Generative diffusion for regional surrogate models from sea-ice simulations

Tobias Sebastian Finn[1], Charlotte Durand[1], Alban Farchi[1], Marc Bocquet[1], Pierre Rampal[2], and Alberto Carrassi[3]

[1]CEREA, École des Ponts and EDF R&D, Île-de-France, France
[2]IGE/CNRS, Grenoble, France
[3]Dept. of Physics and Astronomy "Augusto Righi", University of Bologna, Bologna, Italy

Key Points:

- We introduce the first denoising diffusion model designed for sea-ice physics
- Generative diffusion outperforms deterministic surrogates and retains the sharpness in the forecasts as observed in the targeted simulations
- Our model generates forecasts that exhibit physical consistency between variables in space and time

Corresponding author: Tobias Sebastian Finn, `tobias.finn@enpc.fr`

Abstract

We introduce deep generative diffusion for multivariate and regional surrogate modeling learned from sea-ice simulations. Given initial conditions and atmospheric forcings, the model is trained to generate forecasts for a 12-hour lead time from simulations by the state-of-the-art sea-ice model neXtSIM. For our regional model setup, the diffusion model outperforms as ensemble forecast all other tested models, including a free-drift model and a stochastic extension of a deterministic data-driven surrogate model. The diffusion model additionally retains information at all scales, resolving smoothing issues of deterministic models. Furthermore, by generating physical consistent forecasts, previously unseen for such kind of completely data-driven surrogates, the model can almost match the scaling properties of neXtSIM, which are also observed for real sea ice. With these results, we provide a strong indication that diffusion models can achieve similar results as traditional geophysical models with the significant advantage of being orders of magnitude faster and solely learned from data.

Plain Language Summary

Thanks to generative deep learning, computers can generate images that are almost indistinguishable from real images. We use this technology to forecast the sea-ice for a region North of Svalbard with models that are learned from data, here from simulation data. Doing so, we enhance the accuracy of the model and maintain the sharpness of the forecasts. The learned model further depicts physical processes as similarly observed for the targeted physical-driven model. Therefore, this technology could provide us with the necessary tools to learn faster models from data that have similar properties to those based on physical equations.

1 Introduction

In recent years, surrogate modeling with deep neural networks has made substantial progress in weather forecasting up to 15 days (Keisler, 2022; Pathak et al., 2022; Bi et al., 2023; Lam et al., 2023), which was seen as highly unlikely a few years ago (Dueben & Bauer, 2018; Palmer, 2022; Rasp & Thuerey, 2021). This approach of fully data-driven modeling also gain appeal for other components of the Earth system, like the ocean (W. Xiong et al., 2023; Wang et al., 2024). Usually trained as deterministic surrogates, they target the expected future conditions based on given initial conditions. However, predicting just the expectation can lead to a loss of small-scale information, which in fact is expressed as smoothing of the forecasted fields (e.g., Bonavita, 2023). While the dynamics of the system might be deterministic, the temporal development of the instantiated fields is stochastic, since the initial conditions and/or forcings are insufficient to explain the full temporal development. Such effects can be exacerbated in discrete-continuous processes as found in precipitation (Ravuri et al., 2021) or sea ice (Durand et al., 2023). In this work, we introduce the first generative multivariate surrogate for sea ice that is trained as denoising diffusion model and which can resolve aforementioned issues. This generative surrogate exceeds the performance of deterministic surrogates and allows us to generate an ensemble of plausible future trajectories.

In diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Y. Song, Sohl-Dickstein, et al., 2021), neural networks are trained to map from noise to data by iteratively denoising. Designed to reverse a diffusion process, these models learn to sample based on training data from the true but unknown data distribution. Conditioned on initial conditions and forcings, the diffusion model can generate samples from the conditional distribution of the targeted fields (Batzolis et al., 2021; Saharia et al., 2022). Such conditional diffusion models show promise for different geophysical problems, like for weather prediction (Price et al., 2023; Hua et al., 2024), downscaling and correction of meteorological fields (Mardani et al., 2023; Wan et al., 2023; Zhong et al., 2023), the gener-

ation of ensemble forecasts (T. S. Finn, Disson, et al., 2023; L. Li et al., 2023), or precipitation forecasts (Asperti et al., 2023; Gao et al., 2023; Leinonen et al., 2023).

Beside training diffusion model from scratch, they can be also trained on top of another model, which is then the prior (Lee et al., 2022). Instantiated like this, the so-called residual diffusion model (Mardani et al., 2023) acts as model error correction for the other model, similar to those instantiated for geophysical sea-ice models (T. S. Finn, Durand, et al., 2023; Gregory et al., 2023). In addition to training from scratch, we also train such a residual diffusion model on top of a deterministic surrogate. Since the residual diffusion model performs as well as the one trained from scratch, we show that we can harness diffusion models for model error corrections.

The breakthrough in surrogate modeling for weather prediction can be partially accounted (Ben-Bouallegue et al., 2023; Bocquet, 2023) to the availability of the large reanalysis dataset ERA5 from the ECMWF (Hersbach et al., 2020), which contains weather data at a $1/4°$ resolution from more than 40 years. This large dataset has unlocked the training of neural networks with tens of millions of parameters. To enable a similar effort for sea ice, we rely on more than 20 years of high-resolution free-running sea-ice simulations (Boutin et al., 2023), performed with the state-of-the-art sea-ice model neXtSIM (Rampal et al., 2016; Ólason et al., 2022) coupled to the ocean component of the NEMO modeling framework (Madec, 2008). Differing from the usual approach in weather forecasting, we target a surrogate model for the geophysical model and not a surrogate model for the dynamics as seen by a reanalysis, a subtle but important difference. We train the surrogates for a 12-hour forecast and them for up to 50 days. Since we want to prove the concept and to reduce the computational costs, we instantiate the problem as a challenging regional modeling dataset with 64×64 grid points and unknown lateral boundary conditions; the surrogates have to generate the inflow and outflow of sea ice solely based on the initial conditions and forcings.

Characterized by multifractality and scale-invariance (Marsan et al., 2004; Rampal et al., 2008; Girard et al., 2009), processes in sea ice exhibit a discrete-continuous behavior. Caused by this scale-invariance, fracturing propagates from small-scales to large-scales (Weiss & Schulson, 2009) and can suddenly show up at the resolved scales, here at around 10 km. From the point of view of the resolved scales, this behavior is seemingly stochastic and surrogate models could benefit from a probabilistic formulation (Andersson et al., 2021; Durand et al., 2023) Sea-ice models with brittle rheologies, like neXtSIM, parameterize these processes by introducing a damage variable, which keeps track of the sub-grid scale fracturing of sea ice. As we want to find the best surrogate for the geophysical model, we treat the damage as another predicted variable beside the sea-ice thickness, sea-ice concentration, and the two components of the horizontal sea-ice velocity. Thereby, we are the first providing a surrogate model for the most important sea-ice variables, altogether modeled within one single neural network.

The fracturing process links the deformation of sea ice to the temporal development of the sea-ice thickness and sea-ice concentration. A physical consistent surrogate should represent these links between deformation and other state variables. Caused by their regression-to-the-mean behavior, deterministic surrogate models fail to represent physical consistency (Bonavita, 2023; Kochkov et al., 2023). While we confirm this lack of consistency for our deterministic surrogate, we also show that our diffusion surrogate can represent these aforementioned links. We see the discovery of such capabilities for generative diffusion as important step towards physical consistent surrogates based on deep neural networks.

In Sect. 2, we introduce the dataset used in this study, we explain therein the simulations performed with the geophysical model neXtSIM and the used forcing fields from the ERA5 atmospheric reanalysis. We elaborate the goal and methodology of training our surrogate models in Sect. 3, where we state our used loss functions and parameter-

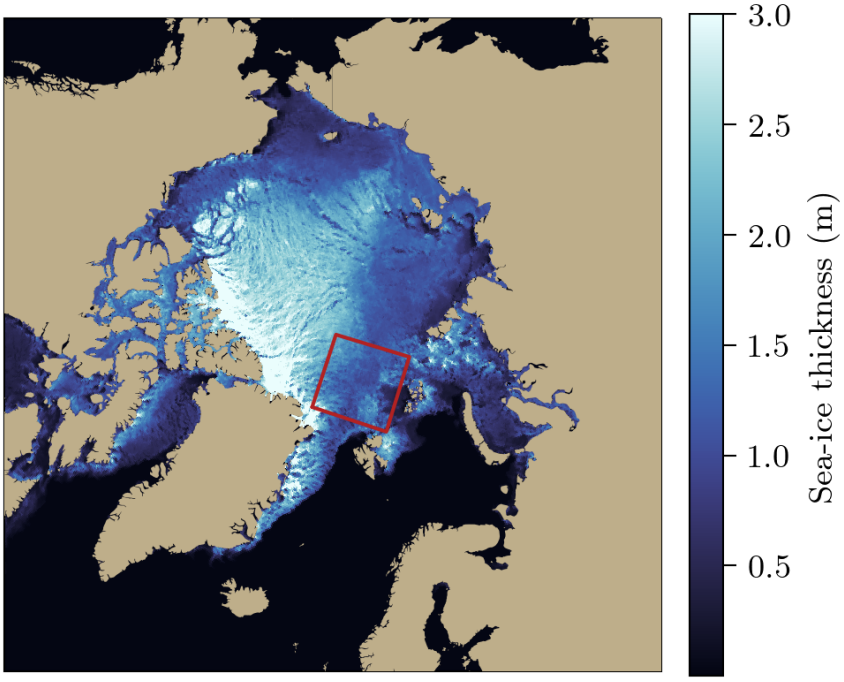

Figure 1. The sea-ice thickness as simulated by neXtSIM for 2015-01-01 03:00 UTC from the validation dataset. The red marked region north of Svalbard depicts the 64 × 64 grid points that are used for the regional setup. The land areas are based on the Natural Earth dataset.

izations to train deterministic and diffusion surrogates. In Sect. 4, we explain our experiments and indicate which hyperparameters were used during the training of the neural networks. We present our results in Sect. 5, while we discuss and summarize these results in Sect. 6. We briefly conclude this study in Sect. 7.

2 Data

The target of this study is to train regional surrogate models on sea-ice simulations from the state-of-the-art sea-ice model neXtSIM. Our data comes from simulations performed with neXtSIM coupled to an ocean model with forcings from the ERA5 atmospheric reanalysis.

Ranging from 1995 to 2018, the dataset is available in six-hourly steps. In accordance with Durand et al. (2023), we train the surrogate model for a 12-hour lead time to increase the signal-to-noise ratio in the data.

The regional dataset contains of a region north of Svalbard with 64×64 grid points, as depicted in Fig. 1. This region has no land masses and is characterized by heavy forcings from ocean currents and temporally changing sea-ice conditions. While the southern-eastern border contains examples of marginal ice zones, the northern part has an inflow of thick sea ice during the winter season. Since our goal is to evaluate the performance of surrogate models in a regionally constrained setup, we impose no lateral boundary conditions in the surrogate model: outflowing sea ice is lost and the type of inflowing sea ice is unknown. The surrogate must learn to generate the type of inflowing sea ice based on the initial conditions and the atmospheric forcings. We elaborate on the dataset of sea-ice states stemming from neXtSIM in Sect. 2.1, while we explain our strategy in using atmospheric forcings from ERA5 in Sect. 2.2.

2.1 Simulations from the sea-ice model neXtSIM

The purely Lagrangian sea-ice model neXtSIM (Rampal et al., 2016) is designed to model sea-ice over regions as large as the whole Arctic. The simulations (Boutin et al., 2023) were performed with the brittle Bingham-Maxwell rheology (Ólason et al., 2022), which builds upon the Maxwell-Elasto-Brittle rheology (Dansereau et al., 2016). The resulting model has been shown to reproduce some properties of sea-ice dynamics, for instance the observed temporal and spatial scaling of the sea-ice deformation over a wide range of scales. (Rampal et al., 2019; Ólason et al., 2022; Bouchat et al., 2022). For more information about neXtSIM, we refer to Rampal et al. (2016); Ólason et al. (2022).

In our used simulations, neXtSIM has been coupled via the OASIS3-MCT coupler (Valcke, 2013; Craig et al., 2017) to OPA, the ocean component of the NEMO modeling framework (Nucleus for European modeling of the Ocean, v3.6, Madec, 2008). In addition to the ocean coupling, the sea ice is driven from the atmosphere by forcings from the deterministic reanalysis run of the ERA5 reanalysis dataset (Hersbach et al., 2020) on an hourly basis. Run at a horizontal resolution of $1/4^\circ \approx 12\,\mathrm{km}$, the coupled model simulates processes over the full Arctic. The curvilinear mesh for the ocean component is given by the regional CREG025 configuration (Talandier & Lique, 2021), while neXtSIM uses a dynamical Lagrangian mesh with remeshing. For a more detailed introduction to the modeling setup, we refer to Boutin et al. (2023).

Our targeted prognostic model variables are the sea-ice thickness (SIT), sea-ice concentration (SIC), sea-ice damage (SID), and sea-ice velocity in x- (SIU) and y-direction (SIV). The model output is interpolated with a conservative scheme from the Lagrangian neXtSIM mesh to the aforemented fixed curvilinear mesh from the ocean model. While SID represents instantaneous values every six hours, all other model variables are averaged on a six-hourly basis. Initialized on 1995-01-01, the coupled model is run up to 2018-12-31. While the first five years are normally treated as spin-up phase (Boutin et al., 2023), we include them into our dataset to increase the data amount, since our goal is here to find the best surrogate for neXtSIM.

2.2 Forcings from the ERA5 reanalysis

The external forcings for our surrogate model are given from the deterministic reanalysis run of the ERA5 dataset (Hersbach et al., 2020), acquired from the Copernicus Climate Change Service (Hersbach et al., 2023). While the neXtSIM simulations are driven by hourly ERA5 output, we use as input into our surrogate model output every 12 hours; our surrogate has less information from the atmosphere than the targeted simulations. As additional constrain, we just use atmospheric forcings, while neglecting forcings from the ocean.

As forcing variables, we choose the 2-meter temperature (T2m), 2-meter specific humidity (Q2m), and the 10-meter wind velocities in meridional (U10m) and zonal (V10m) direction, neglecting other variables like the solar insulation which are used in neXtSIM. These four variables are usually also available on a six-hourly basis in the CMIP6 datasets (Eyring et al., 2016), such that, in the future, we could apply the surrogates to climate projections. All variables are interpolated from the $1/4^\circ$ lat-lon mesh to the curvilinear CREG025 mesh by nearest neighbor interpolation. The wind velocities are rotated from meridional and zonal direction to the native x- and y-direction of the curvilinear grid as internally done within the NEMO modeling framework. Combined with the state variables, we have a total of nine variables (five state variables plus four forcing variables) per six-hourly timestep in our dataset.

3 Surrogate modeling with diffusion models

With the current sea-ice conditions $\mathbf{x}_t$ and current and future atmospheric forcings $\mathbf{f}_{t:t+12\,\mathrm{h}}$, we want to forecast the future sea-ice conditions 12 hours later $\mathbf{x}_{t+12\,\mathrm{h}}$. Here, the sea-ice conditions contain the sea-ice thickness, concentration, damage, velocity in x-direction, and the velocity in y-direction; in total, we have 13 input fields and 5 target fields. For this task, we employ a statistical forecast model $\mathcal{M}_{\boldsymbol{\theta}}(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}})$ with its parameters $\boldsymbol{\theta}$. The forecast model outputs a forecast $\widehat{\mathbf{x}}_{\boldsymbol{\theta},t+12\,\mathrm{h}}$, which should best estimate the true future sea-ice conditions,

$$\mathbf{x}_{t+12\,\mathrm{h}} \approx \widehat{\mathbf{x}}_{t+12\,\mathrm{h}} = \mathbf{x}_t + \mathcal{M}_{\boldsymbol{\theta}}(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}). \tag{1}$$

To get the forecast, the output of the neural network is added to the persistence forecast, as the dynamics are additive and this tends to improve the forecasting results (e.g., Durand et al., 2023; Lam et al., 2023).

We employ as statistical model a deep neural network which predicts all five model variables at the same time. The model parameters $\boldsymbol{\theta}$ are the weights and biases of this deep neural network. We train the neural network by minimizing a loss function with a variant of stochastic gradient descent based on a mini-batch of data samples drawn from the training dataset $(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}, \mathbf{x}_{t+12\,\mathrm{h}}) \sim \mathcal{D}$.

After its training, we can cycle the surrogate model for longer lead times than the trained 12 hours. To do so, the forecasts of the model are clipped to their physical bounds (SIT: $[0, \infty)$, SIC: $[0, 1]$, SID: $[0, 1]$, SIU: $(-\infty, \infty)$, SIV: $(-\infty, \infty)$) and used as initial conditions for the following cycle, e.g., $\widehat{\mathbf{x}}_{t+24\,\mathrm{h}} = \widehat{\mathbf{x}}_{t+12\,\mathrm{h}} + \mathcal{M}_{\boldsymbol{\theta}}(\widehat{\mathbf{x}}_{t+12\,\mathrm{h}}, \mathbf{f}_{t+12\,\mathrm{h}:t+24\,\mathrm{h}})$.

We apply surrogates in four different flavors: first, we train a deterministic surrogate predicting the expected future conditions, as explained in Sect. 3.1. Secondly, we extend the determinstic surrogate to stochastic forecasts by introducing a stochastic term, which is fitted to the validation dataset, as elucidated in Sect. 3.2. Thirdly, we use generative diffusion models as stochastic surrogates to sample from the probability distribution of the future conditions, as introduced in Sect. 3.3. Fourthly, we correct the forecasts of the deterministic surrogate with residual diffusion models, as presented in Sect. 3.4.

3.1 Deterministic surrogate modeling

The deterministic surrogate takes as input the current sea-ice conditions $\mathbf{x}_t$ and forcings $\mathbf{f}_{t:t+12\,\mathrm{h}}$ and is trained to give one single forecast of the future sea-ice conditions. As usual approach to train such deterministic models, we take the mean-squared error (MSE) between the forecast and the true sea-ice conditions after 12 hours as loss function. Since the five predicted variables have different physical meaning, we have to weight the contribution of these variables to the loss, which results into a weighted MSE. The deterministic surrogate is optimized over the K variables with

$$\mathcal{L}_{\mathrm{det}}(\theta) = \sum_{k=1}^{K} w_k \left\| \mathbf{x}_{t+12\,\mathrm{h},k} - \mathbf{x}_{t,k} - \mathcal{M}_{\boldsymbol{\theta},k}(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}) \right\|_2^2, \tag{2}$$

where w_k is the weighting factor for the k-th variable. The weighting factor is kept constant throughout the optimization and set $w_k = \frac{1}{s_k^2}$. s_k^2 is the variance of the dynamics, $\Delta\mathbf{x}_{t+12\,\mathrm{h},k} = \mathbf{x}_{t+12\,\mathrm{h},k} - \mathbf{x}_{t,k}$, estimated over N_{samples} samples and N_{grid} grid points in the training dataset,

$$s_k^2 = \frac{1}{N_{\mathrm{samples}} \cdot N_{\mathrm{grid}} - 1} \sum_{i=1}^{N_{\mathrm{samples}}} \sum_{j=1}^{N_{\mathrm{grid}}} (\Delta x_{t+12\,\mathrm{h},i,j,k} - \overline{\Delta x_{t+12\,\mathrm{h},k}})^2, \tag{3}$$

where $\overline{\Delta x_{t+12\,\mathrm{h},k}}$ corresponds to the mean dynamics for the k-th variable.

As shown in Appendix A1, we can recover Eq. (2) using maximum likelihood estimation and a local Gaussian distribution with the forecast as its mean and a diagonal covariance matrix with s_k^2 on its diagonal. By optimizing Eq. (2), the target of the deterministic surrogate is to predict the expected sea-ice conditions after 12 hours given the initial conditions and forcings, $\widehat{\mathbf{x}}_{t+12\,\mathrm{h}} = \mathbb{E}(\mathbf{x}_{t+12\,\mathrm{h}} \mid \mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}})$.

3.2 Stochastic surrogate modeling

While the deterministic surrogate is trained to imitate an ensemble mean for a 12-hour forecast, cycling such a deterministic surrogate differs from an ensemble mean and can lead to unphysical behavior in the forecasts and to smoothing effects (Bonavita, 2023; Kochkov et al., 2023; Durand et al., 2023). Additionally, although trained by a deterministic loss function, the surrogate model is thought to have stochastic dynamics rather than deterministic ones (Bocquet et al., 2020), based on the underlying Gaussian assumptions of Eq. (2).

Instead of using the deterministic surrogate as single forecast, we can also sample from an assumed Gaussian distribution, here for the i-th ensemble member,

$$\widehat{\mathbf{x}}^{(i)}_{t+12\,\mathrm{h}} = \mathbf{x}_t + \mathcal{M}_{\boldsymbol{\theta}}(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}) + \mathbf{L}\boldsymbol{\epsilon}^{(i)}, \qquad \boldsymbol{\epsilon}^{(i)} \sim \mathcal{N}(0, \mathbf{I}), \tag{4}$$

where $\mathbf{L}$ is matrix factor of the covariance matrix $\mathbf{Q}$, i.e. $\mathbf{Q} = \mathbf{L}\mathbf{L}^\top$, such as the Cholesky decomposition of $\mathbf{Q}$. Comparing Eq. (1) with Eq. (4), we see that we get an additional stochastic term, which should represent the predictive uncertainty.

To apply Eq. (4) for forecasts, we have to find the covariance matrix $\mathbf{Q}$. In this study, we decompose the covariance matrix into a cross-covariance between variables and spatial correlations within a single variable. The spatial correlations are efficiently modeled by using a two-dimensional FFT-based approach, as shown in Appendix A2. To avoid issues with overfitting, we fit the cross-covariance and the spectrum for the spatial correlations on the validation dataset as a post-processing step, after training the deterministic surrogate. This surrogate serves as baseline approach for a stochastic model. Derived from a Gaussian assumption of the forecast distribution, its forecasts are always constrained to this assumption.

3.3 Diffusion models

Besides training neural networks as deterministic forecasts, we also train generative diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Y. Song, Sohl-Dickstein, et al., 2021) for stochastic forecasts to generate samples from the full probability distribution (Mohamed & Lakshminarayanan, 2016; J. Song et al., 2020) without making a Gaussian assumption. The idea behind such diffusion models is to iteratively denoise fields towards forecast samples by starting with fields of pure noise.

