
Bubbleformer: Forecasting Boiling with Transformers

Sheikh Md Shakeel Hassan^{1*} Xianwei Zou¹

Akash Dhruv² Aparna Chandramowlishwaran^{1*}

¹University of California, Irvine ²Argonne National Laboratory

Abstract

Modeling boiling—an inherently chaotic, multiphase process central to energy and thermal systems—remains a significant challenge for neural PDE surrogates. Existing models require future input (e.g., bubble positions) during inference because they fail to learn nucleation from past states, limiting their ability to autonomously forecast boiling dynamics. They also fail to model flow boiling velocity fields, where sharp interface–momentum coupling demands long-range and directional inductive biases.

We introduce **Bubbleformer**, a transformer-based spatiotemporal model that forecasts stable and long-range boiling dynamics including nucleation, interface evolution, and heat transfer without dependence on simulation data during inference. Bubbleformer integrates factorized axial attention, frequency-aware scaling, and conditions on thermophysical parameters to generalize across fluids, geometries, and operating conditions. To evaluate physical fidelity in chaotic systems, we propose interpretable physics-based metrics that evaluate heat flux consistency, interface geometry, and mass conservation. We also release **BubbleML 2.0**, a high-fidelity dataset that spans diverse working fluids (cryogenics, refrigerants, dielectrics), boiling configurations (pool and flow boiling), flow regimes (bubbly, slug, annular), and boundary conditions. Bubbleformer sets new benchmark results in both prediction and forecasting of two-phase boiling flows.

1 Introduction

Boiling is one of the most efficient modes of heat transfer due to the large latent heat of vaporization at liquid-vapor interfaces, making it an attractive solution for ultra-high heat flux applications such as nuclear reactors and next-generation computing infrastructure. Companies such as ZutaCore² and LiquidStack³ are pioneering two-phase and immersion cooling technologies for data centers supporting AI workloads. These industrial efforts reflect a broader trend toward harnessing phase-change phenomena for thermal control in compact, high-density environments. Boiling also holds promise for spacecraft thermal control, but without buoyancy to aid vapor removal, boiling in microgravity faces severe challenges, limiting its current viability. Addressing these limitations will require novel boiling architectures optimized for low gravity, guided by modeling, design optimization, and experimental validation—an iterative process that is computationally intensive and costly. More fundamentally, accurately modeling two-phase pool and flow boiling remains one of the grand challenges in fluid dynamics. The underlying physics is inherently chaotic and multiscale: bubbles nucleate stochastically on heated surfaces, liquid-vapor interfaces continuously deform, coalesce, and break apart, and transitions between flow regimes (e.g., bubbly, slug, annular) occur unpredictably under strong thermal-hydrodynamic coupling. As illustrated in Figure 1, boiling

*Corresponding authors: {sheikhh1, amowli}@uci.edu

²<https://zutacore.com>

³<https://www.liquidstack.com>

systems can span a vast range of spatial and temporal scales and are highly sensitive to boundary conditions, fluid properties, and geometry.

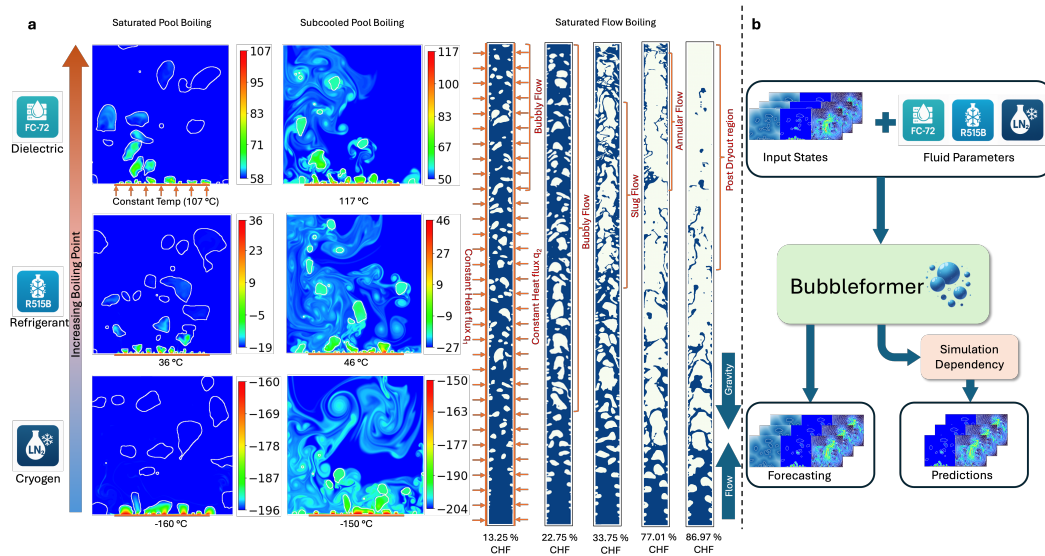


Figure 1: **(a) BubbleML 2.0 dataset:** Visualizes temperature and phase fields across pool and flow boiling configurations. Bubbles nucleate on heated surfaces, deform, coalesce, and transition between regimes, with dynamics strongly coupled to boundary conditions, working fluid, and geometry. **(b) Bubbleformer downstream tasks:** Autonomous *forecasting* of full-field dynamics including nucleation and interface evolution, and *prediction* conditioned on future bubble interfaces.

Recent advances in high-fidelity multiphysics solvers such as Flash-X [15, 10] have enabled simulations of boiling by solving incompressible Navier-Stokes equations in two phases with interface tracking and nucleation models. Although physically accurate, these simulations are computationally expensive, often requiring days on petascale supercomputers to simulate seconds of physical time [11, 12]. This has motivated the development of machine learning (ML) surrogates that learn to approximate spatiotemporal evolution of boiling directly from data. These models promise orders-of-magnitude acceleration, enabling new capabilities in real-time forecasting, parametric studies, and design-space exploration. However, current ML models for boiling [25, 32] exhibit three main limitations: (1) They require future bubble positions as input to predict velocity and temperature fields, making them unsuitable for forecasting, (2) They fail to learn bubble nucleation, a stochastic discontinuous process central to long rollouts, and (3) They fail to predict velocity fields in flow boiling, even when provided with future bubble positions.

