Title

Accelerating phase field simulations through a hybrid adaptive Fourier neural operator with U-net backbone

Journal

npj Computational Materials, 11(1)

ISSN

2057-3960

Authors

Bonneville, Christophe

Bieberdorf, Nathan

Hegde, Arun

et al.

Publication Date

2025

DOI

10.1038/s41524-024-01488-z

npj | computational materials

Article

Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences

https://doi.org/10.1038/s41524-024-01488-z

Accelerating phase field simulations through a hybrid adaptive Fourier neural operator with U-net backbone

Christophe Bonneville[1] ✉, Nathan Bieberdorf[2,3], Arun Hegde[1], Mark Asta[2,3], Habib N. Najm[1], Laurent Capolungo[4] & Cosmin Safta[1]

Prolonged contact between a corrosive liquid and metal alloys can cause progressive dealloying. For one such process as liquid-metal dealloying (LMD), phase field models have been developed to understand the mechanisms leading to complex morphologies. However, the LMD governing equations in these models often involve coupled non-linear partial differential equations (PDE), which are challenging to solve numerically. In particular, numerical stiffness in the PDEs requires an extremely refined time step size (on the order of 10^{-12}s or smaller). This computational bottleneck is especially problematic when running LMD simulation until a late time horizon is required. This motivates the development of surrogate models capable of leaping forward in time, by skipping several consecutive time steps at-once. In this paper, we propose a U-shaped adaptive Fourier neural operator (U-AFNO), a machine learning (ML) based model inspired by recent advances in neural operator learning. U-AFNO employs U-Nets for extracting and reconstructing local features within the physical fields, and passes the latent space through a vision transformer (ViT) implemented in the Fourier space (AFNO). We use U-AFNOs to learn the dynamics of mapping the field at a current time step into a later time step. We also identify global quantities of interest (QoI) describing the corrosion process (e.g., the deformation of the liquid-metal interface, lost metal, etc.) and show that our proposed U-AFNO model is able to accurately predict the field dynamics, in spite of the chaotic nature of LMD. Most notably, our model reproduces the key microstructure statistics and QoIs with a level of accuracy on par with the high-fidelity numerical solver, while achieving a significant 11, 200 × speed-up on a high-resolution grid when comparing the computational expense per time step. Finally, we also investigate the opportunity of using hybrid simulations, in which we alternate forward leaps in time using the U-AFNO with high-fidelity time stepping. We demonstrate that while advantageous for some surrogate model design choices, our proposed U-AFNO model in fully auto-regressive settings consistently outperforms hybrid schemes.

Phase field modeling is a powerful tool for simulating the microstructure and morphological evolution of material systems subjected to various driving forces (e.g., chemical, mechanical, and electrical). The phase field method uses a "diffuse-interface" approach to track phases, interfaces, as well as conserved quantities (e.g., mole fraction of an element). This allows for the temporal evolution of the system to be defined everywhere by a set of partial differential equations (PDEs) representing well-known continuum and interfacial kinetic processes[1–3]. While this avoids the numerical complexity associated with explicitly tracking and updating a sharp interface in addition to providing a route to describing interface reactions, phase field modeling can still require a high-resolution spatial discretization to accurately resolve nano-scale interfacial phenomena. This can render large length- and time-scale phase field simulations prohibitively computationally expensive, thereby limiting the use of phase field modeling to coarse grain microstructure evolution.

[1]Sandia National Laboratories, Livermore, CA, 94550, USA. [2]Materials Sciences Division, Lawrence Berkeley National Laboratory, Berkeley, CA, 94720, USA. [3]Department of Materials Science and Engineering, University of California, Berkeley, CA, 94720, USA. [4]Los Alamos National Laboratory, Los Alamos, NM, 87544, USA. ✉e-mail: cpbonne@sandia.gov

Several approaches have been developed to circumvent the computational expense of phase field simulations. One approach is to use adaptive meshing to set the spatial discretization to be very fine near the interface and much coarser in the bulk phases where longer-range transport processes take place[4]. However, this is not guaranteed to be advantageous in situations where the interface morphology is rapidly changing, since frequent re-initialization of the adaptive mesh would be required. Another approach is to artificially broaden interfaces—from the nanometer-scale to, for example, the micron-scale—and thereby resolve the system with a relatively coarse mesh at interfaces as well as within bulk phases. In this approach, transport equations and effective kinetic parameters are rederived to ensure the phase field simulations evolve consistently with a realistic thin-interface system[5–7]. But any approach using a micron-scale interface is inherently limited in its ability to resolve morphological features that initiate at the nano-scale. And so, while these approaches are valuable for a wide variety of phase field simulations, they are not necessarily suitable for situations with fine-scale, dynamic morphologies.

One such example is the dealloying corrosion of metals, where the infiltration of a corrosive agent into the alloy can rapidly lead to morphologically complex metal structures[8–13]. Understanding corrosion-induced dealloying is particularly critical in energetic applications, such as next-generation molten-salt nuclear reactor design[14,15]. In these aforementioned metal structures, preferential dissolution of the alloy precedes an instability that initiates nano-scale, compositionally heterogeneous features along the interface[9]. In a coupled manner, these features grow while the corrosive agent infiltrates as individual channels into the alloy. The reemergence of the nano-scale interfacial instability can cause these channels to bifurcate and advance tortuously into the alloy, increasing the alloy's topological complexity[16]. Simultaneously, capillary forces act to coarsen the dealloyed structure and reduce local curvatures and overall topological complexity[17]. Competition among these processes sets up a dynamic morphology that underlies the rate of corrosion into the base metal. Phase field simulations for dealloying corrosion, specifically for liquid-metal dealloying, have successfully reproduced the experimentally observed morphologies and uncovered the key mechanisms by which these structures evolve[16,18–20]. To simulate representative domains and time-scales with high resolution, these phase field simulations have generally relied on using high-performance computing resources and advanced time-integration schemes[16,20]. However, the length- and time-scales of such simulations are several orders of magnitude below experiments, so that validation requires extrapolation using scaling laws.

The computational cost of numerical simulations such as the ones involved in liquid-metal dealloying is a major bottleneck. This is particularly true when a large number of forward simulations are required, like in uncertainty quantification[21,22], inverse problems[21,23–25], and design optimization[26–28] applications. This computational challenge has historically motivated the development of surrogate models that are faster than the high-fidelity simulations, albeit being less accurate. This drop in accuracy can be an acceptable compromise when the time gain is significant. A particular class of surrogate models for PDEs, such as reduced-order-models (ROMs) aim to decrease computational cost by reducing the dimensionality of the problem. This can be done by projecting snapshots of high-fidelity simulations into a lower-dimensional latent space. Linear projection methods, such as proper-orthogonal decomposition (POD), are widely used for building accurate ROMs[29–33], and have been successfully applied to a variety of problems, e.g., in fluid mechanics[34–39], structural dynamics[40–42], and control systems[43–45]. More recently, employing machine learning (ML) approaches to compress high-fidelity data snapshots has gained significant interest. Indeed, non-linear mappings, such as neural networks, can yield better performance, especially in advection-dominated problems[46–49].

With recent advances in deep learning and computer vision, a second class of surrogate models for PDEs has emerged, namely neural operators[50–52]. Neural operators learn mappings between function spaces through a resolution-invariant tensor-to-tensor regression. Popular neural operator architectures include Fourier neural operator (FNO)[51], DeepONets[52], and more recently, Laplace neural operator (LNO)[53] and convolutional neural operator (CNO)[54] constructions. In time-dependent problems, neural operators can be employed to predict the physical state at a future time step, given the current state. Often, this strategy is used auto-regressively, i.e., the operator output becomes its own input, allowing for predicting consecutive time steps. This strategy (sometimes referred to as auto-regressive roll-out) is analogous to conventional explicit time integrators. Auto-regressive neural operator roll-outs have the convenient advantage of being naturally well suited for non-parametric initial conditions (e.g., random initial field), unlike more traditional ROM approaches. As such, neural operators have been applied to a variety of problems, including fluid dynamics[51,55,56], electromagnetics and electro-convection[57,58], fracture mechanics[59], weather prediction[60,61], phase field modeling[62,63], and plasma physics[64].

While neural operators have shown great accuracy in the aforementioned papers, recent work suggests that this accuracy can be increased further by combining key neural operator building blocks with U-Nets[65] and Vision transformers (ViTs)[66]. Recent papers proposed to employ the encoder-decoder architecture of a U-Net as a feature extraction mechanism, in combination with a neural operator bottleneck[63,67,68]. Building on this work, ViTO[69] and DiTTO[70] proposed a neural operator architecture based on ViTs to capture interdependence relations within the input field compressed features, where the compression is obtained by a U-Net. In this paper, we propose to build on these ideas by strategically placing an adaptive Fourier neural operator (AFNO)[71] at the intermediate bottleneck of a U-Net. An AFNO is a special implementation of a ViT, where the attention mechanism specific to transformer architectures is computed in the Fourier space. This idea, initially introduced with FNO[51], allows for significant computational efficiency gains during training. We call our proposed model U-AFNO, and show that it is able to capture the phase field predictions of dealloying morphology and rate with excellent accuracy, in spite of the highly chaotic underlying physics.

LMD phase fields can be used to extract local and global quantities of interest (QoIs). Such QoIs, describing, for example, the topology of corroded/dealloyed metal interfaces or the chemical composition within the alloy, may be crucial for decision-making, optimization, or uncertainty quantification purposes. In the present study, we employ several physically relevant QoIs to describe the dealloying process at both local and global scales. Furthermore, we show that our proposed U-AFNO model is accurate, not only according to state-of-the-art error metrics[50–52,72,73], but also in correctly reproducing the aforementioned QoIs, unlike other commonly used neural operator models. Lastly, we also investigate the performance of blending our proposed surrogate model with high-fidelity simulations. In a recent ML-based phase field surrogate study[73], Oommen et al. proposed to augment a U-Net-based model with alternate high-fidelity solver time stepping. After each leap in time through a U-Net forward pass, several of the following time steps are computed with the high-fidelity solver. This hybrid approach still partially relies on the high-fidelity solver, and is consequently significantly slower than fully auto-regressive roll-outs, but allows for greater (and tunable) accuracy. In this paper, we investigate the performance of U-AFNOs with and without hybrid high-fidelity time stepping. In particular, we show that even without hybrid time stepping augmentation, our proposed U-AFNO model can outperform other augmented models, such as U-Nets.

The contributions of this paper are, therefore, twofold: (1) a novel surrogate model for LMD phase field simulation to accurately reproduce the microstructure dynamics and the QoIs describing dealloying processes and (2) a thorough performance analysis of hybrid ML surrogates blended with high-fidelity solvers. Note that we employ the term high-fidelity to refer to simulations obtained from traditional PDE solvers (sometimes referred to as full-order models in the literature), as opposed to surrogate models. In the section "Results and discussion", we first present and discuss the performance of our proposed model in auto-regressive settings (section "Performance of the fully auto-regressive U-AFNO surrogate"), and investigate the

effects of hybrid time stepping (section "Investigating the effects of hybrid time stepping"). Then, in the section "High-fidelity phase field model for liquid-metal dealloying", we describe the mathematical model for LMD processes and provide brief details of our high-fidelity solver implementation. In section "Leaping forward in time with U-AFNO", we introduce key technical details of our proposed U-AFNO surrogate model, and in section "Phase field quantities of interest", we present methods for computing the various local and global QoIs employed in the paper.

Results and discussion

Training

The training set comprises 87 high-fidelity simulations, each containing 100 field snapshots spanning from time $t = 1$ μs to $t = 6$ μs. Our proposed model, the U-AFNO is trained to leap 50,000 (high-fidelity) time steps ahead, i.e., $\Delta\tau = 5 \cdot 10^4 \times \Delta t = 0.05$ μs every forward pass (where Δt is the time step employed in the high-fidelity solver). The test set comprises 20 high-fidelity simulations of similar form. The model is trained for 20 epochs, using the Adam optimizer[74] with a learning rate of 10^{-4}. The code is implemented with Pytorch[75], and the AFNO bottleneck is based on the FourCastNet implementation[60]. For a baseline comparison, we also train a standard U-Net, an AFNO (with no U-Net backbone), and an FNO. Thus, the effects of using a U-Net with and without an AFNO bottleneck, and using an AFNO with and without a U-Net encoder/decoder can be carefully investigated in an ablation study. The high-idelity training, validation, and testing dataset is generated using 128 cores on a Linux cluster (32-socket Intel Xeon Platinum 8176, 2.10 GHz, 12TB DDR4). Each simulation took, on average, 43 h to complete, and generating all the datasets took us about 2 months. All our models are trained on a single Nvidia A100 80GB GPU. The U-AFNO training took 35 h. All the models are trained using a standard ℓ^2 relative error loss[50], averaged across the phase and species field.

In this paper, we run test surrogate simulations for up to time step 6×10^6 (i.e., $t_{\max} = 6\mu s$). We initialize the surrogate simulations with time integration of the high-fidelity model (in the following sections, $n_{\text{in}} = 1{,}000{,}000$ steps). This allows for the instabilities at the interface between the phases to grow and be sufficiently distinguishable across different initial conditions. We note that these instabilities reemerge at later stages of the simulation, and so the surrogate simulations will still see, and be required to emulate these morphological instabilities that can rapidly change the simulation trajectories.

