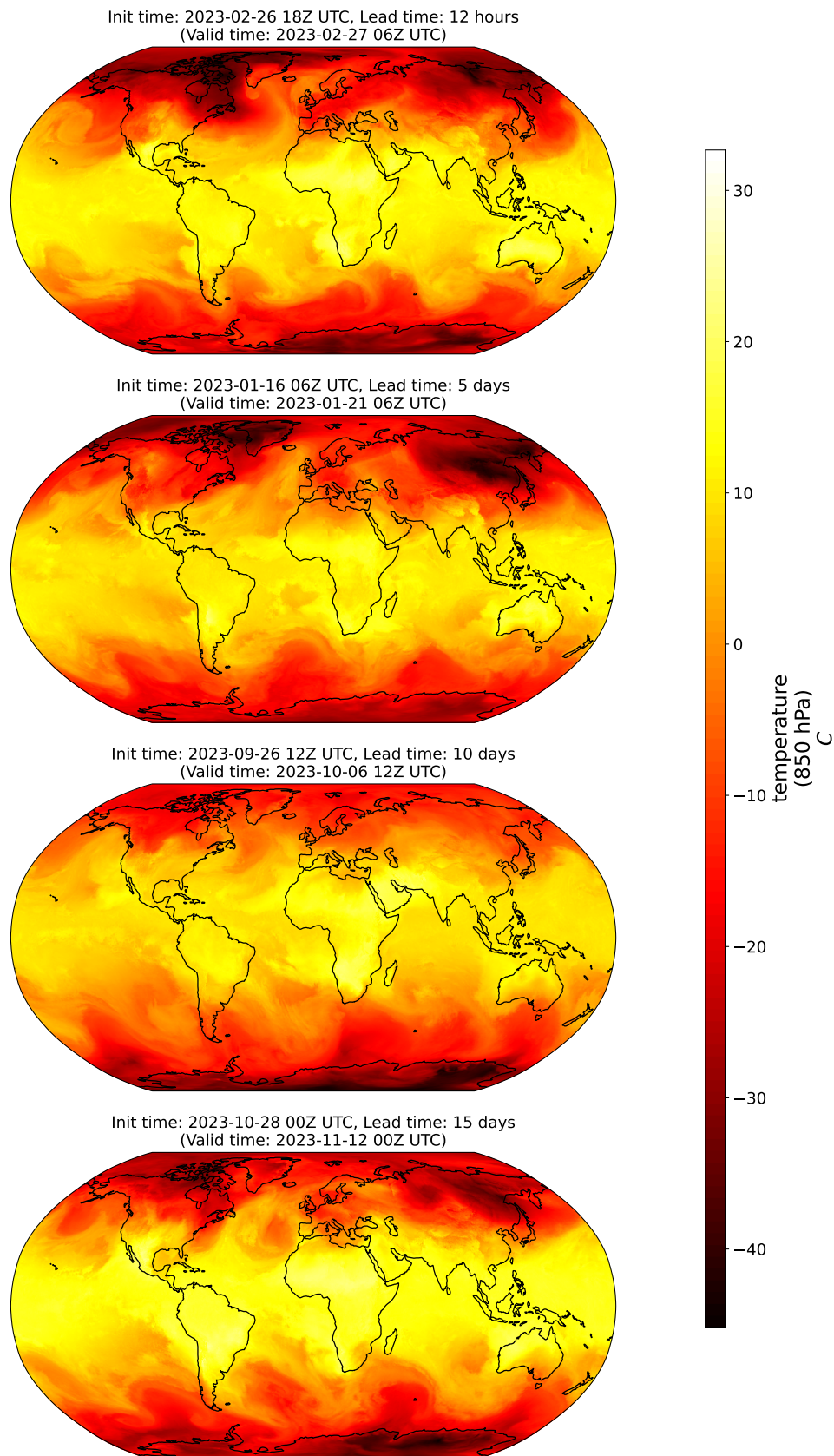Figure A.25 | Visualization of temperature at 300 hPa.