We introduce **Bubbleformer**, a transformer-based spatiotemporal model that forecasts full-field boiling dynamics that includes temperature, velocity, and signed distance fields representing interfaces, setting a new benchmark for ML-based boiling physics. Bubbleformer makes the following core contributions:

- **Beyond prediction to forecasting.** By operating directly on full 5D spatiotemporal tensors and preserving temporal dependencies, Bubbleformer learns nucleation, key to forecasting and predicting long-range dynamics. Unlike prior models that compress time or require future bubble positions, our approach infers them end-to-end.
- **Generalizing across fluids and flow regimes.** We introduce **BubbleML 2.0**, the most comprehensive boiling dataset to date, comprising over 160 high-fidelity simulations across diverse fluids (e.g., cryogenics, refrigerants, dielectrics), boiling configurations (pool and flow), heater geometries (single- or double-sided heating), and flow regimes (bubbly, slug, and annular until dryout). Bubbleformer is conditioned on thermophysical parameters, allowing a single model to generalize across these axes.
- **Physics-based evaluation.** We introduce new interpretable metrics to assess physical fidelity beyond pixel-wise error. These include heat flux divergence, Eikonal equation loss for signed

distance functions, and conservation of vapor mass. Together, these metrics provide a more rigorous evaluation of physical correctness in chaotic boiling systems.

To our knowledge, *Bubbleformer is the first model to demonstrate autonomous, physically plausible forecasting of boiling dynamics*. It sets new state-of-the-art benchmarks on both prediction and forecasting tasks in BubbleML 2.0, representing a significant step toward practical, generalizable ML surrogates for multiphase thermal transport.

2 Problem Statement

Boiling involves the phase change of a liquid into vapor at a heated surface, forming bubbles that enhance turbulence and heat transfer. This phenomenon is highly chaotic: bubbles nucleate unpredictably on the heated surface, then grow, merge, and eventually detach, all while interacting with the surrounding liquid. The boiling system is governed by the incompressible Navier-Stokes equations (for momentum conservation) coupled with an energy transport equation, solved in both the liquid and vapor phases. These equations describe the evolution of the fluid's velocity $\vec{u}$ and pressure fields P along with the temperature field T that captures heat distribution. The *liquid-gas interface* is represented by a level-set function ϕ (signed distance field) that tracks the moving phase boundary. The interface Γ is defined by $\phi = 0$, with $\phi > 0$ in the vapor region and $\phi < 0$ in the liquid. The governing equations are non-dimensionalized using characteristic scales defined in the liquid phase (e.g., capillary length for length scale and the terminal velocity for velocity scale) and key dimensionless numbers such as Reynolds, Prandtl, Froude, Peclet, and Stefan. To ensure consistency across both phases, vapor properties (e.g., density, viscosity, thermal conductivity, and specific heat) are specified relative to the liquid's properties.

Interfacial physics is modeled by enforcing the conservation of mass and energy at Γ , accounting for surface tension and latent heat. Jump conditions for velocity, pressure, and temperature across the interface are implemented using the Ghost Fluid Method (GFM), a numerical scheme that handles sharp discontinuities at the boundary without smearing them. The level-set interface ϕ is updated via an advection (convection) equation, which moves the interface with the local fluid velocity. Evaporation and condensation are governed by differences in interfacial heat flux between the two phases, coupling flow dynamics with thermal effects. The two-phase numerical simulation framework is implemented in Flash-X [15] and follows the formulation in [10]. For completeness, the governing equations and numerical modeling assumptions are provided in Appendix A.

3 Failure Modes of Neural Solvers for Boiling

Recent works [25, 32] have explored neural surrogates trained on Flash-X simulations [15, 10, 11] to model boiling dynamics. These models aim to predict future velocity, temperature, and interface evolution given past physical states. While promising, current architectures exhibit persistent failure modes that limit their ability to forecast real-world systems.

3.1 Simulation Dependency and Failure to Learn Nucleation

Current boiling models are trained to predict velocity and temperature fields given both past physical states and *future* bubble positions. A task in BubbleML [25] is to learn an operator: $\mathcal{G}(\phi_{prev}, \vec{u}_{prev}, T_{prev}, \phi_{next}) = [\vec{u}_{next}, T_{next}]$. Although this task is tractable during supervised training, it introduces a serious limitation during inference: the model requires access to ϕ_{next} , which are not available without running the underlying simulation. This precludes autonomous rollouts and the model is fundamentally unsuitable for forecasting.

To eliminate this dependency, one can attempt to jointly learn bubble evolution: $\mathcal{G}(\phi_{prev}, \vec{u}_{prev}, T_{prev}) = [\phi_{next}, \vec{u}_{next}, T_{next}]$, requiring the model to predict future nucleation events purely from historical data. However, this task proves challenging for current architectures. Bubble nucleation is a discontinuous stochastic phenomena governed by microscale surface physics, contact angles, and thermal boundary layers [23, 58, 11]. In Flash-X, nucleation is modeled algorithmically: new bubbles reappear on the heater surface at nucleation sites with seed radii after specified wait time (time before a new bubble forms after the prior bubble departs). When these conditions are satisfied, re-nucleation is achieved through the union of the new signed

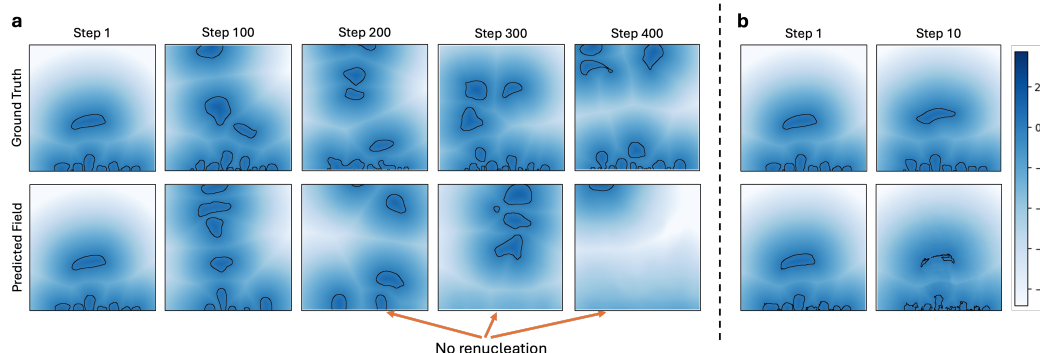


Figure 2: **Failure modes of current ML models in forecasting.** Left: UNet models maintain smooth evolution during autoregressive rollout but fail to nucleate. Right: FNO exhibits early instability.

distance field (due to the newly formed bubble) with the evolving phase field, introducing sharp discontinuities and topological changes.

These reinitializations are non-differentiable and violate assumptions of spatial smoothness. As a result, learning this behavior directly from data proves difficult. We train UNet and Fourier Neural Operator (FNO) models from BubbleML [25] to learn this task. UNet-based models perform well in single-step prediction but fail in autoregressive rollouts, as they do not learn to nucleate new bubbles as shown in Figure 2. FNO models degrade more rapidly, often diverging after a single step. FNO relies on global spectral filters and smooth continuous mappings between input and output Banach spaces [34]. The sharp discontinuities associated with nucleation add "shocks" in the underlying function space, making it difficult for the neural operator models leading to quick divergence during rollout. This is consistent with theoretical analysis showing that neural operators are sensitive to abrupt and high-frequency discontinuous topological changes [6] [35].