Extracting informative quantities of interest

An essential motivation for developing physical simulations (whether high-fidelity or surrogate ones) is that it is often a necessary first step for extracting informative quantities of interest (QoIs) from the physical fields. These QoIs, computed locally and/or globally across the spatial field and over time, may be crucial for optimization or uncertainty quantification and decision-making purposes. In some applications, the extracted QoIs may be more important than the simulated physical field itself. In dealloying corrosion, some QoIs are particularly informative, such as the local curvature[10,12,16,17,76] and the extent (perimeter)[77] of the solid-liquid interface, the penetration depth of the corrosive liquid into the metal[11,19], and the amount of remaining metal alloy, and species throughout the corrosion process. In the following subsections, we evaluate the performance of our model not just for predicting physically consistent phase and species fields over time, but also in recovering the correct QoI dynamics (and associated uncertainty envelope across different initial conditions).

Error metrics

We consider several error metrics to assess the accuracy of our model. While the relative error between the ground truth (i.e., high-fidelity simulation) and the predicted field is a common error metric in the neural operator literature[50–52], it is not directly applicable here due to the chaotic nature of LMD simulations. In chaotic systems, comparing invariant statistical properties, such as field auto-correlations, is more meaningful and informative[72]. In phase field simulations, the field spatial auto-correlation is equivalent to the probability that two points in the field will be part of the same phase[78]. As such, the spatial auto-correlation relative error has been shown to be more relevant for describing the microstructure statistics than other relative error metrics, and has been recently used in neural operator-based phase field models[73]. The spatial auto-correlation relative error is defined as:

$$e_{\text{AC}}(\mathbf{u}, \hat{\mathbf{u}}, t) = \frac{||S_{\hat{\mathbf{u}}\hat{\mathbf{u}}}(\boldsymbol{r}, t) - S_{\mathbf{uu}}(\boldsymbol{r}, t)||_2}{||S_{\mathbf{uu}}(\boldsymbol{r}, t)||_2} \tag{1}$$

where $\mathbf{u}$ can represent either a ground truth species mole fraction or phase field variable at time t, and $\hat{\mathbf{u}}$ represents the corresponding surrogate prediction. S is the corresponding spatial auto-correlation function and $\boldsymbol{r}$ is a vector connecting any two points in the spatial field. Note that, for baseline comparison, we also compute the average auto-correlation relative *discrepancy* between each pair of ground truth simulations contained in the test set. In this later case, equation (1) is used as well, but we employ the term discrepancy rather than error since we only compare ground truth simulations together. This discrepancy metric allows for meaningful judgment of the difference between ground truth dynamics and our surrogate, compared to a high-fidelity simulation with a different random initial condition. To assess the ability of our model to predict the key QoIs described in section "Phase field quantities of interest", we also consider a second set of error metrics, the relative error of each QoI across all time steps. For example, we define the mean-curvature relative error as:

$$e(\boldsymbol{\mu_k}, \hat{\boldsymbol{\mu}}_{\boldsymbol{k}}) = \frac{||\hat{\boldsymbol{\mu}}_{\boldsymbol{k}} - \boldsymbol{\mu_k}||_2}{||\boldsymbol{\mu_k}||_2} \tag{2}$$

where $\boldsymbol{\mu_k}$ is the vector of ground truth mean-curvature at each time step: $\boldsymbol{\mu_k} = (\mu_k(t_0), \ldots, \mu_k(t_{\max}))$ (and $\hat{\boldsymbol{\mu}}_{\boldsymbol{k}}$ is the equivalent vector for the surrogate prediction). In the results shown shortly, $t_0 = 0$ μs and $t_{\max} = 6\mu s$. In a similar fashion, we define the relative error for the curvature standard deviation, the interface perimeter, the maximum penetration depth, the mean ligament height, and the remaining metal mass.

Performance of the fully auto-regressive U-AFNO surrogate

We first investigate the performance of our model in a fully auto-regressive setting. That is, the U-AFNO is called in a recurrent fashion, where the output at any time is immediately re-used as input to predict the field at a later time (without further processing).

$$\mathbf{u}(t + \Delta\tau) \approx \mathcal{G}(\mathbf{u}(t)), \quad \forall t, \tag{3}$$

Where $\mathbf{u}$ represents the physical field and includes the phase field ϕ and the two species field c_A and c_B. $\mathcal{G}$ represents the machine learning surrogate. Figure 1 shows the solid phase ϕ and species A and B predicted with a U-AFNO-B/1 surrogate (where B/1 refers to a 1×1 patch size), at different moments in time from $t = 0$ μs to $t = 6$ μs. The total number of auto-regressive forward passes through the U-AFNO is 100. The LMD physics is chaotic, and contaminating the initial species fields with white noise may lead to widely different microstructure topologies at the end of the simulation (as shown in Fig. 2). Given such chaotic behavior, the surrogate model cannot reproduce the exact ligament shapes, but it captures patterns that are visually consistent with them. Specifically, based on visual inspection, the ligaments of the solid phase exhibit satisfactory thickness, height, and size. Further, the liquid phase penetrates into the metal in a manner consistent with the ground truth results, making its way steadily toward the bottom boundary. The dynamics of both species are similarly consistent; especially at the solid-liquid interface where species A tends to segregate to the growing ligaments while species B is dissolved into the liquid. This behavior, consistent with dealloying phenomena, is correctly reproduced here. This empirical observation can be confirmed by looking at the associated auto-correlation maps, represented in Fig. 3. The predicted

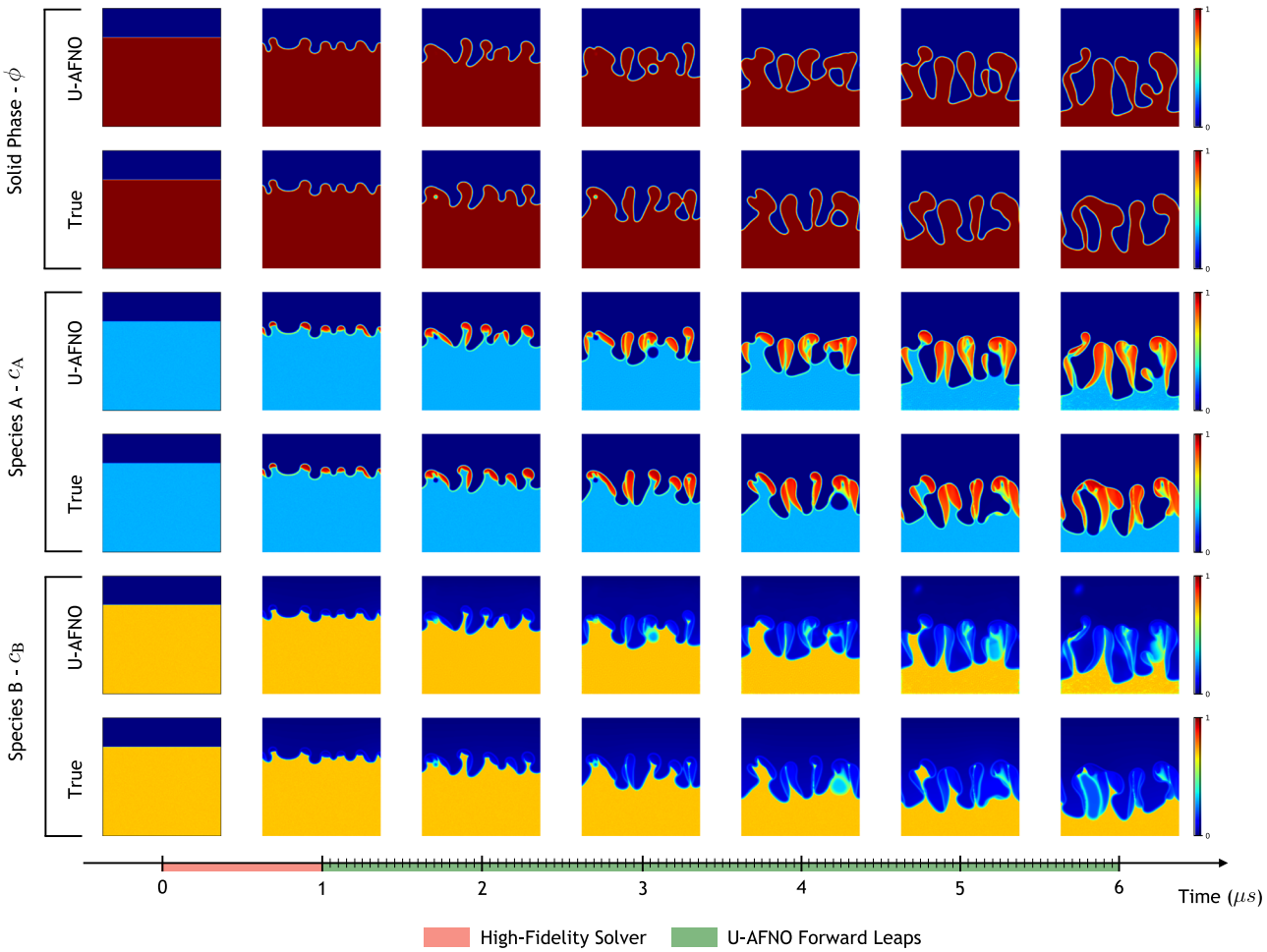

Fig. 1 | U-AFNO-B/1 field predictions in the fully auto-regressive case. The predicted and ground truth fields are shown at $t = (0, 1, 2, 3, 4, 5, 6)$μs. Since the initial 10^6 time steps are computed with the high-fidelity solver, time $t = 0-1$ μs for the U-AFNO and the ground truth are identical.

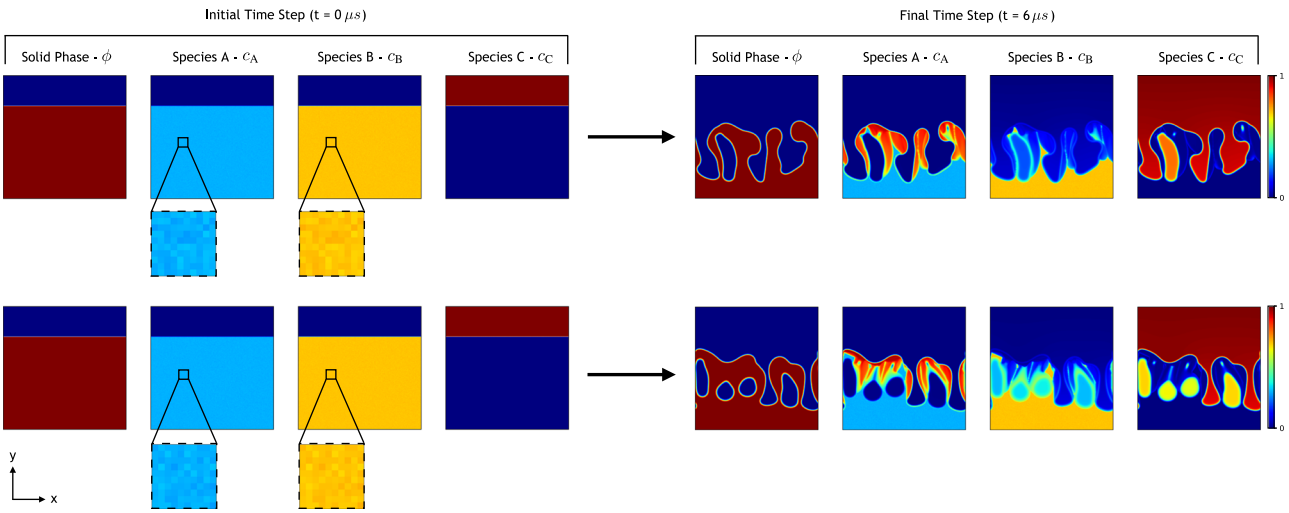

Fig. 2 | Liquid-metal dealloying with two sets of initial conditions. The initial species fields (c_A and c_B at $t = 0$) are contaminated with low-amplitude random white noise (left fields). Given the chaotic nature of the dealloying process, the initial noise perturbation eventually leads to widely different solid phase fields at late time (e.g., right fields, at $t = 6$ μs, after 6 million time steps).