Diffusion models work with $\mathbf{z}_\tau$, a noised version of our targeted fields $\mathbf{x}_{t+12\,\mathrm{h}}$, where τ is a pseudo time going from $\tau = 1$ for pure noise to $\tau = 0$ for cleaned data samples. We parameterize the output of the neural network as

$$\widehat{\mathbf{v}}_{\boldsymbol{\phi}}(\mathbf{z}_\tau, \mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}, \tau), \tag{5}$$

with the neural network parameters $\boldsymbol{\phi}$. The output of the neural network $\widehat{\mathbf{v}}_{\boldsymbol{\phi}}(\cdot)$ corresponds to a surrogate target, internally used within the diffusion model to iteratively denoise the fields (Salimans & Ho, 2022).

During training, we sample data pairs from our training dataset, which then also include samples of our targeted fields. Assuming that these samples are normalized to have mean 0 and standard deviation 1, we increasingly replace the signal in the samples

by Gaussian noise, defining a *variance-preserving* diffusion process,

$$\mathbf{z}_\tau = \alpha_\tau \mathbf{x}_{t+12\,\mathrm{h}} + \sigma_\tau \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{6}$$

where $\mathbf{z}_\tau$ is the noised data sample at pseudo time $\tau \in [0, 1]$ with the signal amplitude α_τ and the noise amplitude σ_τ. We define the signal and noise amplitude in terms of logarithmic signal-to-noise ratio

$$\lambda(\tau) = \log\left(\frac{\alpha_\tau^2}{\sigma_\tau^2}\right), \tag{7}$$

which monotonically decreases with increasing pseudo time. During training, we use a dynamic noise scheduling (D. P. Kingma & Gao, 2023), which is adapted to the approximation error of the neural network and further explained in A4. On the one end, by setting $\lambda(0)$ large enough, we achieve $\alpha_0 \approx 1$ and approximately recover $\mathbf{x}_{t+12\,\mathrm{h}}$ from $\mathbf{z}_0$. On the other end, by setting $\lambda(1)$ small enough, the signal amplitude goes towards zero, $\alpha_1 \approx 0$, and $p(\mathbf{z}_1) \approx \mathcal{N}(\mathbf{0}, \mathbf{I})$ approximately holds (D. Kingma et al., 2021).

To train the diffusion model, we use

$$\mathbf{v}_\tau := \alpha_\tau \boldsymbol{\epsilon} - \sigma_\tau \mathbf{x}_{t+12\,\mathrm{h}} \tag{8}$$

as surrogate target, which has been shown to be more stable during training and sampling for small signal amplitudes (Salimans & Ho, 2022). We optimize our neural network approximation from Eq. (5) by sampling a pseudo time step from a uniform distribution $U(0, 1)$ and minimizing

$$\mathcal{L}_{\mathrm{Diff}}(\boldsymbol{\phi}) = \mathbb{E}_{\tau \sim U(0,1)}\left[w(\tau) \cdot \left(-\frac{d\lambda(\tau)}{d\tau}\right) \cdot (e^{-\lambda(\tau)} + 1)^{-1} \left\| \mathbf{v}_\tau - \widehat{\mathbf{v}}_{\boldsymbol{\phi}}(\mathbf{z}_\tau, \mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}, \tau) \right\|_2^2\right], \tag{9}$$

as loss function with $w(\tau)$ as weighting factor. The multiplicative factor $-\frac{d\lambda_\tau}{d\tau} \cdot (e^{-\lambda_\tau} + 1)^{-1}$ ensures that the loss function optimizes a lower bound on the likelihood of $\mathbf{x}_{t+12\,\mathrm{h}}$ (ELBO, D. Kingma et al., 2021; Y. Song, Durkan, et al., 2021). Although the target $\mathbf{v}_\tau$ is independent from the conditioning information, Eq. (9) optimizes the ELBO of the conditional distribution $p(\mathbf{x}_{t+12\,\mathrm{h}} \mid \mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}})$, as we condiiton the neural network (Batzolis et al., 2021; Saharia et al., 2022).

If the weighting function $w(\tau)$ monotonically increases with increasing pseudo time, the loss function corresponds to the ELBO with additive data augmentation (D. P. Kingma & Gao, 2023), which has been shown to lead to better results (e.g., Karras et al., 2022). As proposed in Salimans and Ho (2022), we use an exponential weighting function

$$w(\tau) = \exp\left(-\frac{\lambda(\tau)}{2}\right), \tag{10}$$

which is monotonically increasing, since $\lambda(\tau)$ decreases with increasing pseudo time.

We generate data samples by drawing fields of random noise $\mathbf{z}_1 = \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and integrating the ordinary differential equation (ODE) that corresponds to the denoising problem (see also A3, Y. Song, Sohl-Dickstein, et al., 2021) with a deterministic second-order Heun integrator (Karras et al., 2022). Within the integration, we make use of the trained neural network by defining the following denoiser function

$$\widehat{D}_\phi(\mathbf{z}_\tau, \mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}) = \alpha_\tau \mathbf{z}_\tau - \sigma_\tau \widehat{\mathbf{v}}_{\boldsymbol{\phi}}(\mathbf{z}_\tau, \mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}, \tau), \tag{11}$$

where $\mathbf{z}_\tau$ corresponds to the states backward integrated from 1 to τ. The denoiser approximates the cleaned states based on all information up to time τ. This approximation is then used within the integration scheme to denoise $\mathbf{z}_\tau$ one integration step further. In the following, we denote $D_\phi(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}, \epsilon)$ as the final integrated solution of the ODE.

The pseudo time steps used for the integration from $\tau = 1$ to $\tau = 0$ are defined by an additional noise scheduling, which can be independent from the one used during training. To reduce the truncation errors, we choose the sampling scheduling as proposed by Karras et al. (2022) and modified by D. P. Kingma and Gao (2023) for wider ranges of λ, also shown in Fig. A2.

By drawing different initial conditions for the ODE, we get different forecasts from the diffusion model. Hence, the forecasts with the diffusion surrogate are inherently stochastic and allow us to create an ensemble of forecasts. In practice, as proposed in Eq. (1), we train the diffusion model to predict the dynamics instead of the states directly. Then, the forecast of the diffusion surrogate for the i-th ensemble member can be described as

$$\widehat{\mathbf{x}}^{(i)}_{t+12\,\mathrm{h}} = \mathbf{x}_t + D_\phi(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}, \boldsymbol{\epsilon}^{(i)}), \qquad \boldsymbol{\epsilon}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{12}$$

3.4 Residual diffusion models

When we directly predict the dynamics for 12 hours with a diffusion model, it must do all the heavy lifting. However, we can also split the dynamics into two different parts: one deterministic and one stochastic part, similarly to what we have done in Sect. 3.2. We leverage this splitting and fit residual diffusion models (Mardani et al., 2023), where the deterministic surrogate serves as prior (Lee et al., 2022).

During training of the residual diffusion model, we replace the target $\mathbf{x}_{t+12\,\mathrm{h}}$ by the residuals of the deterministic surrogate $\mathbf{x}_{t+12\,\mathrm{h}} - \mathcal{M}_\theta(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}})$. We additionally condition the diffusion model on the output of the deterministic surrogate, since it is available before the diffusion model is applied. Beside these changes, we train the diffusion model with the same loss function and weighting as in Eq. (9). The forecast of the residual diffusion surrogate for the i-th ensemble member reads then

$$\widehat{\mathbf{x}}^{(i)}_{t+12\,\mathrm{h}} = \mathbf{x}_t + \mathcal{M}_\theta(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}) + D_\phi(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}, \mathcal{M}_\theta(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}), \boldsymbol{\epsilon}^{(i)}), \tag{13}$$

again with $\boldsymbol{\epsilon}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $D_\phi(\cdot)$ as integrated solution of the diffusion model.

The forecast of the deterministic surrogate is the prior and refined by the diffusion model. As the diffusion model is trained on the residuals of the deterministic surrogate, it can be seen as model error correction. This splitting of the surrogate model into one deterministic and one stochastic part speeds up the convergence of the diffusion model, as illustrated in Fig. 2.

4 Experiments

We perform our experiments with the data as described in Sect. 2 and train neural networks for surrogate modeling as presented in Sect. 3. In these experiments, we want to compare determinstic surrogates to stochastic surrogates, either applied on top of the deterministic ones or trained independently. To make the experiments comparable, we used almost the same neural network architecture and hyperparameters for training of the neural networks.

Our neural network architecture is inspired by the UViT architecture of Hoogeboom et al. (2023), which builds upon the vision transformer (ViT) architecture (Dosovitskiy et al., 2021) for diffusion models (Peebles & Xie, 2023). In the encoding and decoding part of our architecture with a U-form and skip-connection (Ronneberger et al., 2015), we use ConvNeXt blocks (Z. Liu et al., 2022) and two additional types of layers to decrease and increase the spatial dimensionality of our data: we decrease the spatial dimensions by convolution layers with a kernel size of 2 and a stride of 2. To increase the spatial dimensions, the features are interpolated with a nearest neighbor interpolation followed by a convolutional layer with a kernel size of 3. The bottleneck at the bottom

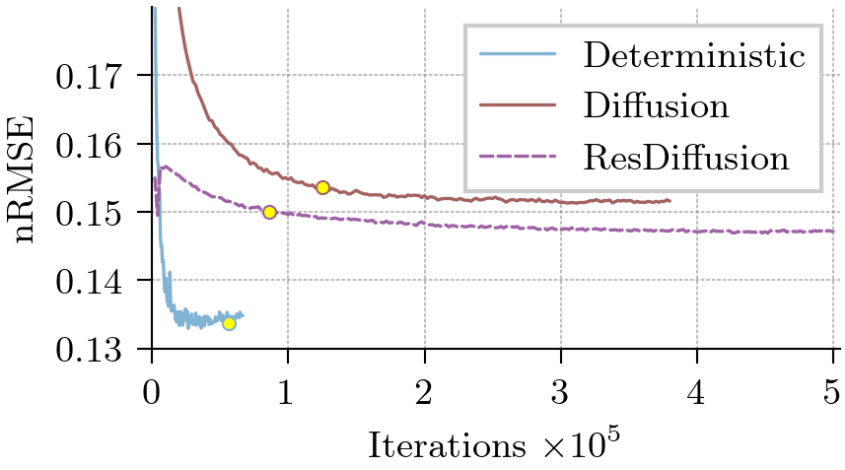

Figure 2. Normalized root-mean-squared error (nRMSE) for a lead time of 12 hours in the validation dataset as function of training iterations for the deterministic model (blue), the diffusion model (red), and the residual diffusion model (dashed, violet). The nRMSE for the diffusion and residual diffusion model are for a single ensemble member. The yellow dots correspond to the model selected by the best validation loss, which is different from the nRMSE for the diffusion and residual diffusion model. Note, the diffusion models are trained with a 2.5× lower learning rate than the deterministic model.

of the UViT architecture consists of transformer blocks (Vaswani et al., 2017), where self-attention layers (Bahdanau et al., 2016) are followed by feed-forward layers to extract global features and mix these features up. The architecture is further explained in A6.

As architectural scaling parameter, we use the number of transformer blocks in the bottleneck layer as similarly done in Hoogeboom et al. (2023). To reduce overfitting, we apply dropout ($p = 0.2$) in these transformer blocks. In addition to dropout, we use data augmentation to artificially increase the training dataset size. As data augmentation, we use random horizontal flip (probability $p = 0.5$), random vertical flips ($p = 0.5$), and random rotations counter-clockwise by 90° ($p = 0.5$). The information about the activated augmentation is given as additional conditioning input to the neural network and linearly embedded. In Appendix B3, we show that this data augmentation improves our results, something also observed for probabilistic and generative models in general (Jun et al., 2020; Karras et al., 2022; Podell et al., 2023). During forecast, we deactivate all data augmentation and give an empty conditioning by zeros to the neural network.

For the diffusion model, we additionally condition the neural network on the pseudo time in terms of $\lambda(\tau)$ and use a fixed sinusoidal embedding (Vaswani et al., 2017). Within the neural network, all embedded information is added together and then transformed into the scale and shift parameters of the normalization layers.

The deterministic model easily overfits on the training dataset, and we found the optimum of 2 transformer blocks. Contrastingly, our diffusion models suffer less from overfitting since they are trained with additive noise. We use 8 transformer blocks for the diffusion model, and yet the model has less overfitting than the deterministic one for the RMSE, as can be seen in Fig. 2. In total, the deterministic model has 7.6×10^6 parameters, while the diffusion models have 19.4×10^6 parameters.

The neural networks are trained on neXtSIM data from 1995 to 2014. The full year 2015 is used as validation dataset and the architectures and hyperparameters are tuned on this dataset. The results in Sect. 5 are calculated on data from 2016 to 2018. All inputs for the neural network are normalized based on the global per-variable mean and standard deviation in the training dataset, while the targets are normalized with the global per-variable mean and standard deviation of the dynamics.

As optimizer, we use AdamW (Loshchilov & Hutter, 2019), which decouples the optimizer Adam (D. P. Kingma & Ba, 2017) from weight decay, which we set as a constant to $\lambda = 0.01$. The learning rate is linearly increased to $\gamma = 5 \times 10^4$ (deterministic) or $\gamma = 2{\times}10^4$ (diffusion) within the first 5000 iterations, and afterwards decreased with a cosine scheduling up to the maximum number of iterations. We optimize all neural networks with a batch size of 256 for a maximum of 1×10^5 iterations (deterministic) or $5{\times}10^5$ iterations (diffusion) with early stopping if the validation loss was not improving. To note, one epoch contains 115 iterations at this batch size. After stopping the training, the best performing model in terms of validation loss is selected, as marked in Fig. 2 by yellow dots.

As training devices, we use an Nvidia RTX A5000 with 24 GB memory and an Nvidia RTX A6000 with 48 GB memory. The models are implemented in Python (Van Rossum, 1995) with PyTorch (Paszke et al., 2019), PyTorch lightning (Falcon et al., 2020), and Hydra (Yadan, 2019). The code for a PyTorch toolbox to instantiate diffusion models is available under `https://github.com/cerea-daml/ddm-dynamical`, while the code for the experiments can be found under `https://github.com/cerea-daml/diffusion-nextsim-regional`. All models are trained in *bfloat16* and evaluated in *float32*.

In total, we compare our four different surrogates with two baseline methods. As first baseline, The persistence forecast constantly predicts the initial conditions, $\widehat{\mathbf{x}}_{t+\Delta t} = \mathbf{x}_t, \forall \Delta t \in [0, \infty)$. In the free-drift model, our second baseline, we calculate the sea-ice velocity based on the atmospheric wind velocity (Thorndike & Colony, 1982; Brunette et al., 2022), which is given in the atmospheric forcings. Using the so-calculated sea-ice velocity, we advect the tracer variables SIT, SIC, and SID with a semi-Lagrangian advection scheme and a linear interpolation, as explained in A5. Per surrogate modeling strategies explained in Sect. 3, we present the results of a single surrogate model.

All models have been tuned for a 12-hour lead time in the validation dataset. For forecasting, the weights in the network of the diffusion models are replaced by their exponential moving average (rate $\gamma = 0.999$) as this can further stabilize diffusion models (Y. Song & Ermon, 2020b). The forecasts of the diffusion models are sampled in 20 integration steps with a second-order Heun integrator and the sampling noise scheduler from Karras et al. (2022), where the limits are set to $\lambda_{\min} = -10$ and $\lambda_{\max} = 15$ by truncation (D. P. Kingma & Gao, 2023). Because of these 20 integration step, the neural network is evaluated 39 times per forecasting step in our diffusion surrogates.

5 Results

In the following, we analyze the results of the diffusion surrogates compared to the deterministic surrogate and its stochastic extension. We start by evaluating the ensemble mean forecasts in terms of their root-mean-square errors (RMSE). Later, we will examine the results for the deterministic and residual diffusion surrogate more in detail. We present additional results, like the evaluation of the ensemble, in Appendix B.

The deterministic surrogate outperforms the persistence forecast and the free-drift model for all model variables, Table 1 and Table 2, showing the efficiency of deep learning for surrogate modeling of sea ice. With only one ensemble member, stochastic surrogates are in general inferior to deterministic surrogates, even for diffusion, and in several cases, they also have an increased error compared to the baseline methods. These

Table 1. Normalized root-mean-squared error (nRMSE) of the ensemble means for the sea-ice thickness (SIT), sea-ice concentration (SIC), sea-ice damage (SID), sea-ice velocity in x-direction (SIU), and sea-ice velocity in y-direction (SIV) after a lead time of 12 hours, averaged across the testing dataset. N is the number of ensemble members and $\overline{\Sigma}$ the average across all five variables. The rows above the line are the nRMSE for the baseline models and below the line for the deep learning surrogates. All scores are normalized by the climatology from the training dataset. The best performing models in a column are marked by bold values.

Experiment	N	SIT	SIC	SID	SIU	SIV	$\overline{\Sigma}$
Persistence	1	0.15	0.19	0.30	0.73	0.69	0.48
Free-drift	1	0.11	0.15	0.21	0.57	0.62	0.40
Deterministic	**1**	**0.07**	**0.09**	**0.15**	**0.18**	**0.18**	**0.14**
Stochastic	1	0.10	0.12	0.19	0.26	0.26	0.20
Diffusion	1	0.09	0.11	0.20	0.20	0.19	0.17
ResDiffusion	1	0.09	0.11	0.20	0.20	0.19	0.17
Stochastic	**16**	**0.07**	**0.09**	**0.15**	**0.19**	**0.18**	**0.15**
Diffusion	**16**	**0.07**	**0.09**	**0.16**	**0.18**	**0.17**	**0.14**
ResDiffusion	**16**	**0.07**	**0.09**	**0.15**	**0.18**	**0.17**	**0.14**

Table 2. NRMSEs after a lead time of 15 days (30 iterations). The columns and rows have the same meaning as Tab. 1.

Experiment	N	SIT	SIC	SID	SIU	SIV	$\overline{\Sigma}$
Persistence	1	0.59	0.89	1.10	1.41	1.45	1.14
Free-drift	1	0.49	0.77	0.86	0.57	0.62	0.68
Deterministic	1	0.41	0.53	0.79	0.41	0.39	0.53
Stochastic	1	0.51	0.63	0.90	0.52	0.51	0.63
Diffusion	1	0.43	0.55	0.81	0.39	0.39	0.54
ResDiffusion	1	0.44	0.56	0.82	0.40	0.38	0.55
Stochastic	16	0.39	0.55	0.74	0.42	0.41	0.52
Diffusion	**16**	**0.37**	**0.49**	**0.70**	**0.36**	**0.36**	**0.47**
ResDiffusion	**16**	**0.37**	**0.48**	**0.69**	**0.36**	**0.35**	**0.47**

stochastic surrogates add noise to the forecast which hurts their performance. With 16 ensemble members, the stochastic surrogates perform similar to the deterministic surrogate after a 12-hour lead time, since the deterministic surrogate targets a mean forecast for this lead time. However, for longer lead times, diffusion with 16 ensemble members outperforms the deterministic surrogate. The trajectory of the deterministic surrogate differs from the ensemble mean of the diffusion runs, see also Fig. 3.

Even though tuned on the validation dataset, the stochastic surrogate only gains performance on longer lead times compared to the deterministic surrogate, as can be seen in Fig. 3. However, residual diffusion outperforms the deterministic model for all variables and lead times, performing similar to diffusion trained from scratch. Residual diffusion seems efficient to correct forecast errors of other models.

Examining the resulting power spectrum in Fig. 4, the deterministic surrogate loses small-scale information, especially for the discrete-continuous sea-ice thickness and damage. Caused by a double penalty effect of the weighted MSE, this loss of information comes

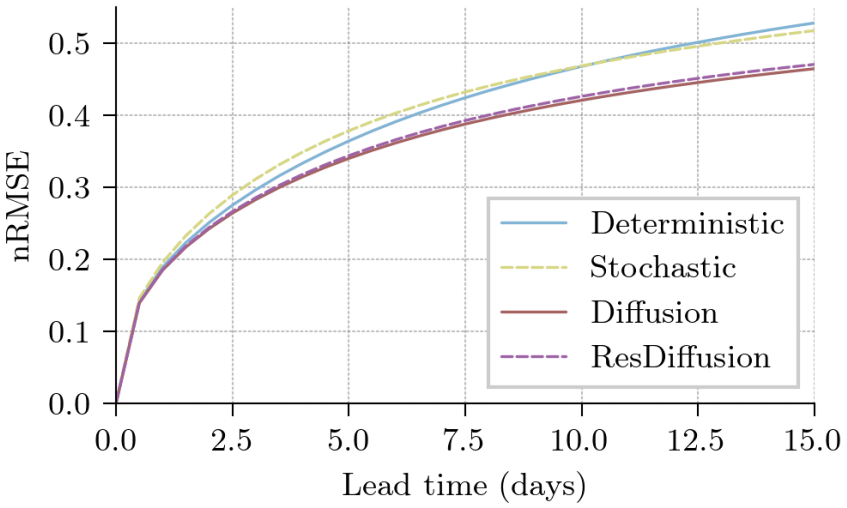

Figure 3. The normalized root-mean-squared error for the deterministic surrogate (blue), the ensemble mean of the stochastic (dashed, yellow), the diffusion (red), and the residual diffusion surrogate (dashed, violet), averaged across all five variables and the testing dataset.

inherently with the optimization of the surrogate. Contrastingly, the residual diffusion surrogate is optimized to generate forecast samples without the Gaussian assumption in data space. Therefore, generative diffusion retains information across all spatial scales, resolving the issues of the deterministic surrogate.

Until now, we have quantitatively analyzed the results averaged across the whole testing dataset. We move on and show results for forecasts started on the 2017-11-10 at 03:00 UTC. With the deterministic and the residual diffusion surrogate, we make a 50-day forecast to showcase their physical consistency and possible problems in the forecasts.

The loss of small-scale information leads to a smoothing of the deterministic forecasts which becomes especially visible for a lead time of 50 days as seen in Fig. 5. The surrogate additionally tends to generate recurring patterns of artificially large strains. Driven by the external forcings and using the deterministic surrogate as base model, the residual diffusion forecast has a similar general structure as the deterministic one, while the strains appear much more realistic. Since small-scale information is retained, generative diffusion keeps the forecasts as sharp as seen for the targeted neXtSIM simulations.

In Fig. 6, we present snapshots of divergence and shear rate, which are estimated based on the gradients in the velocity fields and related to the external stress imposed on the sea ice. Sea ice can be especially deformed where the sea-ice is weaker and its concentration lower. There, convergence leads to ridging and divergence to further thinning of sea ice.

