Performance and Scalability Analysis of a MoE Model for MPDATA on Multi-GPU HPC System

Weronika Folwarska[1][0009−0002−9003−4898], Krzysztof Rojek[1][0000−0002−2635−7345], Kamil Halbiniak[1][0000−0001−9116−8981], and Sergio Iserte[2][0000−0003−3654−7924]

[1] Czestochowa University of Technology, Poland
{weronika.folwarska,krzysztof.rojek,kamil.halbiniak}@pcz.pl
[2] Barcelona Supercomputing Center (BSC-CNS), Spain
sergio.iserte@bsc.es

Abstract. Numerical solvers based on the Multidimensional Positive Definite Advection Transport Algorithm (MPDATA) is widely used in geophysical and fluid dynamics simulations due to its stability and conservation properties. However, their computational cost becomes a limiting factor for long-term integrations and large ensemble simulations on high-resolution grids.

In this work, we investigate a data-driven surrogate modeling approach based on a Mixture-of-Experts (MoE) neural network trained to learn the temporal evolution of MPDATA simulations. The model takes as input three consecutive time steps of multiple three-dimensional prognostic fields and predicts the subsequent time step. Training data are generated using MPDATA, and the model is trained in a distributed manner on the MareNostrum supercomputer.

The proposed MoE architecture enables scalable model capacity through multiple specialized experts coordinated by a gating network, making it well-suited for high-performance computing environments. This work demonstrates the feasibility of combining large-scale deep learning with HPC infrastructures for surrogate modeling of complex numerical schemes.

Keywords: Surrogate modeling · Mixture-of-Experts · MPDATA · High-performance computing · Distributed training

1 Introduction

High-resolution numerical schemes for advection-dominated transport play a central role in geophysical fluid dynamics and related disciplines. Their ability to resolve sharp gradients while preserving key physical properties makes them indispensable in the simulation of tracer transport, mass conservation, and coupled multi-physics processes [14]. However, in three-dimensional configurations, the computational cost of such solvers increases rapidly with grid resolution, dimensionality, and integration length, which poses practical limitations for long simulations, ensemble-based studies, and systematic parameter sweeps [13].

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

Among these schemes, MPDATA has emerged as a widely adopted method due to its robustness [17], exact conservation, and strict preservation of non-negativity. These properties are achieved through an iterative corrective structure, which—while numerically advantageous—introduces a non-negligible per-timestep computational cost. In modern HPC settings, this cost becomes particularly relevant when simulations are executed repeatedly or at scale, motivating the exploration of complementary, data-driven approaches.

In this work, we investigate a neural surrogate model designed to interact with MPDATA-generated data. The model ingests multiple consecutive solver states comprising a scalar field x and three auxiliary fields (f_1, f_2, f_3), and predicts the scalar field at the subsequent time step. Importantly, the surrogate does not attempt to predict the auxiliary fields themselves. While this constitutes a limitation in constructing a fully autonomous solver replacement, it reflects a deliberate modeling choice. The scalar field x represents the primary transported quantity of interest and is subject to strict physical constraints, such as non-negativity and mass conservation. Focusing the surrogate on predicting x allows us to isolate and study the learnability of MPDATA's conservative transport behavior, while retaining the auxiliary fields as conditioning information provided by the numerical solver.

From this perspective, the present model should be understood as a preliminary but physically meaningful step toward hybrid MPDATA–ML workflows. It enables targeted experiments in which the surrogate acts as a corrective or approximative component for the scalar transport update, and it provides a controlled setting for evaluating whether learned predictions can preserve global invariants, such as total mass, when coupled with MPDATA states. At the same time, we acknowledge that the current validation is limited to this reduced setting and does not yet cover the full spectrum of coupled field dynamics.

Beyond model design, the primary contribution of this paper lies in a detailed performance analysis of the proposed surrogate on modern GPU-based HPC infrastructure. In particular, we focus on execution characteristics relevant to large-scale deployment rather than algorithmic accuracy alone.