3.2 Failure in Flow Boiling Velocity Prediction

Flow boiling introduces additional modeling challenges. Unlike pool boiling, which is largely buoyancy-driven and symmetric, flow boiling features directional velocity, thin films, and shear-induced instabilities. Accurate modeling of such flows requires resolving the tight coupling between evolving interface dynamics and momentum transport (i.e., velocity fields). We observe that UNet and FNO models fail to predict velocity in flow boiling datasets. This failure occurs despite training on fine-resolution data and using future interface fields as supervision. We identify two primary contributing factors:

Lack of directional inductive bias. Flow boiling geometries are typically long, narrow rectangular domains (see Figure 1), with a higher resolution in the flow direction than in the cross-stream direction. When FNO's low-pass filters are applied isotropically in Fourier space, they disproportionately suppress high-frequency modes along the flow-direction, resulting in directional aliasing. Similarly, UNet models apply symmetric convolutions across both axes and lack the architectural bias to prioritize dominant features, limiting their ability to resolve streamwise velocity gradients and anisotropic flow patterns.

Insufficient spatiotemporal integration. Unlike pool boiling, flow boiling introduces additional spatiotemporal gradients from the bulk liquid flow. Localized changes in interface topology (e.g., film rupture, bubble coalescence) induce sharp and nonlocal responses in the velocity field [16]. Predicting this behavior requires long-range spatial context and multiscale temporal integration. UNet models, while effective for local pixelwise regression, lack sufficient temporal memory and spatial receptive field to capture these dynamics. FNOs, on the other hand, suffer from spectral oversmoothing, which makes it difficult to isolate sharp interface-driven effects.

These failures are not isolated. Across all flow boiling datasets in BubbleML, we observe that FNO and UNet models fail to converge when learning to predict velocity. This highlights the need for architectures that go beyond spectral operators and spatial attention to incorporate hierarchical, directional, and temporally-aware representations.

4 Bubbleformer

Bubbleformer is a spatiotemporal transformer designed to forecast boiling dynamics across fluids, boiling configurations, geometries, and flow regimes. In contrast to prior surrogates that fail to nucleate (Section 3.1) or generalize to flow boiling (Section 3.2), Bubbleformer integrates structural innovations that enable long rollout, parameter generalization, and high-frequency prediction.

4.1 Model Architecture

Bubbleformer architecture illustrated in Figure 3 consists of four components: hierarchical patch embedding, physics-based parameter conditioning, factorized space-time axial attention, and frequency-aware feature modulation.

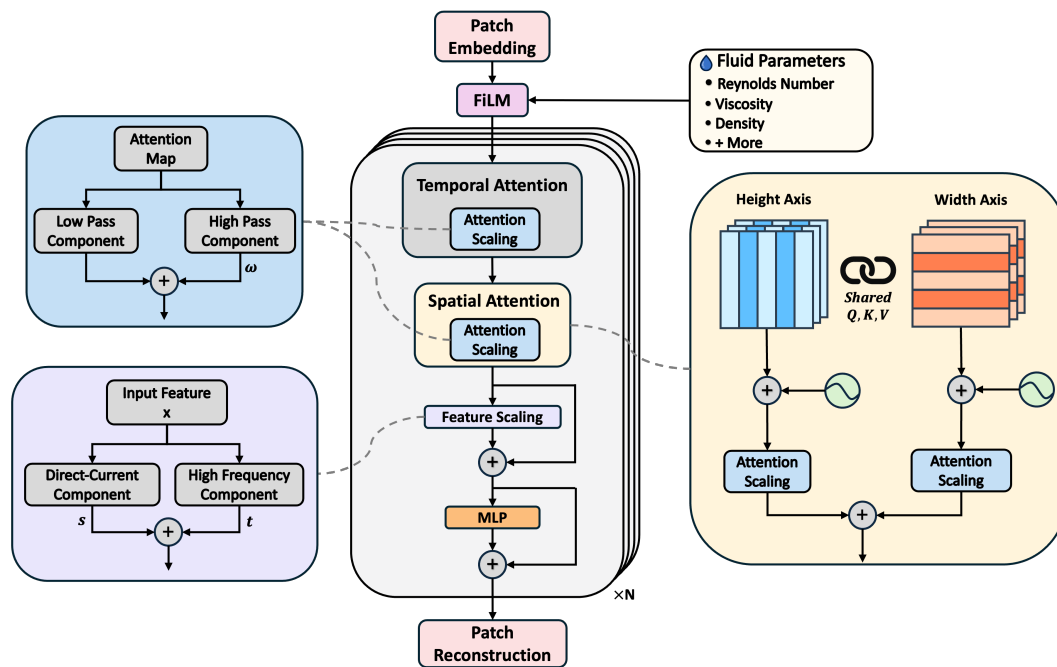


Figure 3: **Bubbleformer architecture.** Input fields are encoded into multiscale spatiotemporal patches, conditioned with fluid-specific parameters, transformed by frequency-scaled temporal and spatial axial attention, and decoded into future velocity, temperature, and interface fields.

Hierarchical Patch Embedding. Bubbleformer applies a hierarchical MLP (hMLP) stem inspired by recent hybrid vision transformers [50] to embed input physical fields as spatiotemporal patches. Each time slice is processed using a series of non-overlapping 2×2 convolutional layers with stride 2, progressively reducing spatial resolution and increasing representational depth. This design differs from prior hMLP implementations that fix resolution to 16×16 patches [50, 41]. Instead, we stack repeated 2×2 kernels at each stage to support flexible generalization across multiple resolutions. Compared to flat patch tokenization in ViT [14], this embedding builds a multiscale feature hierarchy with a stronger inductive bias for boiling dynamics, especially in domains with high aspect-ratio and varying discretizations.

FiLM-based Parameter Conditioning. To generalize across fluids (such as cryogenics, refrigerants, and dielectrics), Bubbleformer conditions embedded feature representations on physical parameters. A feature-wise linear modulation (FiLM) layer [44] applies a learned affine transformation on each channel of the patch embeddings using coefficients derived from a 9D fluid descriptor: Reynolds number, Prandtl number, Stefan number, viscosity, density, thermal conductivity, specific heat capacity, heater temperature and nucleation wait time. A small MLP maps this descriptor to channel-wise scaling and bias terms (γ_c, β_c) , which are applied as: $f'_c = \gamma_c f_c + \beta_c$ for each channel c . This

conditioning allows the model to adapt its internal representations to fluid-specific thermophysical properties, essential for generalization across working fluids. Without such conditioning, the model cannot distinguish between fluids with drastically different boiling characteristics (see Figure 1) resulting in mispredicted nucleation timing, incorrect interface velocities, and failure to preserve heat flux scaling during inference.