The deterministic surrogate is unable to represent the mechanics as observed in the targeted simulation, caused by its loss of small-scale information. The gradients of the velocity and the divergence and shear are smoothed out, leading to fewer pixels with weak and strong deformation. This results into missing grid points with strong shear, divergence, or convergence, as additionally shown in Fig. 7. The connection between strains and weaker sea ice is much more blurry, weakening the link between shear and concentration. While a relation between divergence and change in the sea-ice thickness still ex-

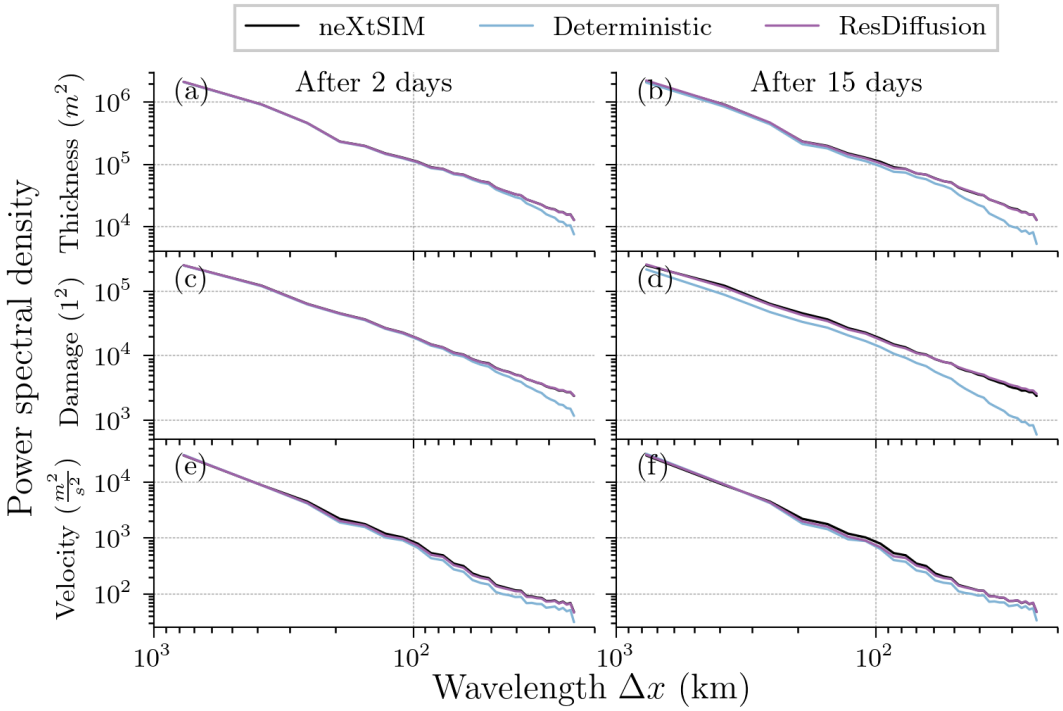

Figure 4. The spectral density of the deterministic and the residual diffusion surrogate for sea-ice thickness (a & b), sea-ice damage (c & d), and sea-ice velocity in x-direction (e & f) after a two days lead time (a, c & e) or a 15 days lead time (b, d & f). The spectra are estimated over the full three-year-long testing dataset.

ists, the thickness change exhibits much longer correlations and artificial ridging and thinning, amplifying the artificial strains. The deterministic surrogate consequently loses its physical consistency to the processes within sea ice.

The diffusion forecast clearly exposes the link between the divergence, shear, and concentration. Compared to the deterministic forecast, the thickness change resembles much more the targeted simulation, with similar correlation lengths. However, the diffusion surrogate results into noisier deformation fields, leading to fewer pixels with low shear, divergence, and convergence than in neXtSIM, see Fig. 7. This issue appears similar to the brightness issues discovered in diffusion models for image and video generation (Everaert et al., 2024; M. Li et al., 2023; Lin et al., 2024; Wu et al., 2023). Nevertheless, the diffusion surrogate can match the probability of strong deformations in neXtSIM.

In Fig. 8, we assess the dependence of the first three moments in the distribution of the total deformation rate on the spatial scale. The total deformation rate is estimated the square-root of the sum of the squared divergence and shear fields. Since the sea-ice velocities in our dataset are six-hourly averaged values, the derived total deformation fields correspond to the total deformation rates within these six hours. The estimated rates have been scaled to daily rates. As we only perform a spatial analysis, we stick to the Eulerian point of view in estimating the deformation (Herman & Glowacki, 2012), differing from the usual analysis of pseudo trajectories (Rampal et al., 2019; Ólason et al., 2022). For the spatial scaling, we coarse-grain the fields by averaging the total deformation rate within an increasing spatial window size. Expecting a power-law scaling of the distributional moments $\langle \dot{\epsilon}^q_{\mathsf{tot}} \rangle \sim L^{-\beta(q)}$, we estimate the scaling exponents $\beta(q)$ with a least-squares regression in log-log space. If the predicted fields are multi-fractal,

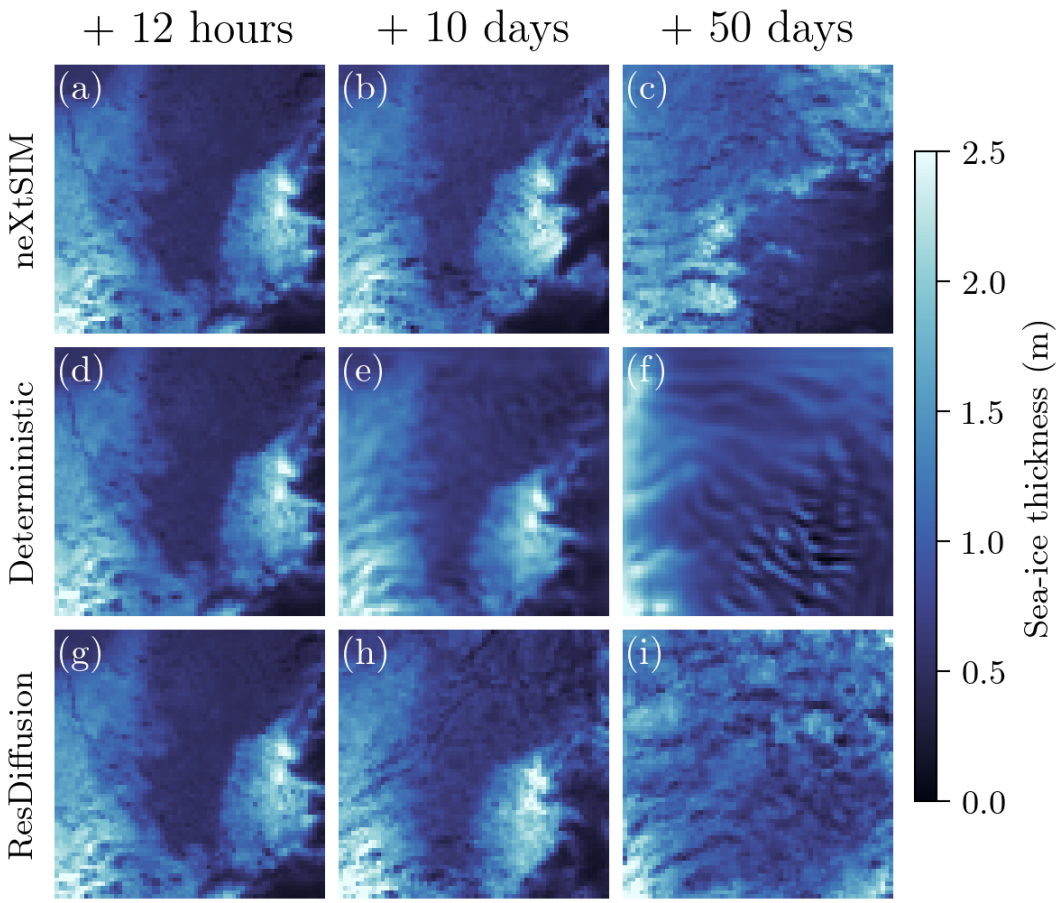

Figure 5. Snapshots of the sea-ice thickness for a forecast started on 2017-11-10 at 03:00 UTC for our target simulation from neXtSIM (a–c), the forecast with the deterministic surrogate (d–f), and the forecast with the residual diffusion surrogate (g–i) with lead times of 12 hours (a, d, & g), 10 days (b, e, & h), and 50 days (c, f, & i).

the exponents should increase with increasing moment (Marsan et al., 2004; Rampal et al., 2008), resulting into a quadratic dependency of the scaling exponents on the moments, also called structure function. We can additionally estimate an uncertainty in the upper bound estimates for the scaling exponent based on the difference between pairs of spatial scales (Rampal et al., 2019).

Simulations with neXtSIM and its brittle rheology can reproduce the scaling laws as observed by satellites (from e.g., Synthetic Aperture Radar images, Rampal et al., 2019; Ólason et al., 2022). Compared to these simulations, the deterministic surrogate shows a much weaker scaling, leading to a flatter structure function, more similar to the one obtained when sea ice is simulated with a standard viscous-plastic rheology (cf., Ólason et al., 2022, Fig. 7).

The noisier deformation fields from residual diffusion result into larger values for the moments than observed in neXtSIM. However, the derived spatial scaling laws are similar to neXtSIM's and quite remarkable in their scaling exponents and the derived structure function. Therefore, generative diffusion shows the ability to forecast spatially multi-fractal processes in the total deformation rate of sea-ice, a diagnostic variable derived from the sea-ice velocity.

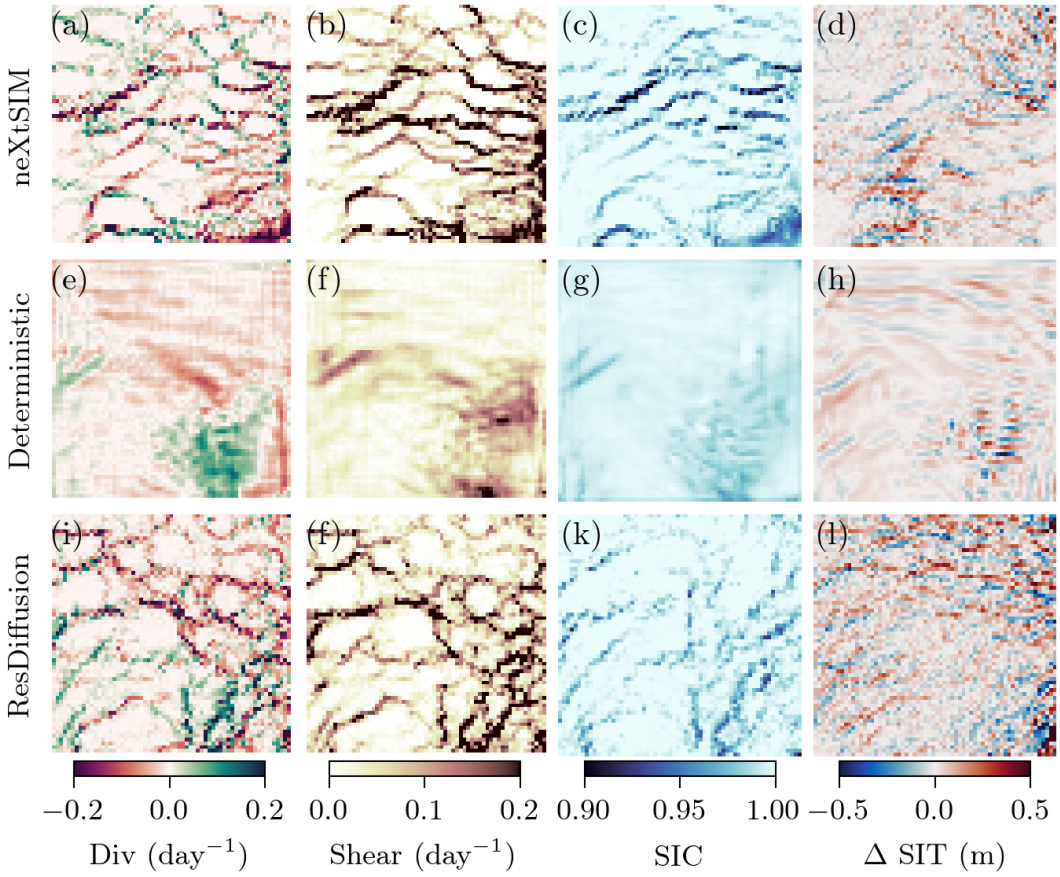

Figure 6. Snapshots of the divergence (a, e, and i), shear rate (b, f, and j), sea-ice concentration (c, g, and k), and change in the sea-ice thickness within 12 hours (d, h, and l) for the neXtSIM simulations (a–d), the deterministic forecast (e–h), and an ensemble member from the the diffusion forecast (i–l), the forecast is valid for 2017-12-30 at 03:00 UTC, a lead time of 50 days.

Events with linear kinematic features can be characterized by a few grid points with strong shear (Ólason et al., 2022). In Fig. 9, we analyze the tail of the shear distribution by tracking its 90-th percentile throughout our 50-day-long trajectories. While neXtSIM can represent such strong shear events, the deterministic surrogate generally fails to do so, leading to much a weaker tail. Contrastingly, the diffusion surrogate has a much smaller bias to neXtSIM, especially visible in the beginning of the trajectories. With unknown lateral boundary conditions, the trajectories between the diffusion surrogate and neXtSIM diverge after a few days. Afterwards, the shear rates of the diffusion follow more closely the ones from the deterministic surrogate, exhibiting However, if supported by the forcings, the diffusion surrogate can show sudden bursts in the shear as similarly observed in neXtSIM, e.g., before December 05. Therefore, the diffusion surrogate indicates a physical consistency in its forecast, something difficult to demonstrate with the deterministic surrogate.

6 Summary and Discussion

In this paper, we introduce the generative diffusion model specifically designed for sea-ice physics. Our model is built as a regional multivariate surrogate model learned

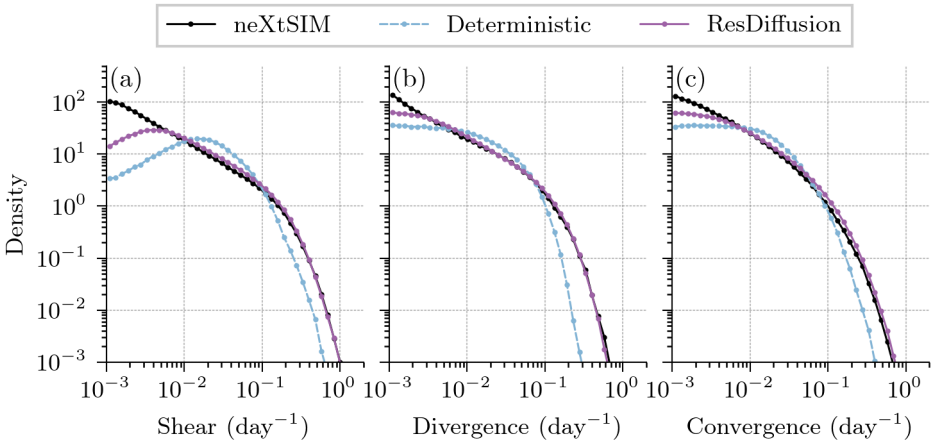

Figure 7. Empirical distributions of (a) divergence, (b) shear, and (c) convergence over the 50-days-long trajectories as in Fig. 5. The histogram for the residual diffusion model is an average across all 16 ensemble members.

from more than 20 years of data provided by the simulation analyzed in (Boutin et al., 2023). We select a region north of Svalbard and use a simulation, where neXtSIM (Rampal et al., 2016; Ólason et al., 2022) is coupled to the ocean component of NEMO (Madec, 2008). We train the diffusion surrogate to predict five different variables related to sea ice for a 12-hour lead time. We compare the diffusion surrogate to other surrogates like a deterministic surrogate trained with a weighted mean-squared error. In our experiments, generative diffusion consistently outperforms the other surrogates.

6.1 Surrogate modeling with diffusion models

The surrogates with generative diffusion are inherently stochastic and allow us to generate an ensemble of trajectories out of a single initial condition. Since the forecast error of its ensemble mean is lower than the error of all other competing models, generative diffusion has a large potential to generate cheap ensembles. The generated ensemble is however poorly calibrated with a too small ensemble spread, as shown in Appendix B5.

In our diffusion experiments, we generate the forecasts with the deterministic version of the second-order Heun integrator and the sampling noise scheduler from Karras et al. (2022), extend to a wider range of noise amplitudes. Out of the initial noise, the samples are generated without adding additional noise. Consequently, this sampler directly exhibits the quality of the diffusion model and of the chosen noise scheduling. As examined in Appendix B4, the diffusion model seems to suffer from an unbalanced training and might be improved by dynamically weighting of the loss function during training. Additionally, the results can be likely further improved by using a sampling noise scheduler adapted to geophysical problems. In the end, there might be a need of finding good sampling parameters and noise schedulers that are specifically tuned for geophysical problems.

The forecasts must be clipped into physical bounds, because otherwise they can become unstable and especially the deterministic surrogate would perform much worse, as shown in Appendix B1. The clipping introduces a bias into the forecasting procedure as the model are trained with an unconstrained criterion, e.g., the mean-squared error. To circumvent this bias, we need to explicitly incorporate the physical bounds into the optimization of the surrogates. A possibility for deterministic models could be to train

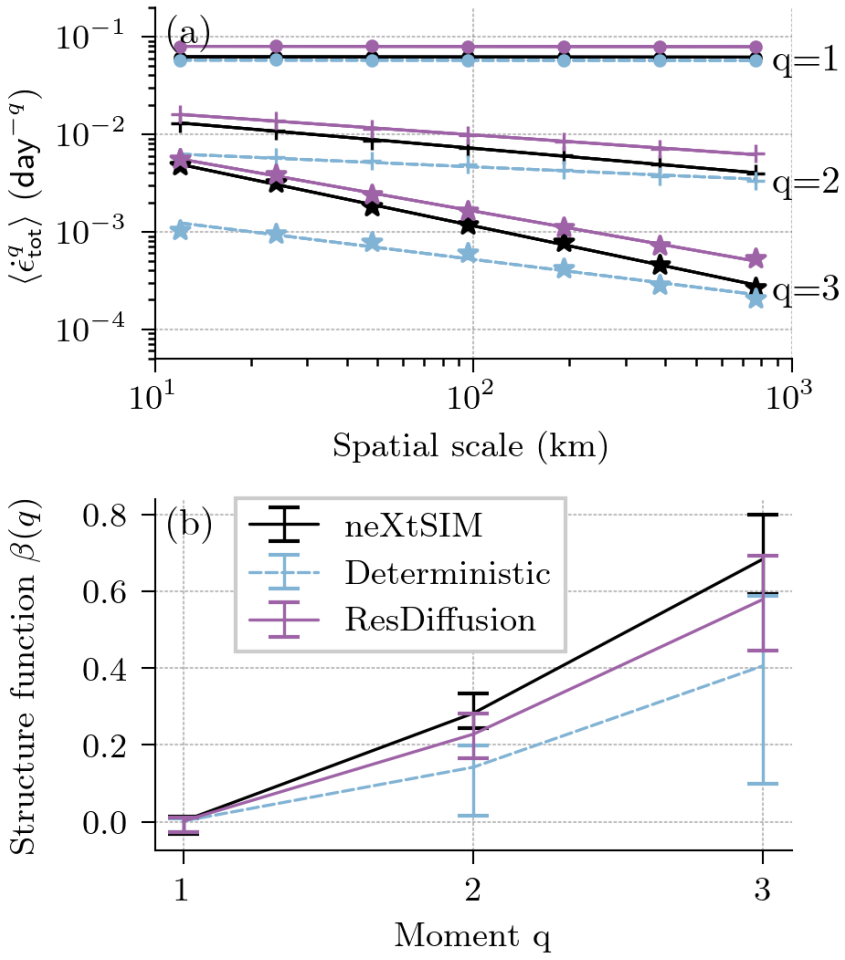

Figure 8. Spatial scaling analysis of the total deformation rate calculated over a timescale of 6 h in the 50-days-long trajectories for the fields from the true simulations (black), the deterministic surrogate (dashed, blue), and the diffusion surrogate (red). (a) Distributional moments of order $q = 1, 2, \text{and} 3$ for the total deformation rate for spatial scales estimated based on coarse-graining of the total deformation fields. The solid lines show the power-law scaling of the moments by the relation $\langle \dot{\epsilon}^q_{\text{tot}} \rangle \sim L^{-\beta(q)}$, where $\beta(q)$ is the scaling factor. (b) The structure functions that corresponds to the estimated scaling factors with error bars indicating a sort of uncertainty in the scaling factors, see also (Rampal et al., 2019).

the neural network by assuming a censored Gaussian distribution. However, for diffusion models, this is an open problem, where only approximative solutions exist (Luo et al., 2023; Fishman, Klarner, De Bortoli, et al., 2023; Fishman, Klarner, Mathieu, et al., 2023).

6.2 Physical consistency of the surrogate models

Training a deterministic surrogate with a (weighted) mean-squared error corresponds to making a local Gaussian assumption around the forecast of the surrogate; the surrogate implicitly targets the mean for the trained lead time, see also Appendix A1. Targetting a mean can result into unphysical and blurry forecasts, a problem that still persists in the newest generation of surrogates for the atmosphere (Bonavita, 2023; Lam et

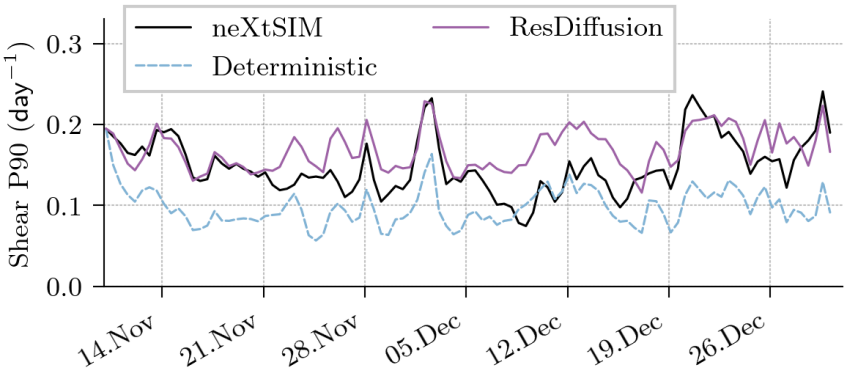

Figure 9. The temporal development of the 90-th percentile in the shear for neXtSIM, the deterministic surrogate and the diffusion surrogate in the 50-day-long trajectories.

al., 2023; Kochkov et al., 2023) and which has been also found for sea-ice surrogates in Durand et al. (2023). If cycled for longer lead times than originally trained for, the forecasted mean is reused as initial conditions for the next cycle, which amplifies the issue.