Contributions. The main contributions of this study are as follows:

- We present a data-driven surrogate model that predicts the MPDATA scalar update from multi-field, multi-timestep solver states, serving as a data-conditioned preliminary component for hybrid MPDATA–ML simulations.
- We demonstrate, through targeted experiments, that the surrogate can reproduce scalar evolution while respecting global mass constraints, despite not explicitly predicting auxiliary vector fields.
- We perform a comprehensive GPU performance analysis on the Barcelona Supercomputing Center GPU cluster, using nodes equipped with 4 GPUs and scaling experiments across multiple nodes.
- We investigate the dependence of training performance on batch size and numerical precision, comparing FP32, BF16, and FP16 arithmetic.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

– We evaluate scalability of the training workflow across multi-GPU and multi-node configurations, identifying practical bottlenecks and efficiency regimes relevant for large-scale surrogate training.

By emphasizing performance characteristics and scalability alongside physical consistency, this work aims to clarify how data-driven surrogate components can be integrated into high-performance simulation pipelines. At the same time, we note that the present study does not yet include a direct end-to-end comparison of execution time between the baseline MPDATA solver and a hybrid configuration with the neural surrogate. While such a quantitative evaluation of runtime speedup is essential for assessing practical benefits, it requires full integration of the surrogate into the simulation loop, including inference and communication overheads. Therefore, we position this work as an initial step focused on scalability and physical fidelity, leaving systematic runtime comparisons for future work.

2 MPDATA Background

Governing equation. MPDATA, originally developed by Smolarkiewicz [17], is an iterative finite-difference scheme for the numerical solution of conservative transport problems. In its classical formulation, the method advances a scalar field $\phi(\mathbf{x}, t)$ governed by the linear advection equation

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{u}\,\phi) = 0, \tag{1}$$

where $\mathbf{u}(\mathbf{x}, t)$ denotes a prescribed velocity field and ϕ represents a non-negative transported quantity such as a mass density or concentration.

Algorithmic structure. The MPDATA scheme is constructed as a sequence of conservative advection passes. The first pass applies a first-order upwind (donor-cell) discretization, which is unconditionally stable but introduces significant numerical diffusion. Subsequent corrective passes apply antidiffusive fluxes designed to cancel the leading-order truncation errors of the baseline scheme. These antidiffusive fluxes are expressed in terms of pseudo-velocities computed from the evolving solution and are applied using the same conservative upwind framework, ensuring stability and consistency across dimensions.

Accuracy and numerical properties. Through the iterative correction process, MPDATA achieves second-order accuracy in both space and time for smooth solutions while remaining largely non-oscillatory near sharp gradients. A defining characteristic of the scheme is its strict preservation of non-negativity: provided the initial scalar field satisfies $\phi(\mathbf{x}, t_0) \geq 0$, the numerical solution remains non-negative at all subsequent time levels under the standard CFL constraint. In addition, MPDATA is exactly conservative, preserving the discrete integral

$$\int_{\Omega} \phi(\mathbf{x}, t)\, \mathrm{d}\mathbf{x} \tag{2}$$

over closed computational domains.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

Scope and applications. These numerical properties have established MPDATA as a widely used transport scheme in geophysical fluid dynamics and related fields [15, 6]. The method forms the advection core of numerous atmospheric and oceanic models, where it is applied to the transport of moisture, chemical species, aerosols, and other tracers. Over time, MPDATA has been extended to handle variable-sign fields, implicit and semi-implicit time integration, and higher-order formulations, while retaining its core conservation and stability properties.

3 Dataset Generation and Input–Output Formulation

The dataset is generated through a controlled, procedural simulation pipeline designed to support supervised learning of short-horizon MPDATA dynamics. The objective is to construct a collection of physically admissible trajectories from which local time-stepping behavior can be inferred.

3.1 Computational domain and initial conditions

A total of 1000 independent simulations are performed on a uniform Cartesian grid of fixed resolution $32 \times 32 \times 32$. While the numerical grid resolution is fixed, the effective physical extent of the domain is varied between simulations by randomly sampling characteristic length scales, thereby introducing geometric variability without changing the discretization.