Factorized Space-Time Axial Attention. To model long-range spatiotemporal dynamics in boiling flows, Bubbleformer combines *factorized space-time attention* [3, 1] with *axial attention* [27, 28, 52]. In factorized space-time attention, temporal attention is applied first independently at each spatial location followed by spatial attention. This preserves the spatial locality during temporal modeling and has proven effective in video transformers [3, 1]. At each Bubbleformer block, temporal attention operates across input timesteps to learn dynamics such as bubble growth, departure, and renucleation. The resulting encoding from temporal attention is then passed to spatial attention. Instead of 2D self-attention, axial attention further decomposes the temporally updated features into two 1D attentions along the height and width axes. This decomposition reduces the overall computational complexity from $\mathcal{O}(HWT)^2$ in joint space-time attention to $\mathcal{O}(H^2 + W^2 + T^2)$, while still maintaining a global receptive field. This design, successfully applied to PDE forecasting [41, 60] addresses limitations in prior architectures. Direction-aware attention captures anisotropic structure—important for flow boiling—where strong gradients develop in the flow direction. Unlike FNO, which mixes spatial modes globally, axial attention preserves local structure and mitigates spectral oversmoothing. Each axis learns shared query, key, and value projections, and we apply T5 relative position embeddings [45] along both temporal and spatial attention blocks.

Attention and Frequency Scaling. To prevent loss of detail in deeper layers, Bubbleformer incorporates frequency-aware attention and feature scaling [53]. Deep transformers often suffer from attention collapse, where repeated application of softmax attention suppresses high-frequency signals, effectively acting as a low-pass filter. This degradation is particularly detrimental in boiling with high-frequency features such as sharp interfaces and condensation vortices. Attention scaling helps mitigate this by modifying attention to act more like an all-pass filter, by separately scaling the low- and high-frequency components of the attention scores. In addition, we also add a feature scaling layer [53] to each spatiotemporal block that explicitly separates the output feature map into low- and high-frequency components, reweights them with separate learnable parameters before recombining them. This acts as an adaptive sharpening filter that helps preserve fine-grained structures essential for modeling phase interfaces and turbulence. Similar high-frequency feature scaling has been applied successfully to improve temperature field prediction in BubbleML datasets using ResUNet and diffusion models [32].

Patch Reconstruction. The patch reconstruction mirrors hierarchical patch embedding in reverse. The spatiotemporal output embeddings are progressively upsampled through k transposed convolution layers to recover the original spatial resolution. This reconstruction produces future predictions of all physical fields—temperature, velocity, and signed distance—at each output timestep.

4.2 Metrics

We adopt and extend the metrics introduced in BubbleML [25] to evaluate both short-term predictive accuracy and long-term physical fidelity. BubbleML includes field-based metrics, such as root mean squared error (RMSE), maximum squared error, relative L2 error, boundary RMSE (BRMSE), RMSE along bubble interfaces (IRMSE), and low/mid/high Fourier mode errors. These metrics provide a comprehensive view of spatial and frequency-domain accuracy, particularly useful for evaluating sharp gradients (e.g., temperature discontinuities across liquid-vapor boundaries) that may be masked by global averaging. To evaluate Bubbleformer’s long-horizon forecasting, we introduce three additional physically interpretable system-level metrics.

Heat Flux Consistency. Boiling is inherently chaotic, and Flash-X simulations represent one possible trajectory of bubble dynamics under given boundary conditions among many. Small deviations in predicted bubble dynamics may lead to increasingly dissimilar fields, yet still preserve physically plausible behavior. To assess system-level consistency, we measure the heat flux distribution across the heater surface over time.

In thermal science, *heat flux* is a system-level indicator of boiling efficiency and its peak value—*critical heat flux* (CHF)—marks the transition to boiling crisis due to the formation of a vapor barrier (see

Figure 1. Accurate prediction of heat flux is essential for safe and efficient design of heat transfer systems [39, 46, 61, 49]. We compute heat flux normal to the heater surface using Fourier's law [30]:

$$q^j = k_\ell \frac{\partial T^j}{\partial n} \Big|_{\text{wall}}, \quad j \in \{t, \dots, t+k\}$$

where k_ℓ is the thermal conductivity of the liquid and T is the temperature field. We accumulate heat flux over time, estimate its empirical distribution ($P_{\text{GT}}(q)$ and $P_{\text{ML}}(q)$) using kernel density estimation, and report mean, standard deviation, and Kullback-Leibler (KL) divergence [4] between simulation and model distributions:

$$D_{KL}(P_{\text{GT}} \| P_{\text{ML}}) = \int P_{\text{GT}}(q) \log \frac{P_{\text{GT}}(q)}{P_{\text{ML}}(q)} dq$$

Eikonal Loss. Phase interfaces (i.e., bubble positions) are represented via a signed distance field $\phi(x, y)$, which must satisfy the Eikonal equation: $|\nabla \phi| = 1$ throughout the domain. To assess geometric correctness of the predicted interfaces, we compute the pointwise Eikonal residual [57] and report its average across all N grid points:

$$\mathcal{L}_{\text{eik}}(\phi) = \frac{1}{k} \sum_{j=t}^{t+k} \frac{1}{N} \sum_{i=1}^N ||\nabla \phi_{\text{ML}}^j(x_i)| - 1| \quad (1)$$

A low Eikonal loss indicates that predicted interfaces preserve the level set property of the signed distance field and conform to physically valid bubble geometries. In practice, we find that losses below 0.1 threshold are sufficient to ensure stable interface evolution under autoregressive rollout.

Mass Conservation. In boiling systems with fixed heater temperature, fluid properties, and surface geometry, the total vapor volume should remain approximately conserved, up to fluctuations from nucleation and condensation. We assess this, we compute the deviation in vapor volume between model predictions and ground truth simulations across the rollout window $[T, T+k]$. We define the relative vapor volume error as:

$$\mathcal{L}_{\text{vol}}(\phi) = \frac{1}{k} \sum_{j=t}^{t+k} \frac{|\sum_{i=1}^N \mathbf{1}_{\{\phi_{\text{ML}}^j(x_i) > 0\}} - \sum_{i=1}^N \mathbf{1}_{\{\phi_{\text{GT}}^j(x_i) > 0\}}|}{\sum_{i=1}^N \mathbf{1}_{\{\phi_{\text{GT}}^j(x_i) > 0\}}}$$

where $\mathbf{1}_{\{\phi(x) > 0\}}$ is an indicator function for the vapor region, inferred from the signed distance field $\phi(x)$ at each timestep. A low $\mathcal{L}_{\text{vol}}(\phi)$ indicates that the model conserves global vapor volume in the domain.

5 BubbleML 2.0 Dataset

BubbleML 2.0 expands the original BubbleML dataset [25] with new fluids, boiling configurations, and flow regimes, enabling the study of generalization across thermophysical conditions and geometries. It adds **over 160 new high-resolution 2D simulations** spanning pool boiling and flow boiling, with diverse physics including saturated, subcooled, and single-bubble nucleation across three fluid types: FC-72 (dielectric), R-515B (refrigerant), and LN_2 (cryogen). In addition to fluid diversity, BubbleML 2.0 introduces new constant heat flux boundary conditions with double-sided heaters to simulate different boiling regimes, including bubbly, slug, and annular until dryout. Simulating these phenomena required advances in numerical methods in Flash-X. Appendix A provides a detailed description of the simulations and validates against experimental data.