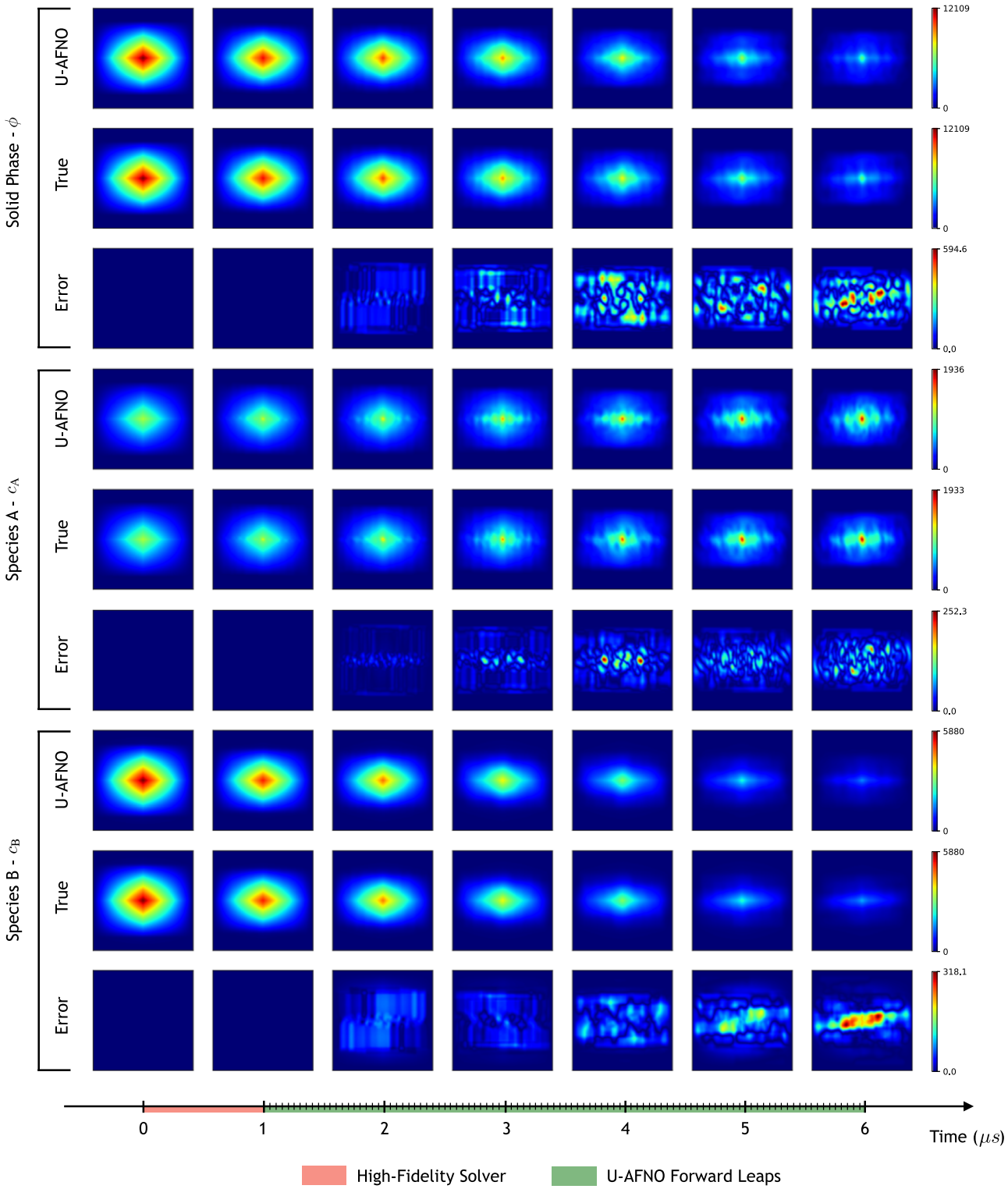

Fig. 3 | U-AFNO field auto-correlation maps in the fully auto-regressive case. The auto-correlation maps correspond to the fields shown in Fig. 1. The error maps are the absolute error between the predicted and ground truth auto-correlation.

auto-correlations closely match the ground truth, indicating that the U-AFNO reproduces invariant field statistics well.

The auto-correlation relative error metrics for each surrogate model are represented in Fig. 4. The auto-correlation error for the solid phase, and species A and B are shown every 500,000 time steps. For each field, the U-AFNO-B/1 clearly outperforms all the baseline models. At the end of the simulation, the relative error for ϕ and c_A does not exceed 15%, while that for c_B does not exceed 20%. Hence, even after 100 forward pass through the U-AFNO model, the error remains remarkably well controlled. Note that monotonic error growth is often expected with neural operators, since these models are inherently explicit (similar to explicit time integrators such as forward Euler). Fully eliminating error growth is still an open research question. Strategies to prevent error build-up, such as physics-informed loss functions, have been proposed[79], but they offer little guarantees and are not feasible here since we are leaping between non-consecutive time steps (making time derivative computations unreliable). When employing a

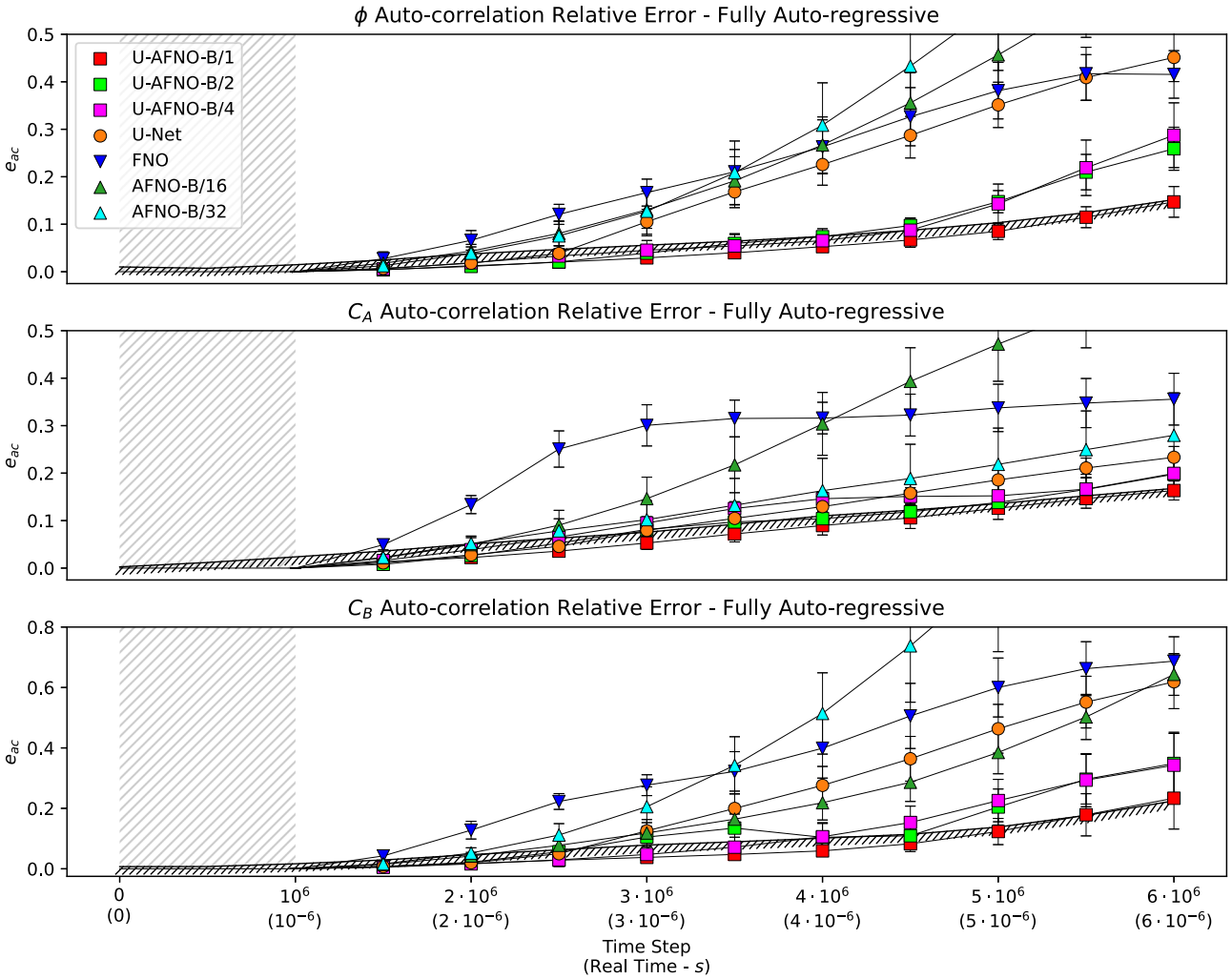

Fig. 4 | Auto-correlation relative errors for fully auto-regressive simulations. Each model is trained to skip 50,000 time steps at a time, but for simplicity we only show the error values every 500,000 steps. The hatched gray regions represent the initial 10^6 time steps computed with the high-fidelity solver. The cross-hatched black lines in each plot represent the average auto-correlation discrepancy between each distinct pair of ground truth simulations in the test set.

larger patch size (2×2 and 4×4), the errors slightly deteriorate, which is expected since larger transformer patch sizes generally yield worse accuracy. Yet, each U-AFNO model outperforms the other baseline models. In particular, they are more accurate than both U-Nets and vanilla AFNOs, confirming that combining these two model architectures can significantly improve prediction accuracy. The U-AFNO-B/1 relative error is on par with the average discrepancy between distinct pairs of ground truth simulation (shown in hatched black). This is significant: in practice, it means that, on the auto-correlation relative error metric, and considering a given high-fidelity simulation, our surrogate will reproduce the dynamics equally well as any other high-fidelity run (on perturbatively different initial condition).

The QoI relative errors are provided in Table 1, and the predicted dynamics of each QoI are shown in Figs. 5, 6 for the U-AFNO-B/1 and the U-Net, respectively. Note that the QoIs and error metrics are all computed across 20 test simulations, therefore the values in Table 1 and curves in Figs. 5, 6 are mean values (with standard deviations). The U-AFNO clearly outperforms the baseline models and predicts the evolution of each QoI with very reasonable accuracy, as it rarely exceeds 5 to 10% relative error (except for the mean ligament height, with a mean relative error of 36.5%). As shown in Fig. 5, our proposed model reproduces the mean curvature and mass envelopes with remarkable accuracy, and is also able to capture the curvature standard deviation, interface perimeter and maximum penetration depth true patterns very well. Conversely, in Fig. 6, the U-Net underestimates the mean curvature and mass envelope values, while overestimating the curvature standard-deviation, interface perimeter and maximum penetration depth. Most notably, with the U-Net, the liquid penetrates into the solid exceedingly fast and reaches the bottom boundary much earlier then it should (the maximum penetration depth steeply rises after 2 μs). This is due to the fact that with the U-Net, the liquid-solid interface tends to collapse at both the right and left edges of the domain. It is not clear why this phenomena occurs, but a possible explanation is that the U-Net fails to accurately capture the effects of the boundary conditions on the field dynamics.

Investigating the effects of hybrid time stepping

Neural operators and adjacent regression models have been extensively used with an auto-regressive roll-out to predict PDE dynamics over time[50–52,60]. While this approach (employed in the section "Performance of the fully auto-regressive U-AFNO surrogate") is simple and fast, errors can grow and be potentially unbounded. This is in part because the outputs of neural operators are generally not guaranteed to lie within the correct manifold of physical fields (especially with fully data-driven operators). In order to slow down error growth, a recently proposed approach[62,73] alternates forward passes through the neural operator (forward leap in time) with high-fidelity time stepping (relaxation steps). This strategy allows for greater flexibility in balancing the trade-off of speed-up vs. accuracy. More relaxation steps means greater reliance on the high-fidelity solver and fewer neural operator forward passes, which is expected to mitigate error build-up (as each new

Table 1 | QoIs relative errors with fully auto-regressive roll-out

Model	Mean curvature	Curvature standard deviation	Interface perimeter	Total mass	c_A mass	c_B mass	Maximum penetration depth	Mean ligament height
U-AFNO-B/1 (Ours)	**0.057 ± 0.019**	**0.117 ± 0.032**	**0.068 ± 0.029**	**0.011 ± 0.006**	**0.023 ± 0.011**	**0.007 ± 0.003**	**0.040 ± 0.017**	0.358 ± 0.238
U-AFNO-B/2 (Ours)	0.082 ± 0.030	0.128 ± 0.044	0.182 ± 0.038	0.032 ± 0.009	0.057 ± 0.017	0.014 ± 0.004	0.055 ± 0.023	0.345 ± 0.096
U-AFNO-B/4 (Ours)	0.058 ± 0.017	0.122 ± 0.030	0.073 ± 0.033	0.036 ± 0.012	0.055 ± 0.019	0.029 ± 0.004	0.052 ± 0.022	**0.328 ± 0.150**
U-Net	0.069 ± 0.022	0.144 ± 0.043	0.202 ± 0.076	0.096 ± 0.015	0.119 ± 0.026	0.033 ± 0.006	0.364 ± 0.037	0.368 ± 0.165
FNO	1.862 ± 0.290	3.798 ± 0.310	0.282 ± 0.073	0.107 ± 0.021	0.164 ± 0.031	0.054 ± 0.019	0.326 ± 0.107	0.841 ± 0.047
AFNO-B/16	0.790 ± 0.092	1.895 ± 0.173	0.132 ± 0.053	0.123 ± 0.026	0.106 ± 0.022	0.133 ± 0.036	0.040 ± 0.016	0.366 ± 0.146
AFNO-B/32	0.777 ± 0.081	2.110 ± 0.269	0.286 ± 0.085	0.121 ± 0.035	0.211 ± 0.049	0.045 ± 0.032	0.123 ± 0.059	0.581 ± 0.109

For U-AFNOs and vanilla AFNOs, B/1, B/2, and B/4 indicate a patch size of 1×1, 2×2, and 4×4, respectively. U-AFNOs can employ significantly smaller patch sizes than AFNOs, since the U-Net encoder architecture downsamples the AFNO input. Note that for fair comparison, every model output is wrapped into the interval [0, 1], as described in section "Leaping forward in time with U-AFNO".