Trained to remove noise that has been artificially added during training, diffusion models learn to generate samples from the data-generating distribution without making a Gaussian assumption or whatsoever in data space. While this implicit sample generation could make the tuning of the model more difficult than explicitly assuming a distribution, it seems to improve the results for ensemble forecasts compared to a simple stochastic extension of the deterministic surrogate.

In addition to training diffusion surrogates from scratch, we also train a residual diffusion model (Mardani et al., 2023) on top of the deterministic surrogate. The generative diffusion then provides the missing stochastic term and can be seen as model error correction for the deterministic surrogate. Residual diffusion converges faster than training a diffusion model from scratch, while achieving similar scores. Therefore, generative diffusion models can be used for stochastic model error corrections, which enables us their use on top of physics-driven geophysical models, as possibly needed for sea-ice models (T. S. Finn, Durand, et al., 2023).

Since diffusion surrogates are trained to sample from the conditional probability distribution, they elegantly circumvent the mean-forecast issues of deterministic surrogates. Diffusion models consequently have the potential to generate physically-consistent trajectories.

Without being explicitly trained for, generative diffusion can match the spectral density of the neXtSIM simulations, even if cycled for longer lead times than the trained 12 hours. Further confirmed by inspecting single snapshots of predicted fields, generative diffusion can completely resolve the smoothing issue for sea-ice surrogates raised by Durand et al. (2023).

Going beyond the visual analysis of predicted fields, we also investigate if the predicted fields exhibit a physical consistency. We concentrate on the sea-ice dynamics in form of the divergence and shear rate as derived from the sea-ice velocity components.

The deterministic surrogate with its regression-to-the-mean leads to smoothing, artificial linear kinematic features, and wrong correlation lengths in the changes of the sea-ice thickness. Additionally exhibiting multi-fractality to a lesser degree than neXtSIM,

deterministic surrogates can hardly represent a physical consistency, as also observed for atmospheric surrogates by Bonavita (2023).

Generative diffusion can represent linear kinematic features as they are observed in neXtSIM. The link between weaker sea ice and divergence and shear is clearly exhibited and also the changes in the sea-ice thickness resemble those observed in neXtSIM. The spatial scaling laws derived from the moments of the total deformation distribution shows a clear multi-fractal signature which is similar to neXtSIM. Since we impose no lateral boundary conditions and constrain the available atmospheric forcing, the trajectories of the diffusion surrogate diverge from the neXtSIM simulations after a few days. Nevertheless, the tails of the derived shear fields indicate that the diffusion surrogate has a similar temporal behavior as neXtSIM. Therefore, diffusion surrogates show their potential for physical-consistent trajectories in our regional setup. However, it is too early to say if these results also hold for larger and even global setups as needed for, e.g., weather forecasts or climate projections.

If diffusion surrogates exhibit such a physical consistency, they might also lead to more stable long-term forecasts/projections. The forecasts of the diffusion surrogate are stable even if we remove clipping, see also Appendix B1. Furthermore, in early tests (not shown), we find that our trained diffusion surrogate can keep predictive power over a time period of two years, while the deterministic model shows this for just half a year. This would confirm results like Kohl et al. (2023) where diffusion surrogate have a superior stability compared to deterministic ones for turbulence modeling. However, its treatment would exceed the frame of this study, and we leave this open for future studies.

6.3 Scalability

One of the important question for diffusion models remains open: their computational scalability to very high-dimensional problems and the reduction of their forecast costs. Since the trained neural network is applied many times for one single forecast step, diffusion surrogates are n-times more expensive than deterministic ones, where n is the number of neural network evaluations, in our case $n = 39$. Additionally, they show their full predictive power if run as ensemble forecasts, which makes them further expensive. Evaluating our deterministic surrogate over the whole testing dataset takes 3 minutes, while the diffusion surrogate takes around 1.5 hours. Compared to classical geophysical models, this is still much cheaper but nevertheless one to two orders of magnitude bigger than for the deterministic surrogate.

Training of a diffusion surrogate is supposedly as expensive as training a deterministic surrogate, since both are trained with a supervised loss function. However, noise injection during training perturbs the gradient, requiring a lower learning rate, and slowing down the training of the diffusion model. Additionally, generative diffusion is trained to denoise for many amplitudes of noise, a multi-task problem (Hang et al., 2023), and we have to train bigger neural networks. On the one hand, this can unlock large-scale training where previously only small neural networks were trainable. On the other hand, these large-scales make their training more expensive. In our case, on one GPU, we trained the deterministic surrogate within 12 hours, while the training of the diffusion models took us several days.

This question of the scaling can prohibit the use of diffusion models for high-resolution and full-Arctic setups. However, the same question is raised for image generation, and there has been progress by integrating diffusion models within a latent space (D. Kingma et al., 2021; Vahdat et al., 2021; Rombach et al., 2022). The latent space is often spanned by a pre-trained autoencoder, which possibly makes the training of the diffusion surrogate more difficult. We could also try to tackle the problem directly in the core of the diffusion model, in the diffusion process: one way is the possible use of consistency models (Y. Song et al., 2023) which impose a consistency restriction on the neural network.

Another way can be rectified flows (X. Liu et al., 2022; Lipman et al., 2023) which abolish the diffusion process for a simpler linear mixing, and which show promise for large-scale image generation (Esser et al., 2024).

Despite these open questions, our results show the benefit of generative diffusion for geophysical modeling and specifically sea-ice physics. Our completely data-driven models exhibit a glimpse of physical consistency with possibly wide-reaching consequences. Hence, we see a huge potential of generative diffusion to resolve currently persisting issues with deterministic surrogates.

7 Conclusions

We introduce the first (denoising) diffusion model for sea ice physics, designed for multivariate surrogate modeling. In this study, we focus on a quantitative and qualitative analysis of the surrogate's properties. Based on our results, we conclude the following:

- Ensemble forecasting with generative diffusion outperforms deterministic surrogate models and their stochastic extensions across all prognostic sea-ice variables. While on par with the deterministic surrogate for the trained 12-hour lead time, the ensemble forecast improves the scores for longer lead times, tested up to 15 days. The training as generative diffusion enables us thereby the use of larger neural networks, which could improve their performance even more.
- Residual diffusion models can be trained as model error correction on top of other forecast models, like a deterministic surrogate. Applied like this, they enable us a stochastic forecast from a previously deterministic predictive model. Combined with a deterministic surrogate, residual diffusion surrogates can converge faster than diffusion surrogates trained from scratch.
- Diffusion surrogates retain information at all scales, enabling them to match the power spectral density of the data. Surrogate modeling with diffusion consequently yield sharp forecasts even for very long lead times, way outside what they were trained for. Diffusion surrogates hence resolves the smoothing issues of deterministic surrogates.
- The forecasts from diffusion surrogates exhibit a higher physical consistency than the deterministic surrogates'. For sea-ice models, diffusion surrogates clearly show the link between deformation, sea-ice concentration, and change in sea-ice thickness. The resulting fields hereby resemble those modeled by neXtSIM and exhibit a multi-fractal scaling behavior similar to that derived from observations.

Therefore, we see a huge potential for generative diffusion to unlock the next step in geophysical surrogate modeling.

Open Research Section

The code for a PyTorch toolbox to instantiate diffusion models is available under `https://github.com/cerea-daml/ddm-dynamical`, while the code for the experiments can be found under `https://github.com/cerea-daml/diffusion-nextsim-regional`. A Zenodo capsule, `https://doi.org/10.5281/zenodo.10949057`, contains the weights of the used neural networks (T. Finn et al., 2024). Extracted from `https://github.com/sasip-climate/catalog-shared-data-SASIP`, the capsule additionally includes the processed neXtSIM and ERA5 data. Disclaimer for the use of the included ERA5 data: the results contain modified Copernicus Climate Change Service information, 2023. Neither the European Commission nor ECMWF is responsible for any use that may be made of the Copernicus information or data it contains.

Author contributions

Conceptualization: TSF, CD, AF, MB. Data curation: TSF, CD. Formal Analysis: TSF. Investigation, Methodology: TSF. Software: TSF. Visualization: TSF. Writing – original draft preparation: TSF. Writing – review & editing: TSF, CD, AF, MB, PR, AC. Funding Acquisition: MB, PR, AC.

Acknowledgments
This study is a contribution to the SASIP project funded under Grant no. 353 by Schmidt Science – a philanthropic initiative that seeks to improve societal outcomes through the development of emerging science and technologies. TSF, CD, AF, and MB additionally received financial support from INSU/CNRS by the project GenD^{2}M (LEFE-MANU) and the project DeepGeneSIS (PNTS). This work was granted access to the HPC resources of IDRIS under the allocations 2023-AD011013069R2 made by GENCI. The authors would like to thank Guillaume Boutin for providing access to the data and other members from the SASIP project which gave helpful comments along the way. An additional thank goes to the Copernicus Climate Change Service to provide access to the ERA5 renalysis dataset (Hersbach et al., 2020, 2023). CEREA is a member of the Institut Pierre-Simon Laplace (IPSL).

Appendix A Additional methods

In this Appendix, we introduce additional methods and an more extensive treatment of the methods introduced in Sect. 3.

A1 Maximum likelihood estimation with a Gaussian assumption

In Sect. 3.1, we have introduced a weighted mean-squared error (MSE) as loss function to optimize the deterministic surrogate model. In the following, we will generalize this loss function in to maximum likelihood estimation and show that the weighted MSE corresponds to a Gaussian assumption for the predictive distribution.

Maximum likelihood estimation is derived from the idea that the future sea-ice conditions $\mathbf{x}_{t+12\,\mathrm{h}}$ are drawn from the true but unknown conditional probability distribution with its density function $p(\mathbf{x}_{t+12\,\mathrm{h}} \mid \mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}})$. This distribution includes the unresolved processes, which remain unexplained given the initial conditions $\mathbf{x}_t$ and the forcings $\mathbf{f}_{t:t+12\,\mathrm{h}}$. Since this distribution is unknown, we use a parameterized version $p_\theta(\mathbf{x}_{t+12\,\mathrm{h}} \mid \mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}})$, where θ denotes the distributional parameters (e.g., the mean and standard deviation of a univariate Gaussian distribution). This parameterized density function describes the likelihood of the future sea-ice conditions in dependence on the distributional parameters.

Our goal is to maximize the likelihood of the trainings data $(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}, \mathbf{x}_{t+12\,\mathrm{h}}) \sim \mathcal{D}$ given the distributional parameters. Since the logarithm is strictly increasing, the optimum of maximizing the likelihood is the same as the one maximizing the log-likelihood. Maximizing the log-likelihood is the same as minimizing the negative log-likelihood, our generalized loss function,

$$\mathcal{L}_{\mathrm{NLL}}(\theta) = -\log p_\theta(\mathbf{x}_{t+12\,\mathrm{h}} \mid \mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}). \tag{A1}$$

As conditional distribution, we assume a univariate Gaussian distribution with its density $\mathcal{N}(\mathbf{x}_{t+12\,\mathrm{h}} \mid \mathbf{x}_t + \mathcal{M}_\theta(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}), \mathbf{s}^2\mathbf{I})$, where the forecast of the deterministic surrogate model is the mean and the covariance is given as diagonal matrix with $\mathbf{s}^2$ on

its diagonal. Given this assumed Gaussian with its density, Eq. (A1) reads,

$$\mathcal{L}_{\text{Gauss},\mathbf{s}}(\theta) = \frac{1}{2}\left\| \frac{\mathbf{x}_{t+12\,\text{h}} - \mathbf{x}_t - \mathcal{M}_\theta(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\text{h}})}{\mathbf{s}} \right\|_2^2 + \frac{1}{2}\sum_{i=1}^{k} \log(s_i^2) + C, \tag{A2}$$

with C, a constant independent of θ and $\mathbf{s}$. By setting a global per-variable constant $\mathbf{s}$, $\log(\mathbf{s}^2)$ becomes a constant, and we can factorize out $\frac{1}{\mathbf{s}^2}$ of the remaining loss function. With such a constant variance as weighting factor, we hence recover the loss function used to optimize the deterministic surrogate model, Eq. (2). Consequently, the deterministic surrogate model is optimized to give predict the mean of a Gaussian distribution after a lead time of 12 hours.

A2 Covariance matrix estimation for the stochastic surrogate

To convert the deterministic surrogate into a stochastic surrogate, we can add noise to the deterministic forecast, as shown in Eq. (4). Since we assume a Gaussian distribution to train the deterministic model, we can naturally assume that the additive noise is also Gaussian distributed with $\mathbf{Q}$ as covariance. We can encode cross-variable and spatial correlations into the covariance, however, we are always confined to the Gaussian noise assumption. In the following, we show how we construct this covariance matrix. We make thereby extensively use of the deterministic forecast residuals after one iteration before the clipping is applied,

$$\mathbf{r} = \mathbf{x}_{t+12\,\text{h}} - \mathbf{x}_t - \mathcal{M}_\theta(\mathbf{x}_t, \mathbf{f}_{t:t+12\,\text{h}}). \tag{A3}$$

We decompose the covariance matrix $\mathbf{Q}$ into two terms: a univariate spatial correlation term and a cross-covariance term between variables. We describe the spatial correlation term by a two-dimensional Fourier spectrum which we impose on drawn random samples and the cross-covariance term by an explicit covariance matrix $\mathbf{Q}_{\text{cross}} \in \mathbb{R}^{5\times5}$.

The univariate spatial correlations are represented by two-dimensional power spectrum. The residuals from the validation dataset are transformed into Fourier space and averaged across all samples in this space. We present the averaged power spectrum transformed back into physical space in Fig. A1, indicating typical textures we expect for the residuals of the five forecasted variables. To circumvent issues with the boundary values, we split the power spectrum into a periodic and smooth component as described in Moisan (2011). We draw random samples from the periodic component by convolution with random Gaussian fields, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Afterwards, the smooth component is added to the random fields. This procedure to synthesize new samples out of known textures by convolution is called asymptotic discrete spot noise (ADSN, Galerne et al., 2011) and also used for generation of random precipitation fields (Seed et al., 2013; Pulkkinen et al., 2019). With this procedure, we efficiently generate samples with spatial correlations extracted from the validation dataset, while still allowing anisotropy.

The cross-covariance term is approximated based on the cross-covariance of the residuals for the i-th and j-th variable, averaged across all N_{samples} samples and N_{grid} grid points,

$$Q_{\text{cross},i,j} \approx \frac{1}{N_{\text{samples}} \cdot N_{\text{grid}} - 1} \sum_{k=1}^{N_{\text{samples}}} \sum_{l=1}^{N_{\text{grid}}} (r_{i,k,l} - \overline{r}_i)(r_{j,k,l} - \overline{r}_j), \tag{A4}$$

with $\overline{r}_i = \frac{1}{N_{\text{samples}} \cdot N_{\text{grid}}} \sum_{k=1}^{N_{\text{samples}}} \sum_{l=1}^{N_{\text{grid}}} r_{i,k,l}$. We show the estimated cross-covariance, decomposed into correlations and standard deviation, in Tab. A1a. To avoid spurious correlations, we take the estimated cross-covariance rounded to two decimals and suppress all correlations below 0.05. After the random fields are added to the deterministic forecast, the forecasts are clipped into their physical bounds, reducing the ensemble

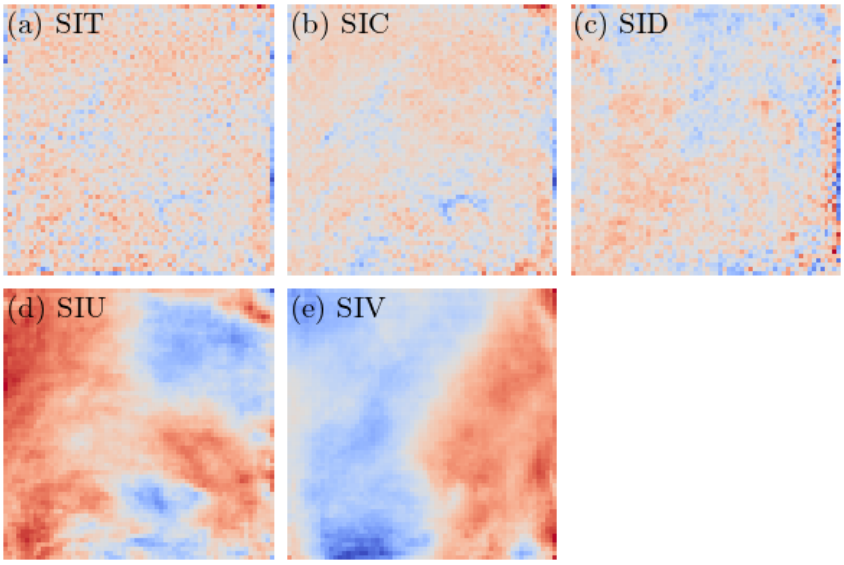

Figure A1. Textures extracted from the power spectrum of the residuals, averaged in Fourier space across all samples in the validation dataset. The random perturbations are generated based on a convolution with random Gaussian noise. The sea-ice thickness (a) shows almost no spatial correlations, while the velocity components (d) and (e) exhibit quite long correlations.

Table A1. Cross-correlations and standard deviations (σ) as estimated based on the residuals in the validation dataset (a) or as used for sampling (b).

(a)	SIT	SIC	SID	SIU	SIV
SIT	1.00	0.57	-0.05	0.01	-0.00
SIC	0.57	1.00	0.01	0.00	0.01
SID	-0.05	0.01	1.00	0.00	0.00
SIU	0.01	0.00	0.00	1.00	-0.07
SIV	-0.00	0.01	0.00	-0.07	1.00
σ	0.05	0.01	0.02	0.02	0.02

(b)	SIT	SIC	SID	SIU	SIV
SIT	1.00	0.57	-0.05	0.00	0.00
SIC	0.57	1.00	0.00	0.00	0.00
SID	-0.05	0.00	1.00	0.00	0.00
SIU	0.00	0.00	0.00	1.00	-0.06
SIV	0.00	0.00	0.00	-0.06	1.00
σ	0.05	0.02	0.02	0.02	0.02

spread. To counteract this reduced spread, we artificially inflate the standard deviations by factors. The modeled cross-covariance is shown in Tab. A1b.

We have tested several different methods to generate the noise but achieved hardly a stochastic surrogate that consistently outperforms the deterministic forecast.

A3 Score-based diffusion models

In Sect. 3.3, we briefly introduced our formulation of diffusion models. Here, we extend this formulation and give a stochastic differential equation (SDE) point of view.

We define in Eq. (6) a *variance-preserving* diffusion process, where the signal is progressively replaced by noise,

$$\mathbf{z}_\tau = \alpha_\tau \mathbf{x}_{t+12\,\mathrm{h}} + \sigma_\tau \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{6}$$

By defining a new variable $\widetilde{\mathbf{z}}_\tau = \frac{\mathbf{z}_\tau}{\alpha_\tau}$, we can do a change of variables and convert into a *variance-exploding* process, where noise is progressively added to the signal,

$$\widetilde{\mathbf{z}}_\tau = \mathbf{x}_{t+12\,\mathrm{h}} + \frac{\sigma_\tau}{\alpha_\tau}\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{A5}$$

In the following, we will describe $\frac{\sigma_\tau}{\alpha_\tau} = \widetilde{\sigma}_\tau = e^{-\lambda(\tau)}$ as the amount of noise added to the signal. Note, differently to Karras et al. (2022), we assume that our data is normalized to unit standard deviation, $\widetilde{\sigma}_{\mathsf{data}} = 1$. The noised state from the *variance-exploding* process can be equivalently written as

$$\widetilde{\mathbf{z}}_\tau \sim q(\widetilde{\mathbf{z}}_\tau \mid \mathbf{x}_{t+12\,\mathrm{h}}) = \mathcal{N}\left(\mathbf{x}_{t+12\,\mathrm{h}}, (\widetilde{\sigma}_\tau)^2\mathbf{I}\right). \tag{A6}$$

This diffusion process can be described by the following stochastic differential equation (SDE, Y. Song, Sohl-Dickstein, et al., 2021; Karras et al., 2022),

$$d\widetilde{\mathbf{z}} = g(\tau)d\mathbf{w}, \tag{A7}$$

where $g(\tau)$ is the diffusion term and $d\mathbf{w}$ a Wiener process, i.e. infinitesimal small Gaussian noise. Using the definition of the variance exploding process Eq. (A5), the diffusion term is given as

$$g(\tau)^2 = \frac{d}{d\tau}\log\left(1 + e^{-\lambda(\tau)}\right). \tag{A8}$$

Corresponding to the SDE that describes the diffusion process, there is a reversed SDE for the denoising process (Anderson, 1982; Y. Song, Sohl-Dickstein, et al., 2021),

$$d\widetilde{\mathbf{z}} = -g(\tau)^2\nabla_{\widetilde{\mathbf{z}}}\log p_\tau(\widetilde{\mathbf{z}})d\tau + g(\tau)d\widetilde{\mathbf{w}}, \tag{A9}$$

where $d\tau$ is an infinitesimal pseudo time step and $d\widetilde{\mathbf{w}}$ a Wiener process, both running in negative pseudo time direction. $\nabla_{\widetilde{\mathbf{z}}}\log p_\tau(\widetilde{\mathbf{z}})$ is the so-called score function. Instead of solving the SDE, we can solve the following probability flow ordinary differential equation (ODE, Y. Song, Sohl-Dickstein, et al., 2021), which results into the same marginals as Eq. (A9),

$$d\widetilde{\mathbf{z}} = -\frac{1}{2}g(\tau)^2\nabla_{\widetilde{\mathbf{z}}}\log p_\tau(\widetilde{\mathbf{z}})d\tau. \tag{A10}$$

To solve the denoising problem by either integrating the SDE or the ODE, we need access to the score function, which we approximate with a deep neural network in practice. Our target is thus to best estimate the weight and biases of the neural network θ such that

$$s_\theta(\widetilde{\mathbf{z}}, \tau) \approx \nabla_{\widetilde{\mathbf{z}}}\log p_\tau(\widetilde{\mathbf{z}}) \tag{A11}$$

holds for all pseudo time steps $\tau \in [0, 1]$. As loss function, we can make use of denoising score matching (DSM, Vincent, 2011; Y. Song & Ermon, 2020a),

$$\mathcal{L}(\theta) = \mathbb{E}_{\tau\sim\mathcal{U}(0,1)}\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\left[\widetilde{w}(\tau)\left\|s_\theta(\widetilde{\mathbf{z}}_\tau, \tau) - \nabla_{\widetilde{\mathbf{z}}_\tau}\log q(\widetilde{\mathbf{z}}_\tau \mid \mathbf{x}_{t+12\,\mathrm{h}})\right\|_2^2\right], \tag{A12}$$

with weighting $\widetilde{w}(\tau)$ and uniform distribution $\mathcal{U}(0, 1)$ with 0 and 1 as bounds. Choosing as weighting $\widetilde{w}(\tau) = \frac{d\lambda(\tau)}{d\tau}\widetilde{\sigma}_\tau$, ensures that the loss function Eq. (A12) maximizes a lower-bound on the data likelihood (Y. Song, Durkan, et al., 2021). Given the definition of the *variance-exploding* diffusion process, Eq. (A5), the denoising score function can be easily expressed as

$$\nabla_{\widetilde{\mathbf{z}}_\tau}\log q(\widetilde{\mathbf{z}}_\tau \mid \mathbf{x}_{t+12\,\mathrm{h}}) = -\frac{\epsilon}{\widetilde{\sigma}_\tau}. \tag{A13}$$

The denoising score matching loss function can be then optimized with Monte-Carlo sampling of the time and noise, converting the time into $\lambda(\tau)$ as defined by the noise scheduler and the noise into $\widetilde{\mathbf{z}}_\tau$ by Eq. (A5).