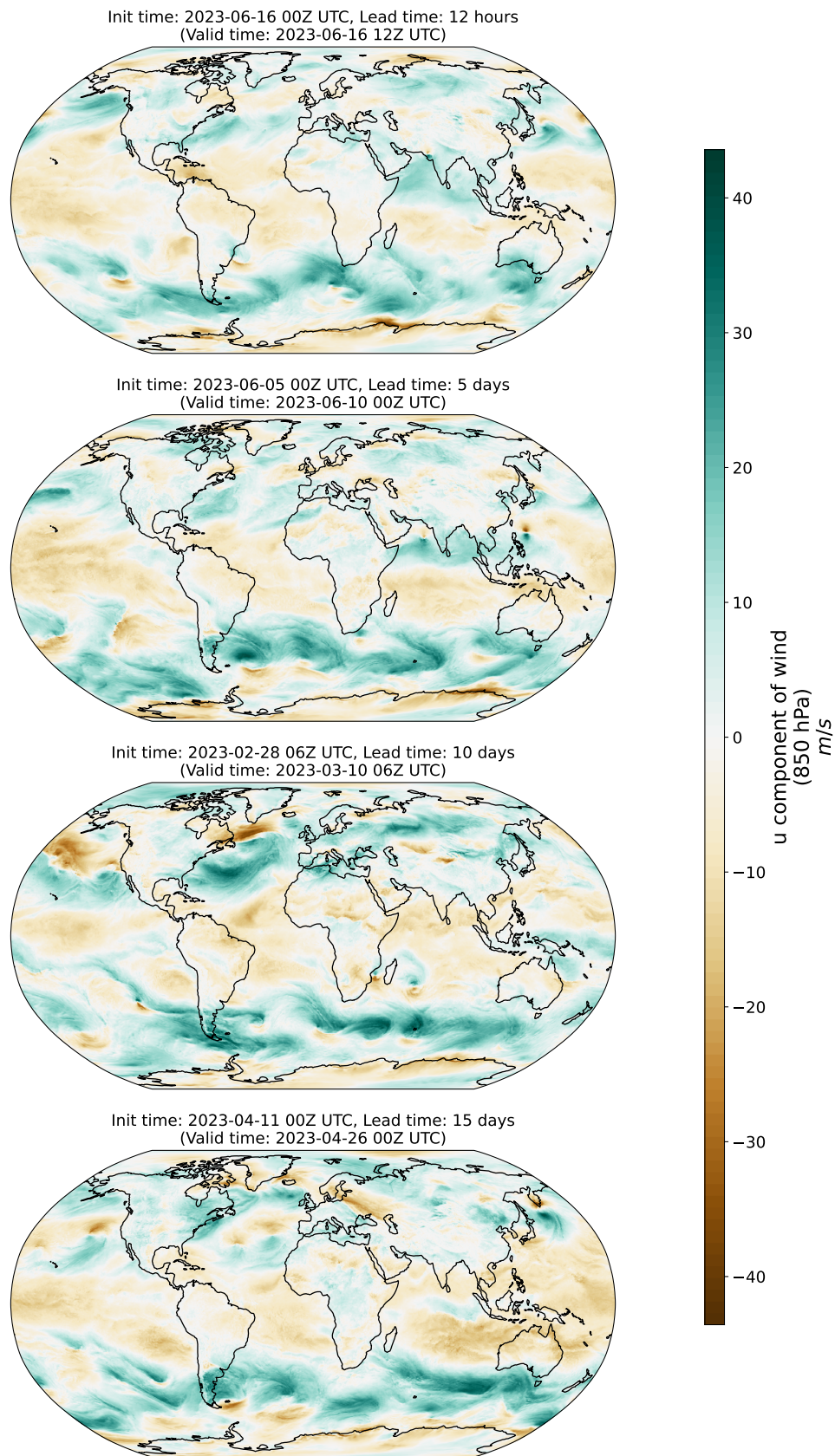Figure A.26 | Visualization of temperature at 850 hPa.