Table I summarizes the dataset. All simulations are performed using Flash-X and stored in HDF5 format. Spatial and temporal resolution vary across fluids based on differences in characteristic scales, and adaptive mesh refinement (AMR) is used where needed for computational efficiency. Simulations performed on AMR grids are interpolated to regular grids with the same discretization as the other datasets. We first apply linear interpolation, followed by nearest-neighbor interpolation to resolve boundary NaN values. Additional details, including boundary conditions and data access instructions, are provided in Appendix B.

Table 1: **Summary of BubbleML 2.0 datasets and their parameters.** Δt is the temporal resolution in non-dimensional time which depends on the characteristic length and velocity for each fluid, as calculated in Table 3. PB: pool boiling. FB: flow boiling.

Type - Physics - Fluid	Sims	Domain (mm^d)	Resolution Spatial	Δt	Time steps	Size (GB)
PB - Single Bubble - FC72	11	4.38×6.57	192×288	0.2	2000	50
PB - Single Bubble - R515B	11	6.48×9.72	192×288	0.2	2000	50
PB - Saturated - FC72	20	11.68×11.68	512×512	0.1	2000	320
PB - Saturated - R515B	20	17.28×17.28	512×512	0.1	2000	320
PB - Saturated - LN2	20	16.96×16.96	512×512	0.1	2000	320
PB - Subcooled - FC72	20	11.68×11.68	512×512	0.1	2000	320
PB - Subcooled - R515B	20	17.28×17.28	512×512	0.1	2000	320
PB - Subcooled - LN2	20	16.96×16.96	512×512	0.1	2000	320
FB - Inlet Velocity - FC72	15	30.66×3.65	1344×160	0.1	2000	320
FB - Constant Heat Flux - FC72	6	117.5×5.1	5152×224	0.1	1000	420

BubbleML 2.0 follows FAIR data principles [54] and includes metadata and physical parameters for every simulation as outlined in Appendix B.2. The dataset enables benchmarking of generalization across fluids, boundary conditions, flow regimes, and supports training long-horizon forecasting models such as Bubbleformer.

6 Results and Discussion

6.1 Training

We train Bubbleformer to forecast future physical fields in boiling systems, including bubble positions (signed distance field ϕ), temperature field (T), and velocity vector field ($\vec{u}$). Given k past frames $[\phi, T, \vec{u}]_{t-k:t-1}$, the model predicts the next k frames $[\phi, T, \vec{u}]_{t:t+k-1}$ in a bundled fashion [5]. Models are trained in a supervised manner by minimizing the sum of relative L2 norms across the predicted physical fields, $[\phi^*, \vec{u}^*, T^*]_{t:t+k-1}$. The single-step loss is given by:

$$\mathcal{L} = \frac{1}{k} \sum_{i=t}^{t+k-1} \left(\frac{\|\phi_i - \phi_i^*\|_2}{\|\phi_i\|_2} + \frac{\|T_i - T_i^*\|_2}{\|T_i\|_2} + \frac{\|\vec{u}_i - \vec{u}_i^*\|_2}{\|\vec{u}_i\|_2} \right)$$

We train two Bubbleformer models, *small* (29M parameters) and *large* (115M parameters), using a prediction window of $k = 5$ steps and train for 250 epochs using the Lion optimizer [7] and a warm-up cosine scheduler. Additional architectural and training details, including hyperparameter settings, are provided in Appendix C.

6.2 Re-nucleation and Forecasting

In autoregressive forecasting, a basic requirement for all models is the nucleation of new bubbles on the heater surface when old ones depart. While prior models fail to renucleate, Bubbleformer successfully learns this behavior, capturing the temporal dynamics required for periodic bubble formation and maintaining stable rollouts over extended horizons.

We observe that the re-nucleation process learned by Bubbleformer is stochastic: predictions initially align closely with simulations, but diverge gradually over time due to the chaotic nature of the system. Despite this divergence, the model's forecasts continue to respect physical laws. As shown in Figure 4, over a 200-step rollout on an unseen flow boiling trajectory, Bubbleformer predicted fields remain physically well-behaved. The system level quantities are conserved: the predicted heat flux distribution remain close to that of the simulation, the bubble positions are valid signed distance fields, and mass conservation is closely followed. These results show that Bubbleformer does not merely replicate a simulation trajectory, but learns to generate valid realizations of the underlying boiling process.

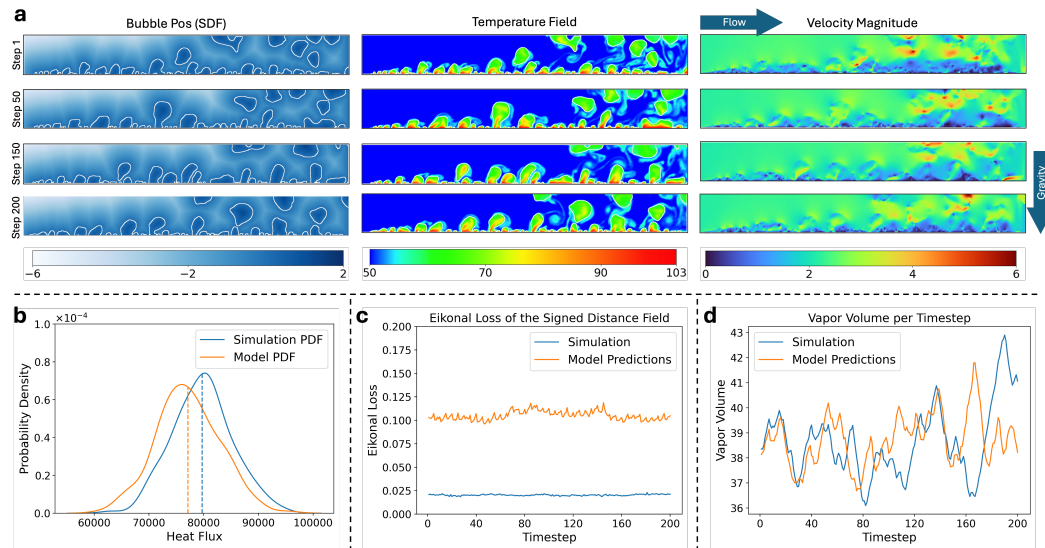


Figure 4: **Flow Boiling Forecasting.** (a) Rollout for a Bubbleformer-S model on an unseen flow boiling trajectory (FC-72, input flow scale = 2.2). (b) Comparison of predicted vs. ground-truth heat flux PDFs. (c) Per-frame Eikonal loss of forecasted signed distance fields representing bubble positions. (d) Vapor volume in the boiling domain over time for both model and simulation.