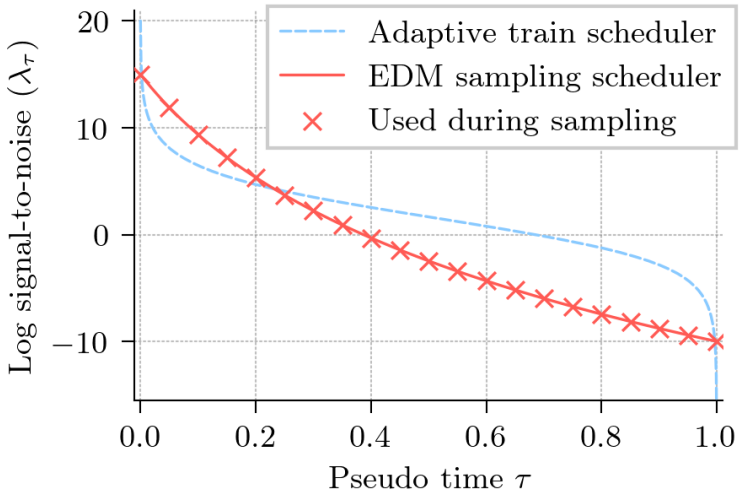

Figure A2. Two noise schedulers defining the log signal-to-noise ratio as function of the pseudo time. They are either adapated during the training process (blue dashed curve, D. P. Kingma & Gao, 2023) or fixed for sampling (red solid curve) as proposed by Karras et al. (2022) and modified by D. P. Kingma and Gao (2023). For the integration of the denoising ODE, the diffusion model is evaluated at 21 time steps as indicated by the red crosses.

Our used loss function Eq. (9) is a special case of the DSM loss Eq. (A12). Consequently, by setting

$$\nabla_{\widetilde{\mathbf{z}}_\tau} \log q(\widetilde{\mathbf{z}}_\tau \mid \mathbf{x}_{t+12\,\mathrm{h}}) \approx -\frac{\sigma_\tau \alpha_\tau \widetilde{\mathbf{z}}_\tau + \alpha_\tau \widehat{\mathbf{v}}_{\boldsymbol{\phi}}(\alpha_\tau \widetilde{\mathbf{z}}_\tau, \mathbf{x}_t, \mathbf{f}_{t:t+12\,\mathrm{h}}, \tau)}{\sigma_\tau}, \tag{A14}$$

we can approximate the denoising score, which can be used within the integration of the SDE, Eq. (A9), or the ODE, Eq. (A10).

A4 Adaptive noise scheduling

For the diffusion model, we need noise schedulers for training and for forecasting. In this study, we apply two different noise schedulers: an adaptive scheduler for training and a fixed one for forecasting. Instances of these noise schedulers are shown in Fig. A2.

The fixed noise scheduler for forecasting corresponds to the sampling scheduler of Karras et al. (2022),

$$\lambda(\tau) = -2\rho \log\left(\sigma_{\max}^{\frac{1}{\rho}} + (1-\tau)(\sigma_{\min}^{\frac{1}{\rho}} - \sigma_{\max}^{\frac{1}{\rho}})\right), \tag{A15}$$

where $\rho = 7$ is the form factor and $\sigma_{\min} = 0.002$ and $\sigma_{\max} = 80$ the minimum and maximum amplitude of noise added during the diffusion process. The support of this noise scheduler is only in the range $\lambda \in [-8.76, 12.43]$ and we extend this range to $\lambda \in [-10, 15]$ by truncation as proposed in D. P. Kingma and Gao (2023).

For training, we make use of the adaptive noise scheduler as introduced in D. P. Kingma and Gao (2023). Building upon variational diffusion models (D. Kingma et al., 2021), where the noise scheduler is learned together with the neural network, the idea is to adapt the scheduler to the loss function that is used for the training of the diffusion model.

As a reminder, the loss function for our diffusion model reads

$$\mathcal{L}_{\text{Diff}}(\boldsymbol{\phi}) = \mathbb{E}_{\tau\sim U(0,1)}\left[w(\tau)\cdot\left(-\frac{d\lambda(\tau)}{d\tau}\right)\cdot(e^{-\lambda(\tau)}+1)^{-1}\left\|\mathbf{v}_\tau - \widehat{\mathbf{v}}_{\boldsymbol{\phi}}(\mathbf{z}_\tau, \mathbf{x}_t, \mathbf{f}_{t:t+12\,\text{h}}, \tau)\right\|_2^2\right]. \quad (9)$$

During training, we convert a sampled time step with the noise scheduler into the log-signal-to-noise ratio $\lambda(\tau)$. The resulting ratio distribution reads then $p(\lambda(\tau)) = -\frac{d\lambda(\tau)}{d\tau}$ (D. P. Kingma & Gao, 2023). Consequently, the multiplicative weighting factor is $-\frac{d\lambda(\tau)}{d\tau} = \frac{1}{p(\lambda(\tau))}$ and the ratio distribution acts as importance sampling distribution. With a change of variables from τ to λ, the loss function results into

$$\mathcal{L}_{\text{Diff}}(\boldsymbol{\phi}) = \mathbb{E}_{\lambda\sim p(\lambda)}\left[\frac{w(\lambda)}{p(\lambda)}\cdot(e^{-\lambda}+1)^{-1}\left\|\mathbf{v}_\lambda - \widehat{\mathbf{v}}_{\boldsymbol{\phi}}(\mathbf{z}_\lambda, \mathbf{x}_t, \mathbf{f}_{t:t+12\,\text{h}}, \lambda)\right\|_2^2\right]. \quad (A16)$$

The diffusion model should be optimized over the whole range from $\lambda_{\min}$ to $\lambda_{\max}$. To focus the optimization on noise amplitudes where the weighted error is large, we set the ratio distribution to

$$p(\lambda) \propto \left[w(\lambda)\cdot(e^{-\lambda}+1)^{-1}\left\|\mathbf{v}_\lambda - \widehat{\mathbf{v}}_{\boldsymbol{\phi}}(\mathbf{z}_\lambda, \mathbf{x}_t, \mathbf{f}_{t:t+12\,\text{h}}, \lambda)\right\|_2^2\right]. \quad (A17)$$

As proposed in D. P. Kingma and Gao (2023), we approximate the distribution by tracking an exponential moving average of the weighted errors in the diffusion model. To track the weighted errors, we make use of 100 equal-distant bins going from $\lambda_{\min}$ to $\lambda_{\max}$. Given a λ-value, we determine the corresponding i-th bin, estimate the local error of diffusion model, and update the value of the bin by exponential moving average,

$$l_i^{\text{new}} = 0.999\cdot l_i^{\text{old}} + 0.001\cdot w(\lambda)\cdot(e^{-\lambda}+1)^{-1}\left\|\mathbf{v}_\lambda - \widehat{\mathbf{v}}_{\boldsymbol{\phi}}(\mathbf{z}_\lambda, \mathbf{x}_t, \mathbf{f}_{t:t+12\,\text{h}}, \lambda)\right\|_2^2. \quad (A18)$$

After updating the errors of the bins with a mini-batch of data, we construct an empirical distribution function, where the tracked values are proportional to the probability of the bin. This empirical distribution function then provides the mapping from λ to pseudo-time. To obtain the inverted mapping from pseudo-time to λ, we evaluate the empirical distribution function at the bin bounds and construct a piece-wise linear function that interpolates between two support values.

While this construction of the training noise scheduler seems difficult compared to a fixed scheduler, it provides the advantage that there are almost no tuning factors, except the rate for the exponential moving average. Additionally, this adaptive noise scheduler seems to improve the optimization of diffusion models (D. P. Kingma & Gao, 2023) as the model is preferably trained at noise amplitudes with large errors.

A5 Free-drift model

The ice velocity $\mathbf{u}_i$ is then analytically given as

$$\mathbf{u}_i = \alpha e^{-i\theta_i}\mathbf{u}_a + \mathbf{u}_w, \quad (A19)$$

where $\alpha = \sqrt{\frac{\rho_a C_a}{\rho_w C_w}}$ is a transfer coefficient and θ_i is the combined turning angle. Following the values of (Rampal et al., 2019; Boutin et al., 2023), we obtain $\alpha \approx 0.0174$ and $\theta_i \approx 25°$ as values. Since we exclusively have atmospheric forcings, the additional velocity term coming from the ocean is unknown and we neglect it by setting $\mathbf{u}_w = 0$. To estimate the grid-point-based sea-ice velocity with Eq. (A19) for times between two available lead times (every 12 hours), we linearly interpolate the atmospheric velocities in time and estimate the sea-ice velocities based on these interpolated values.

To advect the SIT, SIC, and damage with given sea-ice velocities, we construct a two-dimensional advection scheme, solving

$$\frac{\partial s(\mathbf{x},t)}{\partial t} + \mathbf{u}(\mathbf{x},t)\frac{\partial s(\mathbf{x},t)}{\partial \mathbf{x}} = 0, \quad (A20)$$

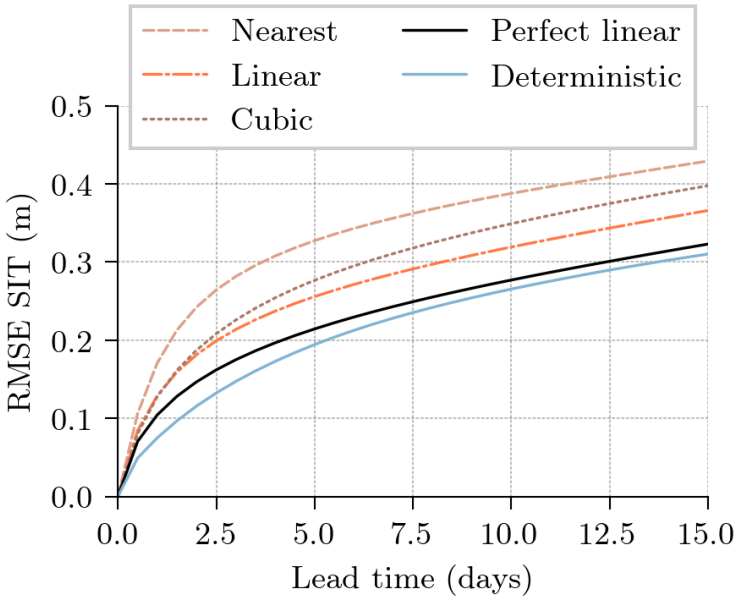

Figure A3. Comparison of the forecasting error for the sea-ice thickness between different interpolation methods for a cycled semi-Lagrangian advection scheme. *nearest* uses a nearest neighbor interpolation, *linear* a bilinear interpolation, *cubic* a bicubic interpolation, *perfect linear* a bilinear interpolation with a perfect knowledge of the sea-ice velocities. For reference, *deterministic* are the results from the deterministic surrogate model.

for the general tracer $s(\mathbf{x}, t)$ and velocity $\mathbf{u}(\mathbf{x}, t)$ at position $\mathbf{x}$ and time t. We solve Eq. (A20) from a Lagrangian perspective, satisfying

$$s(\mathbf{x}, t + \Delta t) = s(\mathbf{x} - \delta, t), \tag{A21}$$

for a time difference Δt and the displacement δ. The displacement corresponds to the velocities integrated from time t to time $t + \Delta t$.

We use a backward semi-Lagrangian integration scheme, where we start at time $\Delta t = 12\,\text{h}$ and take $dt = 1200\,\text{s}$ steps. At time $t+\Delta t-n\cdot dt$, where n is the integration step, we estimate the sea-ice velocity with Eq. (A19) for all grid points and take the nearest neighboring grid points to the backward advected position. The velocities are kept constant for a window of dt, and we advect the positions further backward in time, until we reach $n \cdot dt = \Delta t$.

Each grid point $\mathbf{x}$ at time $t + \Delta t$ has then a corresponding displaced grid point $\mathbf{x}-\delta$ at time t. Since the initial conditions at time t are only known at the original grid point position, we have to interpolate the initial conditions from the grid point positions $\mathbf{x}$ to the displaced positions $\mathbf{x} - \delta$. We test three different schemes to achieve this interpolation: a simple nearest neighbor interpolation, a bilinear interpolation, and a bicubic interpolation, with results shown for the sea-ice thickness in Fig. A3.

From all three interpolation scheme, the bilinear interpolation is the most stable and performs the best across all lead times. While initially the bicubic interpolation has a similar RMSE as the bilinear interpolation, it is more unstable because of oscillations, a well-known problem of higher-order interpolation schemes.

The difference between a bilinear interpolation and a bilinear interpolation with a perfect knowledge of the sea-ice velocities every 12 hours is on a similar scale as the

difference between the nearest neighbor and the bilinear interpolation. Furthermore, the deterministic surrogate model outperforms all free-drift model version, even if the velocities are perfectly known. The approximations of a pre-defined α and θ factor and the neglection of the ocean velocities do not change the general results. Therefore, we use as baseline method the semi-Lagrangian free-drift model with the linear interpolation.

A6 UViT neural network architecture

As neural network architecture, we use a UViT architecture (Hoogeboom et al., 2023), where we combine ConvNeXt blocks with transformer blocks, see also Fig. A4 for a general schematic of the architecture and the two different blocks. The number of parameters per block and the input and output dimensions are given in Table A2 for the deterministic model and in Table A3 for the diffusion model. In the following, we will briefly explain the blocks, for more details we refer to the official implementation, `https://github.com/cerea-daml/diffusion-nextsim-regional/blob/main/diffusion_nextsim/network.py`.

The initial projection expand the input channels to 64 latent features with a convolution that uses a 1×1 kernel. On top of these extracted features, we apply a U-Net-like architecture (Ronneberger et al., 2015), where three downsampling blocks are followed by n transformer blocks and three upsampling blocks. This way the architecture can extract features across four different scales. Shortcut connections between downsampling blocks and upsampling blocks enable the network to maintain the initial sharpness of the fields.

Throughout the network, we make use of layer normalization conditioned on the inputted labels from the data augmentation and, in the case of diffusion models, the pseudo time. The conditioning information determines hereby the affine scaling and shifting parameters of the normalization (Perez et al., 2017). The inputted labels are linearly embedded, while we extract features from the pseudo time by sinusoidal features (Vaswani et al., 2017) and a small MLP afterwards. The linear embedding and the extracted features are added together and activated by a Gelu before they are projected into the affine parameters.

Each downsampling block includes two ConvNeXt blocks and a downsampling layer. The ConvNeXt blocks (Z. Liu et al., 2022) try to imitate transformer blocks with purely convolutional layers: first, spatial features are extracted with convolutions, group-wise operation (no mixing of the feature channels) and a 7×7 kernel. After extracting spatial features, the features are normalized by conditioned layer normalization. Secondly, a small multi-layered perceptron (MLP) with a Gaussian error linear unit (Gelu, Hendrycks & Gimpel, 2016) as activation in-between mixes the channels point-wise. Using residual connections (He et al., 2015), the input from the ConvNeXt block is added to its output with a learnable gamma scaling (Bachlechner et al., 2020; De & Smith, 2020). After the second ConvNeXt block, before the downsampling layer, conditioned layer normalization is applied to normalize the extracted features, which stabilizes the downsampling operation (Z. Liu et al., 2022). The downsampling layer halves the field size and doubles the number of channels by a learnable convolution with a 2×2 kernel and a stride size of 2.

The transformer blocks combine multi-head attention with a MLP (Vaswani et al., 2017). We use pre-layer normalization (R. Xiong et al., 2020), where the multi-head attention and MLP block are started by a conditioned layer normalization. Additionally, we regularize both blocks by incorporating dropout into the attention and MLP with a probability of $p = 0.2$. For the multi-head attention, A 1×1 convolution layer extracts the needed values, keys, and queries. Multiplied to attention weights, the keys and queries are used to reweight the extracted values. Using 8 different heads per self-attention, the multi-head attention can learn to attend to different parts of the data. The output of

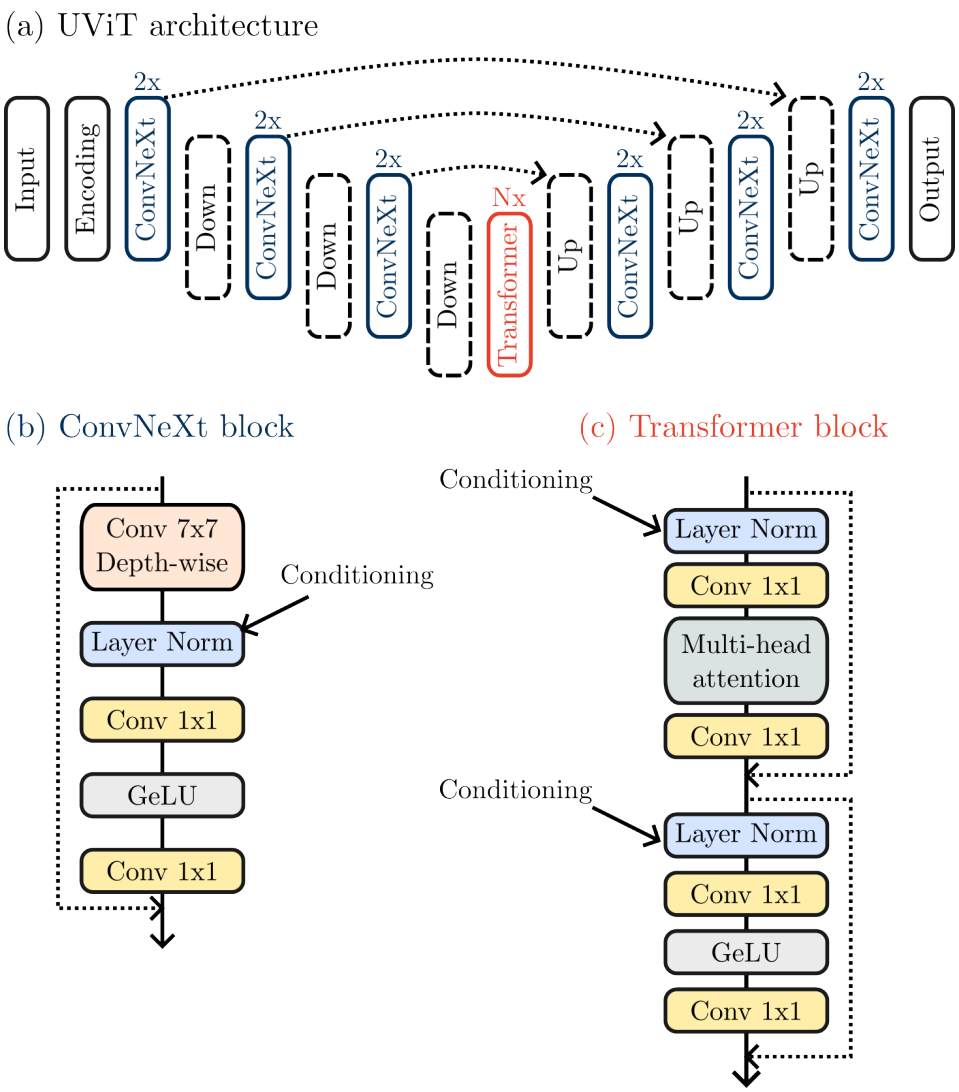

Figure A4. (a) The instantiated UViT architecture with (b) ConvNeXt (blue) and (c) transformer blocks (red). The dashed arrows indicate shortcut and residual connections. In the architecture, the ConvNeXt blocks are repeated twice, while the number of transformer blocks is kept as scaling parameter repeated N-times (for the deterministic surrogate $N = 2$, for the diffusion surrogate $N = 8$).

the multi-head attention is projected back into feature space and added to the input of the attention block by a learnable gamma factor. The following MLP is constructed as the MLP within the ConvNeXt block, mixing the channels up and extracting additional non-linear features.

The upsampling blocks mirror the downsampling blocks as close as possible: an upsampling layer is followed by two ConvNeXt blocks. Before upsampling, the data is normalized by conditioned layer normalization. To upsample, we use nearest neighbour interpolation, doubling the field size. Concatenated to the shortcut connections, the interpolated fields are convolved with a 3×3 kernel. We use this interpolation followed by convolution scheme to avoid checkerboard artifacts which can be caused by transposed convolutions (Odena et al., 2016).

Table A2. The U-Vit architecture as used for the deterministic surrogate model. Each layer and block is shown by its number of parameters, the number of input channels n_{in}, the number of output channels n_{out}, and the grid dimensions of the output in x- and y-direction, n_x and n_y, respectively. In total, the network has 7.6×10^6 parameters.

Stage	Operation	Params	n_{in}	n_{out}	n_x	n_y
Embedding	Labels	256	4	64	1	1
Input	1×1 Conv	896	13	64	64	64
Down 1	ConvNeXt	19 904	64	64	64	64
	ConvNeXt	19 904	64	64	64	64
	Down	41 216	64	128	32	32
Down 2	ConvNeXt	56 192	128	128	32	32
	ConvNeXt	56 192	128	128	32	32
	Down	147 968	128	256	16	16
Down 3	ConvNeXt	177 920	256	256	16	16
	ConvNeXt	177 920	256	256	16	16
	Down	558 080	256	512	8	8
Bottleneck	Transformer	1 710 080	512	512	8	8
	Transformer	1 710 080	512	512	8	8
Up 1	Up	1 836 288	512	256	16	16
	ConvNeXt	177 920	256	256	16	16
	ConvNeXt	177 920	256	256	16	16
Up 2	Up	475 776	256	128	32	32
	ConvNeXt	56 192	128	128	32	32
	ConvNeXt	56 192	128	128	32	32
Up 3	Up	127 296	128	64	64	64
	ConvNeXt	19 904	64	64	64	64
	ConvNeXt	19 904	64	64	64	64
Output	LayerNorm	128	64	64	64	64
	relu	–	64	64	64	64
	1×1 Conv	325	64	5	64	64

For the output, the extracted features from the last upsampling block are normalized by layer normalization without conditioning and activated by a rectified linear unit (relu). Here, we replace Gelu by relu as this can help to represent discrete-continuous behavior for sea-ice applications (T. S. Finn, Durand, et al., 2023). These activated features are then combined by a 1×1 convolution to the output channels.