The scalar field $x(\mathbf{x}, t)$ represents a non-negative density-like quantity and is initialized to zero everywhere. On this background, a random number of spherical density inclusions (between 1 and 8) is embedded. For each sphere:

- The radius is sampled as a fraction of the domain extent, subject to a minimum size of three grid points to ensure adequate numerical resolution.
- The sphere center is sampled uniformly within the interior of the domain, with a margin equal to the radius to guarantee full containment.
- The density amplitude is defined by prescribing an edge value drawn from a bounded interval and a central enhancement capped to enforce an upper density limit.
- The internal density profile varies smoothly from the boundary value to the central value, avoiding sharp discontinuities.

Spheres may overlap or form clusters, producing a wide range of initial configurations spanning isolated structures, interacting blobs, and merged density regions.

In addition to the scalar field, three auxiliary fields f_1, f_2, and f_3 are defined on the same grid. Together, these fields represent the components of a vector quantity that enters the MPDATA update (e.g., an advecting velocity or a forcing-related field). All auxiliary fields are initialized consistently with the scalar field and subsequently evolved by the numerical scheme.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

3.2 Time integration using MPDATA

Each initial configuration is advanced for 50 discrete time steps using the MPDATA scheme. During the integration, the scalar field x is transported according to the governing advection dynamics, while MPDATA enforces strict non-negativity and exact conservation of the total scalar mass. The auxiliary fields f_1, f_2, and f_3 evolve consistently with the solver configuration and are retained as part of the prognostic state.

At each time step, the full solver state

$$\mathbf{X}^t = \{x^t,\ f_1^t,\ f_2^t,\ f_3^t\}$$

is stored for subsequent dataset construction.

4 Interaction Between MPDATA and the Data-Driven Model

Overview. We consider a hybrid solver in which an established advection scheme (MPDATA) produces high-fidelity 3D fields on a regular mesh, and a learned model provides a data-driven correction or closure for the scalar field. The solver state at each discrete physical time t comprises four 3D fields defined on a $32 \times 32 \times 32$ Cartesian mesh: a positive scalar field $x(\mathbf{r}, t) \geq 0$ and a three-component vector force $\mathbf{f}(\mathbf{r}, t) = (f_1, f_2, f_3)(\mathbf{r}, t)$. The learned model takes the MPDATA output at three consecutive time levels $t - 2\Delta t$, $t - \Delta t$, t (i.e., twelve 3D arrays: four physical quantities $\times$ three timesteps) and returns a prediction for the scalar field at the next time $t + \Delta t$ (one 3D array). Formally, with $\mathcal{S}_\tau = \{x(\cdot, \tau),\ f_1(\cdot, \tau),\ f_2(\cdot, \tau),\ f_3(\cdot, \tau)\}$, the model implements

$$\hat{x}(\cdot, t + \Delta t) = \mathcal{M}\big(\mathcal{S}_{t-2\Delta t},\ \mathcal{S}_{t-\Delta t},\ \mathcal{S}_t\big), \tag{3}$$

where $\hat{x}$ denotes the model prediction.

Coupling modes. Two principal coupling strategies are envisaged for production use:

1. **A posteriori correction (post-step)**: MPDATA advances the full state to time $t + \Delta t$ and the model computes a corrective increment δx that is applied to the MPDATA scalar, i.e. $x \leftarrow x + \delta x$. This mode is simple to integrate and preserves MPDATA's timestep stability properties while allowing the model to learn corrections (e.g., sub-grid mixing, numerical diffusion compensation).
2. **Predictor-corrector (in-line)**: the model predicts $\hat{x}(t + \Delta t)$, and MPDATA uses $\hat{x}$ as input for subsequent physics (for example, forcing-dependent terms). This mode couples physics tightly and can reduce split-operator errors, but requires more careful stability checks.