Additional discussion of all forecasting models are provided in Appendix C. We also validate our proposed metrics on a deterministic single-bubble simulation study in Appendix C.3.

6.3 Prediction of Velocity and Temperature

We evaluate Bubbleformer on the supervised prediction task introduced in BubbleML [25], modeling temperature and velocity fields in subcooled pool boiling. As shown in Figure 5, Bubbleformer outperforms the best-performing models in BubbleML—UNet_{mod} and Factorized Fourier Neural Operator (FFNO)—achieving state-of-the-art accuracy across all reported metrics, including relative L2 error. To assess long-horizon prediction stability, we extend the autoregressive rollout from 400 to 800 timesteps—doubling the original evaluation setting in BubbleML. Bubbleformer maintains accurate predictions over this extended horizon, while baseline models exhibit growing instability.

We attribute Bubbleformer’s performance to its spatiotemporal attention, which enables it to resolve both sharp, non-smooth interfaces characteristic of boiling flows and long-range dependencies. While FFNO and UNet_{mod} suffer large error spikes during violent bubble detachment events and exhibit a steadily growing error thereafter, Bubbleformer maintains a uniformly low error across the entire 800-step rollout. Notably, it remains stable even up to 2000 rollout steps, this demonstrates that our spatiotemporal attention architecture is robust to localized topological changes and yields stable long-horizon predictions. A complete listing of error metrics for each model and dataset pairing can be found in Appendix D.

7 Conclusions and Limitations

We introduce Bubbleformer, a spatiotemporal transformer for forecasting boiling dynamics across fluids, geometries, and regimes. Bubbleformer integrates axial attention, frequency-aware scaling, and fluid-conditioned FiLM layers to jointly predict and forecast velocity, temperature, and interface fields. Our results show that Bubbleformer outperforms prior models in both short-term accuracy and long-horizon stability, while preserving physical consistency as measured by system-level metrics such as heat flux, Eikonal loss, and vapor mass conservation. To support generalization, we introduce BubbleML 2.0—a diverse, high-resolution dataset from over 160 simulations.

Limitations. The current Bubbleformer architecture cannot natively operate on AMR grids, necessary for simulating fluids such as water or large real-world domains. Interpolation to uniform grids can

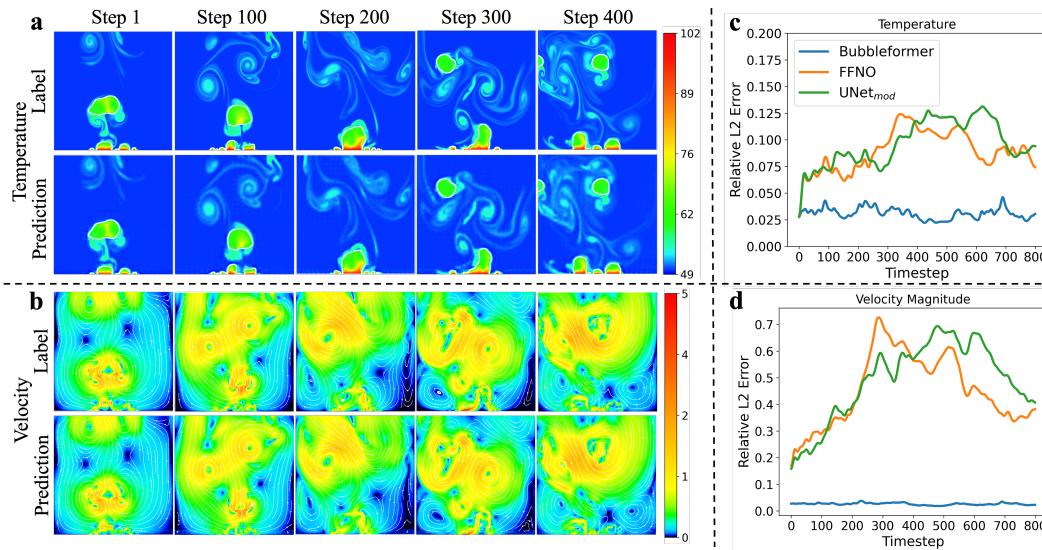


Figure 5: **Subcooled Pool Boiling Prediction.** (a) and (b) Predicted temperature and velocity fields from the Bubbleformer-S model on an unseen subcooled pool boiling trajectory for FC-72. (c) and (d) Relative L2 error over 800 rollout timesteps for temperature and velocity magnitude (combined x and y components), comparing Bubbleformer-S, FFNO, and UNet_{mod}.

introduce numerical errors that can make model training unstable. Extending the model to directly support AMR inputs—via hierarchical encoding or point-based attention—is an important direction.

Current models are specialized for one type of physics. Combining datasets from different physics (e.g., subcooled and saturated pool boiling) remains challenging. Incorporating patch-level routing via mixture-of-experts models may improve scalability and generalization.

Training on high-resolution datasets requires a large amount of GPU memory, primarily due to the storage of activations during backpropagation. While activation checkpointing can alleviate memory usage, it increases computation time. Neural operators, with their resolution-invariant properties, offer a potential solution by enabling training at lower resolutions. However, their current formulations struggle to capture the sharp discontinuities associated with nucleation events. Advancing neural operators to handle such features is a promising direction for future research in forecasting boiling dynamics.

Acknowledgments and Disclosure of Funding

The authors gratefully acknowledge funding support from the Office of Naval Research(ONR) MURI under Grant No. N000142412575, with Dr. Mark Spector serving as the program officer. We also thank the Research Cyberinfrastructure Center at the University of California, Irvine, for the GPU computing resources on the HPC3 cluster. This work was partially supported by the Scientific Discovery through Advanced Computing (SciDAC) program via the Office of Nuclear Physics and the Office of Advanced Scientific Computing Research in the Office of Science at the U.S. Department of Energy, and the Laboratory Directed Research and Development (LDRD) program at Argonne National Lab under contract number DE-AC02-06CH11357.

References

- [1] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6836–6846, 2021.
- [2] N. Basu, G. R. Warrier, and V. K. Dhir. Wall heat flux partitioning during subcooled flow boiling: Part 1—model development. *Journal of Heat Transfer*, 127(2):131–140, 03 2005.