Appendix B Additional results

In Sect. 5, we concentrate on the performance of a single diffusion model without justifying certain hypothesizes. In the following, we present additional results for the deterministic surrogate and the diffusion surrogate to provide a complete picture. Note, compared to the results in the main manuscript, we show results with the diffusion surrogate instead with the residual diffusion surrogate to point towards possible issues with generative diffusion trained from scratch.

B1 Surrogate modeling without clipping

To apply our surrogates, we clip the values for the sea-ice thickness, sea-ice concentration, and damage into physical bounds. However, the surrogates are trained for

Table A3. The U-Vit architecture as used for the diffusion surrogate model. The columns have the same meaning as in Table A2. In total, the network has 19.4×10^6 parameters. The residual diffusion model has five more input channels, which increases the number of the parameters for the input layer to 1 536. Note, the number of parameters is increased for the same layer compared to the deterministic model as the embedding size is increased from 64 to 128.

Stage	Operation	Params	n_{in}	n_{out}	n_x	n_y
Embedding	Labels	512	4	128	1	1
	Time MLP	82 176	1	128	1	1
Input	1×1 Conv	1 216	18	64	64	64
Down 1	ConvNeXt	28 096	64	64	64	64
	ConvNeXt	28 096	64	64	64	64
	Down	49 408	64	128	32	32
Down 2	ConvNeXt	72 576	128	128	32	32
	ConvNeXt	72 576	128	128	32	32
	Down	164 352	128	256	16	16
Down 3	ConvNeXt	210 688	256	256	16	16
	ConvNeXt	210 688	256	256	16	16
	Down	590 848	256	512	8	8
Bottleneck	Transformer	1 841 152	512	512	8	8
	Transformer	1 841 152	512	512	8	8
	Transformer	1 841 152	512	512	8	8
	Transformer	1 841 152	512	512	8	8
	Transformer	1 841 152	512	512	8	8
	Transformer	1 841 152	512	512	8	8
	Transformer	1 841 152	512	512	8	8
	Transformer	1 841 152	512	512	8	8
Up 1	Up	1 901 824	512	256	16	16
	ConvNeXt	210 688	256	256	16	16
	ConvNeXt	210 688	256	256	16	16
Up 2	Up	508 544	256	128	32	32
	ConvNeXt	72 576	128	128	32	32
	ConvNeXt	72 576	128	128	32	32
Up 3	Up	143 680	128	64	64	64
	ConvNeXt	28 096	64	64	64	64
	ConvNeXt	28 096	64	64	64	64
Output	LayerNorm	128	64	64	64	64
	relu	–	64	64	64	64
	1×1 Conv	453	64	5	64	64

unclipped values, which leads to a inconsistency between training and application of the surrogates.

In Fig. B1, we compare the deterministic surrogate with and without clipping, both version are based on the same model, trained for no clipping. While the unclipped surrogate performs initially as well as the clipped one, it becomes easily unstable, leading to a rapid error increases within several days. In the end, the unclipped surrogate performs much worse than the clipped one, showing the need of clipping.

Contrastingly, the diffusion surrogate is always stable and clipping has almost no impact on its scores. This confirms the results from Kohl et al. (2023), where they show for a turbulent flow that diffusion surrogates are much more stable than deterministic ones. Nevertheless, training a diffusion surrogate without explicitly taking physical con-

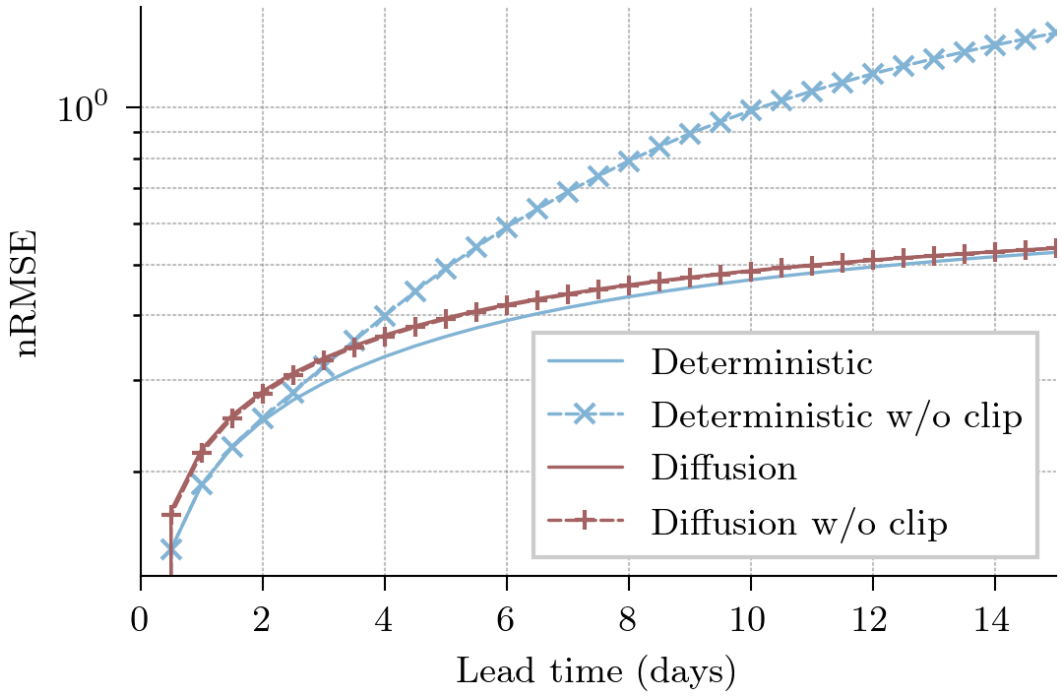

Figure B1. Effect of clipping on the nRMSE of the deterministic (blue) and diffusion (red) surrogate averaged all variables and all samples in the testing dataset. Without clipping (dotted), the deterministic surrogate become easily unstable, while the diffusion one remains stable. Note, the scores for a single ensemble member are shown here, and the nRMSE has a logarithmic scale.

straints into account can introduce a bias into the surrogate, which could lead to sub-optimal results.

B2 Impact of sea-ice damage

In this manuscript, our goal is to learn a surrogate model for the dynamics of neXtSIM, a geophysical model. Therefore, we forecast with our surrogates all prognostic variables available in our dataset, even the sea-ice damage. Originally introduced as memory for past stresses and to simulate the existence of subgrid-scale cracks and leads (Girard et al., 2011), its mechanics are somewhat artifical, acting like an additional latent variable. In our dataset, the damage is treated differently than the other variables and kept as instantaneous variable, while all other are averaged within a 6-hour window. Furthermore, there are no observational equivalents to the damage variable and there is no similar output in the CMIP6 dataset (Eyring et al., 2016). This raises the question if the sea-ice damage variable is needed and if it can improve the surrogate model if cycled over several days.

As comparison, we trained an additional deterministic surrogate by leaving the damage variable out, while keeping everything else the same. In Fig. B2, we compare the forecast error with increasing lead time between the two deterministic surrogates.

For the two shown variables, sea-ice thickness, and sea-ice velocity, the surrogate without damage slightly improves the error compared to the one with damage. However, the difference is smaller than the difference between the deterministic and diffusion surrogate. This result holds also for the other not shown variables. One of the reasons why the neural network performs better without damage might be that the number of its tasks

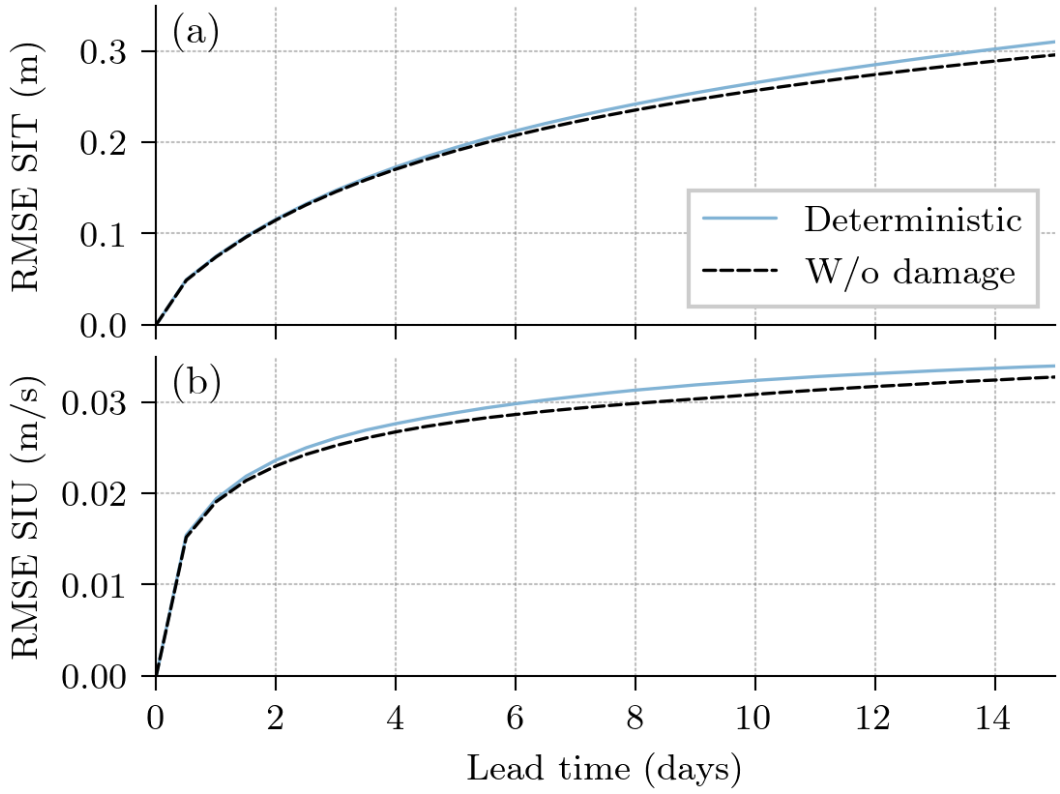

Figure B2. Comparison of root-mean-squared error (RMSE) with the deterministic surrogate with (blue) and without the forecast of damage (blacked, dashed) for (a) the sea-ice thickness and (b) the sea-ice velocity in x-direction, averaged across all samples in the testing dataset.

is reduced from five to four, freeing capacity to better forecast the other variables. Additionally, we have seen that the instabilities if clipping is deactivated, see also Appendix B1, are reduced for the surrogate without damage (not shown). Consequently, if the goal is to get the best possible forecast, independent of the goal to best emulate the geophysical model, we can recommend to use a surrogate without prognostic damage. This can improve the scores, make the model more stable and simplify the evaluation procedure. However, since our goal was to find an emulator for neXtSIM, we kept the deterministic model with predicting the damage.

B3 Impact of data augmentation

A way to artificially increase the data amount is to apply data augmentation. In data augmentation, the drawn samples from the dataset are randomly distorted by given transformations. During the training of our surrogates, we apply random horizontal flipping with a probability of $p = 0.5$, random vertical flipping with $p = 0.5$, and random rotation by 90° with $p = 0.5$. This should help the surrogates to learn features that are invariant to flipping and to rotations, possibly providing an additional physical prior information. During inference time, when we forecast, we deactivate any data augmentation. Applying this data augmentation helps us to reduce the amount of overfitting present in our surrogate model, as illustrated in Fig. B3, when comparing the green to the violet curve. Although the final loss might be lower with data augmentation, the time until convergence is increased.

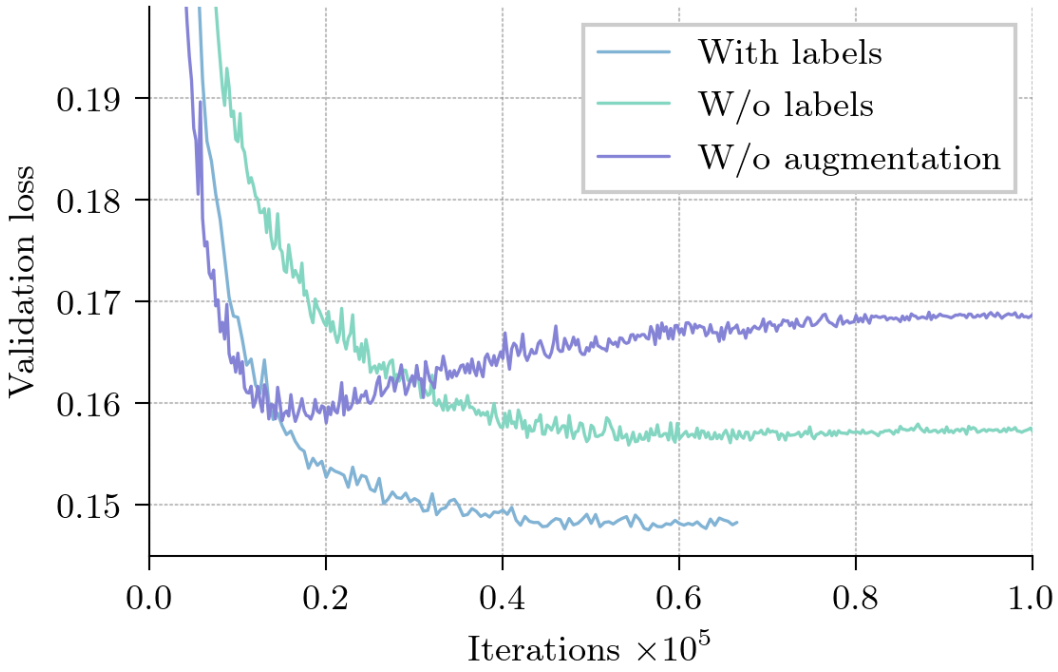

Figure B3. The validation loss of the deterministic surrogate with augmentation and labels (blue), the surrogate without labels (green), and the surrogate without augmentation and labels (violet).

In addition to the initial conditions and external forcing, we can also give the surrogate information about the data augmentation. The surrogate is conditioned by providing label information about which augmentation is activated. This label information is then linearly embedded and influences the affine transformations in the normalization layers. During inference time, we use an empty label vector, filled with zeros. This distributional augmentation approach (Jun et al., 2020) allows us to see the augmentation as data-dependent regularizer or as additional tasks on which the surrogate is trained on. This labelling helps generative modeling in settings with a low amount of data and is also used in some of the state-of-the-art diffusion models (Karras et al., 2022). In our case, the deterministic surrogate reached with this additional labelling the lowest validation loss. Furthermore, this labelling resolves the issues with the speed of convergence when data augmentation is applied. Therefore, we use distributional augmentation during the training of our surrogates.

B4 Weighting in the diffusion surrogate

The diffusion model optimizes a weighted mean-squared error in predicting $\mathbf{v}$, see also Eq. (8). As weighting factor, we use an exponential weighting, while the additional density of the noise scheduler is adapted to a binarized exponential moving average during training, see also Appendix A4. The target data is normalized to mean 0 and standard deviation 1 by per-variable statistics estimated based on the climatology of the dynamics. Consequently, the contribution of the five different variables is implicitly weighted by these climatological statistics.

Variables like the sea-ice velocities might be better constrained by the initial conditions and forcings and easier to predict than others, resulting into smaller errors, Fig. B4. Their contribution to the total loss is then downweighted. The diffusion model would

be more optimized for the other variables, which could lead to problems with the calibration of the surrogate, as shown in Sec. 5.

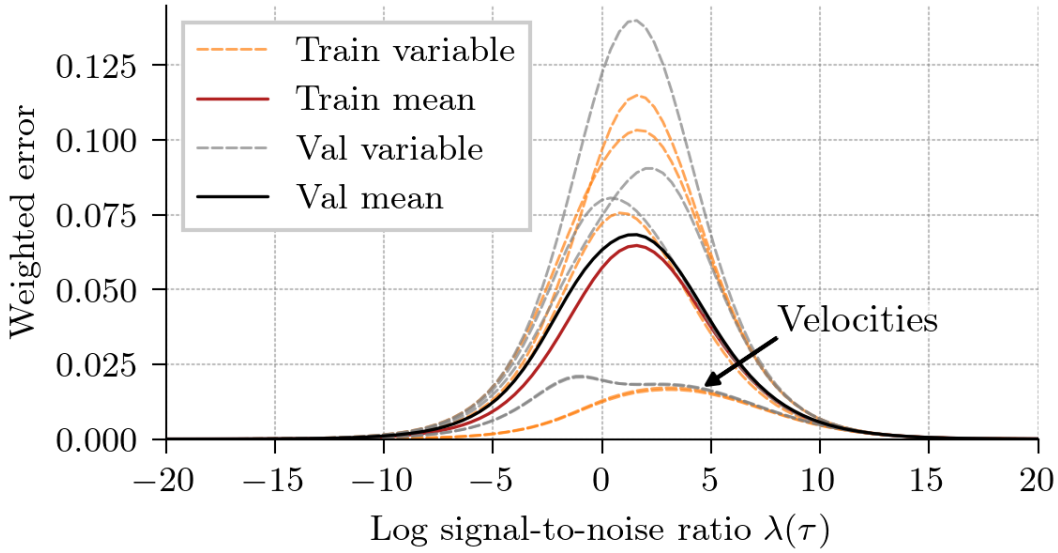

Figure B4. The error of the diffusion model in predicting $\mathbf{v}$ (see also Eq. (8)) for a randomly selected data batch of 1024 samples in the training dataset (orange and red) and in the validation dataset (grey and black), weighted by an exponential weighting as used for the training of the diffusion model. Since the data is normalized by the climatology, also the different error terms are implicitly weighted by this climatology. The five different variables (orange and grey) show in general an unequal error behavior, which is absorbed by the averaged loss. Additionally, the validation errors are generally higher than the training errors, indicating slight overfitting.

One way to tackle such problems could be to alter the weighting for different variables, as similarly done in GenCast (Price et al., 2023). Inspired by the solution of maximum likelihood estimation, we can also weight the different contributions by the expected error for a given variable as proposed in Rybkin et al. (2020) and used in T. S. Finn, Durand, et al. (2023) for model error corrections. In the end, the density of the noise scheduler would not have one single value per λ bin but one for each variable, proportional to the error of this variable within the given bin. The λ values resulting out of the noise scheduler would be still given by the average of all variables, its density is shown by the red and black line in Fig. B4.

Fig. B4 shows a different behavior between the training loss and validation loss, especially for the sea-ice velocities. The training loss is additionally slightly smaller than the validation loss, possibly indicating overfitting, which is also discussed in Appendix B6.

B5 Evaluation of the diffusion ensemble

Here, we discuss the calibration of the ensembles steming from the stochastic surrogate and the diffusion surrogate models. In Fig. B5, we show the spread-skill ratio for the ensembles and the rank histograms of the diffusion ensemble for the sea-ice thickness and the sea-ice velocity in x-direction.

Dissecting the ensembles shows their underdispersion with a decreasing spread-skill ratio for an increasing lead time, as shown in Fig. B5. Since sea ice is heavily driven by

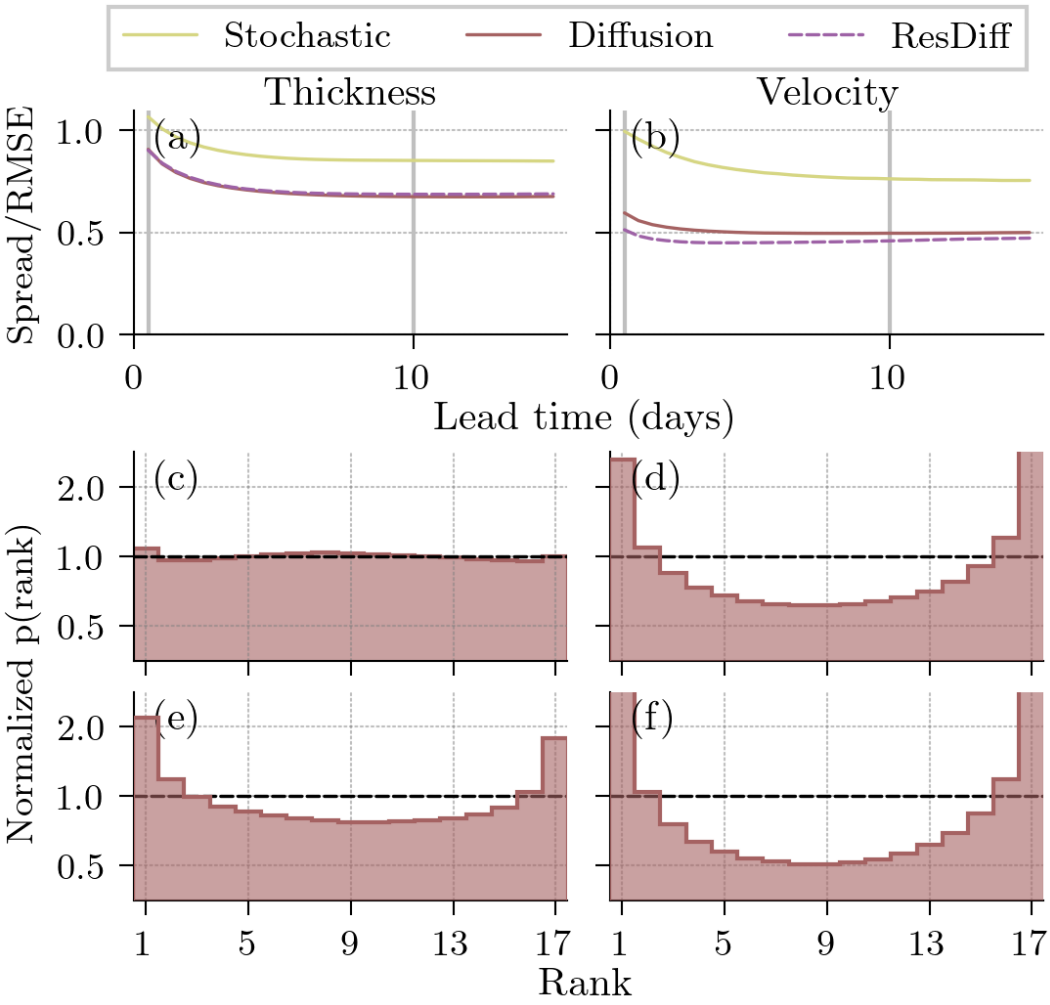

Figure B5. The spread-skill ratio (a & b) and rank histograms (c–f) with the stochastic surrogate (yellow), diffusion surrogate (red) and the residual diffusion surrogate (dashed, violet) for the sea-ice concentration (a, c, & e) and the the sea-ice velocity in x-direction (b, d & f). The spread-skill ratio (a & b) are estimated by ratio of the square-root of the averaged ensemble variance to the square-root of the mean-squared error. The rank histograms are for a lead time of 12 hours (c & d) and a lead time of 10 days (e & f) and normalized by the expected density, $\frac{1}{17}$. All metrics are averaged across all samples and grid points in the testing dataset.

the external forcings, the instantiated models are dissipative, something also observed in geophysical sea-ice models (Chen et al., 2023; Cheng et al., 2023). The models must additionally generate the lateral boundary conditions, which further increases their dissipative behavior. These two factors lead to the reduction of the ensemble spread with lead time.

