A-MFO-NET: BOUNDARY-CONDITIONED GNN–FNO–GNN OPERATORS FOR LONG-HORIZON ROOM-SCALE 3D THERMAL-FLUID SIMULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-horizon three-dimensional thermal-fluid simulation on unstructured meshes requires both global information propagation and geometry-sensitive local reconstruction. We introduce A-MFO-Net, a boundary-conditioned GNN–FNO–GNN operator whose graph stages weight edge-conditioned messages using source and destination states and directed edge geometry. On a held-out 22,000-cell extended-room data-center trajectory (90% ACU cooling, 100% server workload), we predict five state channels over 400 steps and 20 seconds. Training is predominantly on 200-step room-scale trajectories, with one shorter extended-room and one longer room-scale case, so the test probes their joint transfer rather than isolated geometry or horizon extrapolation. All seven models use 200 training windows per epoch and seeds 42, 43, and 44. At the final step, A-MFO-Net's relative-L_2 errors (mean $\pm$ sample standard deviation) are $(6.97 \pm 0.18)\%$ for velocity, $(31.56 \pm 4.75)\%$ for pressure, and $(5.05 \pm 0.15)\%$ for temperature: 68.6% and 22.8% below the strongest external velocity and pressure comparators, respectively. A-MFO-Net has the lowest mean velocity and pressure errors across all seven models, while GAOT has the lowest mean temperature error. FNO has $(39.24 \pm 0.99)\%$ velocity error and highly variable pressure error, supporting the implemented graph-containing pipeline but not isolating attention or neighbor aggregation. A-MFO-Net completes the 400-step evaluation in 64.0 ± 1.3 seconds.

1 INTRODUCTION

Predicting airflow and heat transfer around server racks and air-conditioning units (ACUs) is important for data-center thermal management. Energy-efficient operation motivates alternatives to repeated room-level computational fluid dynamics (CFD) solves (Dayarathna et al., 2016; Athavale et al., 2019). Yet a transient field surrogate must do more than estimate temperatures at a few rack locations: it must propagate coupled three-dimensional velocity, pressure, and temperature fields. Our held-out extended-room task contains 22,000 mesh cells represented as graph nodes and five state channels—110,000 scalar state values per frame—over 400 autoregressive forecast steps. This is a substantial high-dimensional test because each step reuses the preceding forecast, local equipment jets and thermal gradients must be reconstructed, and errors can accumulate over the 20-second trajectory (Lippe et al., 2023). Prescribed non-interior states are restored at each step, so the task is a boundary-conditioned flow forecast rather than a free forecast of every state value.

Graph-based simulators represent irregular spatial discretizations naturally. Graph Network-based Simulators learn dynamics through message passing (Sanchez-Gonzalez et al., 2020), and MeshGraphNets apply this idea to mesh-based simulation (Pfaff et al., 2021). However, local message passing needs multiple layers or scales to communicate across distant mesh regions, a recognized bottleneck for high-resolution simulations (Fortunato et al., 2022). Fourier neural operators (FNOs) supply global spectral mixing on regular grids (Li et al., 2021); geometry-adapted FNOs and geometry-informed neural operators extend spectral processing to non-Cartesian domains (Li et al., 2023a;b). These complementary mechanisms motivate a local–global operator for the room-scale transient setting. GINO is a closely related graph–Fourier operator, but its published evaluation predicts pressure for steady-state 3-D vehicle aerodynamics rather than forecasting five fields over 400 transient steps (Li et al., 2023b). We ran a feasibility pilot on our transient data with the same GPU and 200 training

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

windows per epoch. Ten GINO epochs took about 24 hours; by comparison, 100 A-MFO-Net epochs took 3.14 ± 0.06 hours. We did not complete a 100-epoch, three-seed GINO evaluation and therefore report no GINO accuracy or claim an accuracy advantage over it.

We study A-MFO-Net, a boundary-conditioned GNN–FNO–GNN operator. Graph stages encode mesh-cell states and reconstruct local detail around equipment; an intermediate FNO processes a regular latent grid. The graph stages score incoming messages using source and destination states together with directed edge geometry. Boundary-category features identify walls, flow cells, ACU inlets/outlets, and server inlets/outlets, while recorded states are restored outside the flow mask during autoregressive rollout.

Our contributions are twofold. First, we develop and evaluate a boundary-conditioned local–global operator with geometry-aware node–edge message weighting for five-channel 3-D forecasting on a held-out 22,000-cell extended-room mesh over 400 autoregressive steps (20 seconds), using the full mesh at evaluation rather than only the sampled training cells. Second, on the same frozen split, 200-window-per-epoch budget, and three seeds used by all compared models, A-MFO-Net reduces final-step relative-L_2 error by 68.6% for velocity (6.97% versus 22.20% for GAOT (Wen et al., 2025)) and 22.8% for pressure (31.56% versus 40.88% for Transolver++ (Luo et al., 2025)), the strongest external comparator for each field. A-MFO-Net has lower mean error than the attention-disabled MFO-Net in all three fields, while GAOT has the lowest mean temperature error (4.31%; Table 1). FNO has $(39.24 \pm 0.99)\%$ velocity error, providing an architecture-level test of the graph stages as a package, not an isolation of attention.

2 Related work

Data-center surrogates have long targeted fast thermal prediction. Tashiro et al. built a network model for future temperature distributions (Tashiro et al., 2015). Athavale et al. compared neural networks, support-vector regression, Gaussian processes, and proper orthogonal decomposition for rack-inlet temperatures using room-level CFD data (Athavale et al., 2019); Saiyad et al. modeled rack and facility thermal variables with neural networks (Saiyad et al., 2021). Tong et al. developed a control-oriented, time-varying state-space model for rack-temperature prediction (Tong et al., 2023). More directly related to our output fields, SC-PIGNN3D already demonstrated transient three-dimensional velocity, pressure, and temperature prediction in a data-center hot aisle (Wang et al., 2025). A-MFO-Net is therefore not presented as the first 3-D data-center surrogate. Instead, its evaluation brings together a 22,000-cell unstructured mesh, five coupled state channels, an extended equipment layout, and 400 sequential forecast updates. This joint spatial and temporal burden differs from predicting a few rack temperatures and makes local errors consequential over the full rollout (Lippe et al., 2023).

Boundary and geometric information also have substantial prior art. Boundary Graph Neural Networks model particle–boundary interactions in three-dimensional granular flows (Mayr et al., 2023); multiscale mesh GNNs have reconstructed three-dimensional fluid fields from coarse solutions (Barwey et al., 2025). Graph attention (Veličković et al., 2018), edge-featured graph learning (Gong & Cheng, 2019), and joint node–edge feature processing (Yang & Li, 2020) predate our node–edge score. Geometry-informed and graph-based operators such as GINO, RIGNO, and GAOT address irregular-domain PDE learning through different constructions (Li et al., 2023b; Mousavi et al., 2025; Wen et al., 2025). Long-horizon neural PDE work separately emphasizes that one-step quality need not translate into stable autoregressive behavior (Lippe et al., 2023; Koehler et al., 2024). Our focus is the boundary-conditioned integration of local graph processing and global spectral mixing, assessed by final-step field errors over a demanding 3-D thermal-fluid rollout.