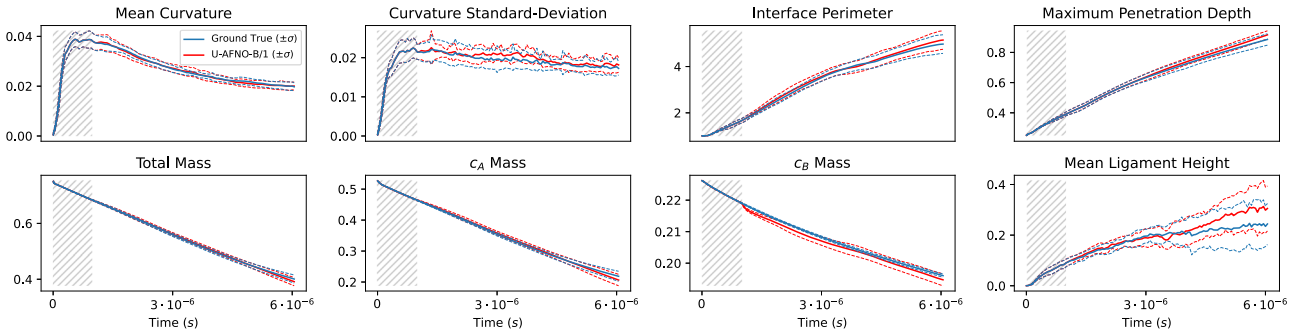

Fig. 5 | QoI dynamics predicted with the U-AFNO-B/1 (fully auto-regressive case). For each QoI, the solid blue line is the mean QoI value over time, taken across all the ground truth simulations in the test set (and the dotted lines are the corresponding standard deviations). The solid red lines represent the QoI dynamics predicted by the U-AFNO (averaged over all test surrogate simulations).

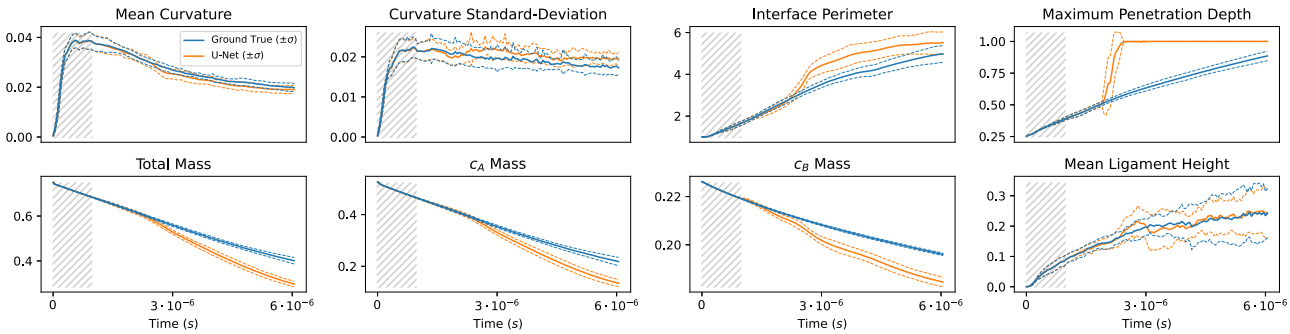

Fig. 6 | QoI dynamics predicted with the U-Net (fully auto-regressive case). For each QoI, the solid blue line is the mean QoI value over time, taken across all the ground truth simulations in the test set (and the dotted lines are the corresponding standard deviations). The solid red lines represent the QoI dynamics predicted by the U-Net (averaged over all test surrogate simulations).

operator forward pass inevitably introduces some error), but also takes more time. Conversely, fewer relaxation steps mean faster surrogate predictions but potentially larger error accumulation in the long term. In this section, we investigate the performance of U-AFNOs and baseline models with and without these high-fidelity relaxation steps (in the former case, we employ the term hybrid simulation) to understand better when hybrid time stepping may be beneficial.

We use the same trained models as in the fully auto-regressive case (leaping 50,000 time steps at a time), but after each forward pass through the surrogate model, we now employ the high-fidelity solver to advance in time for a limited number of relaxation time steps. For both the U-AFNO and the baseline models, we run test simulations for 1000, 10,000, and 50,000 relaxation time steps. Figures 7, 8 show the auto-correlation relative errors with and without relaxation steps, for the U-AFNO and the U-Net, respectively. With the U-AFNO, the accuracy of the surrogate simulation does not significantly improve when using high-fidelity relaxation time stepping. While this may seem surprising, it is, in fact, reasonable since the margin for improvement is limited. Indeed, as seen in the fully auto-regressive case, the U-AFNO average error is already on par with the average auto-correlation discrepancy between distinct ground truth simulations. Thus, reducing the error further with a model that still relies on surrogate modeling (whether moderately or not) is unlikely to succeed.

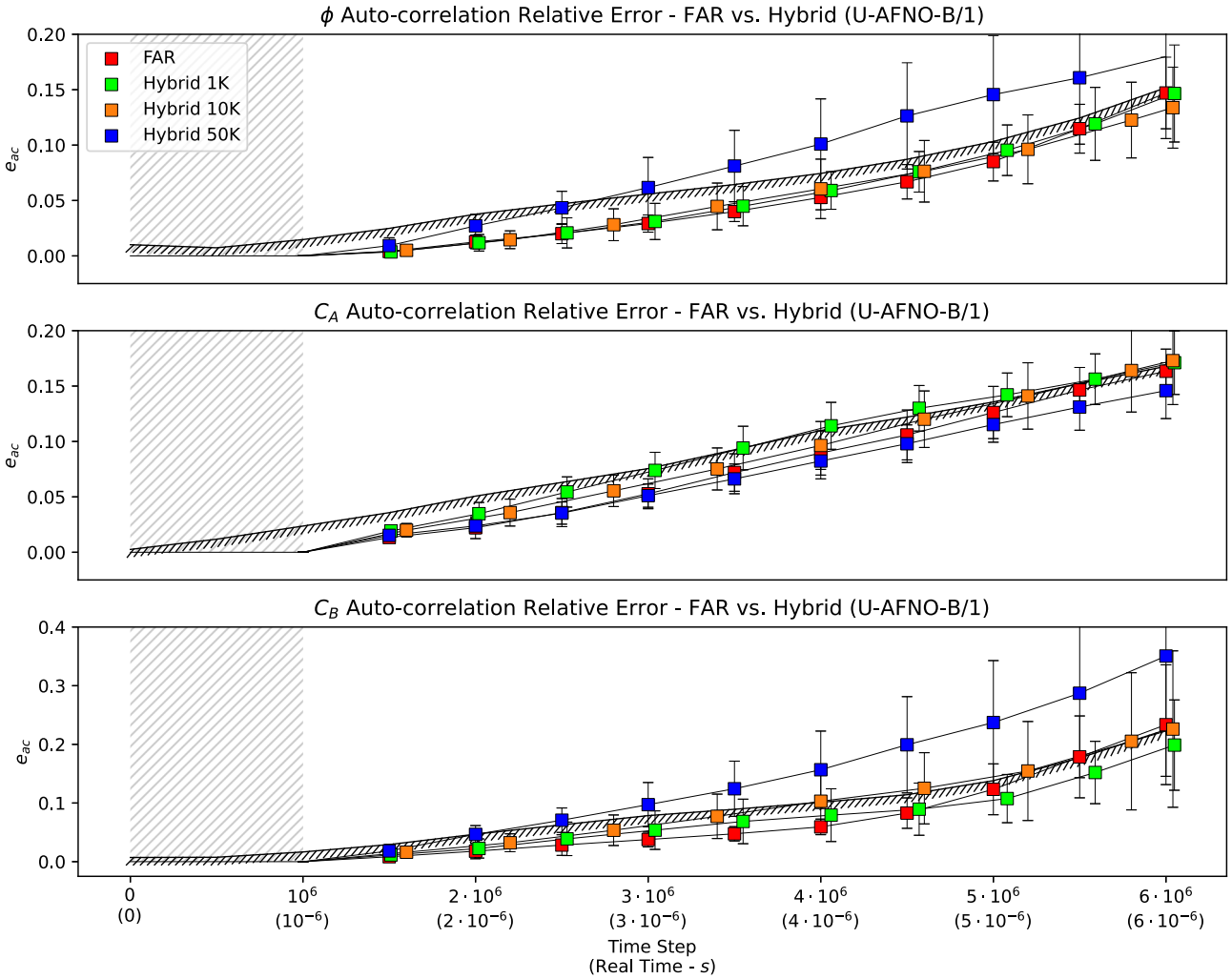

Fig. 7 | Auto-correlation relative errors for the U-AFNO-B/1, using either fully auto-regressive simulations, or hybrid simulations with 1000, 10,000, and 50,000 relaxation time steps. The cross-hatched black lines in each plot represent the average auto-correlation discrepancy between each distinct pair of ground truth simulations in the test set. Note that when using either 1000 or 10,000 relaxation steps, the timeline of the available snapshot is progressively shifting. For example, in the fully auto-regressive case (or the 50,000 relaxation steps case), the 1,050,000th step and the 1,100,000th step will be consecutively available. Conversely, with 10,000 relaxation steps, the 1,050,000th, 1,060,000th, and 1,110,000th steps will be consecutively available. This is the reason why the 1000 relaxation steps (green) and 10,000 relaxation step (orange) markers are slightly shifted.

For the U-Net however, we observe a different trend. As seen in Fig. 8, the error significantly decreases when employing high-fidelity relaxation steps, likely because there is more room for improvement (compared to the fully auto-regressive case). With just 1000 relaxation time steps, the errors sharply decrease (especially for ϕ and c_B), indicating that even just a few relaxation time steps are sufficient to correct the field physics. When increasing the number of relaxation time steps, the errors eventually decrease further, but with diminishing returns. This is expected since (as discussed in the previous paragraph), it is unlikely for any surrogate model to achieve significantly lower errors than the average auto-correlation discrepancy between ground truth simulations. The observation that blending U-Net forward passes with high-fidelity time stepping to improve accuracy is very consistent with findings in Oommen et al.[73]. However, as seen with our proposed U-AFNO model and in Fig. 7, the gains from employing high-fidelity time stepping may vary depending on the specific surrogate model architecture.

Tables 2–4 show the QoI relative errors for 1000, 10,000, and 50,000 high-fidelity relaxation time steps, respectively. While the U-AFNO still generally outperforms the baseline models (in particular with a patch size of 1×1), the improvements compared to the fully auto-regressive case are marginal, and sometimes the accuracy even degrades (as already observed with the auto-correlation errors in Fig. 7). For example, the relative error for the mass of species B increases from 0.7% in the fully auto-regressive case to 3.2% in the hybrid (1000 relaxation steps) case. With some of the baseline models, such as the FNO and the vanilla AFNOs, the accuracy slightly improves between the fully auto-regressive case and the hybrid case (with 1000 relaxation steps), but then significantly deteriorates with 10,000 and 50,000 relaxation steps. In these latter two cases, the metal tends to be consumed by the corrosive liquid much faster than it normally should, and fully disappears early on. As a result, the liquid-metal interface is not properly defined anymore, and the QoIs depending on this interface are unavailable (e.g., curvature QoIs). Intuitively, the accuracy should improve when employing more and more relaxation time steps, as a larger and larger portion of the simulation is computed through the high-fidelity solver. However, as discussed earlier, this assertion does not always hold true. Except for the vanilla U-Net, the accuracy either remains stable, or worsens. Additional investigation as to why and when accuracy may degrade is left to future work, but we conjecture that this degradation is linked to the robustness of the high-fidelity solver to un-physical initial conditions. Since the neural operator is purely data-driven and no physical constraints are enforced, the neural operator outputs may violate some underlying assumptions of the high-fidelity solver, leading to instability or un-physical