While initially quite well-calibrated for the tracer variables, e.g., for the shown sea-ice thickness, the ensemble spread is too small compared to the errors for the velocities. This might be a result out of balancing issues during the training of the diffusion surrogates. The loss terms for the different variables are implicitly weighted by their climatology because of data normalization, whereas the velocities seem to be easier to forecast than the tracers, see also Appendix B4. As a consequence, the contribution of the

velocities to the total loss is smaller than that of the tracers, and the model seems unbalanced. Consequently, the system's dissipative behavior and possible balancing issue seem to cause the poorly calibrated ensemble for the diffusion surrogate.

B6 Overfitting in the diffusion surrogate

Diffusion models optimize the ELBO on the targetted data, minimizing the Kullback-Leibler divergence between the true generating distribution and the distribution as approximated by the diffusion model. The loss function shows the quality of the whole distribution, while the RMSE only measures the performance of the first moment. Finding the best model in terms from RMSE might consequently differ from the best model in terms of loss function. This mismatch between network calibration and accuracy has been also observed in neural networks for classification (Nguyen et al., 2015; Guo et al., 2017; Minderer et al., 2021).

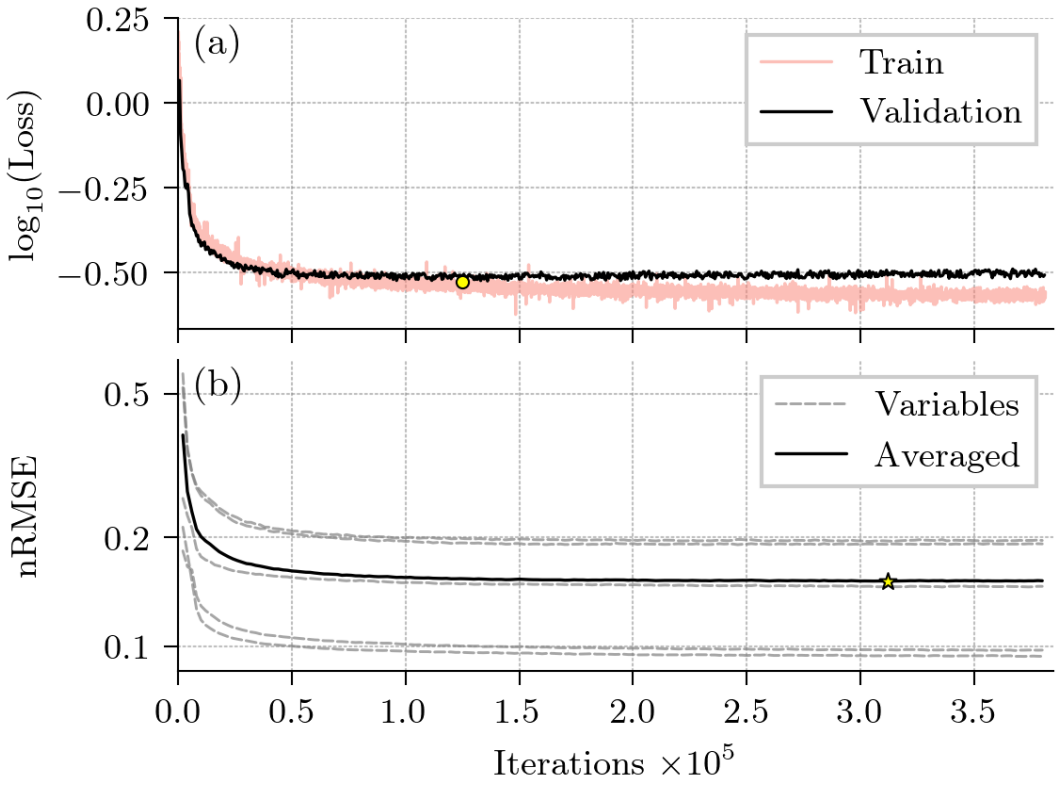

Figure B6. (a) The logarithm of the loss function for the training dataset (red) and validation dataset (black), (b) the nRMSE for a 12-hour lead time of the five predicted variables (grey) and as average across the five variables (black) in the validation dataset. The yellow dot represents the lowest validation loss and the yellow star the lowest nRMSE. While the loss indicates an onset of overfitting at 1.25×10^5 iterations, the nRMSE exhibits almost no overfitting.

In Fig. B6, we show the difference between selecting the best model with the loss (a) and with the MSE (b). The loss in the validation dataset shows sign of overfitting much earlier than the RMSE in the same dataset. Higher moments of the distribution become worse while the first moment still improves with higher number of iterations. In the end, it seems like there is a trade-off between optimizing the model in terms of RMSE or in terms of predicted distribution.

References

Anderson, B. D. O. (1982, May). Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, *12*(3), 313–326. Retrieved 2023-05-14, from https://www.sciencedirect.com/science/article/pii/0304414982900515 doi: 10.1016/0304-4149(82)90051-5

Andersson, T. R., Hosking, J. S., Pérez-Ortiz, M., Paige, B., Elliott, A., Russell, C., ... Shuckburgh, E. (2021, August). Seasonal Arctic sea ice forecasting with probabilistic deep learning. *Nature Communications*, *12*(1), 5124. Retrieved 2022-06-27, from http://www.nature.com/articles/s41467-021-25257-4 (Number: 1 Publisher: Nature Publishing Group) doi: 10.1038/s41467-021-25257-4

Asperti, A., Merizzi, F., Paparella, A., Pedrazzi, G., Angelinelli, M., & Colamonaco, S. (2023, September). *Precipitation nowcasting with generative diffusion models.* arXiv. Retrieved 2024-03-15, from http://arxiv.org/abs/2308.06733 (arXiv:2308.06733 [physics]) doi: 10.48550/arXiv.2308.06733

Bachlechner, T., Majumder, B. P., Mao, H. H., Cottrell, G. W., & McAuley, J. (2020, June). *ReZero is All You Need: Fast Convergence at Large Depth.* arXiv. Retrieved 2022-11-23, from http://arxiv.org/abs/2003.04887 (arXiv:2003.04887 [cs, stat]) doi: 10.48550/arXiv.2003.04887

Bahdanau, D., Cho, K., & Bengio, Y. (2016, May). *Neural Machine Translation by Jointly Learning to Align and Translate.* arXiv. Retrieved 2024-02-19, from http://arxiv.org/abs/1409.0473 (arXiv:1409.0473 [cs, stat]) doi: 10.48550/arXiv.1409.0473

Batzolis, G., Stanczuk, J., Schönlieb, C.-B., & Etmann, C. (2021, November). *Conditional Image Generation with Score-Based Diffusion Models.* arXiv. Retrieved 2024-02-16, from http://arxiv.org/abs/2111.13606 (arXiv:2111.13606 [cs, stat]) doi: 10.48550/arXiv.2111.13606

Ben-Bouallegue, Z., Clare, M. C. A., Magnusson, L., Gascon, E., Maier-Gerber, M., Janousek, M., ... Pappenberger, F. (2023, November). *The rise of data-driven weather forecasting.* arXiv. Retrieved 2024-03-02, from http://arxiv.org/abs/2307.10128 (arXiv:2307.10128 [physics]) doi: 10.48550/arXiv.2307.10128

Bi, K., Xie, L., Zhang, H., Chen, X., Gu, X., & Tian, Q. (2023, July). Accurate medium-range global weather forecasting with 3D neural networks. *Nature*, *619*(7970), 533–538. Retrieved 2024-03-02, from https://www.nature.com/articles/s41586-023-06185-3 (Publisher: Nature Publishing Group) doi: 10.1038/s41586-023-06185-3

Bocquet, M. (2023). Surrogate modeling for the climate sciences dynamics with machine learning and data assimilation. *Frontiers in Applied Mathematics and Statistics*, *9*. Retrieved 2023-08-13, from https://www.frontiersin.org/articles/10.3389/fams.2023.1133226

Bocquet, M., Brajard, J., Carrassi, A., & Bertino, L. (2020). Bayesian inference of chaotic dynamics by merging data assimilation, machine learning and expectation-maximization. *Foundations of Data Science*, *2*(1), 55. Retrieved 2022-09-21, from https://www.aimsciences.org/article/doi/10.3934/fods.2020004 (tex.ids= bocquet_bayesian_2020-1 arXiv: 2001.06270 [physics, stat] institution: Foundations of Data Science publisher: American Institute of Mathematical Sciences) doi: 10.3934/fods.2020004

Bonavita, M. (2023, November). *On some limitations of data-driven weather forecasting models.* arXiv. Retrieved 2023-12-07, from http://arxiv.org/abs/2309.08473 (arXiv:2309.08473 [physics, stat]) doi: 10.48550/arXiv.2309.08473

Bouchat, A., Hutter, N., Chanut, J., Dupont, F., Dukhovskoy, D., Garric, G., ... Wang, Q. (2022). Sea Ice Rheology Experiment (SIREx): 1. Scaling and Statistical Properties of Sea-Ice Deformation Fields. *Journal of Geophysi-*

cal Research: Oceans, *127*(4), e2021JC017667. Retrieved 2022-09-29, from https://onlinelibrary.wiley.com/doi/abs/10.1029/2021JC017667 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2021JC017667) doi: 10.1029/2021JC017667

Boutin, G., Ólason, E., Rampal, P., Regan, H., Lique, C., Talandier, C., ... Ricker, R. (2023, February). Arctic sea ice mass balance in a new coupled ice–ocean model using a brittle rheology framework. *The Cryosphere*, *17*(2), 617–638. Retrieved 2023-12-20, from https://tc.copernicus.org/articles/17/617/2023/ (Publisher: Copernicus GmbH) doi: 10.5194/tc-17-617-2023

Brunette, C., Tremblay, L. B., & Newton, R. (2022, February). A new state-dependent parameterization for the free drift of sea ice. *The Cryosphere*, *16*(2), 533–557. Retrieved 2024-02-06, from https://tc.copernicus.org/articles/16/533/2022/ (Publisher: Copernicus GmbH) doi: 10.5194/tc-16-533-2022

Chen, Y., Smith, P., Carrassi, A., Pasmans, I., Bertino, L., Bocquet, M., ... Dansereau, V. (2023, October). Multivariate state and parameter estimation with data assimilation on sea-ice models using a Maxwell-Elasto-Brittle rheology. *EGUsphere*, 1–36. Retrieved 2023-12-06, from https://egusphere.copernicus.org/preprints/2023/egusphere-2023-1809/ (Publisher: Copernicus GmbH) doi: 10.5194/egusphere-2023-1809

Cheng, S., Chen, Y., Aydoğdu, A., Bertino, L., Carrassi, A., Rampal, P., & Jones, C. K. R. T. (2023, April). Arctic sea ice data assimilation combining an ensemble Kalman filter with a novel Lagrangian sea ice model for the winter 2019–2020. *The Cryosphere*, *17*(4), 1735–1754. Retrieved 2023-09-10, from https://tc.copernicus.org/articles/17/1735/2023/ (Publisher: Copernicus GmbH) doi: 10.5194/tc-17-1735-2023

Craig, A., Valcke, S., & Coquart, L. (2017, September). Development and performance of a new version of the OASIS coupler, OASIS3-MCT_3.0. *Geoscientific Model Development*, *10*(9), 3297–3308. Retrieved 2024-03-04, from https://gmd.copernicus.org/articles/10/3297/2017/gmd-10-3297-2017.html (Publisher: Copernicus GmbH) doi: 10.5194/gmd-10-3297-2017

Dansereau, V., Weiss, J., Saramito, P., & Lattes, P. (2016, July). A Maxwell elasto-brittle rheology for sea ice modelling. *The Cryosphere*, *10*(3), 1339–1359. Retrieved 2021-11-16, from https://tc.copernicus.org/articles/10/1339/2016/ (publisher: Copernicus GmbH) doi: 10.5194/tc-10-1339-2016

De, S., & Smith, S. L. (2020, December). *Batch Normalization Biases Residual Blocks Towards the Identity Function in Deep Networks.* arXiv. Retrieved 2022-11-23, from http://arxiv.org/abs/2002.10444 (arXiv:2002.10444 [cs, stat]) doi: 10.48550/arXiv.2002.10444

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... Houlsby, N. (2021, June). *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.* arXiv. Retrieved 2024-02-19, from http://arxiv.org/abs/2010.11929 (arXiv:2010.11929 [cs]) doi: 10.48550/arXiv.2010.11929

Dueben, P. D., & Bauer, P. (2018, October). Challenges and design choices for global weather and climate models based on machine learning. *Geoscientific Model Development*, *11*(10), 3999–4009. Retrieved 2023-11-10, from https://gmd.copernicus.org/articles/11/3999/2018/ (Publisher: Copernicus GmbH) doi: 10.5194/gmd-11-3999-2018

Durand, C., Finn, T. S., Farchi, A., Bocquet, M., & Òlason, E. (2023, August). Data-driven surrogate modeling of high-resolution sea-ice thickness in the Arctic. *EGUsphere*, 1–38. Retrieved 2023-09-13, from https://egusphere.copernicus.org/preprints/2023/egusphere-2023-1384/ (Publisher: Copernicus GmbH) doi: 10.5194/egusphere-2023-1384

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., ... Rombach,

R. (2024, March). *Scaling Rectified Flow Transformers for High-Resolution Image Synthesis.* arXiv. Retrieved 2024-03-10, from http://arxiv.org/abs/2403.03206 (arXiv:2403.03206 [cs]) doi: 10.48550/arXiv.2403.03206

Everaert, M. N., Fitsios, A., Bocchio, M., Arpa, S., Süsstrunk, S., & Achanta, R. (2024). Exploiting the Signal-Leak Bias in Diffusion Models. In (pp. 4025–4034). Retrieved 2024-03-28, from https://openaccess.thecvf.com/content/WACV2024/html/Everaert_Exploiting_the_Signal-Leak_Bias_in_Diffusion_Models_WACV_2024_paper.html

Eyring, V., Bony, S., Meehl, G. A., Senior, C. A., Stevens, B., Stouffer, R. J., & Taylor, K. E. (2016, May). Overview of the Coupled Model Intercomparison Project Phase 6 (CMIP6) experimental design and organization. *Geoscientific Model Development*, *9*(5), 1937–1958. Retrieved 2024-03-04, from https://gmd.copernicus.org/articles/9/1937/2016/gmd-9-1937-2016.html (Publisher: Copernicus GmbH) doi: 10.5194/gmd-9-1937-2016

Falcon, W., Borovec, J., Wälchli, A., Eggert, N., Schock, J., Jordan, J., ... Bakhtin, A. (2020, May). *PyTorchLightning: 0.7.6 release.* Zenodo. Retrieved 2022-08-22, from https://zenodo.org/record/3828935 doi: 10.5281/zenodo.3828935

Finn, T., Durand, C., Farchi, A., Bocquet, M., Rampal, P., & Carrassi, A. (2024, April). *Dataset and neural network weights to the paper: "Generative diffusion for regional surrogate models from sea-ice simulations".* Zenodo. Retrieved 2024-04-10, from https://zenodo.org/records/10949057 doi: 10.5281/zenodo.10949057

Finn, T. S., Disson, L., Farchi, A., Bocquet, M., & Durand, C. (2023, October). Representation learning with unconditional denoising diffusion models for dynamical systems. *EGUsphere*, 1–39. Retrieved 2023-12-06, from https://egusphere.copernicus.org/preprints/2023/egusphere-2023-2261/ (Publisher: Copernicus GmbH) doi: 10.5194/egusphere-2023-2261

Finn, T. S., Durand, C., Farchi, A., Bocquet, M., Chen, Y., Carrassi, A., & Dansereau, V. (2023, July). Deep learning subgrid-scale parametrisations for short-term forecasting of sea-ice dynamics with a Maxwell elasto-brittle rheology. *The Cryosphere*, *17*(7), 2965–2991. Retrieved 2023-07-21, from https://tc.copernicus.org/articles/17/2965/2023/ (Publisher: Copernicus GmbH) doi: 10.5194/tc-17-2965-2023

Fishman, N., Klarner, L., De Bortoli, V., Mathieu, E., & Hutchinson, M. (2023, April). *Diffusion Models for Constrained Domains.* Retrieved 2024-01-12, from https://arxiv.org/abs/2304.05364v1

Fishman, N., Klarner, L., Mathieu, E., Hutchinson, M., & de Bortoli, V. (2023, November). *Metropolis Sampling for Constrained Diffusion Models.* arXiv. Retrieved 2024-01-10, from http://arxiv.org/abs/2307.05439 (arXiv:2307.05439 [cs]) doi: 10.48550/arXiv.2307.05439

Galerne, B., Gousseau, Y., & Morel, J.-M. (2011, January). Random Phase Textures: Theory and Synthesis. *IEEE Transactions on Image Processing*, *20*(1), 257–267. Retrieved 2024-03-27, from https://ieeexplore.ieee.org/abstract/document/5484588 (Conference Name: IEEE Transactions on Image Processing) doi: 10.1109/TIP.2010.2052822

Gao, Z., Shi, X., Han, B., Wang, H., Jin, X., Maddix, D. C., ... Wang, B. (2023, November). PreDiff: Precipitation Nowcasting with Latent Diffusion Models.. Retrieved 2024-03-15, from https://openreview.net/forum?id=Gh67ZZ6zkS&referrer=%5Bthe%20profile%20of%20Xingjian%20Shi%5D(%2Fprofile%3Fid%3D~Xingjian_Shi1)

Girard, L., Bouillon, S., Weiss, J., Amitrano, D., Fichefet, T., & Legat, V. (2011, January). A new modeling framework for sea-ice mechanics based on elasto-brittle rheology. *Annals of Glaciology*, *52*(57), 123–132. Retrieved 2024-03-21, from https://www.cambridge.org/core/journals/annals-of-glaciology/

article/new-modeling-framework-for-seaice-mechanics-based-on-elastobrittle-rheology/AB25948077AD472BDEC1694917CE7718 doi: 10.3189/172756411795931499

Girard, L., Weiss, J., Molines, J. M., Barnier, B., & Bouillon, S. (2009). Evaluation of high-resolution sea ice models on the basis of statistical and scaling properties of Arctic sea ice drift and deformation. *Journal of Geophysical Research: Oceans*, *114*(C8). Retrieved 2021-11-24, from https://onlinelibrary.wiley.com/doi/abs/10.1029/2008JC005182 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2008JC005182) doi: 10.1029/2008JC005182

Gregory, W., Bushuk, M., Adcroft, A., Zhang, Y., & Zanna, L. (2023). Deep Learning of Systematic Sea Ice Model Errors From Data Assimilation Increments. *Journal of Advances in Modeling Earth Systems*, *15*(10), e2023MS003757. Retrieved 2024-03-02, from https://onlinelibrary.wiley.com/doi/abs/10.1029/2023MS003757 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2023MS003757) doi: 10.1029/2023MS003757

Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017, August). *On Calibration of Modern Neural Networks.* arXiv. Retrieved 2023-11-22, from http://arxiv.org/abs/1706.04599 (arXiv:1706.04599 [cs]) doi: 10.48550/arXiv.1706.04599

Hang, T., Gu, S., Li, C., Bao, J., Chen, D., Hu, H., ... Guo, B. (2023). Efficient Diffusion Training via Min-SNR Weighting Strategy. In (pp. 7441–7451). Retrieved 2024-03-14, from https://openaccess.thecvf.com/content/ICCV2023/html/Hang_Efficient_Diffusion_Training_via_Min-SNR_Weighting_Strategy_ICCV_2023_paper.html

He, K., Zhang, X., Ren, S., & Sun, J. (2015, December). Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]*. Retrieved 2019-11-15, from http://arxiv.org/abs/1512.03385 (arXiv: 1512.03385)

Hendrycks, D., & Gimpel, K. (2016). *Gaussian error linear units (gelus).* arXiv. Retrieved from http://arxiv.org/abs/1606.08415 doi: 10.48550/arXiv.1606.08415

Herman, A., & Glowacki, O. (2012, December). Variability of sea ice deformation rates in the Arctic and their relationship with basin-scale wind forcing. *The Cryosphere*, *6*(6), 1553–1559. Retrieved 2024-03-14, from https://tc.copernicus.org/articles/6/1553/2012/ (Publisher: Copernicus GmbH) doi: 10.5194/tc-6-1553-2012

Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., ... others (2023). ERA5 hourly data on single levels from 1940 to present. *Copernicus climate change service (c3s) climate data store (cds)*, *10*(10.24381). (Publisher: ECMWF Reading, UK) doi: https://doi.org/10.24381/cds.adbb2d47

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., ... Thépaut, J.-N. (2020). The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, *146*(730), 1999–2049. Retrieved 2021-05-30, from http://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.3803 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3803) doi: https://doi.org/10.1002/qj.3803

Ho, J., Jain, A., & Abbeel, P. (2020, December). *Denoising Diffusion Probabilistic Models.* arXiv. Retrieved 2022-06-14, from http://arxiv.org/abs/2006.11239 (tex.ids= ho2020, ho2020a, ho2020b arXiv: 2006.11239 [cs, stat] number: arXiv:2006.11239)

Hoogeboom, E., Heek, J., & Salimans, T. (2023, January). *simple diffusion: End-to-end diffusion for high resolution images.* arXiv. Retrieved 2023-11-02, from http://arxiv.org/abs/2301.11093 (arXiv:2301.11093 [cs, stat]) doi: 10