3 Method

3.1 Problem formulation

Let $\mathbf{x}_i^t = [u_{x,i}^t, u_{y,i}^t, u_{z,i}^t, p_i^t, T_i^t] \in \mathbb{R}^5$ be the state at mesh cell i and time t, and let $\mathbf{X}^t = (\mathbf{x}_i^t)_{i=1}^N$ be the full N-cell state. Every cell has normalized coordinate $\mathbf{q}_i \in \mathbb{R}^3$ and six-channel boundary feature $\mathbf{b}_i \in [0,1]^6$. Point-level one-hot boundary labels are averaged onto mesh cells, so the resulting cell features can be fractional. Let $\mathcal{N}(i)$ contain the K incoming nearest neighbors of cell i, and

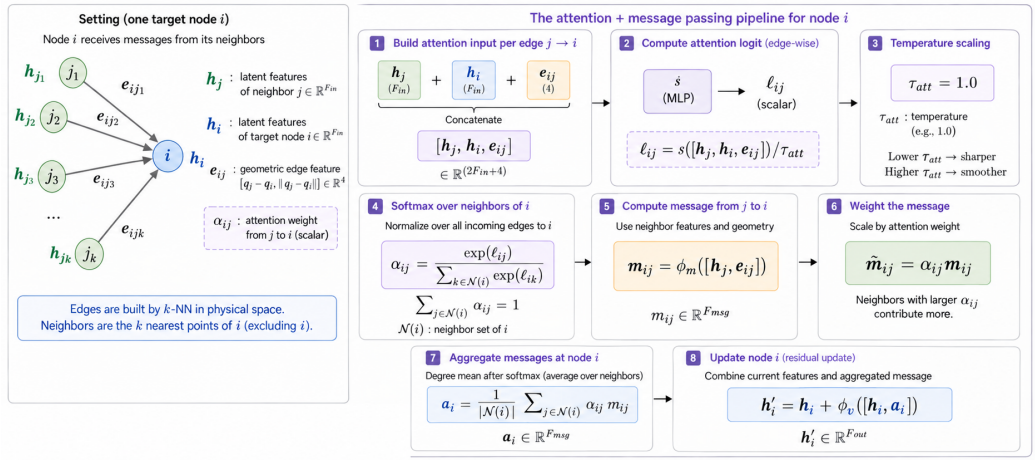

Figure 1: Node–edge score weighting for destination mesh cell i. Directed geometry $\mathbf{e}_{ij}$ and latent states $\mathbf{h}_j, \mathbf{h}_i$ determine scalar logit ℓ_{ij}; neighbor-wise softmax gives α_{ij}. Edge-conditioned messages $\mathbf{m}_{ij}$ are aggregated with the implemented degree mean before the residual update. The reported configuration uses attention temperature $\tau_{\text{att}} = 1.0$.

let $\mathcal{E} = \{(j, i) : j \in \mathcal{N}(i)\}$ be the directed edge set. We write the edge feature from j to i as $\mathbf{e}_{ij} = [\mathbf{q}_j - \mathbf{q}_i, \|\mathbf{q}_j - \mathbf{q}_i\|] \in \mathbb{R}^4$, containing three relative-coordinate channels and distance. We learn the one-step map

$$\widehat{\mathbf{X}}^{t+1} = \mathcal{F}_\theta(\{\mathbf{x}_i^t, \mathbf{q}_i, \mathbf{b}_i\}_{i=1}^N, \mathcal{E}), \tag{1}$$

and apply it autoregressively at test time. The six boundary categories are wall, flow/interior, ACU inlet, ACU outlet, server inlet, and server outlet.

3.2 Local–Global Mesh Operator

Figure 2 summarizes A-MFO-Net. A graph encoder first maps $[\mathbf{x}_i^t, \mathbf{q}_i, \mathbf{b}_i]$ to a latent feature at each mesh cell. We splat these features to a 32^3 regular grid with trilinear weights, concatenate a static signed-distance field, and apply a 3-D FNO. The resulting grid features are sampled back to the original cells with the same trilinear interpolation mechanism and decoded by a second graph network. Thus, the FNO supplies a global communication path while the graph stages retain local mesh and boundary information.

Both graph stages use edge-conditioned messages and the same learned node–edge scoring form. For a destination cell i,

$$\begin{aligned}
\ell_{ij} &= s([\mathbf{h}_j, \mathbf{h}_i, \mathbf{e}_{ij}])/\tau_{\text{att}}, \\
\alpha_{ij} &= \frac{\exp(\ell_{ij})}{\sum_{k \in \mathcal{N}(i)} \exp(\ell_{ik})}, \\
\mathbf{m}_{ij} &= \phi_m([\mathbf{h}_j, \mathbf{e}_{ij}]), \\
\mathbf{a}_i &= \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} \alpha_{ij} \mathbf{m}_{ij}, \\
\mathbf{h}'_i &= \mathbf{h}_i + \phi_v([\mathbf{h}_i, \mathbf{a}_i]).
\end{aligned} \tag{2}$$

Here, s, ϕ_m, and ϕ_v are learned MLPs, ℓ_{ij} is the attention logit, and $\tau_{\text{att}} = 1.0$ is the attention temperature. The score depends on the two endpoint states and directed geometry, whereas the message value depends on the source state and edge geometry. The explicit degree mean shown in Eq. 2 is the implemented aggregation: although $\sum_{j \in \mathcal{N}(i)} \alpha_{ij} = 1$, the coefficients of the messages in the mean sum to $1/|\mathcal{N}(i)|$. This is an implementation-level design choice, not a new attention primitive.