In the experiments described here, we use the *post-step* mode (model-as-corrector). The proposed integration strategy for the hybrid deployment is shown in Figure 1.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

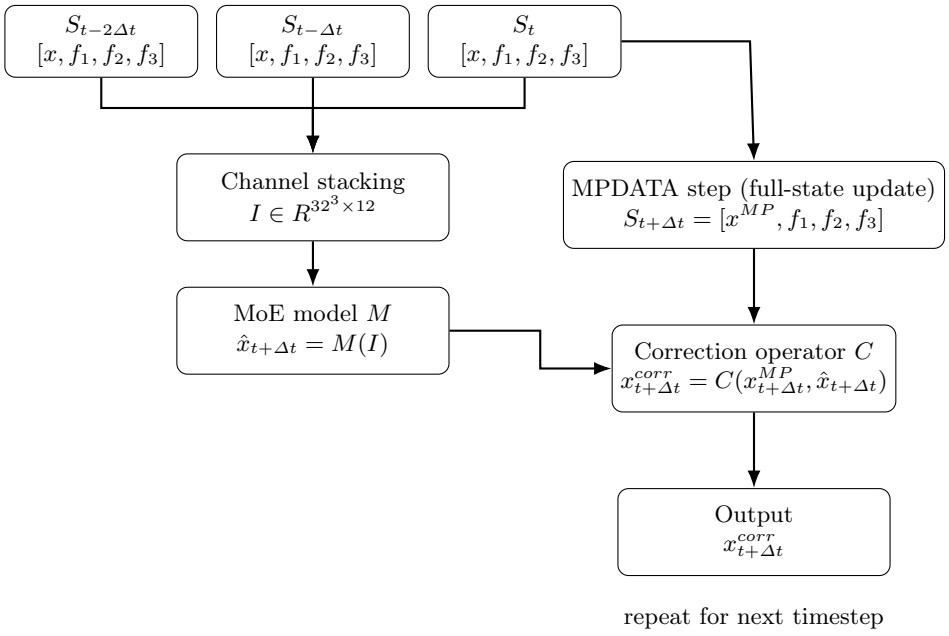

Fig. 1: Proposed integration strategy for the hybrid deployment.

Data representation. All fields are resampled to the same Eulerian grid and stacked into a channel dimension. The model uses channels-last tensors of shape $(32, 32, 32, 12)$ ($12 = 4$ physical fields $\times$ 3 time levels). The generator used in training performs an axis permutation so that stored NPZ arrays map into contiguous, GPU-friendly memory layouts; the training pipeline transforms batches into 'tf.data.Dataset' objects with prefetching to maximize device utilization.

5 Model Architecture and Training Details

Design goals. The model is designed to (i) exploit spatial locality in 3D structured grids, (ii) provide high representational capacity while controlling computation by sparsely activating specialization via an explicit MoE, and (iii) enforce physics-inspired priors (positivity where required, approximate mass conservation, and penalization of spurious mass outside a predefined support). The implementation conforms to the code listing provided and is summarized below.

High-level architecture. The network $\mathcal{M}$ is a local, voxel-wise MoE where the gating network predicts per-voxel weights that select at most two experts to contribute at that voxel. The pipeline is:

1. **Input:** The network input is a channels-last tensor $I \in R^{32\times32\times32\times12}$, corresponding to four physical fields (one scalar and three force components) over three consecutive time levels.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

2. **Gating network:** A $1\times1\times1$ 3D convolution produces per-voxel gating logits $L(\mathbf{r}) \in R^E$ for E experts. A *top-2 masking* operation is applied: for each voxel, only the two largest logits are retained via a binary mask, while the remaining expert channels are suppressed. The final gate values are obtained by multiplying a full softmax over all experts with this top-2 mask, yielding sparse, non-renormalized gating weights.
3. **Expert subnetworks:** The model contains E parallel expert subnetworks, each implemented as a compact 3D U-Net-like block [21]. Every expert processes the full input tensor and produces a multi-channel feature map in $R^{32\times32\times32\times n_f}$. All experts share the same topology but have independent parameters; expert 0 uses a ReLU activation, while the remaining experts use GELU activations.
4. **Mixture assembly:** Each expert feature map is multiplied voxel-wise by its corresponding gating weight (broadcast over channels). The weighted expert outputs are then summed to form a single fused feature representation.
5. **Readout head:** The fused representation is passed through two additional 3D convolutional layers. The final layer is a $1 \times 1 \times 1$ convolution producing a single-channel scalar field with linear activation and `float32` precision.

