Transactions of the Korean Nuclear Society Spring Meeting
Jeju, Korea, May 6-8, 2026

Operator Learning of Bubble Interface Evolution in Nucleate Boiling

UngJin Na[1], Gyudong Lee[1], Hungyo Oh[2], Taeil Kim[1], Sungjin Kwon[3], HangJin Jo[1,4,*]

[1] *Department of Mechanical Engineering, Pohang University of Science and Technology (POSTECH), 77 Cheongam-Ro, Nam-gu, Pohang, Gyeongbuk 37673, Republic of Korea*
[2] *Laboratory for Waste Management, Paul Scherrer Institute (PSI), Villigen 5232, Switzerland*
[3] *Korea Institute of Fusion Technology, 169-148 Gwahak-ro, Yuseong-gu, Daejeon 34133, Republic of Korea*
[4] *Division of Advanced Nuclear Engineering, Pohang University of Science and Technology (POSTECH), 77 Cheongam-Ro, Nam-gu, Pohang, Gyeongbuk 37673, Republic of Korea*
[*] *Corresponding author: jhj04@postech.ac.kr*

Keywords: Nucleate boiling; Bubble-liquid interface; Neural operator; Lattice Boltzmann method

1. Introduction

The liquid-vapor phase field during nucleate boiling undergoes nonlinear interface evolution, commonly referred to as bubble dynamics, in which vapor bubbles repeatedly grow and depart from the heated surface [1-2]. However, the nonlinear aspects of the dynamics are often underrepresented in spatio-temporally averaged analyses, while examples of such aspects include variations in bubble size induced by preceding bubbles or coalescence events [3-5].

Although nonlinear equations governing a physical system can be discretized, the resulting algebraic system remains nonlinear and thus requires iterative solvers to obtain a solution. Furthermore, even minor changes in the source forcing term necessitate repeating the entire solution derivation process from scratch, even when a solution has already been successfully obtained for a previous case. In this regard, recent advancements in neural operators, such as FNO and DeepONet, have demonstrated their potential to learn the underlying physics of such nonlinear fields [6-7]. However, standard neural operators which map the entire field in a single forward pass, may rapidly accumulate interface errors under the autoregressive rollout setting, where the updated phase field may exhibit topological instabilities, void-fraction divergence, or unphysical coalescence behavior. Additionally, physics-informed variants of such neural operators may be limited in their applicability because directly imposing the full set of governing equations on the sharp, topology-changing bubble interface remains highly challenging [8-9].

Here, operator learning of bubble interface evolution in a nucleate boiling simulation dataset is presented. The proposed neural operator, which integrates resolvent formalism with iterative refinement, successfully learns the dataset generated from lattice Boltzmann method (LBM) simulations [10]. Comparative analysis is also performed against the baseline models FNO and DeepONet. The presented model demonstrates its capacity to suppress fine-scale interface residuals and restrict the accumulation of repeated topological errors during the autoregressive rollout process.

2. Methodology

2.1. Dataset Acquisition

The datasets used in this study are from the LBM boiling simulations [10]. In LBM, mesoscopic distribution functions stream along a regular lattice and collide locally at each time step, updating the macroscopic density, velocity, and thermal fields from their moments to recover the macroscopic equations through Chapman-Enskog analys [11]. For two-phase simulations, LBM can capture the entire process from nucleation to bubble growth without remeshing, using a fixed Cartesian lattice. The datasets are provided as 2D fields on a 1001×1411 lattice, and each frame contains a two-dimensional phase map and velocities.