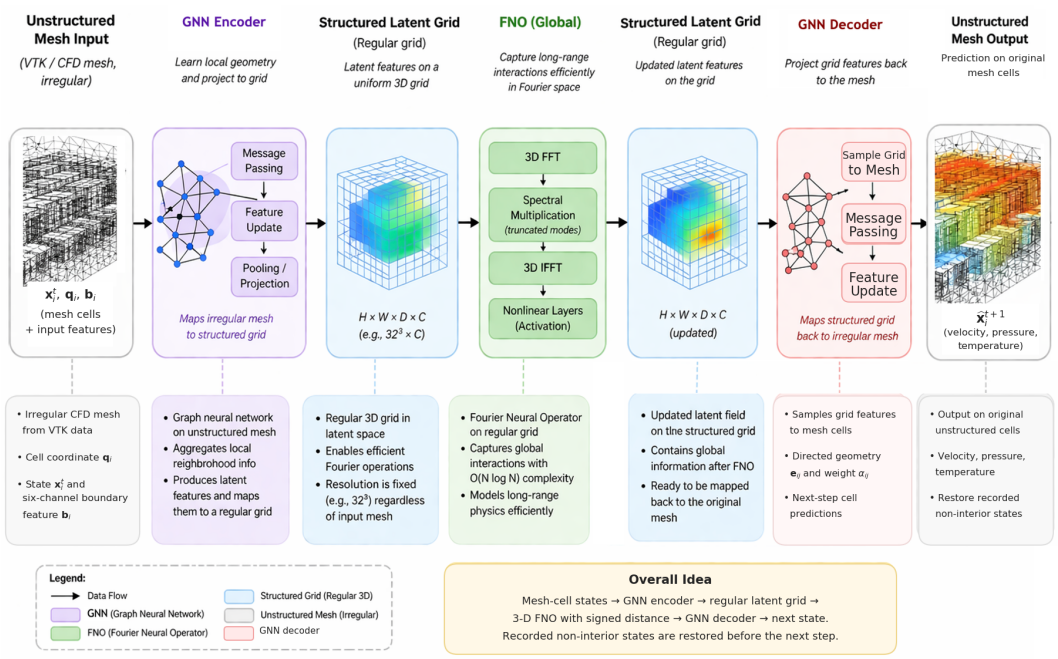

Figure 2: A-MFO-Net maps mesh-cell states $\mathbf{x}_i^t$, coordinates $\mathbf{q}_i$, and boundary features $\mathbf{b}_i$ to a regular 32^3 latent grid. A static signed-distance field enters the 3-D FNO; updated grid features are sampled back to mesh cells and decoded to $\widehat{\mathbf{x}}_i^{t+1}$. Recorded ground-truth states in non-interior cells are restored before the next rollout step. The diagram uses the same directed edge geometry $\mathbf{e}_{ij}$ and attention weight α_{ij} as Eq. 2.

3.3 Boundary treatment and training

We train with a three-step autoregressive unroll. The training loss is restricted to flow cells, identified by a value greater than 0.5 in the flow/interior boundary-feature channel. During rollout, recorded ground-truth values replace predictions outside this mask before the next autoregressive input. The same conditioning rule is applied to every evaluated model.

4 Experimental protocol

The held-out extended-room data-center case uses 90% ACU cooling and 100% server workload. Its domain spans $3.6\,\mathrm{m} \times 3.17\,\mathrm{m} \times 9\,\mathrm{m}$ along the x, y, and z axes, respectively, with eight server racks and two ACUs. The room-scale domain spans $3.6\,\mathrm{m} \times 3.17\,\mathrm{m} \times 6\,\mathrm{m}$ and contains four racks and one ACU. Each grid box shown in Figure 3 represents a $0.6\,\mathrm{m} \times 0.6\,\mathrm{m} \times 0.6\,\mathrm{m}$ spatial module. The test trajectory contains 401 states spanning 20 seconds (400 forecast transitions at a 0.05-second cadence) and approximately 22,000 mesh cells per state; an explicit manifest keeps this case out of training and validation. Of the 14 training trajectories, 12 are room-scale cases with 200 transitions, one is an extended-room case with 200 transitions at 80% ACU cooling and 100% server workload, and one is a room-scale case with 400 transitions at 100% ACU cooling and 0% server workload. Validation comprises two additional room-scale cases with 200 transitions. Training is therefore predominantly on 200-step room-scale data, while also including one shorter extended-room example and one longer room-scale example. The held-out test probes their joint transfer to the extended-room, 90%-cooling, full-server, 400-step setting; it does not isolate unseen-geometry or unseen-horizon generalization by itself. All comparison runs reuse the same manifest-defined case split.

For A-MFO-Net, the two graph processors each have width 64 and two message-passing layers; the FNO uses 64 latent channels and four spectral layers. We construct an eight-neighbor graph, map to a 32^3 grid, and sample 2,048 mesh cells per training item. A-MFO-Net training uses FP32, batch size two, 100 epochs, and three-step autoregressive unrolls. All seven reported models, including

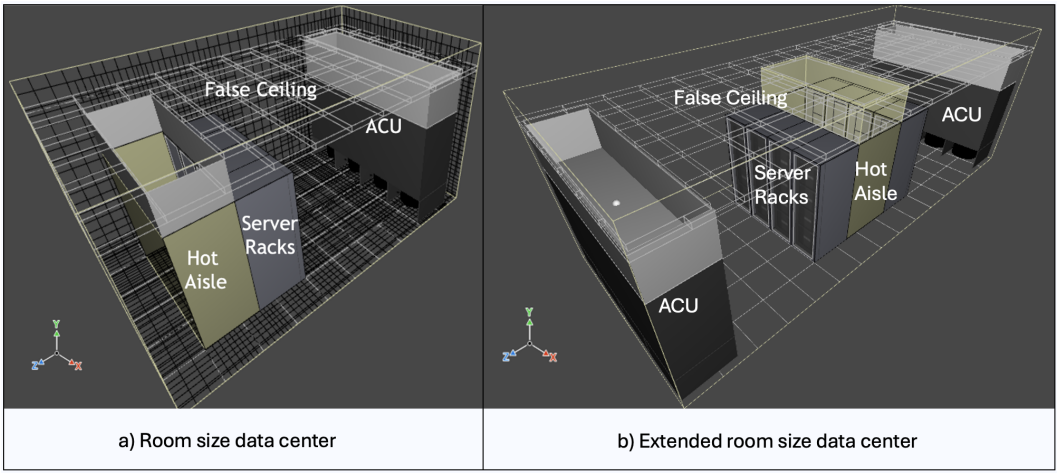

Figure 3: The two data-center simulation geometries. (a) The room-scale domain measures $3.6\,\mathrm{m} \times 3.17\,\mathrm{m} \times 6\,\mathrm{m}$ along (x, y, z) and contains four server racks and one air-conditioning unit (ACU). (b) The extended-room domain measures $3.6\,\mathrm{m} \times 3.17\,\mathrm{m} \times 9\,\mathrm{m}$ and contains eight racks and two ACUs. Each grid box shown represents a $0.6\,\mathrm{m} \times 0.6\,\mathrm{m} \times 0.6\,\mathrm{m}$ spatial module. The image labels the hot aisles and false ceiling; coordinate triads show the shared axis convention.

MeshGraphNet, use the same frozen split and exactly 200 selected three-step training windows per epoch for each of three runs (seeds 42, 43, and 44), irrespective of source-trajectory length. Each test rollout uses the complete 22,000-cell mesh for 400 autoregressive transitions. We report relative-L_2 error at the final forecast step 400 ($t = 20$ s), using the same evaluator for every model. Figures 4 and 5 provide qualitative snapshots at $t = 1$, 10, and 20 seconds (forecast steps 20, 200, and 400).

Table 1 reports final-step means and sample standard deviations over seeds 42, 43, and 44 for every model. The seven rollout-error curves in Appendix Figure 8 and the qualitative snapshots visualize the seed-42 member of those same three-run experiments; the curves therefore do not represent three-seed means.

For field f at forecast step t, relative-L_2 error is

$$\mathrm{RelL}_2^{(f,t)}(\%) = 100\sqrt{\frac{\sum_{i=1}^{N} \|\widehat{\mathbf{x}}_{i,f}^{t} - \mathbf{x}_{i,f}^{t}\|_2^2}{\sum_{i=1}^{N} \|\mathbf{x}_{i,f}^{t}\|_2^2}}. \tag{3}$$

GAOT, MeshGraphNet, Transolver++, FIGConvUNet, FNO, and the attention-disabled MFO-Net control use the same data reader, case partition, state channels, boundary handling, 200-window-per-epoch budget, three training seeds, and 400-step evaluator as A-MFO-Net. MFO-Net retains the graph and FNO stages but disables learned attention in both graph processors. FNO retains the 32^3 spectral core but replaces the graph stages with fixed mesh–grid interpolation and a pointwise head; Appendix B specifies the architectural differences and plots prediction error throughout the rollout.