- [3] G. Bertasius, H. Wang, and L. Torresani. Is space-time attention all you need for video understanding? In *Proceedings of the International Conference on Machine Learning (ICML)*, July 2021.
- [4] C. M. Bishop and H. Bishop. *Deep Learning: Foundations and Concepts*. Springer, 2024.
- [5] J. Brandstetter, D. E. Worrall, and M. Welling. Message passing neural PDE solvers. In *International Conference on Learning Representations*, 2022.
- [6] P. Chauhan, S. E. Choutri, M. Ghattassi, N. Masmoudi, and S. E. Jabari. Neural operators struggle to learn complex pdes in pedestrian mobility: Hughes model case study, 2025.
- [7] X. Chen, C. Liang, D. Huang, E. Real, K. Wang, Y. Liu, H. Pham, X. Dong, T. Luong, C.-J. Hsieh, Y. Lu, and Q. V. Le. Symbolic discovery of optimization algorithms, 2023.
- [8] H. Chi-Yeh and P. Griffith. The mechanism of heat transfer in nucleate pool boiling—part i: Bubble initiation, growth and departure. *International Journal of Heat and Mass Transfer*, 8(6):887–904, 1965.
- [9] V. Devahdhanush, I. Mudawar, H. K. Nahra, R. Balasubramaniam, M. M. Hasan, and J. R. Mackey. Experimental heat transfer results and flow visualization of vertical upflow boiling in earth gravity with subcooled inlet conditions – in preparation for experiments onboard the international space station. *International Journal of Heat and Mass Transfer*, 188:122603, 2022.
- [10] A. Dhruv. A vortex damping outflow forcing for multiphase flows with sharp interfacial jumps. *Journal of Computational Physics*, 511:113122, 2024.
- [11] A. Dhruv, E. Balaras, A. Riaz, and J. Kim. A formulation for high-fidelity simulations of pool boiling in low gravity. *International Journal of Multiphase Flow*, 120:103099, 2019.
- [12] A. Dhruv, E. Balaras, A. Riaz, and J. Kim. An investigation of the gravity effects on pool boiling heat transfer via high-fidelity simulations. *International Journal of Heat and Mass Transfer*, 180:121826, 2021.
- [13] A. Dhruv and S. M. S. Hassan. Github:Lab-Notebooks/Flash-X-Development: 2025.04.
- [14] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.
- [15] A. Dubey, K. Weide, J. O’Neal, A. Dhruv, S. Couch, J. A. Harris, T. Klosterman, R. Jain, J. Rudi, B. Messer, M. Pajkos, J. Carlson, R. Chu, M. Wahib, S. Chawdhary, P. M. Ricker, D. Lee, K. Antypas, K. M. Riley, C. Daley, M. Ganapathy, F. X. Timmes, D. M. Townsley, M. Vanella, J. Bachan, P. M. Rich, S. Kumar, E. Endeve, W. R. Hix, A. Mezzacappa, and T. Papatheodore. Flash-X: A multiphysics simulation software instrument. *SoftwareX*, 19:101168, 2022.
- [16] V. Ganesan. Development of a finite volume general two-phase navier-stokes solver for direct numerical simulations on cut-cells with sharp fixed interface. Master’s thesis, Purdue University, 2018.
- [17] V. Ganesan, R. Patel, J. Hartwig, and I. Mudawar. Review of databases and correlations for saturated flow boiling heat transfer coefficient for cryogenics in uniformly heated tubes, and development of new consolidated database and universal correlations. *International Journal of Heat and Mass Transfer*, 179:121656, 2021.
- [18] V. Ganesan, R. Patel, J. Hartwig, and I. Mudawar. Universal critical heat flux (chf) correlations for cryogenic flow boiling in uniformly heated tubes. *International Journal of Heat and Mass Transfer*, 166:120678, 2021.
- [19] V. Ganesan, R. Patel, J. Hartwig, and I. Mudawar. Universal correlations for post-chf saturated and superheated flow film boiling heat transfer coefficient, minimum heat flux and rewet temperature for cryogenic fluids in uniformly heated tubes. *International Journal of Heat and Mass Transfer*, 195:123054, 2022.

- [20] V. Ganesan, R. Patel, J. Hartwig, and I. Mudawar. Development of two-phase frictional pressure gradient correlation for saturated cryogenic flow boiling in uniformly heated tubes. *International Journal of Heat and Mass Transfer*, 220:124901, 2024.
- [21] R. J. George, J. Zhao, J. Kossaifi, Z. Li, and A. Anandkumar. Incremental spatial and spectral learning of neural operators for solving large-scale PDEs. *Transactions on Machine Learning Research*, 2024.
- [22] F. Gibou, L. Chen, D. Nguyen, and S. Banerjee. A level set based sharp interface method for the multiphase incompressible Navier-Stokes equations with phase change. *Journal of Computational Physics*, 222(2):536–555, 2007.
- [23] L. Gilman and E. Baglietto. A self-consistent, physics-based boiling heat transfer modeling framework for use in computational fluid dynamics. *International Journal of Multiphase Flow*, 95:35–53, 2017.
- [24] J. K. Gupta and J. Brandstetter. Towards multi-spatiotemporal-scale generalized PDE modeling. *arXiv preprint arXiv:2209.15616*, 2022.
- [25] S. M. S. Hassan, A. Feeney, A. Dhruv, J. Kim, Y. Suh, J. Ryu, Y. Won, and A. Chandramowlishwaran. Bubbleml: A multiphase multiphysics dataset and benchmarks for machine learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 418–449. Curran Associates, Inc., 2023.
- [26] D. Hendrycks and K. Gimpel. Gaussian error linear units (gelus), 2023.
- [27] J. Ho, N. Kalchbrenner, D. Weissenborn, and T. Salimans. Axial attention in multidimensional transformers, 2019.
- [28] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 603–612, 2019.
- [29] M. J. Inanlu, V. Ganesan, N. V. Upot, C. Wang, Z. Suo, K. Fazle Rabbi, P. Kabirzadeh, A. Bakhshi, W. Fu, T. S. Thukral, et al. Unveiling the fundamentals of flow boiling heat transfer enhancement on structured surfaces. *Science Advances*, 10(45):eadp8632, 2024.
- [30] F. P. Incropera, D. P. DeWitt, T. L. Bergman, and A. S. Lavine. *Fundamentals of Heat and Mass Transfer*. John Wiley & Sons, New York, 6th edition, 2007.
- [31] M. Kang, R. P. Fedkiw, and X.-D. Liu. A boundary condition capturing method for multiphase incompressible flow. *Journal of Scientific Computing*, 15(3):323–360, Sep 2000.
- [32] S. Khodakarami, V. Oommen, A. Bora, and G. E. Karniadakis. Mitigating spectral bias in neural operators via high-frequency scaling for physical systems, 2025.
- [33] N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, and A. Anandkumar. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24(89):1–97, 2023.
- [34] N. B. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. M. Stuart, and A. Anandkumar. Neural operator: Learning maps between function spaces. *CoRR*, abs/2108.08481, 2021.
- [35] S. Lanthaler, R. Molinaro, P. Hadorn, and S. Mishra. Nonlinear reconstruction for operator learning of PDEs with discontinuities. In *The Eleventh International Conference on Learning Representations*, 2023.
- [36] E. W. Lemmon, I. H. Bell, M. Huber, and M. McLinden. Nist standard reference database 23: reference fluid thermodynamic and transport properties-refprop, version 10.0, national institute of standards and technology. *Standard Reference Data Program, Gaithersburg*, pages 45–46, 2018.