Under the simulation conditions, the grid spacing was set to $\Delta x_{\mathrm{solver}} = 5 \times 10^{-5}\,\mathrm{m}$, and the timestep was $\Delta t_{\mathrm{solver}} \approx 1.25 \times 10^{-5}\,\mathrm{s}$, therefore the time difference between frames was $2.5 \times 10^{-2}\,\mathrm{s}$. The gravitational acceleration in the LBM simulation was $g_{\mathrm{lattice}} = 3.0506 \times 10^{-5}$ based on this grid spacing and timestep. As for the temperature boundary condition, a constant wall temperature was imposed at the bottom surface. The wall superheat was set to $\Delta T = 0.2371 T_c$. The thermodynamic behavior of the phase change was modeled using the Peng-Robinson equation of state [12].

$$p_{\mathrm{EOS}} = \frac{\rho R T}{1 - b\rho} - \frac{a\,\phi(T)\,\rho^2}{1 + 2b\rho - b^2\rho^2},$$

where

$$\phi(T) = \Big[1 + (0.37464 + 1.54226\omega - 0.26992\omega^2)\Big(1 - \sqrt{T/T_c}\Big)\Big]^2,$$

$$a = 0.45724 R^2 T_c^2 / p_c,$$
$$b = 0.778 R T_c / p_c,$$
$$\omega = 0.4392.$$

The CPU used for the simulation was AMD Ryzen Threadripper 3990X 64-core Processor.

2.2. Neural Operator Implementation

We denote the binary phase field by $\phi_t \in \{\text{liquid }(0),\ \text{vapor }(1)\}^{H\times W}$, and the flow field by $u_t = (u_x, u_y) \in (\mathbb{R}^2)^{H\times W}$. The binary phase fields are acquired from the LBM results to train the model for recursive future prediction with neural operators, also known as an autoregressive rollout. As the bubble-interface evolution follows a discrete-time propagator $T_{\Delta t}$ that maps the current state to the next state, where each state consists of the binary phase map ϕ_t and the velocity field u_t, the one-step autoregressive update is defined by

$$(\phi_{t+1}, u_{t+1}) = G_\theta(\phi_t, u_t)\,.$$

For phase-indicated two-phase flow with phase-change effects, we adopt the following effective transport, continuity, and momentum equations:

$$\partial_t \phi + u \cdot \nabla \phi = S_\phi,$$

$$\nabla \cdot u = S_\phi \left(1 - \frac{\rho_{\mathrm{v}}}{\rho_{\mathrm{l}}}\right),$$

and

$$\rho(\phi)\Big(\partial_t u + (u\cdot\nabla)u\Big) = -\nabla p + \rho(\phi)\,\nu(\phi)\nabla^2 u + F_\sigma(\phi) + F_b(\phi) + F_{pc}(\phi),$$

where S_ϕ denotes the local phase-change source term, and the continuity equation accounts for phase-change-induced volumetric variation through the density ratio $\rho_{\mathrm{v}}/\rho_{\mathrm{l}}$. Here, $\rho(\phi)$ and $\nu(\phi)$ are phase-dependent material properties, and $F_\sigma(\phi)$, $F_b(\phi)$, and $F_{pc}(\phi)$ denote interfacial, buoyancy, and additional phase-change-related forcing terms, respectively.

Let

$$\mathcal{L}_{\phi_t} u := \frac{1}{\rho(\phi_t)} \nabla\cdot\!\big(\mu(\phi_t)\nabla u\big), \qquad \mu(\phi_t) = \rho(\phi_t)\nu(\phi_t).$$

Then the effective momentum update term is defined as

$$q_t^{\mathrm{mom}}(\phi_t) := -(u_t\cdot\nabla)u_t + \frac{1}{\rho(\phi_t)}\Big(-\nabla p_t + F_\sigma(\phi_t) + F_b(\phi_t) + F_{pc}(\phi_t)\Big),$$

so that the momentum equation becomes

$$\partial_t u - \mathcal{L}_{\phi_t} u = q_t^{\mathrm{mom}}(\phi_t).$$

Using a backward-Euler discretization for the linear operator gives

$$\big(I - \Delta t\,\mathcal{L}_{\phi_t}\big)u_{t+1} = u_t + \Delta t\, q_t^{\mathrm{mom}}(\phi_t).$$

Accordingly, the exact one-step target implied by the data is

$$q_t^{\mathrm{true}} = \frac{\big(I - \Delta t\,\mathcal{L}_{\phi_t}\big)u_{t+1} - u_t}{\Delta t},$$

and the residual learning target is

$$\tau_t = q_t^{\mathrm{true}} - q_t^{\mathrm{pred}}.$$

The framework was built using PyTorch 2.0.0, CUDA 11.7, Python 3.10.9, and cuDNN 8.9.7.