5 Results

At forecast step 400 ($t = 20$ s), A-MFO-Net's mean relative-L_2 errors are $(6.97 \pm 0.18)\%$ for velocity, $(31.56 \pm 4.75)\%$ for pressure, and $(5.05 \pm 0.15)\%$ for temperature (Table 1). Relative to the strongest external comparator for each field, these are 68.6% less velocity error than GAOT (22.20%) and 22.8% less pressure error than Transolver++ (40.88%). A-MFO-Net has the lowest mean velocity and pressure errors overall, while GAOT has the lowest mean temperature error, $(4.31 \pm 0.39)\%$.

The attention-disabled MFO-Net control has final-step means and sample standard deviations of $(7.48 \pm 0.20)\%$ for velocity, $(33.42 \pm 5.34)\%$ for pressure, and $(5.42 \pm 0.16)\%$ for temperature. A-MFO-Net's mean is lower in all three fields. Disabling attention also removes the scorer and

Table 1: Relative-L_2 error at the final forecast step 400 ($t = 20$ s) on the extended-room data-center test trajectory (90% ACU cooling, 100% server workload). Entries are mean $\pm$ sample standard deviation over seeds 42, 43, and 44; all seven models use 200 training windows per epoch. Bold marks the lowest mean per field. MFO-Net disables attention, and FNO is the graph-free control described in Appendix B.

Model	Velocity (%)	Pressure (%)	Temperature (%)
A-MFO-Net	**(6.97 $\pm$ 0.18)%**	**(31.56 $\pm$ 4.75)%**	(5.05 $\pm$ 0.15)%
MFO-Net	(7.48 $\pm$ 0.20)%	(33.42 $\pm$ 5.34)%	(5.42 $\pm$ 0.16)%
FNO	(39.24 $\pm$ 0.99)%	(168.83 $\pm$ 204.73)%	(8.67 $\pm$ 0.63)%
GAOT	(22.20 $\pm$ 1.17)%	(48.04 $\pm$ 7.32)%	**(4.31 $\pm$ 0.39)%**
MeshGraphNet	(45.75 $\pm$ 3.10)%	(70.68 $\pm$ 11.70)%	(14.27 $\pm$ 2.00)%
Transolver++	(53.93 $\pm$ 9.07)%	(40.88 $\pm$ 6.87)%	(11.40 $\pm$ 2.82)%
FIGConvUNet	(57.30 $\pm$ 3.90)%	(79.40 $\pm$ 17.10)%	(8.77 $\pm$ 1.67)%

changes effective message normalization under the implemented degree-mean aggregation, so these differences cannot be attributed solely to learned edge scoring.

The matched FNO control removes graph message passing and uses fixed mesh–grid interpolation with pointwise projection. Its final-step errors are $(39.24 \pm 0.99)\%$ for velocity, $(168.83 \pm 204.73)\%$ for pressure, and $(8.67 \pm 0.63)\%$ for temperature. A-MFO-Net has lower error for all three fields in each paired seed. The large FNO pressure standard deviation reflects one run with 404.77% final-step error; the other two yield 63.68% and 38.04%. Thus, the pressure mean is not representative of a stable control. The comparison supports the implemented graph-containing pipeline, but does not isolate neighbor aggregation or attention: the graph-free model also changes its pointwise projections, FNO input, and parameter count (Appendix B).

The qualitative comparison is split across Figures 4 and 5 to keep the time-resolved panels legible. Figure 4 shows A-MFO-Net, its attention-disabled counterpart, and FNO; Figure 5 shows GAOT, MeshGraphNet, Transolver++, and FIGConvUNet, each alongside the same ground truth. The snapshots sample forecast steps 20, 200, and 400; Table 1 gives the final-step quantitative comparison, also visualized in Appendix Figure 7. The models retain the broad room-scale circulation structure while displaying distinct local arrow and pressure-plane patterns.

The error maps in Appendix Figure 6 show spatially heterogeneous discrepancies. Velocity and temperature errors concentrate in repeated equipment-level bands and jet-transition regions, while pressure error is more diffuse and remains visible along both the room-length and equipment-height directions. This pattern is consistent with the difficulty of reconstructing sharp local gradients around cooling outlets and rack passages, although the maps alone do not establish a causal source. Clipping affects visualization only and is disclosed in the caption.

Across three seeds, the complete A-MFO-Net 400-step evaluation takes 64.0 ± 1.3 seconds, including VTK input/output, boundary restoration, and metric calculation. This is an end-to-end evaluation wall time, not model-only latency. Standard deviations reflect training-run variation on one held-out trajectory, not test-case uncertainty; Appendix Figure 8 shows how prediction error evolves over the full rollout.

6 Limitations and scope

The three-seed standard deviations characterize training variation on one test trajectory, not variation across geometries or operating conditions; Appendix Figure 8 shows seed 42 only. All models use 200 windows per epoch, but learning rates and parameter counts differ. FNO changes spectral inputs and pointwise projection as well as removing graph stages; disabling attention in MFO-Net also changes message normalization. Neither comparison isolates one mechanism. One FNO run has a large pressure error, making its pressure mean unstable.

Because training includes a 200-step extended-room case and a 400-step room-scale case, the held-out 400-step extended-room test combines geometry, operating-condition, and horizon transfer rather than establishing either independently.