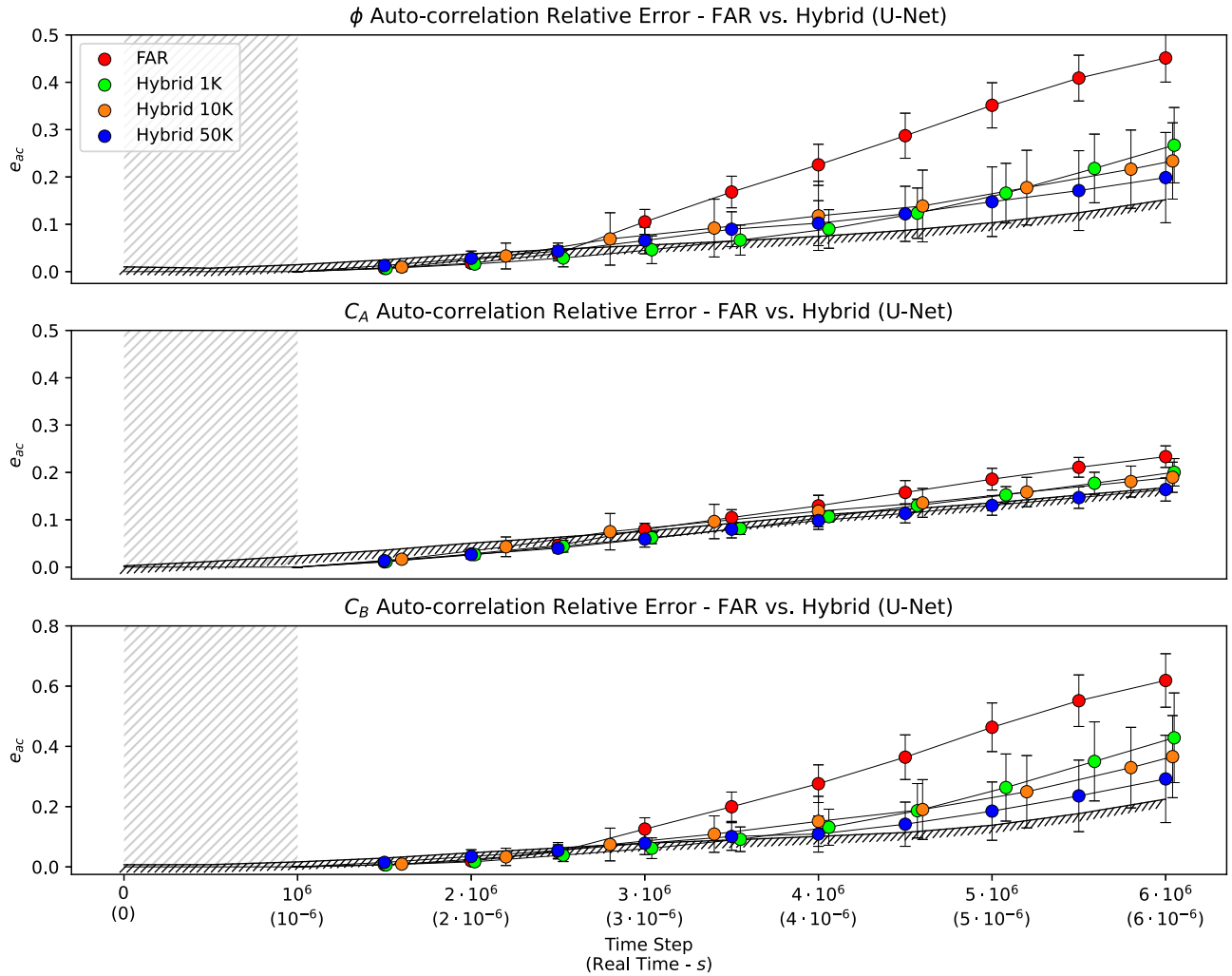

Fig. 8 | Auto-correlation relative errors for the U-Net, using either fully auto-regressive simulations, or hybrid simulations with 1000, 10,000, and 50,000 relaxation time steps. The cross-hatched black lines in each plot represent the average auto-correlation discrepancy between each distinct pair of ground truth simulations in the test set.

behavior build-up by the high-fidelity solver itself. For instance, in the high-fidelity LMD simulations, periodic boundary conditions are enforced between the left and right edges of the domain. However, these boundary conditions are not hard-enforced during the neural operator training. Thus, the neural operator outputs, which serve as initial conditions for the high-fidelity solver, may not be strictly periodic, leading to unexpected behaviors.

As a general conclusion on hybrid simulations, while our findings are consistent with early results from Oommen et al.[73], it is clear that further investigations are necessary, as the benefits from hybrid time stepping may highly depend on the machine learning surrogate architecture, the high-fidelity solver implementation, and ultimately, the underlying physical application. We conjecture that this may be particularly true in chaotic systems.

Speed-up performance

The high-fidelity solver average wall clock run time is 0.026 s per time step (on 128 CPU cores). That is, 1303 s per 50,000 steps. The average wall clock run time for a forward pass through the U-AFNO (with a 1 × 1 patch size) is 0.116 s on a single GPU. Thus, the speed-up to compute 50,000 time steps is about 11,200×. The U-AFNO run time is quasi-negligible compared to the high-fidelity solver run time, so the speed-up of a full surrogate simulation (in the fully auto-regressive case) comes down to the time taken to initialize the simulation (equivalent to a 6× speed up in the present example). Table 5 shows the parameter count, inference speed (i.e., one forward pass), and training time of each U-AFNO and baseline model. The U-AFNO models are slightly slower than the baselines (for both forward pass and training time), but the difference is negligible, especially when compared to the high-fidelity numerical solver. Besides, the U-AFNO prediction accuracy is significantly better than any of the baseline, which largely outweighs the (slightly) slower inference time. Note that augmenting the surrogate simulation with high-fidelity relaxation steps would considerably lower the speed-up gains, and as discussed in the previous paragraph, the U-AFNO already achieves excellent accuracy without needing such hybrid time stepping.

Discussion

We have introduced U-AFNOs, a new ML-based surrogate model for fast prediction of time-dependent PDEs. This architecture was applied to phase field simulations, specifically the dealloying corrosion of metals, where the infiltration of a corrosive liquid into the alloy can rapidly lead to morphologically complex metal structures. Employed within an auto-regressive rollout, our proposed model is able to reproduce the correct LMD patterns, and accurately captures invariant statistics, even in high-chaotic simulations. Our work supports the idea that combining U-Nets and vision transformers may provide very promising architectures for operator learning. Furthermore, we have identified meaningful QoIs to accurately describe the LMD dynamics and showed that they can be accurately predicted by our model. We have also investigated the relevance of augmenting auto-regressive

Table 2 | QoIs relative errors with a hybrid surrogate/high-fidelity roll-out

Model	Mean curvature	Curvature standard deviation	Interface perimeter	Total mass	c_A mass	c_B mass	Maximum penetration depth	Mean ligament height
U-AFNO-B/1 (Ours)	**0.054 ± 0.017**	**0.114 ± 0.035**	**0.083 ± 0.045**	**0.017 ± 0.009**	**0.028 ± 0.012**	0.032 ± 0.006	**0.031 ± 0.019**	**0.317 ± 0.221**
U-AFNO-B/2 (Ours)	0.056 ± 0.017	0.138 ± 0.042	0.139 ± 0.047	0.036 ± 0.011	0.060 ± 0.019	0.015 ± 0.003	0.059 ± 0.024	0.355 ± 0.220
U-AFNO-B/4 (Ours)	0.060 ± 0.017	0.117 ± 0.037	0.105 ± 0.063	0.019 ± 0.013	0.046 ± 0.021	0.030 ± 0.005	0.036 ± 0.021	0.350 ± 0.201
U-Net	0.066 ± 0.024	0.145 ± 0.052	0.129 ± 0.062	0.040 ± 0.021	0.067 ± 0.032	**0.012 ± 0.006**	0.104 ± 0.098	0.381 ± 0.202
FNO	0.735 ± 0.077	1.961 ± 0.215	0.342 ± 0.053	0.342 ± 0.062	0.375 ± 0.060	0.320 ± 0.059	0.266 ± 0.104	0.759 ± 0.073
AFNO-B/16	0.137 ± 0.039	0.471 ± 0.105	0.406 ± 0.051	0.250 ± 0.021	0.185 ± 0.033	0.336 ± 0.019	0.054 ± 0.038	0.433 ± 0.167
AFNO-B/32	0.202 ± 0.034	0.651 ± 0.113	0.357 ± 0.080	0.268 ± 0.057	0.229 ± 0.051	0.329 ± 0.066	0.096 ± 0.041	0.400 ± 0.123

Here, **1000 high-fidelity** relaxation time steps are employed after each forward pass through the surrogate model. In some cases, the surrogate simulation is too inaccurate to exhibit a properly defined liquid-metal interface, in which case any QoI depending on this interface is unavailable (listed as N/A).

Table 3 | QoIs relative errors with a hybrid surrogate/high-fidelity roll-out

Model	Mean curvature	Curvature standard deviation	Interface perimeter	Total mass	c_A mass	c_B mass	Maximum penetration depth	Mean ligament height
U-AFNO-B/1 (Ours)	**0.056 ± 0.016**	**0.114 ± 0.038**	**0.084 ± 0.045**	**0.013 ± 0.007**	0.035 ± 0.015	0.028 ± 0.006	0.042 ± 0.025	**0.261 ± 0.129**
U-AFNO-B/2 (Ours)	**0.056 ± 0.018**	0.116 ± 0.033	0.089 ± 0.047	0.023 ± 0.010	0.045 ± 0.019	**0.018 ± 0.004**	0.047 ± 0.021	0.270 ± 0.194
U-AFNO-B/4 (Ours)	0.057 ± 0.015	0.123 ± 0.051	0.110 ± 0.054	0.025 ± 0.014	**0.034 ± 0.018**	0.029 ± 0.006	**0.040 ± 0.027**	0.291 ± 0.166
U-Net	0.060 ± 0.018	0.125 ± 0.037	0.087 ± 0.033	0.044 ± 0.030	0.062 ± 0.035	0.021 ± 0.015	0.101 ± 0.109	0.336 ± 0.165
FNO	N/A	N/A	N/A	0.540 ± 0.055	0.540 ± 0.040	0.534 ± 0.079	N/A	N/A
AFNO-B/16	N/A	N/A	N/A	0.580 ± 0.124	0.522 ± 0.117	0.652 ± 0.127	N/A	N/A
AFNO-B/32	N/A	N/A	N/A	0.791 ± 0.020	0.703 ± 0.024	0.843 ± 0.015	N/A	N/A

Here, **10,000 high-fidelity** relaxation time steps are employed after each forward pass through the surrogate model. In some cases, the surrogate simulation is too inaccurate to exhibit a properly defined liquid-metal interface, in which case any QoI depending on this interface is unavailable (listed as N/A).

Table 4 | QoIs relative errors with a hybrid surrogate/high-fidelity roll-out

Model	Mean curvature	Curvature standard deviation	Interface perimeter	Total mass	c_A mass	c_B mass	Maximum penetration depth	Mean ligament height
U-AFNO-B/1 (Ours)	0.061 ± 0.018	**0.107 ± 0.036**	0.092 ± 0.042	**0.029 ± 0.013**	0.053 ± 0.019	**0.014 ± 0.004**	**0.058 ± 0.028**	0.272 ± 0.132
U-AFNO-B/2 (Ours)	**0.051 ± 0.013**	0.108 ± 0.037	**0.076 ± 0.032**	0.034 ± 0.011	0.041 ± 0.016	0.021 ± 0.004	0.059 ± 0.027	0.271 ± 0.151
U-AFNO-B/4 (Ours)	0.055 ± 0.017	**0.107 ± 0.036**	0.109 ± 0.051	0.041 ± 0.012	0.048 ± 0.018	0.018 ± 0.005	**0.058 ± 0.024**	**0.263 ± 0.152**
U-Net	0.065 ± 0.027	0.110 ± 0.039	0.102 ± 0.064	0.030 ± 0.019	**0.039 ± 0.023**	0.017 ± 0.006	0.083 ± 0.067	0.266 ± 0.101
FNO	N/A	N/A	N/A	0.400 ± 0.076	0.394 ± 0.073	0.414 ± 0.075	N/A	N/A
AFNO-B/16	N/A	N/A	N/A	0.493 ± 0.108	0.443 ± 0.124	0.635 ± 0.228	N/A	N/A
AFNO-B/32	N/A	N/A	N/A	0.734 ± 0.027	0.669 ± 0.030	0.799 ± 0.020	N/A	N/A

Here, **50,000 high-fidelity** relaxation time steps are employed after each forward pass through the surrogate model. In some cases, the surrogate simulation is too inaccurate to exhibit a properly defined liquid-metal interface, in which case any QoI depending on this interface is unavailable (listed as N/A).

surrogate simulations with hybrid high-fidelity time stepping. We showed that the potential gains in accuracy are not systematic, and may be dependent on the architecture of the ML surrogate and/or the high-fidelity solver (i.e., problem-dependent).

Methods

High-fidelity phase field model for liquid-metal dealloying

Consider a model dealloying system, where a binary alloy composed of two species, denoted as "A" and "B", is in contact with a liquid dealloying agent of pure species "C". The model is parameterized to represent a liquid-metal dealloying system, following refs. 16,20. The solid and liquid phases are tracked using a non-conserved phase field variable ϕ, where $\phi = 1$ in the solid phase, $\phi = 0$ in the liquid phase, and $0 < \phi < 1$ represents their diffuse interface. Conserved phase field variables are used to track each species, with c_A, c_B, and c_C representing the mole fraction of species A, B, and C, respectively. Everywhere in the system, $c_A + c_B + c_C = 1$.