- [37] Q. Lhoest, A. Villanova del Moral, Y. Jernite, A. Thakur, P. von Platen, S. Patil, J. Chaumond, M. Drame, J. Plu, L. Tunstall, J. Davison, M. Šaško, G. Chhablani, B. Malik, S. Brandeis, T. Le Scao, V. Sanh, C. Xu, N. Patry, A. McMillan-Major, P. Schmid, S. Gugger, C. Delangue, T. Matussière, L. Debut, S. Bekman, P. Cistac, T. Goehringer, V. Mustar, F. Lagunas, A. Rush, and T. Wolf. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics.
- [38] Z. Li, N. B. Kovachki, K. Azizzadenesheli, B. liu, K. Bhattacharya, A. Stuart, and A. Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021.
- [39] G. Liang and I. Mudawar. Pool boiling critical heat flux (chf)–part 1: Review of mechanisms, models, and correlations. *International Journal of Heat and Mass Transfer*, 117:1352–1367, 2018.
- [40] P. Lippe, B. S. Veeling, P. Perdikaris, R. E. Turner, and J. Brandstetter. Modeling accurate long rollouts with temporal neural PDE solvers. In *ICML Workshop on New Frontiers in Learning, Control, and Dynamical Systems*, 2023.
- [41] M. McCabe, B. R.-S. Blancard, L. H. Parker, R. Ohana, M. Cranmer, A. Bietti, M. Eickenberg, S. Golkar, G. Krawezik, F. Lanusse, M. Pettee, T. Tesileanu, K. Cho, and S. Ho. Multiple physics pretraining for spatiotemporal surrogate models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- [42] M. McCabe, P. Harrington, S. Subramanian, and J. Brown. Towards stability of autoregressive neural operators. *arXiv preprint arXiv:2306.10619*, 2023.
- [43] A. Mukherjee and S. G. Kandlikar. Numerical study of single bubbles with dynamic contact angle during nucleate pool boiling. *International Journal of Heat and Mass Transfer*, 50(1-2):127–138, 2007.
- [44] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. C. Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.
- [45] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023.
- [46] S. M. Rassoulinejad-Mousavi, F. Al-Hindawi, T. Soori, A. Rokoni, H. Yoon, H. Hu, T. Wu, and Y. Sun. Deep learning strategies for critical heat flux detection in pool boiling. *Applied Thermal Engineering*, 190:116849, 2021.
- [47] Y. Sato and B. Niceno. Nucleate pool boiling simulations using the interface tracking method: Boiling regime from discrete bubble to vapor mushroom region. *International Journal of Heat and Mass Transfer*, 105:505 – 524, 2017.
- [48] Y. Sato and B. Niceno. Pool boiling simulation using an interface tracking method: From nucleate boiling to film boiling regime through critical heat flux. *International Journal of Heat and Mass Transfer*, 125:876 – 890, 2018.
- [49] K. N. R. Sinha, V. Kumar, N. Kumar, A. Thakur, and R. Raj. Deep learning the sound of boiling for advance prediction of boiling crisis. *Cell Reports Physical Science*, 2(3), 2021.
- [50] H. Touvron, M. Cord, A. El-Nouby, J. Verbeek, and H. Jégou. Three things everyone should know about vision transformers, 2022.
- [51] A. Tran, A. Mathews, L. Xie, and C. S. Ong. Factorized fourier neural operators. In *The Eleventh International Conference on Learning Representations*, 2023.
- [52] H. Wang, Y. Zhu, B. Green, H. Adam, A. Yuille, and L.-C. Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *European conference on computer vision*, pages 108–126. Springer, 2020.

- [53] P. Wang, W. Zheng, T. Chen, and Z. Wang. Anti-oversmoothing in deep vision transformers via the fourier domain analysis: From theory to practice. In *International Conference on Learning Representations*, 2022.
- [54] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, et al. The fair guiding principles for scientific data management and stewardship. *Scientific data*, 3(1):1–9, 2016.
- [55] R. J. Williams and D. Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2):270–280, 1989.
- [56] L. Xiao, Y. Zhuang, X. Wu, J. Yang, Y. Lu, Y. Liu, and X. Han. A review of pool-boiling processes based on bubble-dynamics parameters. *Applied Sciences*, 13(21), 2023.
- [57] H. Yang, Y. Sun, G. Sundaramoorthi, and A. Yezzi. Steik: Stabilizing the optimization of neural signed distance functions and finer shape representation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 13993–14004. Curran Associates, Inc., 2023.
- [58] M. Yazdani, T. Radcliff, M. Soteriou, and A. A. Alahyari. A high-fidelity approach towards simulation of pool boiling. *Physics of Fluids*, 28(1):1–30, 2016.
- [59] L. Zeng, J. Klausner, and R. Mei. A unified model for the prediction of bubble detachment diameters in boiling systems—i. pool boiling. *International Journal of Heat and Mass Transfer*, 36(9):2261–2270, 1993.
- [60] P. Zhang, M. P. Laiu, M. Norman, D. Stefanski, and J. Gounley. Matey: multiscale adaptive foundation models for spatiotemporal physical systems, 2024.
- [61] X. Zhao, K. Shirvan, R. K. Salko, and F. Guo. On the prediction of critical heat flux using a physics-informed machine learning-aided framework. *Applied Thermal Engineering*, 164:114540, 2020.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: [\[Yes\]](#)

Justification: Our claims are backed by extensive failure mode analysis of current neural solvers for boiling² and extensive physics-based evaluation of our model⁴.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: Current limitations of our work is discussed in ⁷.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: Our results and claims are based on our experimental observations. We do not make any theoretical claims.

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We have a well documented GitHub repository with code to load our dataset, and run our models to reproduce the results reported in our paper.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.

- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [\[Yes\]](#)

Justification: We also release our code under the MIT License and our dataset under the Creative Commons Attribution 4.0 International License to ensure free and fair use. Our code can be found at [our Github page](#). The dataset is hosted at [Huggingface](#).

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Train-test splits, hyperparameters and data generation configurations are discussed in Appendix sections [B](#), [C](#) and [D](#).

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Due to computational limitations, multiple runs could not be performed before the submission deadline

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: This is discussed in Appendix sections [C](#), [D](#).

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: We follow the code of ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: We foresee no immediate negative societal impact. Positive societal impact is a promising future research direction.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: We foresee no possible misuse of our models, dataset or code.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We release our code using MIT License and dataset using Creative Commons Attribution 4.0 International License.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We provide well documented code that shows how our dataset can be used.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: Our research does not involve human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: Our research does not involve human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: Apart from simple writing and coding assistance, LLMs are not used in this work. We do not think that the use of LLMs for coding assistance impacted the core methodology, scientific rigorousness or originality of this research.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.