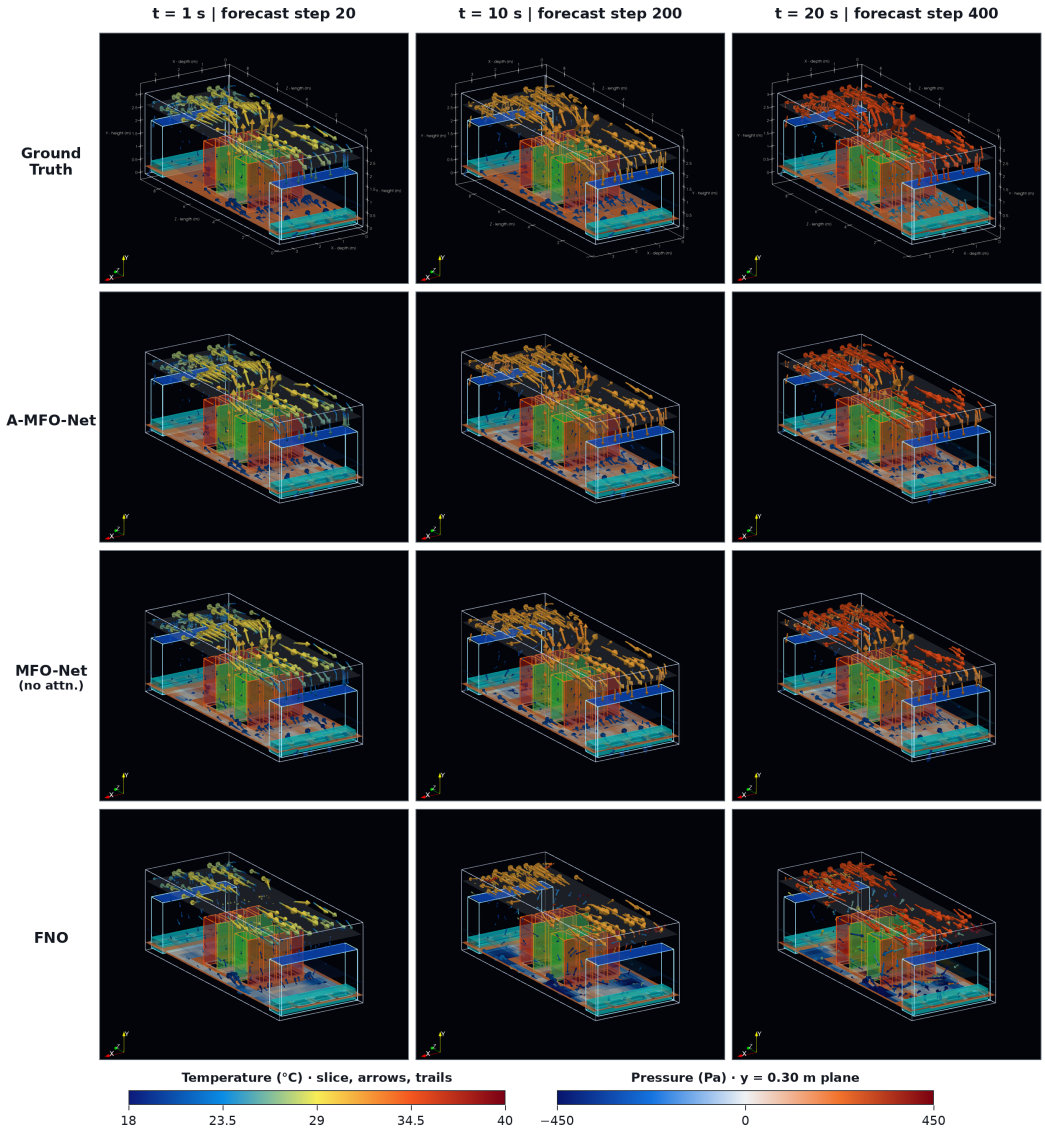

Figure 4: Extended-room rollout comparison (first of two figures): ground truth, A-MFO-Net, MFO-Net with attention disabled, and FNO at $t = 1$, 10, and 20 seconds (forecast steps 20, 200, and 400; 0.05-second cadence; 90% ACU cooling and 100% server workload). These qualitative panels show seed-42 outputs; quantitative results aggregate seeds 42, 43, and 44. The moving-arrow visualization uses 260 forecast-flow markers and 80 shared, prescribed ceiling-return context markers (20 per z band); the latter are not model predictions. Arrow and trail colors encode temperature on a shared 18–40°C scale; displayed arrow lengths are capped at 15 m/s. The temperature slice is at $y = 0.80$ m, and pressure is shown on the $y = 0.30$ m plane (-450–450 Pa). Equipment key: blue/cyan boxes = ACU inlet/outlet; red/green boxes = server inlet/outlet; orange wireframes = eight racks; pale slab = false ceiling; white wireframe = room envelope. Rack and ceiling outlines are inferred from solid-wall geometry.

The A-MFO-Net evaluation time is descriptive, not a controlled throughput comparison. More held-out conditions are needed.

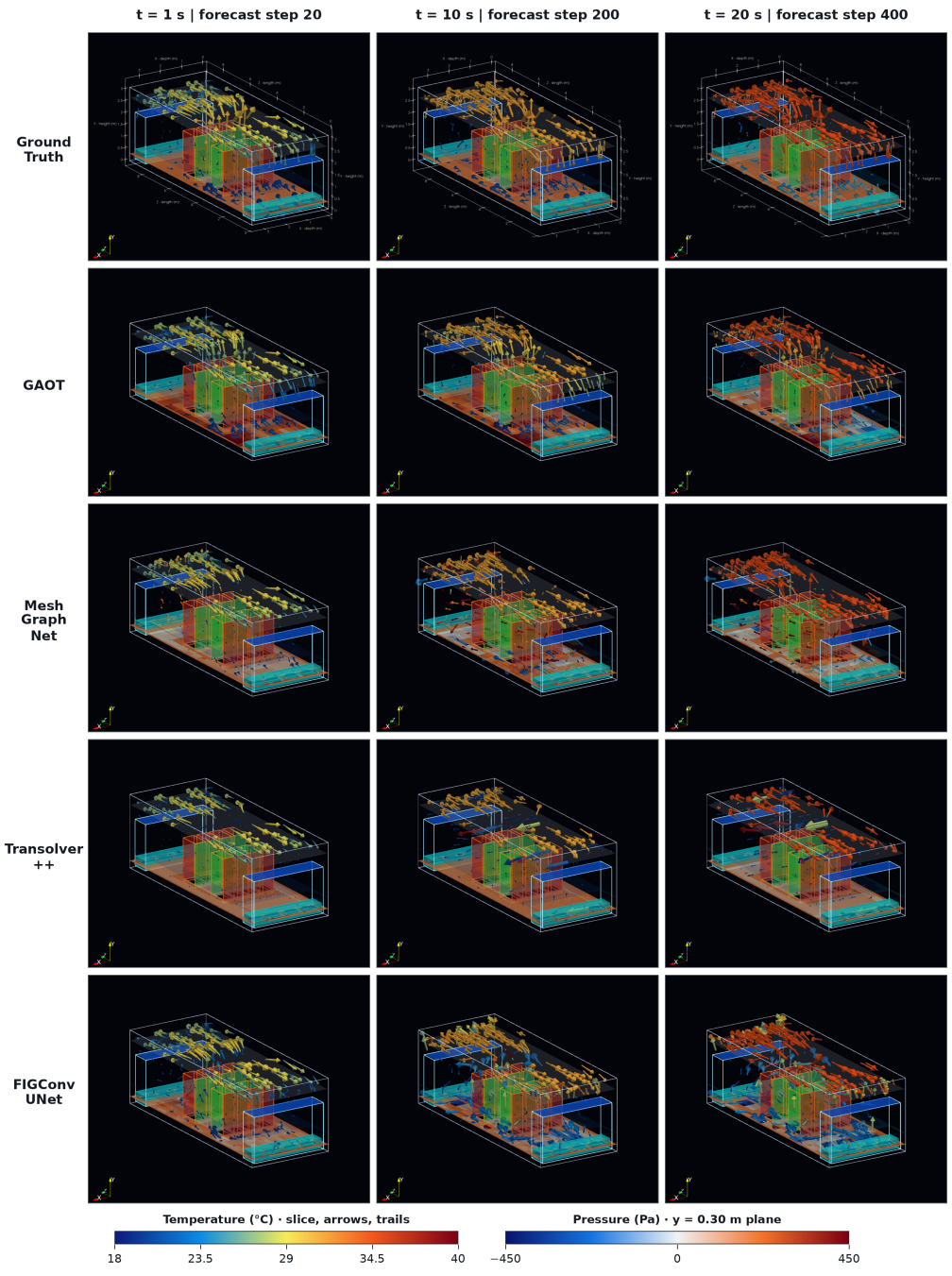

Figure 5: Extended-room rollout comparison (second of two figures): ground truth, GAOT, MeshGraphNet, Transolver++, and FIGConvUNet at $t = 1$, 10, and 20 seconds (forecast steps 20, 200, and 400). These qualitative panels show seed-42 outputs; the ground-truth row is repeated for direct comparison. Moving-arrow markers, prescribed ceiling-return context, equipment outlines, temperature and pressure slices, camera, and shared color limits match Figure 4.