While the system, in principle, exhibits dynamics in 3-dimensional (3D) space, we focus here on a 2D context for computational convenience,

Table 5 | U-AFNO and baseline models parameter count and computational efficiency

Model	Parameter count	Inference speed	Training time
U-AFNO (B/1) (Ours)	$66.3 \cdot 10^6$	0.116 s	35 h
U-AFNO (B/2) (Ours)	$65.1 \cdot 10^6$	0.087 s	29 h
U-AFNO (B/4) (Ours)	$69.2 \cdot 10^6$	0.081 s	27 h
U-Net	$7.8 \cdot 10^6$	0.077 s	26 h
FNO	$19.1 \cdot 10^6$	0.049 s	30 h
AFNO (B/16)	$60.5 \cdot 10^6$	0.035 s	26 h
AFNO (B/32)	$63.4 \cdot 10^6$	0.033 s	25 h

Table 6 | Thermodynamic and kinetic parameters used in the high-fidelity phase field simulations[16,20]

Parameter	Value	Parameter	Value
T	1775 K	σ_{sl}	$0.2\ J \cdot m^{-2}$
η	4×10^{-9} m	κ	$2.4 \times 10^{-9}\ J \cdot m^{-1}$
L_A	$2.82 \times 10^9\ J \cdot m^{-3}$	L_B	$1.89 \times 10^9\ J \cdot m^{-3}$
L_C	$1.84 \times 10^9\ J \cdot m^{-3}$	T_A	3290 K
T_B	1941 K	T_C	1358 K
V_a	$0.01 \times 10^{-27}\ m^3$	Ω_{AC}	$1.44 \times 10^{10}\ J \cdot m^{-3}$
$\tilde{M}_\phi$	$12.0\ m \cdot s^{-1} \cdot GPa^{-1}$	D_{liq}	$7 \times 10^{-9}\ m^2 \cdot s^{-1}$

presuming uniformity in the third dimension. The initial condition for the system is shown in Fig. 2. The spatial domain Ω is defined as a square of width 102.4 nm ($\Omega = [0, 102.4] \times [0, 102.4]$). The binary alloy is initialized in the lower part of the domain, while the liquid phase is on top. The alloy's initial composition is nominally (c_A, c_B) = (0.3, 0.7) everywhere, with superimposed white noise sampled from a uniform distribution in the interval $[-0.025, 0.025]$ at each grid-point. The liquid is initialized as pure $c_C = 1$. Dirichlet boundary conditions of $\phi = 0$ and $c_C = 1$ are enforced along the top edge of the simulation. Neumann boundary conditions (zero normal gradient) are enforced along the bottom edge of the simulation. Periodic boundary conditions are enforced on the left and right edges of the domain. We discretize the system using a regular 512×512 grid, with a grid spacing of $\Delta x = 0.2$ nm.

The evolution of the phase field is defined using

$$\frac{\partial \phi}{\partial t} = -\tilde{M}_\phi \frac{\pi^2}{8\eta} \frac{\delta F}{\delta \phi} \tag{4}$$

where $\tilde{M}_\phi$ is the mobility of the solid-liquid interface, η is the diffuse interface width, and $\delta F/\delta\phi$ is the functional derivative of the free energy functional, F, with respect to the solid phase[3,80]. Conserved-species mole fraction evolution is governed by the continuity equation

$$\frac{\partial c_i}{\partial t} = \nabla \cdot \sum_{j=A,B} M_{ij}(\phi) \nabla \left(\frac{\delta F}{\delta c_j} \right) \tag{5}$$

with a phase-dependent solute mobility M_{ij} and the functional derivative $\delta F/\delta c_j$. Eq. (5) is solved for species A and B evolution, while species C can always be determined by the condition $c_C = 1 - c_A - c_B$. The summation in Eq. (5) is only accomplished for species A and B since $\delta F/\delta c_j$ represents the diffusion potential of species j with respect to species C. The solute mobility is given by

$$M_{ij}(\phi) = D_{liq}(1-\phi) \frac{V_a}{kT} c_i(\delta_{ij} - c_j) \tag{6}$$

where the diffusivity is assumed to be negligibly small in the solid phase and smoothly increases to D_{liq} in the liquid phase. V_a is an atomic volume, k is the Boltzmann constant, and T is temperature.

The free energy function is defined as

$$F = \int_\Omega (f_{phase}(\phi) + f_{chem}(\phi, c_A, c_B)) dV \tag{7}$$

with the free energy density f_{phase} that sets the diffuse interface, defined as

$$f_{phase} = \frac{4\sigma_{sl}}{\eta} \left[\frac{\eta^2}{\pi^2} (\nabla\phi)^2 + \phi(1-\phi) \right] \tag{8}$$

where σ_{sl} is the interfacial energy of a pure solid-liquid interface. The chemical energy density f_{chem} is given by a regular solution model, which has an ideal entropy of mixing and non-ideal enthalpy of mixing that depends quadratically on mole fractions.

$$f_{chem} = \sum_{i=A,B,C} \left[\frac{kT}{V_a} c_i \ln(c_i) \right] + \Omega_{AC} c_A c_C + h(\phi) \Delta g_{sl} + \frac{1}{2} \sum_{i=A,B,C} \left[\kappa (\nabla c_i)^2 \right] \tag{9}$$

where Ω_{AC} is the excess enthalpy of mixing between species A and C, $h(\phi)$ is an interpolation function, Δg_{sl} is the difference between solid and liquid reference chemical energies, and κ is the energy penalty for spatial gradients of each species. The interpolation function, defined to smoothly vary between $h(\phi = 1) = 1$ in the solid and $h(\phi = 0) = 0$ in the liquid, is given by

$$h(\phi) = \frac{1}{2} + \frac{2}{\pi} \left[(2\phi - 1)\sqrt{\phi(1-\phi)} + \frac{1}{2} \arcsin(2\phi - 1) \right] \tag{10}$$

while Δg_{sl} is defined as

$$\Delta g_{sl} = \sum_{i=A,B,C} \left[c_i L_i \left(\frac{T - T_i}{T_i} \right) \right] \tag{11}$$

where L_i is the latent heat of melting and T_i is the melting temperature for each species i.

The parameters for the model are given in Table 6, following refs. 16,20. In this work, species A is assumed to have a very small solubility with the liquid C bath, which is set by the large, positive excess enthalpy of mixing $\Omega_{AC} \gg 0$ and the large melting temperature of species A, $T_A \gg T$. This leads to the dealloying condition where species B is selectively dissolved from the alloy into the liquid C bath, while A rearranges laterally along the interface to form a topologically-complex solid structure. Importantly, diffusion along the interface (where $0 < \phi < 1$) is significantly fast, according to Eq. (6), to enable reorganization of species A. We assume the dealloying is rate-limited by diffusion in the liquid phase (based on refs. 11,16), and therefore set the interface mobility $\tilde{M}_\phi$ to be very large so that dissolution is fast compared to diffusion in the liquid.

In the high-fidelity phase field simulations, non-conserved phases are updated using a simple forward Euler time-integration of Eq. (4). However, for conserved species, Eq. (5) is numerically very stiff, and using forward Euler would require prohibitively small time steps to maintain numerical stability. This is, in part, due to the fine spatial resolution of the numerical mesh and also the fourth derivative on each species field in Eq. (5) that arises from the square of the gradient in Eq. (9). This species gradient penalty together with the mixing enthalpy Ω_{AC} in Eq. (9) are required for capturing the characteristics of the interfacial composition separation (i.e., spinodal decomposition) that underlies the dealloying corrosion and morphology selection[19]. Instead of using forward Euler, conserved species are updated using a semi-implicit time-integration of Eq. (5), which is based on the approach of ref. 81 and uses spectral methods. The reader is referred to ref. 20 for details of this semi-implicit time-integration scheme and

implementation. These high-fidelity phase field simulations use a time-step size of $\Delta t = 10^{-12}$ s.

Leaping forward in time with U-AFNO

In this section, we present technical details of our proposed U-AFNO model. We first cover attention mechanisms and Fourier neural operators, and further motivate the need for a U-Net backbone augmentation. Vision transformers (ViT)[66] have become widely popular in computer vision[82,83], and recent attempts to use them for operator learning have shown promising early results[69,84–86]. Originally introduced for natural language processing applications[87], the key idea of transformers is to compute a similarity metric (attention) across sub-samples of the training data (tokenization). In ViTs, this is done by splitting the input image (denoted X) into N smaller patches $x \in \mathbb{R}^d$ (each patch is reformatted into a vector of dimension d). The input image can thus be represented by a tensor $X \in \mathbb{R}^{N\times d}$. The attention can be computed with the following expression:

$$Att : \mathbb{R}^{N\times d} \mapsto \mathbb{R}^{N\times d} \qquad Att(X) = \sigma\left(\frac{XW_q(XW_k)^\top}{\sqrt{d}}\right)XW_v \tag{12}$$

Where σ is an activation function, generally taken as softmax. The query, key, and value matrices, respectively W_q, W_k, and W_v are learnable weights. Since ViT are designed for 2D image inputs, Eq. (12) complexity is quadratic in nature. Thus, the self-attention mechanism of ViTs scales poorly with the patch size and the resolution of the input image (or, equivalently, in this paper, the input physical field). This is particularly problematic for two reasons. First, high-resolution images of physical fields are sometimes necessary to capture small-scale local features (as is the case in this paper with LMD). Second, a small patch size is desirable as it generally yields higher prediction accuracy[88]. As a result, training ViTs can be computationally challenging. To alleviate this issue, Adaptive Fourier Neural Operators (AFNOs)[71] have been proposed. The main idea of AFNOs is to formalize Eq. (12) as a kernel summation. Each row of $Att(X)$ can be re-written as:

$$Att(X)(s,:) = \sum_{t=1}^{N} X(t,:)\cdot \underbrace{K(s,t)W_v}_{\kappa(s,t)} \qquad K = \sigma\left(\frac{XW_q(XW_k)^\top}{\sqrt{d}}\right) \tag{13}$$

Where $t \in [\![1, N]\!]$ represents a dummy row index (in this section only). By assuming that the kernel is a Green's kernel, i.e., $\kappa(s, t) = \kappa(s - t)$, self-attention can be seen as a global convolution operation. Building on an idea originally introduced with Fourier neural operators (FNOs)[51], the attention operation is performed in the Fourier space, which allows reducing the convolution to a simpler and computationally faster element-wise matrix product:

$$Att(X)(s,:) \Longleftrightarrow \mathcal{F}^{-1}(\mathcal{F}(\kappa)\cdot\mathcal{F}(X))(s,:) \tag{14}$$

Where $\mathcal{F}$ refers to the discrete FFT operation. Additional details on AFNOs can be found in ref. 71. It should be noted that unlike other popular neural operator models, such as FNOs[51] and DeepONets[52], AFNOs are not neural operators in the true sense. Indeed, neural operators are generally considered to be resolution-independent. While the operation described in Eq. (14) is resolution-independent by construction, in AFNOs the input image is processed through an embedding layer before computing attention. This embedding layer is generally fully connected, with a fixed dimension, resulting in the loss of resolution invariance. Note that resolution invariance could, in principle, be restored by adopting a similar strategy as in DeepOnets[52]. In particular, the U-AFNO could be used as a branch network to process field quantities, and be connected with a trunk network to process Cartesian grid information (using, e.g., a fully connected or convolutional neural network).

AFNOs have been successfully employed to build surrogates in a variety of physical simulation problems, such, as and most notably, weather forecasting through auto-regressive roll-out[60,61]. However, AFNOs are prone to exhibit squared-shape artifacts and mismatches between patch interfaces in the predicted field. In our experiments, we have found this to be particularly true with the binary phase fields describing LMD simulations. To address this issue and ensure both meaningful feature extraction and smooth outputs, we propose to combine a simple AFNO with a U-Net backbone (U-AFNO). U-Nets[65] employs an encoder-decoder architecture, with U-shaped successive convolutions, downsampling, and upsampling layers, and skipped connections between the encoder and decoder. U-Nets have proven to be highly successful in image regression tasks, and combining the strengths of U-Nets with neural operators has been recently found to be useful[63,67,68,73]. In particular, ViTO[69] and DiTTO[70] studies have proposed to use vision transformer blocks within the standard U-Net bottleneck, along with a trunk network for enforcing resolution independence (as originally introduced with DeepONets[52]). In the present study, we adopt a similar approach but better suited for high-resolution fields by replacing the U-Net bottleneck with an AFNO. Note that downsampling the input image through the U-Net encoder has the convenient advantage of allowing for patch sizes of 1×1 (without downsampling, such a small patch size would be computationally intractable).

The architecture of the U-AFNO is represented in Fig. 9. The input field has dimensions $3\times 512\times 512$ (the three channels correspond to ϕ, c_A, and c_B, respectively—note that c_C is not needed here since its computation knowing c_A and c_B is trivial), and is downsampled to dimensions $256\times 64\times 64$ through the U-Net encoder. The AFNO bottleneck is composed of 12 back-to-back Attention blocks, each of which employs 16 attention heads, a patch size of 1×1, a skipped connection, and two fully connected layers with 3072 neurons each. We pick these architecture settings based on the hyperparameters recommended in refs. 65,71. The AFNO output is then upsampled back into the original dimension through the U-Net decoder. Both the U-Net and the AFNO layers use GeLU activation functions, and the final output of the U-AFNO is passed through a sigmoid (logistic) function. This guarantees that the U-AFNO predictions are bounded between 0 and 1, and hence consistent with the phase field physics.