.48550/arXiv.2301.11093

Hua, Z., He, Y., Ma, C., & Anderson-Frey, A. (2024, February). *Weather Prediction with Diffusion Guided by Realistic Forecast Processes.* arXiv. Retrieved 2024-03-02, from http://arxiv.org/abs/2402.06666 (arXiv:2402.06666 [physics]) doi: 10.48550/arXiv.2402.06666

Jun, H., Child, R., Chen, M., Schulman, J., Ramesh, A., Radford, A., & Sutskever, I. (2020, November). Distribution Augmentation for Generative Modeling. In *Proceedings of the 37th International Conference on Machine Learning* (pp. 5006–5019). PMLR. Retrieved 2024-02-19, from https://proceedings.mlr.press/v119/jun20a.html (ISSN: 2640-3498)

Karras, T., Aittala, M., Aila, T., & Laine, S. (2022, October). *Elucidating the Design Space of Diffusion-Based Generative Models.* arXiv. Retrieved 2022-10-27, from http://arxiv.org/abs/2206.00364 (arXiv:2206.00364 [cs, stat]) doi: 10.48550/arXiv.2206.00364

Keisler, R. (2022, February). *Forecasting Global Weather with Graph Neural Networks.* arXiv. Retrieved 2024-03-02, from http://arxiv.org/abs/2202.07575 (arXiv:2202.07575 [physics]) doi: 10.48550/arXiv.2202.07575

Kingma, D., Salimans, T., Poole, B., & Ho, J. (2021). Variational Diffusion Models. In *Advances in Neural Information Processing Systems* (Vol. 34, pp. 21696–21707). Curran Associates, Inc. Retrieved 2022-09-11, from https://proceedings.neurips.cc/paper/2021/hash/b578f2a52a0229873fefc2a4b06377fa-Abstract.html (tex.ids= kingma_variational_2021-1)

Kingma, D. P., & Ba, J. (2017, January). Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*. Retrieved 2020-09-26, from http://arxiv.org/abs/1412.6980 (arXiv: 1412.6980)

Kingma, D. P., & Gao, R. (2023, September). *Understanding Diffusion Objectives as the ELBO with Simple Data Augmentation.* arXiv. Retrieved 2023-10-05, from http://arxiv.org/abs/2303.00848 (arXiv:2303.00848 [cs, stat]) doi: 10.48550/arXiv.2303.00848

Kochkov, D., Yuval, J., Langmore, I., Norgaard, P., Smith, J., Mooers, G., ... Hoyer, S. (2023, November). *Neural General Circulation Models.* arXiv. Retrieved 2024-01-09, from http://arxiv.org/abs/2311.07222 (arXiv:2311.07222 [physics]) doi: 10.48550/arXiv.2311.07222

Kohl, G., Chen, L.-W., & Thuerey, N. (2023, September). *Turbulent Flow Simulation using Autoregressive Conditional Diffusion Models.* arXiv. Retrieved 2023-10-27, from http://arxiv.org/abs/2309.01745 (arXiv:2309.01745 [physics]) doi: 10.48550/arXiv.2309.01745

Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Alet, F., ... Battaglia, P. (2023, November). Learning skillful medium-range global weather forecasting. *Science*, *0*(0), eadi2336. Retrieved 2023-11-15, from https://www.science.org/doi/10.1126/science.adi2336 (Publisher: American Association for the Advancement of Science) doi: 10.1126/science.adi2336

Lee, S.-g., Kim, H., Shin, C., Tan, X., Liu, C., Meng, Q., ... Liu, T.-Y. (2022, February). *PriorGrad: Improving Conditional Denoising Diffusion Models with Data-Dependent Adaptive Prior.* arXiv. Retrieved 2023-12-14, from http://arxiv.org/abs/2106.06406 (arXiv:2106.06406 [cs, eess, stat]) doi: 10.48550/arXiv.2106.06406

Leinonen, J., Hamann, U., Nerini, D., Germann, U., & Franch, G. (2023, April). *Latent diffusion models for generative precipitation nowcasting with accurate uncertainty quantification.* arXiv. Retrieved 2024-01-19, from http://arxiv.org/abs/2304.12891 (arXiv:2304.12891 [physics]) doi: 10.48550/arXiv.2304.12891

Li, L., Carver, R., Lopez-Gomez, I., Sha, F., & Anderson, J. (2023, October).

SEEDS: Emulation of Weather Forecast Ensembles with Diffusion Models. arXiv. Retrieved 2023-11-30, from http://arxiv.org/abs/2306.14066 (arXiv:2306.14066 [physics]) doi: 10.48550/arXiv.2306.14066

Li, M., Qu, T., Yao, R., Sun, W., & Moens, M.-F. (2023, October). Alleviating Exposure Bias in Diffusion Models through Sampling with Shifted Time Steps.. Retrieved 2024-03-28, from https://openreview.net/forum?id=ZSD3MloKe6

Lin, S., Liu, B., Li, J., & Yang, X. (2024). Common Diffusion Noise Schedules and Sample Steps Are Flawed. In (pp. 5404–5411). Retrieved 2024-03-28, from https://openaccess.thecvf.com/content/WACV2024/html/Lin_Common_Diffusion_Noise_Schedules_and_Sample_Steps_Are_Flawed_WACV_2024_paper.html

Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M., & Le, M. (2023, February). *Flow Matching for Generative Modeling.* arXiv. Retrieved 2024-03-14, from http://arxiv.org/abs/2210.02747 (arXiv:2210.02747 [cs, stat]) doi: 10.48550/arXiv.2210.02747

Liu, X., Gong, C., & Liu, Q. (2022, September). *Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow.* arXiv. Retrieved 2024-03-14, from http://arxiv.org/abs/2209.03003 (arXiv:2209.03003 [cs]) doi: 10.48550/arXiv.2209.03003

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022, March). *A ConvNet for the 2020s.* arXiv. Retrieved 2022-06-13, from http://arxiv.org/abs/2201.03545

Loshchilov, I., & Hutter, F. (2019, January). *Decoupled Weight Decay Regularization.* arXiv. Retrieved 2024-02-19, from http://arxiv.org/abs/1711.05101 (arXiv:1711.05101 [cs, math]) doi: 10.48550/arXiv.1711.05101

Luo, G., Dunlap, L., Park, D. H., Holynski, A., & Darrell, T. (2023, May). *Diffusion Hyperfeatures: Searching Through Time and Space for Semantic Correspondence.* arXiv. Retrieved 2023-06-15, from http://arxiv.org/abs/2305.14334 (arXiv:2305.14334 [cs]) doi: 10.48550/arXiv.2305.14334

Madec, G. (2008). *NEMO ocean engine* (Project report No. 1288-1619). Institut Pierre-Simon Laplace (IPSL). (Series: 27)

Mardani, M., Brenowitz, N., Cohen, Y., Pathak, J., Chen, C.-Y., Liu, C.-C., ... Pritchard, M. (2023, September). *Generative Residual Diffusion Modeling for Km-scale Atmospheric Downscaling.* arXiv. Retrieved 2023-10-30, from http://arxiv.org/abs/2309.15214 (arXiv:2309.15214 [physics]) doi: 10.48550/arXiv.2309.15214

Marsan, D., Stern, H., Lindsay, R., & Weiss, J. (2004, October). Scale Dependence and Localization of the Deformation of Arctic Sea Ice. *Physical Review Letters*, *93*(17), 178501. Retrieved 2024-03-13, from https://link.aps.org/doi/10.1103/PhysRevLett.93.178501 (Publisher: American Physical Society) doi: 10.1103/PhysRevLett.93.178501

Minderer, M., Djolonga, J., Romijnders, R., Hubis, F., Zhai, X., Houlsby, N., ... Lucic, M. (2021). Revisiting the Calibration of Modern Neural Networks. In *Advances in Neural Information Processing Systems* (Vol. 34, pp. 15682–15694). Curran Associates, Inc. Retrieved 2024-03-28, from https://proceedings.neurips.cc/paper/2021/hash/8420d359404024567b5aefda1231af24-Abstract.html

Mohamed, S., & Lakshminarayanan, B. (2016, October). Learning in Implicit Generative Models. *arXiv:1610.03483 [cs, stat]*. Retrieved 2019-06-27, from http://arxiv.org/abs/1610.03483 (arXiv: 1610.03483)

Moisan, L. (2011, February). Periodic Plus Smooth Image Decomposition. *Journal of Mathematical Imaging and Vision*, *39*(2), 161–179. Retrieved 2024-03-27, from https://doi.org/10.1007/s10851-010-0227-1 doi: 10.1007/s10851-010-0227-1

Nguyen, A., Yosinski, J., & Clune, J. (2015). Deep Neural Networks Are Easily Fooled: High Confidence Predictions for Unrecognizable Images. In (pp. 427–436). Retrieved 2024-03-28, from https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Nguyen_Deep_Neural_Networks_2015_CVPR_paper.html

Odena, A., Dumoulin, V., & Olah, C. (2016, October). Deconvolution and Checkerboard Artifacts. *Distill*, *1*(10), e3. Retrieved 2022-08-18, from http://distill.pub/2016/deconv-checkerboard doi: 10.23915/distill.00003

Palmer, T. (2022, April). *A Vision for Numerical Weather Prediction in 2030.* arXiv. Retrieved 2024-03-15, from http://arxiv.org/abs/2007.04830 (arXiv:2007.04830 [physics]) doi: 10.48550/arXiv.2007.04830

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 32* (pp. 8024–8035). Curran Associates, Inc. Retrieved from http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Pathak, J., Subramanian, S., Harrington, P., Raja, S., Chattopadhyay, A., Mardani, M., ... Anandkumar, A. (2022, February). *FourCastNet: A Global Data-driven High-resolution Weather Model using Adaptive Fourier Neural Operators.* arXiv. Retrieved 2023-08-13, from http://arxiv.org/abs/2202.11214 (arXiv:2202.11214 [physics]) doi: 10.48550/arXiv.2202.11214

Peebles, W., & Xie, S. (2023, March). *Scalable Diffusion Models with Transformers.* arXiv. Retrieved 2024-02-19, from http://arxiv.org/abs/2212.09748 (arXiv:2212.09748 [cs]) doi: 10.48550/arXiv.2212.09748

Perez, E., Strub, F., de Vries, H., Dumoulin, V., & Courville, A. (2017, December). *FiLM: Visual Reasoning with a General Conditioning Layer.* arXiv. Retrieved 2023-02-28, from http://arxiv.org/abs/1709.07871 (arXiv:1709.07871 [cs, stat]) doi: 10.48550/arXiv.1709.07871

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., ... Rombach, R. (2023, July). *SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis.* arXiv. Retrieved 2023-10-05, from http://arxiv.org/abs/2307.01952 (arXiv:2307.01952 [cs]) doi: 10.48550/arXiv.2307.01952

Price, I., Sanchez-Gonzalez, A., Alet, F., Ewalds, T., El-Kadi, A., Stott, J., ... Willson, M. (2023, December). *GenCast: Diffusion-based ensemble forecasting for medium-range weather.* arXiv. Retrieved 2024-01-09, from http://arxiv.org/abs/2312.15796 (arXiv:2312.15796 [physics]) doi: 10.48550/arXiv.2312.15796

Pulkkinen, S., Nerini, D., Pérez Hortal, A. A., Velasco-Forero, C., Seed, A., Germann, U., & Foresti, L. (2019, October). Pysteps: an open-source Python library for probabilistic precipitation nowcasting (v1.0). *Geoscientific Model Development*, *12*(10), 4185–4219. Retrieved 2024-03-27, from https://gmd.copernicus.org/articles/12/4185/2019/ (Publisher: Copernicus GmbH) doi: 10.5194/gmd-12-4185-2019

Rampal, P., Bouillon, S., Ólason, E., & Morlighem, M. (2016, May). neXtSIM: a new Lagrangian sea ice model. *The Cryosphere*, *10*(3), 1055–1073. Retrieved 2022-06-05, from https://tc.copernicus.org/articles/10/1055/2016/ (publisher: Copernicus GmbH) doi: 10.5194/tc-10-1055-2016

Rampal, P., Dansereau, V., Olason, E., Bouillon, S., Williams, T., Korosov, A., & Samaké, A. (2019, September). On the multi-fractal scaling properties of sea ice deformation. *The Cryosphere*, *13*(9), 2457–2474. Retrieved 2023-10-15, from https://tc.copernicus.org/articles/13/2457/2019/ (Publisher: Copernicus GmbH) doi: 10.5194/tc-13-2457-2019

Rampal, P., Weiss, J., Marsan, D., Lindsay, R., & Stern, H. (2008). Scaling properties of sea ice deformation from buoy dispersion analysis. *Journal of Geophysical Research: Oceans*, *113*(C3). Retrieved 2024-03-13, from https://onlinelibrary.wiley.com/doi/abs/10.1029/2007JC004143 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2007JC004143) doi: 10.1029/2007JC004143

Rasp, S., & Thuerey, N. (2021, February). Data-driven medium-range weather prediction with a Resnet pretrained on climate simulations: A new model for WeatherBench. *Journal of Advances in Modeling Earth Systems*, *13*(2). Retrieved 2021-05-21, from http://arxiv.org/abs/2008.08626 (arXiv: 2008.08626) doi: 10.1029/2020MS002405

Ravuri, S., Lenc, K., Willson, M., Kangin, D., Lam, R., Mirowski, P., ... Mohamed, S. (2021, September). Skilful precipitation nowcasting using deep generative models of radar. *Nature*, *597*(7878), 672–677. Retrieved 2022-05-18, from http://www.nature.com/articles/s41586-021-03854-z (Number: 7878 Publisher: Nature Publishing Group) doi: 10.1038/s41586-021-03854-z

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-Resolution Image Synthesis With Latent Diffusion Models. In (pp. 10684–10695). Retrieved 2023-11-16, from https://openaccess.thecvf.com/content/CVPR2022/html/Rombach_High-Resolution_Image_Synthesis_With_Latent_Diffusion_Models_CVPR_2022_paper.html

Ronneberger, O., Fischer, P., & Brox, T. (2015, May). U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv:1505.04597 [cs]*. Retrieved 2022-03-08, from http://arxiv.org/abs/1505.04597 (arXiv: 1505.04597)

Rybkin, O., Daniilidis, K., & Levine, S. (2020, June). Simple and Effective VAE Training with Calibrated Decoders. *arXiv:2006.13202 [cs, eess, stat]*. Retrieved 2020-07-03, from http://arxiv.org/abs/2006.13202 (arXiv: 2006.13202)

Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D. J., & Norouzi, M. (2022). Image Super-Resolution Via Iterative Refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–14. (Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence) doi: 10.1109/TPAMI.2022.3204461

Salimans, T., & Ho, J. (2022, June). *Progressive Distillation for Fast Sampling of Diffusion Models.* arXiv. Retrieved 2022-10-12, from http://arxiv.org/abs/2202.00512 (arXiv:2202.00512 [cs, stat]) doi: 10.48550/arXiv.2202.00512

Seed, A. W., Pierce, C. E., & Norman, K. (2013). Formulation and evaluation of a scale decomposition-based stochastic precipitation nowcast scheme. *Water Resources Research*, *49*(10), 6624–6641. Retrieved 2024-03-20, from https://onlinelibrary.wiley.com/doi/abs/10.1002/wrcr.20536 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wrcr.20536) doi: 10.1002/wrcr.20536

Sohl-Dickstein, J., Weiss, E. A., Maheswaranathan, N., & Ganguli, S. (2015, November). Deep Unsupervised Learning using Nonequilibrium Thermodynamics. *arXiv:1503.03585 [cond-mat, q-bio, stat]*. Retrieved 2022-02-24, from http://arxiv.org/abs/1503.03585 (arXiv: 1503.03585)

Song, J., Meng, C., & Ermon, S. (2020, October). Denoising Diffusion Implicit Models.. Retrieved 2024-02-17, from https://openreview.net/forum?id=St1giarCHLP

Song, Y., Dhariwal, P., Chen, M., & Sutskever, I. (2023, May). *Consistency Models.* arXiv. Retrieved 2023-12-21, from http://arxiv.org/abs/2303.01469 (arXiv:2303.01469 [cs, stat]) doi: 10.48550/arXiv.2303.01469

Song, Y., Durkan, C., Murray, I., & Ermon, S. (2021, October). *Maximum Likelihood Training of Score-Based Diffusion Models.* arXiv. Retrieved 2023-09-05, from http://arxiv.org/abs/2101.09258 (arXiv:2101.09258 [cs, stat]) doi:

10.48550/arXiv.2101.09258

Song, Y., & Ermon, S. (2020a, October). *Generative Modeling by Estimating Gradients of the Data Distribution.* arXiv. Retrieved 2023-11-30, from http://arxiv.org/abs/1907.05600 (arXiv:1907.05600 [cs, stat]) doi: 10.48550/arXiv.1907.05600

Song, Y., & Ermon, S. (2020b). Improved Techniques for Training Score-Based Generative Models. In *Advances in Neural Information Processing Systems* (Vol. 33, pp. 12438–12448). Curran Associates, Inc. Retrieved 2024-04-08, from https://proceedings.neurips.cc/paper/2020/hash/92c3b916311a5517d9290576e3ea37ad-Abstract.html

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., & Poole, B. (2021, February). *Score-Based Generative Modeling through Stochastic Differential Equations.* arXiv. Retrieved 2023-05-03, from http://arxiv.org/abs/2011.13456 (arXiv:2011.13456 [cs, stat]) doi: 10.48550/arXiv.2011.13456

Talandier, C., & Lique, C. (2021, December). *CREG025.L75-NEMO_r3.6.0.* Zenodo. Retrieved 2024-03-04, from https://zenodo.org/records/5802028 doi: 10.5281/zenodo.5802028

Thorndike, A. S., & Colony, R. (1982). Sea ice motion in response to geostrophic winds. *Journal of Geophysical Research: Oceans*, *87*(C8), 5845–5852. Retrieved 2024-02-06, from https://onlinelibrary.wiley.com/doi/abs/10.1029/JC087iC08p05845 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/JC087iC08p05845) doi: 10.1029/JC087iC08p05845

Vahdat, A., Kreis, K., & Kautz, J. (2021, December). *Score-based Generative Modeling in Latent Space.* arXiv. Retrieved 2022-09-11, from http://arxiv.org/abs/2106.05931 (arXiv:2106.05931 [cs, stat]) doi: 10.48550/arXiv.2106.05931

Valcke, S. (2013, March). The OASIS3 coupler: a European climate modelling community software. *Geoscientific Model Development*, *6*(2), 373–388. Retrieved 2024-03-04, from https://gmd.copernicus.org/articles/6/373/2013/ (Publisher: Copernicus GmbH) doi: 10.5194/gmd-6-373-2013

Van Rossum, G. (1995, May). *Python tutorial, Technical Report CS-R9526* (Tech. Rep.). Amsterdam: Centrum voor Wiskunde en Informatica (CWI).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017, December). Attention Is All You Need. *arXiv:1706.03762 [cs]*. Retrieved 2019-11-20, from http://arxiv.org/abs/1706.03762 (tex.ids: vaswani_attention_2017-1 arXiv: 1706.03762)

Vincent, P. (2011, July). A Connection Between Score Matching and Denoising Autoencoders. *Neural Computation*, *23*(7), 1661–1674. (Conference Name: Neural Computation) doi: 10.1162/NECO_a_00142

Wan, Z. Y., Baptista, R., Chen, Y.-f., Anderson, J., Boral, A., Sha, F., & Zepeda-Núñez, L. (2023, October). *Debias Coarsely, Sample Conditionally: Statistical Downscaling through Optimal Transport and Probabilistic Diffusion Models.* arXiv. Retrieved 2023-12-12, from http://arxiv.org/abs/2305.15618 (arXiv:2305.15618 [physics]) doi: 10.48550/arXiv.2305.15618

Wang, X., Wang, R., Hu, N., Wang, P., Huo, P., Wang, G., ... Song, J. (2024, February). *XiHe: A Data-Driven Model for Global Ocean Eddy-Resolving Forecasting.* arXiv. Retrieved 2024-03-02, from http://arxiv.org/abs/2402.02995 (arXiv:2402.02995 [physics] version: 2) doi: 10.48550/arXiv.2402.02995

Weiss, J., & Schulson, E. M. (2009, October). Coulombic faulting from the grain scale to the geophysical scale: lessons from ice. *Journal of Physics D: Applied Physics*, *42*(21), 214017. Retrieved 2024-03-04, from https://dx.doi.org/10.1088/0022-3727/42/21/214017 doi: 10.1088/0022-3727/42/21/214017

Wu, T., Si, C., Jiang, Y., Huang, Z., & Liu, Z. (2023, December). *FreeInit: Bridg-*

ing Initialization Gap in Video Diffusion Models. arXiv. Retrieved 2024-03-28, from http://arxiv.org/abs/2312.07537 (arXiv:2312.07537 [cs]) doi: 10.48550/arXiv.2312.07537

Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., ... Liu, T. (2020, November). On Layer Normalization in the Transformer Architecture. In *International Conference on Machine Learning* (pp. 10524–10533). PMLR. Retrieved 2021-05-27, from http://proceedings.mlr.press/v119/xiong20b.html (ISSN: 2640-3498)

Xiong, W., Xiang, Y., Wu, H., Zhou, S., Sun, Y., Ma, M., & Huang, X. (2023, August). *AI-GOMS: Large AI-Driven Global Ocean Modeling System.* arXiv. Retrieved 2024-03-15, from http://arxiv.org/abs/2308.03152 (arXiv:2308.03152 [physics]) doi: 10.48550/arXiv.2308.03152

Yadan, O. (2019). *Hydra - A framework for elegantly configuring complex applications.* Retrieved from https://github.com/facebookresearch/hydra (tex.howpublished: Github)

Zhong, X., Chen, L., Liu, J., Lin, C., Qi, Y., & Li, H. (2023, October). *FuXi-Extreme: Improving extreme rainfall and wind forecasts with diffusion model.* arXiv. Retrieved 2024-03-15, from http://arxiv.org/abs/2310.19822 (arXiv:2310.19822 [physics, stat]) doi: 10.48550/arXiv.2310.19822

Ólason, E., Boutin, G., Korosov, A., Rampal, P., Williams, T., Kimmritz, M., ... Samaké, A. (2022). A New Brittle Rheology and Numerical Framework for Large-Scale Sea-Ice Models. *Journal of Advances in Modeling Earth Systems*, *14*(8), e2021MS002685. Retrieved 2023-12-20, from https://onlinelibrary.wiley.com/doi/abs/10.1029/2021MS002685 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2021MS002685) doi: 10.1029/2021MS002685