7 Conclusion

We presented A-MFO-Net, a boundary-conditioned GNN–FNO–GNN operator for 3-D thermal-fluid rollout on an unstructured mesh. On the held-out 22,000-cell extended-room case (90% ACU

cooling, 100% server workload), it predicts five state channels over 400 steps (20 seconds). Final-step relative-L_2 errors are $(6.97 \pm 0.18)\%$ for velocity, $(31.56 \pm 4.75)\%$ for pressure, and $(5.05 \pm 0.15)\%$ for temperature: 68.6% less velocity and 22.8% less pressure error than the strongest external comparator for each field. A-MFO-Net has the lowest mean velocity and pressure errors, and GAOT the lowest mean temperature error. FNO has higher error in every paired seed and field, supporting the implemented graph-containing pipeline without isolating any one graph component. The MFO-Net ablation trails A-MFO-Net in all three mean errors but also changes message normalization, so it does not establish a causal attention benefit. A-MFO-Net's complete evaluation takes about 64 seconds. Broader held-out tests and more targeted controls remain important future work.

AI USE STATEMENT

Generative-AI tools were used to assist with constructing this manuscript and visually polishing the network-pipeline figure. We did not use generative AI to generate synthetic data, establish theoretical results, or generate the reported numerical measurements. All reported numerical values were computed from saved model outputs using local scripts. The human authors are responsible for reviewing the final technical content, citations, figures, disclosure, and submission materials, and for the final content of this work.

ETHICS STATEMENT

This study uses simulated CFD trajectories and does not involve human participants, personal data, or clinical decisions. The proposed model is evaluated as a research surrogate rather than as an autonomous controller. Any deployment that informs cooling or operational decisions should be independently validated under the target operating conditions, because rollout errors and the use of recorded non-interior ground-truth states can limit transfer beyond the reported protocol.

REPRODUCIBILITY STATEMENT

All seven models share the frozen split, 200 training windows per epoch, seeds 42, 43, and 44, and the 400-step evaluator. Table 1 reports three-seed final-step statistics; Appendix Figure 8 shows seed 42 only. Training configurations, seed-level metrics, and figure scripts are retained author-side. The source archive compiles the paper but excludes raw CFD data, the runtime, and seed-level outputs.

REFERENCES

J. Athavale, M. Yoda, and Y. Joshi. Comparison of data driven modeling approaches for temperature prediction in data centers. *International Journal of Heat and Mass Transfer*, 135:1039–1052, 2019. doi: 10.1016/j.ijheatmasstransfer.2019.02.041. URL https://doi.org/10.1016/j.ijheatmasstransfer.2019.02.041.

Shivam Barwey, Pinaki Pal, Saumil Patel, Riccardo Balin, Bethany Lusch, Venkatram Vishwanath, Romit Maulik, and Ramesh Balakrishnan. Mesh-based super-resolution of fluid flows with multiscale graph neural networks. *Computer Methods in Applied Mechanics and Engineering*, 443:118072, 2025. doi: 10.1016/j.cma.2025.118072. URL https://doi.org/10.1016/j.cma.2025.118072.

Miyuru Dayarathna, Yonggang Wen, and Rui Fan. Data center energy consumption modeling: A survey. *IEEE Communications Surveys & Tutorials*, 18(1):732–794, 2016. doi: 10.1109/COMST.2015.2481183. URL https://doi.org/10.1109/COMST.2015.2481183.

Meire Fortunato, Tobias Pfaff, Peter Wirnsberger, Alexander Pritzel, and Peter Battaglia. MultiScale MeshGraphNets, 2022. URL https://arxiv.org/abs/2210.00612.

Liyu Gong and Qiang Cheng. Exploiting edge features for graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9211–9219, 2019. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Gong_

Exploiting_Edge_Features_for_Graph_Neural_Networks_CVPR_2019_paper.html.

Felix Koehler, Simon Niedermayr, Rüdiger Westermann, and Nils Thuerey. APEBench: A benchmark for autoregressive neural emulators of partial differential equations. In *Advances in Neural Information Processing Systems*, volume 37, pp. 120252–120310. Curran Associates, Inc., 2024. doi: 10.52202/079017-3822. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/d9875ebcf74bccdc5076acab0dbee62c-Abstract-Datasets_and_Benchmarks_Track.html.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=c8P9NQVtmnO.

Zongyi Li, Daniel Zhengyu Huang, Burigede Liu, and Anima Anandkumar. Fourier neural operator with learned deformations for PDEs on general geometries. *Journal of Machine Learning Research*, 24(388):1–26, 2023a. URL https://jmlr.org/papers/v24/23-0064.html.

Zongyi Li, Nikola Kovachki, Chris Choy, Boyi Li, Jean Kossaifi, Shourya Otta, Mohammad Amin Nabian, Maximilian Stadler, Christian Hundt, Kamyar Azizzadenesheli, and Animashree Anandkumar. Geometry-informed neural operator for large-scale 3D PDEs. In *Advances in Neural Information Processing Systems*, volume 36, pp. 35836–35854. Curran Associates, Inc., 2023b. doi: 10.52202/075280-1556. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/70518ea42831f02afc3a2828993935ad-Abstract-Conference.html.

Phillip Lippe, Bas Veeling, Paris Perdikaris, Richard E. Turner, and Johannes Brandstetter. PDE-Refiner: Achieving accurate long rollouts with neural PDE solvers. In *Advances in Neural Information Processing Systems*, volume 36, pp. 67398–67433. Curran Associates, Inc., 2023. doi: 10.52202/075280-2946. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d529b943af3dba734f8a7d49efcb6d09-Abstract-Conference.html.

Huakun Luo, Haixu Wu, Hang Zhou, Lanxiang Xing, Yichen Di, Jianmin Wang, and Mingsheng Long. Transolver++: An accurate neural solver for PDEs on million-scale geometries. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 41432–41449. PMLR, 2025. URL https://proceedings.mlr.press/v267/luo25o.html.

Andreas Mayr, Sebastian Lehner, Arno Mayrhofer, Christoph Kloss, Sepp Hochreiter, and Johannes Brandstetter. Boundary graph neural networks for 3D simulations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 9099–9107, 2023. doi: 10.1609/aaai.v37i8.26092. URL https://ojs.aaai.org/index.php/AAAI/article/view/26092.

Sepehr Mousavi, Shizheng Wen, Levi E. Lingsch, Maximilian Herde, Bogdan Raonic, and Siddhartha Mishra. RIGNO: A graph-based framework for robust and accurate operator learning for PDEs on arbitrary domains. In *Advances in Neural Information Processing Systems*, volume 38, pp. 150039–150101. Curran Associates, Inc., 2025. doi: 10.52202/085713-5018. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/dcb91f43033bb1d367d1848806dee98d-Abstract-Conference.html.

Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter W. Battaglia. Learning mesh-based simulation with graph networks. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=roNqYL0_XP.

Anashusen Saiyad, Asif Patel, Yogesh Fulpagare, and Atul Bhargav. Predictive modeling of thermal parameters inside the raised floor plenum data center using artificial neural networks. *Journal of Building Engineering*, 42:102397, 2021. doi: 10.1016/j.jobe.2021.102397. URL https://doi.org/10.1016/j.jobe.2021.102397.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter W. Battaglia. Learning to simulate complex physics with graph networks. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8459–8468, 2020. URL https://proceedings.mlr.press/v119/sanchez-gonzalez20a.html.

Shinya Tashiro, Yuya Tarutani, Go Hasegawa, Yutaka Nakamura, Kazuhiro Matsuda, and Morito Matsuoka. A network model for prediction of temperature distribution in data centers. In *2015 IEEE 4th International Conference on Cloud Networking*, pp. 261–266, 2015. doi: 10.1109/CloudNet.2015.7335319. URL https://doi.org/10.1109/CloudNet.2015.7335319.

Xiaoxi Tong, Jiaqiang Wang, Weiwei Liu, Hodo-Abalo Samah, Quan Zhang, and Linfeng Zhang. A time-varying state-space model for real-time temperature predictions in rack-based cooling data centers. *Applied Thermal Engineering*, 230:120737, 2023. doi: 10.1016/j.applthermaleng.2023.120737. URL https://doi.org/10.1016/j.applthermaleng.2023.120737.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

Yidi Wang, Aik Beng Ng, Simon See, Daniel Wang, and Frank Guan. Simulating 3D thermal fluid dynamics in data centers with soft-constrained physics-informed graph neural network. In *Proceedings of the SIGGRAPH Asia 2025 Posters*, 2025. doi: 10.1145/3757374.3771448. URL https://doi.org/10.1145/3757374.3771448.

Shizheng Wen, Arsh Kumbhat, Levi E. Lingsch, Sepehr Mousavi, Yizhou Zhao, Praveen Chandrashekar, and Siddhartha Mishra. Geometry aware operator transformer as an efficient and accurate neural surrogate for PDEs on arbitrary domains. In *Advances in Neural Information Processing Systems*, volume 38, pp. 155423–155501. Curran Associates, Inc., 2025. doi: 10.52202/085713-5197. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/e45a448dfa778f6d62729a7bc8633c06-Abstract-Conference.html.

Yulei Yang and Dongsheng Li. NENN: Incorporate node and edge features in graph neural networks. In *Proceedings of the 12th Asian Conference on Machine Learning*, volume 129 of *Proceedings of Machine Learning Research*, pp. 593–608. PMLR, 2020. URL https://proceedings.mlr.press/v129/yang20a.html.

A Extended-room error visualizations

Figure 6 shows A-MFO-Net's final-step spatial errors on two mesh slices. Figure 7 visualizes the relative-L_2 results summarized in Table 1, including the MFO-Net control.

B Graph-free FNO control and rollout-error trajectories

The FNO control uses the same five state channels, normalized cell coordinates, six boundary features, signed-distance grid, 14/2/1 case partition, training seeds, 100 epochs, 200 selected three-step training windows per epoch, and boundary-conditioned 400-step evaluator as A-MFO-Net. It splats cell features onto the 32^3 grid using fixed trilinear weights, supplies grid occupancy and signed distance to a four-layer, 64-channel 3-D FNO, and interpolates its output back to cells for a pointwise residual head. It does not read graph edges or compute node-to-node messages. Training samples 2,048 cells per item; testing uses all 22,000 cells. This is an architecture-level graph-free control, not an isolated removal of neighbor aggregation: the FNO input changes from 65 to 16 channels, its pointwise head differs from the graph decoder, and parameter counts are not matched.

Figure 8 plots relative-L_2 error at every forecast step. In the displayed paired run, FNO has larger step-400 velocity, pressure, and temperature errors than A-MFO-Net. The pressure curve also places A-MFO-Net below MFO-Net at step 400, consistent with their aggregate mean ordering in Table 1. The large FNO pressure variability in the table reflects another run and is not represented by this curve. These observations do not identify which change between the two architectures is responsible;

a pointwise-encoder/decoder control with matched spectral input would be needed for a narrower graph-stage claim. The attention-disabled control changes effective message normalization as well as learned scoring.