Phase field quantities of interest

The first collection of QoIs we consider involves the curvature of the liquid-metal interface. We formally define the interface in this 2D system as any spatial coordinate $(x, y) \in \Omega$ such that $\phi(t, x, y) = 0.5$, $\forall\ t$. The interface coordinates may then be described at any time t as a parameterized curve, $\gamma(s, t) = (x(s, t), y(s, t))$ with $s \in [0, \ell] \subset \mathbb{R}$, and the signed curvature along the interface is:

$$k(s,t) = \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{3/2}} \qquad x' \equiv \frac{\partial x}{\partial s} \quad y' \equiv \frac{\partial y}{\partial s} \tag{15}$$

The curvature is defined locally, but we can quantify it globally by considering statistics such as the mean and standard deviation of the absolute curvature over the entire interface, at any moment in time:

$$\mu_k(t) = \int_0^\ell |k(s,t)|\,\mathrm{d}s \qquad \sigma_k(t) = \left[\int_0^\ell (|k(s,t)| - \mu_k(t))^2 \mathrm{d}s\right]^{1/2} \tag{16}$$

Furthermore, we also define the interface perimeter at any time as:

$$p(t) = \int_0^\ell \sqrt{(x')^2 + (y')^2}\,\mathrm{d}s \tag{17}$$

Eqs. (15), (16) and (17) are all computed discretely using finite difference. Figure 10 represents the general approach to compute the curvature locally at each time step, and extract statistical quantities to represent more global curvature dynamics over time. Note that in liquid-metal dealloying, metal ligaments may fully separate from the rest of the solid phase and form

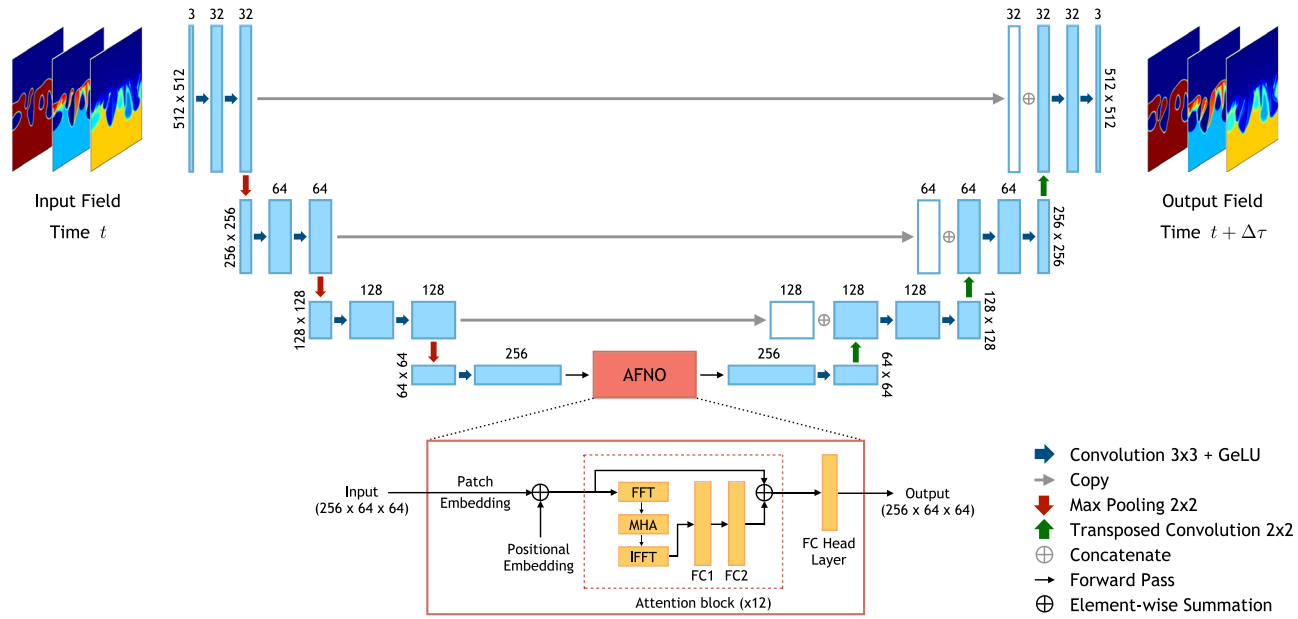

Fig. 9 | U-AFNO Architecture. The model takes as input the field at time t and outputs the field at a future time step $t+\Delta\tau$. The U-Net color scheme is analogous to the one employed in the original U-Net paper[65]. The abbreviations in the AFNO layers are the following: FFT fast Fourier transform, IFFT inverse fast Fourier transform, MHA multi-head attention, FC fully connected.

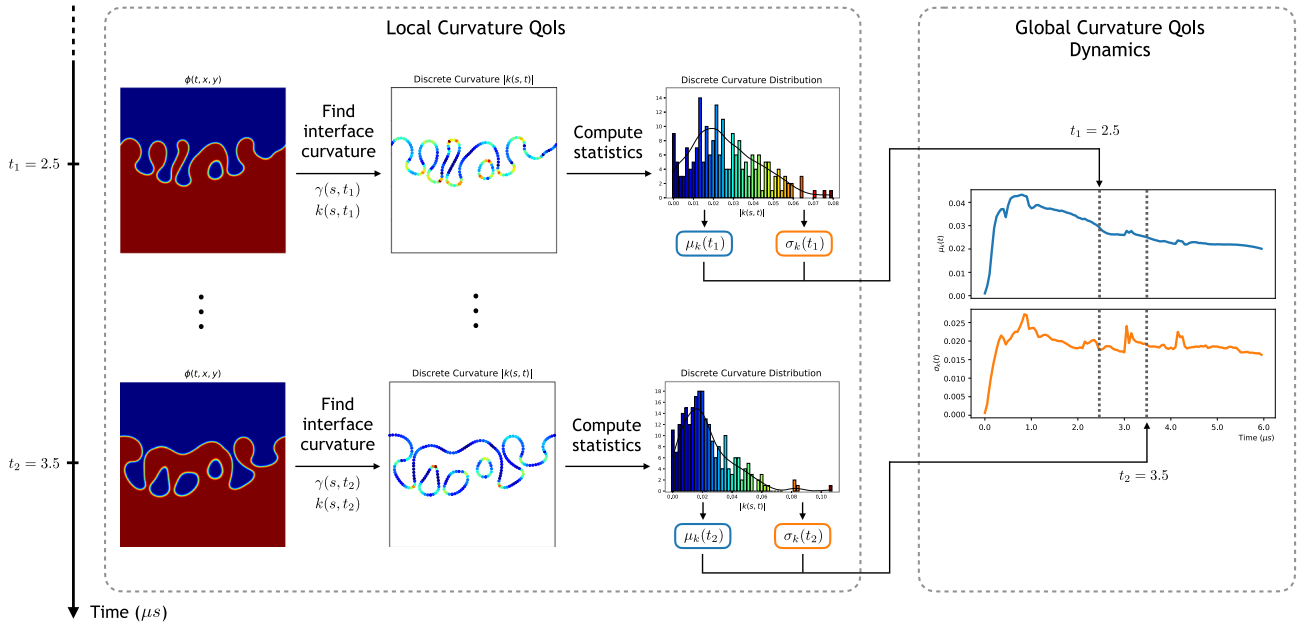

Fig. 10 | Curvature QoIs. At each time t, the parametric curve $\gamma(s,t)$ delineating the interface between the two phases is computed, from which we compute the curvature $k(s,t)$. Histograms of the curvature distribution at two different times ($t=2.5$ μs and $t=3.5$ μs) are shown, from which means and standard deviations are estimated and computed over time (blue and orange plots).

islands. Similarly, some metal ligaments may rejoin and close-up, trapping liquid bubbles inside the metal. In some cases, these features can shrink via capillary forces until they are eventually eliminated. During such events, the mean and standard deviation of the curvature is naturally skewed upward, before sharply dropping once an island or bubble disappears. Such patterns can be observed in Fig. 10.

The second collection of QoIs is considered to describe corrosive liquid penetration within the metal. As the species C-rich liquid progressively penetrates into the metal, the latter reorganizes to form narrow ligaments that typically grow as the liquid penetrates further into the bulk metal. We can quantify the average ligament length over time, as well as the maximum penetration depth, by considering the local extrema of the parametric curve representing the interface (along the vertical axis). The set of local minima and maxima at time t are defined as:

$$\mathcal{S}_{\min}(t)=\{y(s,t)\mid y'=0 \quad \text{and} \quad y''>0\} \qquad \mathcal{S}_{\max}(t)=\{y(s,t)\mid y'=0 \quad \text{and} \quad y''<0\} \tag{18}$$

Each local maximum is representative of a ligament tip $y-$ axis coordinate, while each local minimum is representative of a ligament base $y-$axis

coordinate. Thus, we approximate the mean ligament height μ_d as the difference between the mean of the set of local minima and local maxima, thus

$$\mu_d(t) = \mathbb{E}[\mathcal{S}_{\max}(t)] - \mathbb{E}[\mathcal{S}_{\min}(t)]. \tag{19}$$

We also define the maximum penetration depth as

$$\max_p(t) = 1 - \min(\mathcal{S}_{\min}(t)). \tag{20}$$

The third and last set of QoIs considered describes the dynamics of lost material throughout the corrosion process. We define the total "mass" of metal, species A, and species B as the integrals over the spatial domain of ϕ, c_A, and c_B, respectively, thus

$$m_\phi(t) = \int_\Omega \phi(t,x,y)\,\mathrm{d}x\mathrm{d}y \qquad m_A(t) = \int_\Omega c_A(t,x,y)\,\mathrm{d}x\mathrm{d}y \qquad m_B(t) = \int_\Omega c_B(t,x,y)\,\mathrm{d}x\mathrm{d}y \tag{21}$$

The actual masses in proper mass units are related to these quantities with proportionality constants, these being the metal density in the case of m_ϕ, and the molar weights for m_A and m_B.

Data availability

The datasets used in the current study will be made available from the corresponding author on reasonable request.

Code availability

The code used in the current study will be made available from the corresponding author on reasonable request.

Received: 12 July 2024; Accepted: 24 November 2024;
Published online: 13 January 2025

References

1. Chen, L.-Q. Phase-field models for microstructure evolution. *Ann. Rev. Mater. Res.* **32**, 113–140 (2002).
2. Karma, A. In *Encyclopedia of Materials: Science and Technology* (eds Buschow, K. J. et al.) (Elsevier, 2001).
3. Steinbach, I. Phase-field models in materials science. *Model. Simul. Mater. Sci. Eng.* **17**, 073001 (2009).
4. Provatas, N., Goldenfeld, N. & Dantzig, J. Adaptive mesh refinement computation of solidification microstructures using dynamic data structures. *J. Comput. Phys.* **148**, 265–290 (1999).
5. Karma, A. & Rappel, W.-J. Phase-field method for computationally efficient modeling of solidification with arbitrary interface kinetics. *Phys. Rev. E* **53**, R3017–R3020 (1996).
6. Karma, A. & Rappel, W.-J. Quantitative phase-field modeling of dendritic growth in two and three dimensions. *Phys. Rev. E* **57**, 4323–4349 (1998).
7. Karma, A. Phase-field formulation for quantitative modeling of alloy solidification. *Phys. Rev. Lett.* **87**, 115701 (2001).
8. Harrison, J. & Wagner, C. The attack of solid alloys by liquid metals and salt melts. *Acta Metall.* **7**, 722–735 (1959).
9. Erlebacher, J., Aziz, M. J., Karma, A., Dimitrov, N. & Sieradzki, K. Evolution of nanoporosity in dealloying. *Nature* **410**, 450–453 (2001).
10. McCue, I., Karma, A. & Erlebacher, J. Pattern formation during electrochemical and liquid metal dealloying. *MRS Bull.* **43**, 27–34 (2018).
11. McCue, I., Gaskey, B., Geslin, P.-A., Karma, A. & Erlebacher, J. Kinetics and morphological evolution of liquid metal dealloying. *Acta Mater.* **115**, 10–23 (2016).
12. Wada, T., Geslin, P.-A., Wei, D. & Kato, H. Partial liquid metal dealloying to synthesize nickel-containing porous and composite ferrous and high-entropy alloys. *Commun. Mater.* **4**, 43 (2023).
13. Liu, X. et al. Formation of three-dimensional bicontinuous structures via molten salt dealloying studied in real-time by in situ synchrotron x-ray nano-tomography. *Nat. Commun.* **12**, 1–12 (2021).
14. Calderoni, P. & Cabet, C. In *Nuclear Corrosion Science and Engineering* (ed. Féron, D.) Ch. 23 (Woodhead Publishing, 2012).
15. Roper, R. et al. Molten salt for advanced energy applications: a review. *Ann. Nucl. Energy* **169**, 108924 (2022).
16. Geslin, P.-A., Mccue, I., Gaskey, B., Erlebacher, J. & Karma, A. Topology-generating interfacial pattern formation during liquid metal dealloying. *Nat. Commun.* **6**, 8887 (2015).
17. chen Karen Chen-Wiegart, Y. et al. Structural evolution of nanoporous gold during thermal coarsening. *Acta Mater.* **60**, 4972–4981 (2012).
18. Lai, L., Gaskey, B., Chuang, A., Erlebacher, J. & Karma, A. Topological control of liquid-metal-dealloyed structures. *Nat. Commun.* **13**, 2918 (2022).
19. Lai, L., Geslin, P.-A. & Karma, A. Microstructural pattern formation during liquid metal dealloying: phase-field simulations and theoretical analyses. *Phys. Rev. Mater.* **6**, 093803 (2022).
20. Bieberdorf, N., Asta, M. & Capolungo, L. Grain boundary effects in high-temperature liquid-metal dealloying: a multi-phase field study. *npj Comput. Mater.* **9**, 127 (2023).
21. Biegler, L. et al. Large-scale inverse problems and quantification of uncertainty. http://hdl.handle.net/10754/656260 (2010).
22. Smith, R. C. *Uncertainty Quantification - Theory, Implementation, and Applications* (SIAM, 2013).
23. Galbally, D., Fidkowski, K., Willcox, K. & Ghattas, O. Non-linear model reduction for uncertainty quantification in large-scale inverse problems. *Int. J. Numer. Methods Eng.* **81**, 1581–1608 (2010).
24. Fountoulakis, V. & Earls, C. Duct heights inferred from radar sea clutter using proper orthogonal bases. *Radio Sci.* **51**, 1614–1626 (2016).
25. Peherstorfer, B., Willcox, K. & Gunzburger, M. Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *Siam Rev.* **60**, 550–591 (2018).
26. Wang, S., Sturler, E. D. & Paulino, G. H. Large-scale topology optimization using preconditioned Krylov subspace methods with recycling. *Int. J. Numer. Methods Eng.* **69**, 2441–2468 (2007).
27. White, D., Choi, Y. & Kudo, J. A dual mesh method with adaptivity for stress-constrained topology optimization. *Struct. Multidisc. Optim.* **61**, 749–762 (2020).
28. Forrester, A., Sobester, A. & Keane, A. *Engineering Design via Surrogate Modelling: A Practical Guide* (John Wiley & Sons, 2008).
29. Berkooz, G., Holmes, P. & Lumley, J. L. The proper orthogonal decomposition in the analysis of turbulent flows. *Annu. Rev. Fluid Mech.* **25**, 539–575 (1993).
30. Rozza, G., Huynh, D. & Patera, A. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Arch. Comput. Methods Eng.* **15**, 1–47 (2007).
31. Benner, P., Gugercin, S. & Willcox, K. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Rev.* **57**, 483–531 (2015).
32. Benner, P. et al. *Model Reduction and Approximation: Theory and Algorithms* (SIAM, 2017).
33. Safonov, M. G. & Chiang, R. Y. A schur method for balanced model reduction. In *1988 American Control Conference* 1036–1040 (1988).
34. Stabile, G. & Rozza, G. Finite volume POD-galerkin stabilised reduced order methods for the parametrised incompressible navier–stokes equations. *Comput. Fluids* **173**, 273–284 (2018).
35. Iliescu, T. & Wang, Z. Variational multiscale proper orthogonal decomposition: Navier-stokes equations. *Numer. Methods Partial Differ. Equ.* **30**, 641–663 (2014).
36. Copeland, D. M., Cheung, S. W., Huynh, K. & Choi, Y. Reduced order models for lagrangian hydrodynamics. *Comput. Methods Appl. Mech. Eng.* **388**, 114259 (2022).