3. Results and Discussion

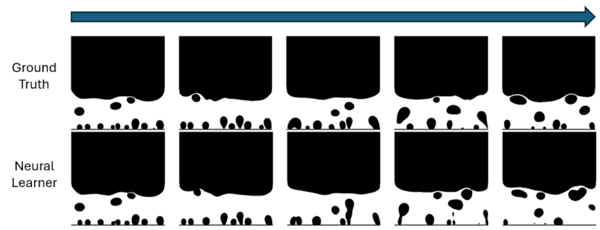

Figure 1: Snapshots of the model prediction.

Figure 1 shows the autoregressive behavior of the learned result. The overall phase field is similar to the baseline. Differences are observed in bubble coalescence and the nucleation of new bubbles, but the fine-scale behavior near the interface is captured reasonably well.

4. Conclusions

Learning bubble-interface evolution is a challenging task. In particular, autoregressive rollout has been prohibitively difficult despite its importance. The resolvent formalism with iterative refinement demonstrates the ability to recover the interfacial phenomena of bubble evolution in nucleate boiling, indicating the benefits of using this formalism for operator-learning tasks.

5. Acknowledgements

This research was supported by the National Research Council of Science & Technology (NST) grant funded by the Korea government (MSIT) (No. GTL25031-000).

References

[1] Dhir, V. K. (1998). Boiling heat transfer. Annual review of fluid mechanics, 30(1), 365-401.

[2] Dhir, V. K., Abarajith, H. S., & Li, D. (2007). Bubble dynamics and heat transfer during pool and flow boiling. Heat Transfer Engineering, 28(7), 608-624.

[3] Shoji, M., & Takagi, Y. (2001). Bubbling features from a single artificial cavity. International journal of heat and mass transfer, 44(14), 2763-2776.

[4] Shoji, M. (2002). Nonlinear bubbling and micro-convection at a submerged orifice. Tsinghua Science and Technology, 7(2), 97-108.

[5] Shoji, M. (2004). Studies of boiling chaos: a review. International Journal of Heat and Mass Transfer, 47(6-7), 1105-1128.

[6] Lu, L., Jin, P., Pang, G., Zhang, Z., & Karniadakis, G. E. (2021). Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. Nature machine intelligence, 3(3), 218-229.

[7] Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., & Anandkumar, A. (2020).

Fourier neural operator for parametric partial differential equations. arXiv preprint arXiv:2010.08895.

[8] Li, Z., Zheng, H., Kovachki, N., Jin, D., Chen, H., Liu, B., ... & Anandkumar, A. (2024). Physics-informed neural operator for learning partial differential equations. ACM/IMS Journal of Data Science, 1(3), 1-27.

[9] Wang, S., Wang, H., & Perdikaris, P. (2021). Learning the solution operator of parametric partial differential equations with physics-informed DeepONets. Science advances, 7(40), eabi8605.

[10] Oh, H., & Jo, H. (2024). Turbulent flow boiling simulation based on pseudopotential lattice boltzmann method with developed wall boundary treatments and unit conversion. Applied Thermal Engineering, 248, 123148.

[11] X. He and L.-S. Luo, Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation, Physical Review E 56 (1997), 6811-6817.

[12] Yuan, P., & Schaefer, L. (2006). Equations of state in a lattice Boltzmann model. Physics of fluids, 18(4).