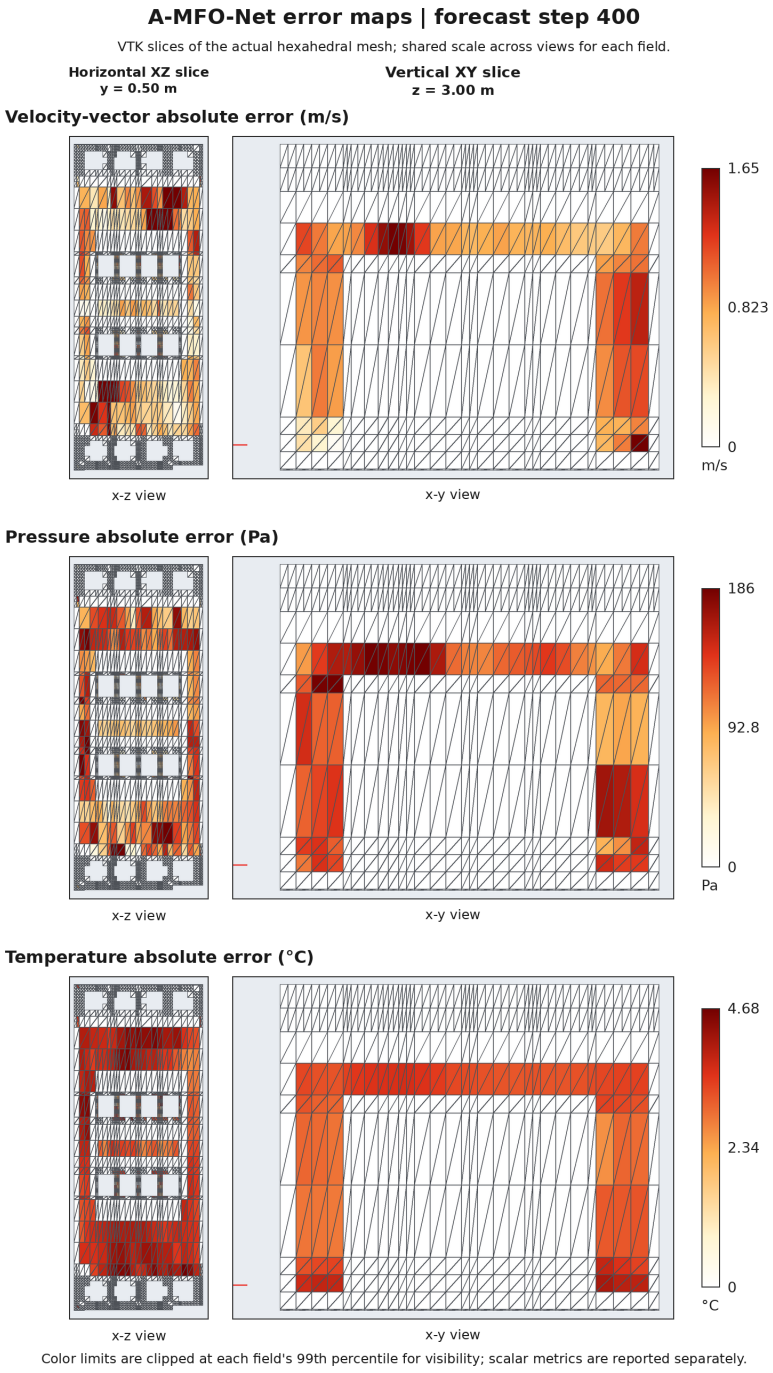

Figure 6: VTK mesh-colored A-MFO-Net error maps at forecast step 400 ($t = 20$ seconds) on two slices of the actual hexahedral grid: a horizontal XZ slice at $y = 0.50$ m and a vertical XY slice at $z = 3.00$ m. Velocity is the norm of the vector error; pressure and temperature are absolute errors in Pa and °C. Visible cell edges expose mesh resolution, and each field shares one color scale across both views. Upper limits are clipped at the field's 99th percentile for visibility; clipping affects only the visualization, while scalar results are reported in Table 1.

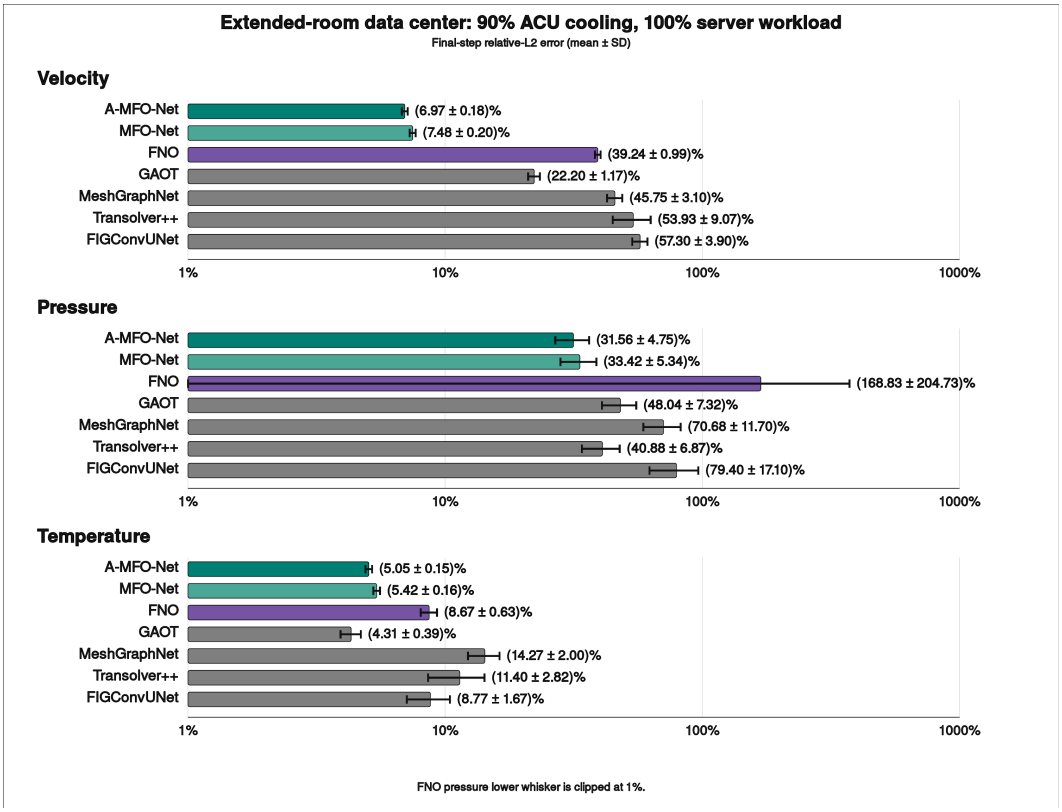

Figure 7: Relative-L_2 error at forecast step 400 ($t = 20$ s) of the extended-room rollout at 90% ACU cooling and 100% server workload. Bars show the three-seed means and sample standard deviations in Table 1 (200 training windows per epoch). The horizontal axis is logarithmic; the FNO pressure mean minus one standard deviation lies below zero and its left whisker is clipped at the 1% plotting boundary.

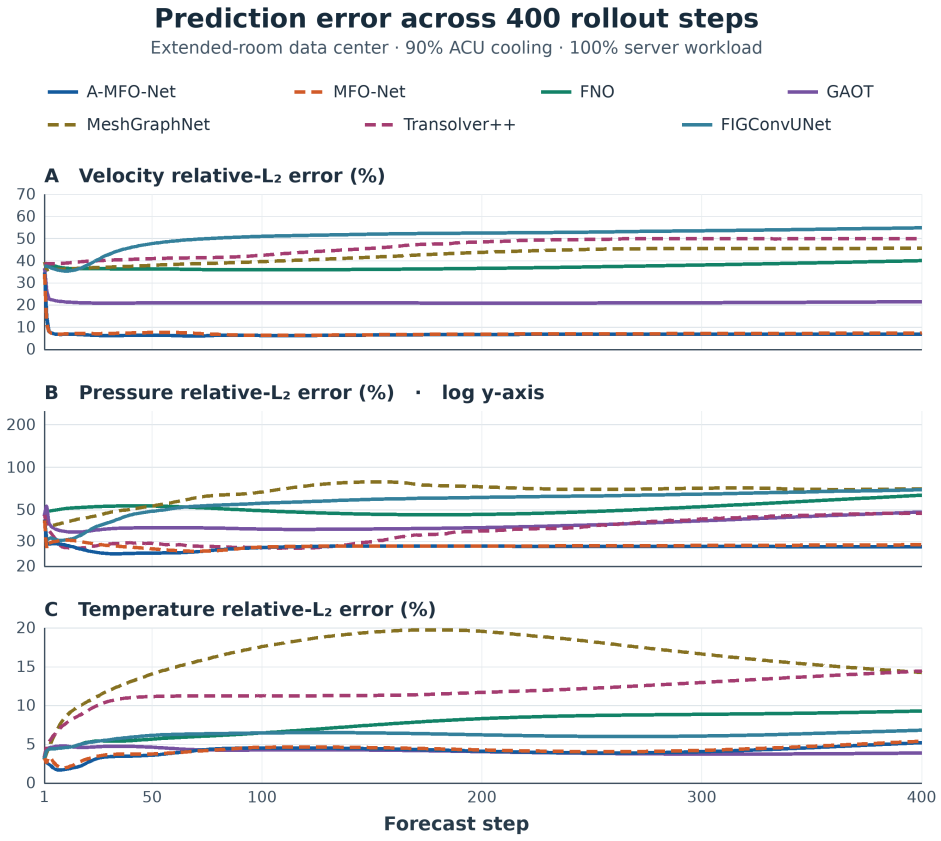

Figure 8: Relative-L_2 error across forecast steps 1–400 on the held-out extended-room data-center trajectory, rather than training loss. Panels show velocity, pressure, and temperature error in percent. The pressure axis is logarithmic. Curves show seed 42 from the same three-run training protocol, without uncertainty bands; Table 1 gives three-seed final-step variability.