37. Cheung, S. W., Choi, Y., Copeland, D. M. & Huynh, K. Local lagrangian reduced-order modeling for rayleigh-taylor instability by solution manifold decomposition. *J. Comput. Phys.* **472**, 111655 (2023).
38. McLaughlin, B., Peterson, J. & Ye, M. Stabilized reduced order models for the advection–diffusion–reaction equation using operator splitting. *Comput. Math. Appl.* **71**, 2407–2420 (2016).
39. Holmes, P. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry* (Cambridge Univ. Press, 2012).
40. Amsallem, D., Cortial, J., Carlberg, K. & Farhat, C. A method for interpolating on manifolds structural dynamics reduced-order models. *Int. J. Numer. Methods Eng.* **80**, 1241–1258 (2009).
41. Georgiou, I. T. & Schwartz, I. B. Dynamics of large scale coupled structural/mechanical systems: a singular perturbation/proper orthogonal decomposition approach. *SIAM J. Appl. Math.* **59**, 1178–1207 (1999).
42. Guyan, R. J. Reduction of stiffness and mass matrices. *AIAA J.* **3**, 380–380 (1965).
43. Antoulas, A. C. *Approximation of Large-Scale Dynamical Systems* (SIAM, 2005).
44. Krener, A. J. *Analysis and Design of Nonlinear Control Systems: In Honor of Alberto Isidori* (Springer, 2008).
45. Strazzullo, M., Ballarin, F., Mosetti, R. & Rozza, G. Model reduction for parametrized optimal control problems in environmental marine sciences and engineering. *SIAM J. Sci. Comput.* **40**, B1055–B1079 (2018).
46. Lee, K. & Carlberg, K. T. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *J. Computat. Phys.* **404**, 108973 (2020).
47. Kutz, J. N. Deep learning in fluid dynamics. *J. Fluid Mech.* **814**, 1–4 (2017).
48. Bonneville, C., Choi, Y., Ghosh, D. & Belof, J. L. Gplasdi: Gaussian process-based interpretable latent space dynamics identification through deep autoencoder. *Comput. Methods Appl. Mech. Eng.* **418**, 116535 (2024).
49. Bonneville, C. et al. A comprehensive review of latent space dynamics identification algorithms for intrusive and non-intrusive reduced-order-modeling. Preprint at arXiv:2403.10748 (2024).
50. Kovachki, N. B. et al. Neural operator: learning maps between function spaces with Application to PDEs. *J. Mach. Learn. Res.* **24**, 1–97 (2023).
51. Li, Z. et al. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations* (2021).
52. Lu, L., Jin, P., Pang, G., Zhang, Z. & Karniadakis, G. E. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nat. Mach. Intell.* **3**, 218–229 (2021).
53. Cao, Q., Goswami, S. & Karniadakis, G. E. Lno: Laplace neural operator for solving differential equations. *Nat. Mach. Intell.* **6**, 631–640 (2024).
54. Raonić, B. et al. Convolutional neural operators for robust and accurate learning of PDEs. In *NIPS '23: Proceedings of the 37th International Conference on Neural Information Processing Systems* (2023).
55. Choubineh, A., Chen, J., Wood, D. A., Coenen, F. & Ma, F. Fourier neural operator for fluid flow in small-shape 2d simulated porous media dataset. *Algorithms*. https://www.mdpi.com/1999-4893/16/1/24 (2023).
56. Li, Z., Peng, W., Zelong, Y. & Wang, J. Fourier neural operator approach to large eddy simulation of three-dimensional turbulence. *Theor. Appl. Mech. Lett.* **12**, 100389 (2022).
57. Pornthisan, N. & Markidis, S. Fast electromagnetic field pattern calculation with fourier neural operators. In *Computational Science – ICCS 2023* 247–255 (eds Mikyška, J. et al.) (Springer Nature, 2023).
58. Cai, S., Wang, Z., Lu, L., Zaki, T. A. & Karniadakis, G. E. Deepm&mnet: Inferring the electroconvection multiphysics fields based on operator approximation by neural networks. *J. Comput. Phys.* **436**, 110296 (2021).
59. Goswami, S., Yin, M., Yu, Y. & Karniadakis, G. E. A physics-informed variational deeponet for predicting crack path in quasi-brittle materials. *Comput. Methods Appl. Mech. Eng.* **391**, 114587 (2022).
60. Pathak, J. et al. Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators. Preprint at arXiv:2202.11214 (2022).
61. Kurth, T. et al. Fourcastnet: Accelerating global high-resolution weather forecasting using adaptive fourier neural operators. In *Proc. Platform for Advanced Scientific Computing Conference*, PASC '23 (Association for Computing Machinery, 2023).
62. Oommen, V., Shukla, K., Goswami, S., Dingreville, R. & Karniadakis, G. Learning two-phase microstructure evolution using neural operators and autoencoder architectures. *npj Comput. Mater.* **8**, 190 (2022).
63. Wen, G., Li, Z., Azizzadenesheli, K., Anandkumar, A. & Benson, S. M. U-fno – an enhanced fourier neural operator-based deep-learning model for multiphase flow. *Adv. Water Resour.* **163**, 104180 (2022).
64. Gopakumar, V. et al. Fourier neural operator for plasma modelling. Preprint at arXiv:2302.06542. (2023).
65. Ronneberger, O., Fischer, P. & Brox, T. U-net: convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, 234–241 (2015).
66. Dosovitskiy, A. et al. An image is worth 16x16 words: transformers for image recognition at scale. In *International Conference on Learning Representations* (2021).
67. Diab, W. & Al-Kobaisi, M. U-deeponet: U-net enhanced deep operator network for geologic carbon sequestration. *Sci. Rep.* **14**, 21298 (2024).
68. Li, Z., Peng, W., Yuan, Z. & Wang, J. Long-term predictions of turbulence by implicit U-Net enhanced Fourier neural operator. *Phys. Fluids* **35**, 075145 (2023).
69. Ovadia, O., Kahana, A., Stinis, P., Turkel, E. & Karniadakis, G. E. Vito: Vision transformer-operator. *Comput. Methods Appl. Mech. Eng.* **428**, 117109 (2024).
70. Ovadia, O. et al. Real-time inference and extrapolation via a diffusion-inspired temporal transformer operator (ditto). Preprint at arXiv:2307.09072 (2023).
71. Guibas, J. et al. Adaptive fourier neural operators: efficient token mixers for transformers. In *International Conference on Learning Representations* (2022).
72. Li, Z. et al. Learning dissipative dynamics in chaotic systems. In *36th Conference on Neural Information Processing Systems* (2022).
73. Oommen, V., Shukla, K., Desai, S., Dingreville, R. & Karniadakis, G. E. Rethinking materials simulations: blending direct numerical simulations with neural operators. *npj Comput. Mater.* **10**, 145 (2024).
74. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. In *International Conference on Learning Representations* (2015).
75. Paszke, A. et al. Pytorch: an imperative style, high-performance deep learning library. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems* (2019).
76. Vakili, S., Steinbach, I. & Varnik, F. On the numerical evaluation of local curvature for diffuse interface models of microstructure evolution. *Procedia Comput. Sci.* **108**, 1852–1862 (2017).
77. Tran, A., Liu, D., Tran, H. & Wang, Y. Quantifying uncertainty in the process-structure relationship for al–cu solidification. *Model. Simul. Mater. Sci. Eng.* **27**, 064005 (2019).
78. Herman, E., Stewart, J. A. & Dingreville, R. A data-driven surrogate model to rapidly predict microstructure morphology during physical vapor deposition. *Appl. Math. Model.* **88**, 589–603 (2020).
79. Li, Z. et al. Physics-informed neural operator for learning partial differential equations. *ACM/JMS J. Data Sci.* **1**, (2024).
80. Steinbach, I. & Pezzolla, F. A generalized field method for multiphase transformations using interface fields. *Phys. D.* **134**, 385–393 (1999).

81. Badalassi, V. E., Ceniceros, H. D. & Banerjee, S. Computation of multiphase systems with phase field models. *J. Comput. Phys.* **190**, 371–397 (2003).
82. Han, K. et al. A survey on vision transformer. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**, 87–110 (2023).
83. Khan, S. et al. Transformers in vision: a survey. *ACM Comput. Surv.* **54**, 1–41 (2022).
84. Hao, Z. et al. Dpot: auto-regressive denoising operator transformer for large-scale pde pre-training. Preprint at arXiv:2403.03542 (2024).
85. Varghese, A. J., Bora, A., Xu, M. & Karniadakis, G. E. Transformerg2g: adaptive time-stepping for learning temporal graph embeddings using transformers. *Neural Netw.* **172**, 106086 (2024).
86. Li, Z., Meidani, K. & Farimani, A. B. Transformer for partial differential equations' operator learning. In *Transactions on Machine Learning Research* (2023).
87. Vaswani, A. et al. Attention is all you need. In *31st Conference on Neural Information Processing Systems* (2017).
88. Beyer, L. et al. Flexivit: one model for all patch sizes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023).

Acknowledgements

This work was supported by the US Department of Energy, Office of Nuclear Energy, and Office of Science, Office of Advanced Scientific Computing Research through the Scientific Discovery through Advanced Computing project on Simulation of the Response of Structural Metals in Molten-Salt Environment. This article has been co-authored by employees of National Technology and Engineering Solutions of Sandia, LLC under Contract No. DE-NA0003525 with the US Department of Energy (DOE). The employees co-own right, title and interest in and to the article and are responsible for its contents. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish, or reproduce the published form of this article or allow others to do so, for United States Government purposes. The DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan https://www.energy.gov/downloads/doe-public-access-plan. Sandia Release Number: SAND2024-07895O. Los Alamos National Laboratory, United States, an affirmative action/equal opportunity employer, is operated by Triad National Security, LLC, for the National Nuclear Security Administration of the US Department of Energy under Contract No. 89233218CNA000001. Lawrence Berkeley National Laboratory is supported by the DOE Office of Science under contract no. DE-AC02-05CH11231. This study made use of computational resources of the National Energy Research Scientific Computing Center (NERSC), which is also supported by the Office of Basic Energy Sciences of the US Department of Energy under the same contract number.

Author contributions

C.B. conceptualized, designed, implemented the machine learning surrogate code and the numerical tools for computing the QoIs, trained the models, and run the experiments. A.H. provided technical guidance and developed earlier proof-of-concepts for the hybrid simulation study. N.B., M.A., and L.C. provided technical guidance and help on the LMD physics and high-fidelity data generation. C.S. provided technical guidance and help for generating high-fidelity data and running hybrid simulations. C.S., M.A., L.C., and H.N. provided technical guidance, supervised, and funded the project. C.B., N.B., A.H., M.A., L.C., H.N., and C.S. wrote the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Christophe Bonneville.