Init time: 2023-06-16 00Z UTC, Lead time: 12 hours
(Valid time: 2023-06-16 12Z UTC)

Init time: 2023-06-05 00Z UTC, Lead time: 5 days
(Valid time: 2023-06-10 00Z UTC)

Init time: 2023-02-28 06Z UTC, Lead time: 10 days
(Valid time: 2023-03-10 06Z UTC)

Init time: 2023-04-11 00Z UTC, Lead time: 15 days
(Valid time: 2023-04-26 00Z UTC)

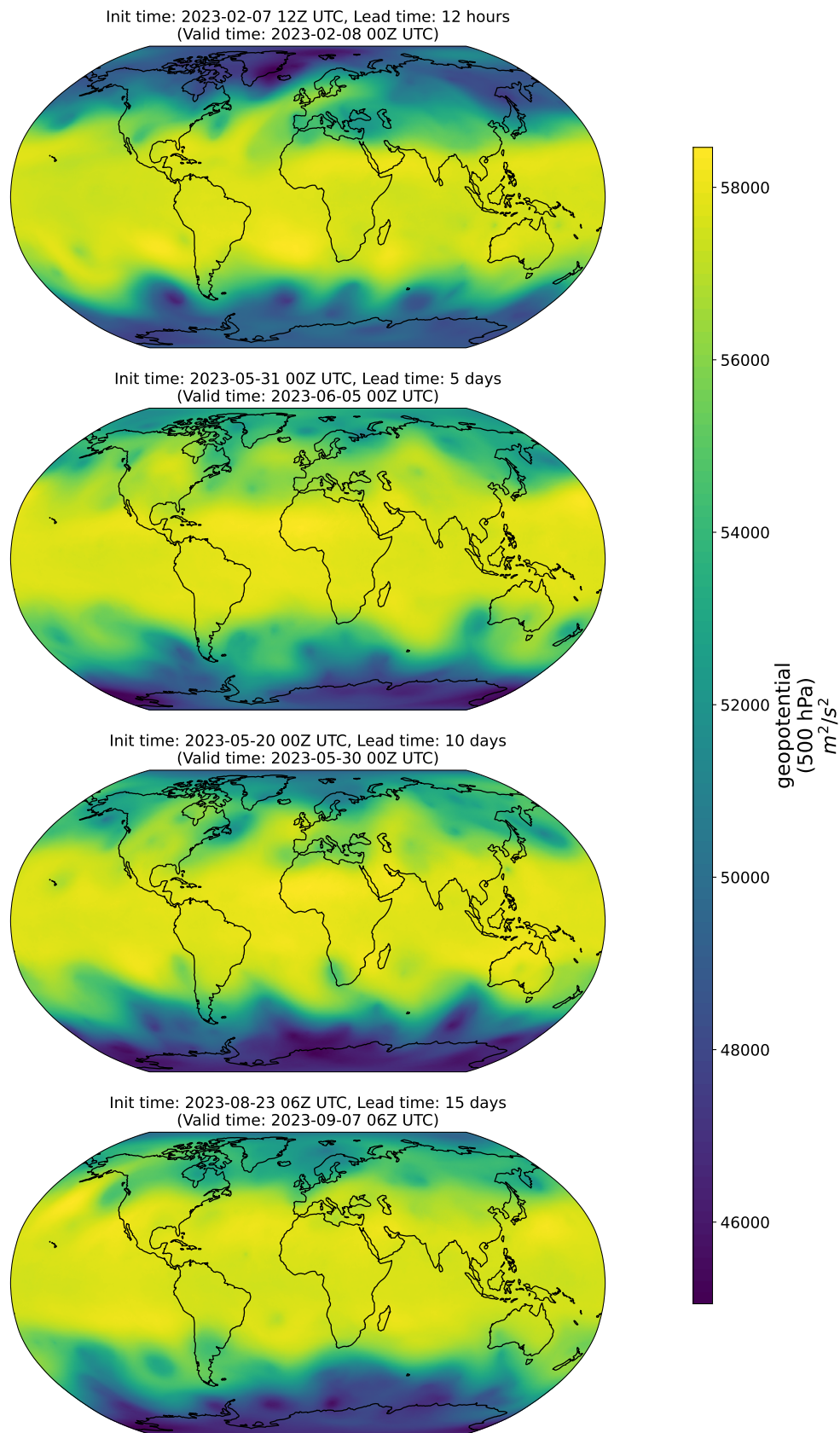Figure A.27 | Visualization of u component of wind at 850 hPa.
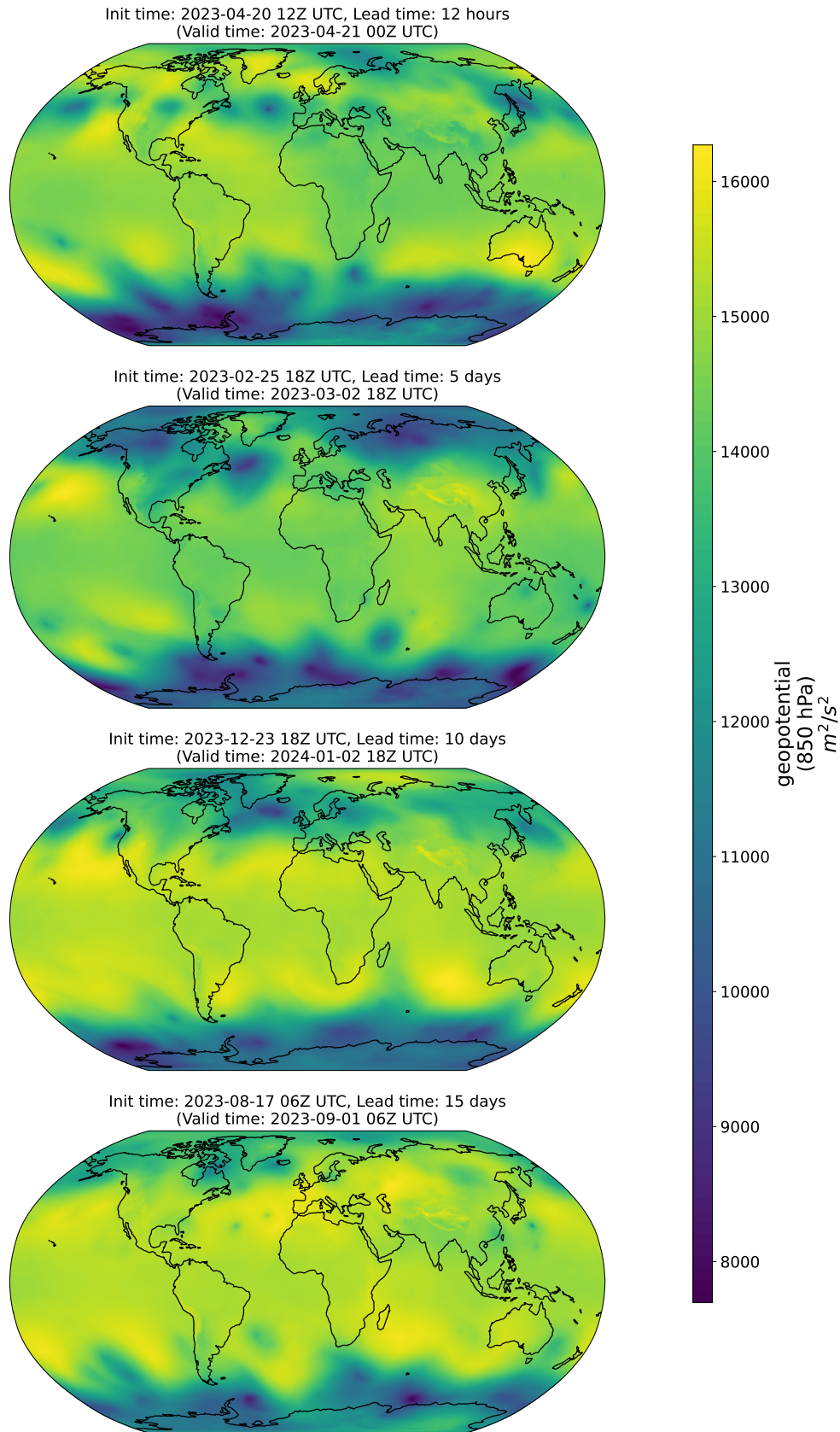
Figure A.28 | Visualization of geopotential at 500 hPa.

Figure A.29 | Visualization of geopotential at 850 hPa.