Mathematical formulation of gating and MoE. Let $I(\mathbf{r})$ be the input tensor at voxel $\mathbf{r}$. The gating logits are produced by a linear convolution

$$L(\mathbf{r}) = W_g * I(\mathbf{r}) \in R^E,$$

where $*$ denotes a $1 \times 1 \times 1$ convolution. Denote by $\mathcal{E}_i(I)$ the output of expert i (a scalar per voxel). The implemented top-2 selection computes the two largest entries of $L(\mathbf{r})$, forms a binary mask $m_i(\mathbf{r}) \in \{0, 1\}$ that is 1 for the top two indices and 0 otherwise, and then constructs gate weights

$$g_i(\mathbf{r}) = \frac{\exp(L_i(\mathbf{r}))}{\sum_{j=1}^{E} \exp(L_j(\mathbf{r}))}\, m_i(\mathbf{r}).$$

The fused output is

$$\hat{x}(\mathbf{r}) \;=\; \sum_{i=1}^{E} g_i(\mathbf{r})\, \mathcal{E}_i(I)(\mathbf{r}).$$

This design enforces sparsity (only two experts act per voxel) while retaining differentiability during training.

Expert block specification. Each expert is a shallow 3D U-Net block (function `make_expert` in code):

- 3D conv (n filters, kernel 3^3, activation = GELU or ReLU as noted) $\rightarrow$ skip connection.
- MaxPool3D(2) downsample.
- 3D conv ($2n$ filters, kernel 3^3).
- UpSampling3D(2).

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

– Concatenate skip and upsampled feature maps.
– 3D conv (n filters, kernel 3^3).

The base filter count is a hyperparameter ($base\_filters$, default 48). The readout path performs an additional 3D *conv* ($base\_filters/2$ filters) and a final $1 \times 1 \times 1$ *conv* to one channel (linear).

Loss function and physical priors. To encode the desiderata of positivity (where appropriate) and mass conservation, the training objective uses a composite loss (function *sphere_loss* in the code). Let $y(\mathbf{r})$ be the ground-truth scalar (target) and $\hat{y}(\mathbf{r})$ the model prediction. Define the indicator mask $\chi(\mathbf{r}) = \mathbf{1}\{y(\mathbf{r}) > 0\}$ for voxels belonging to the object of interest. The implemented loss is shown in Equation 4.

$$\begin{aligned}\mathcal{L} = \alpha \frac{1}{|\Omega|} \sum_{\mathbf{r}} \chi(\mathbf{r}) |y - \hat{y}| \\ + \beta \frac{1}{B} \sum_{b=1}^{B} \Big| \sum_{\mathbf{r}} \chi_b(\mathbf{r}) (\hat{y}_b - y_b)(\mathbf{r}) \Big| \\ + \gamma \frac{1}{|\Omega|} \sum_{\mathbf{r}} (1 - \chi(\mathbf{r})) \hat{y}(\mathbf{r})^2\end{aligned} \tag{4}$$

Data pipeline and implementation notes. Training data are stored in NPZ files and organized into input–target pairs for the standard splits ($train/val/test$). A custom data sequence class is used to stream data efficiently during training, employing memory-mapped file access to minimize host memory overhead. Each batch consists of input tensors of shape $(B, 32, 32, 32, 12)$, corresponding to three consecutive time levels of four physical fields, and target tensors of shape $(B, 32, 32, 32, 1)$ representing the scalar field at the next timestep.

The sequence is wrapped in a `tf.data` pipeline, which enables dataset repetition, optional shuffling during training, and asynchronous prefetching to overlap data transfer with GPU computation. This setup ensures stable throughput and efficient utilization of accelerator resources. As the dataset is configured to repeat indefinitely, the number of steps per epoch is explicitly controlled during training to maintain consistent epoch semantics.

Hyperparameters and practical choices. Unless stated otherwise, experiments use $E = 6$ experts, a base width of $n = 48$ convolutional filters per expert, and a batch size of 16. Most experts employ GELU activations to provide smooth, non-linear responses well suited for modeling complex, weakly non-linear transport phenomena. In addition, one expert uses a ReLU activation, explicitly biasing this subnetwork toward representations that preserve non-negativity. Given that the predicted scalar quantity is positive-defined, the inclusion of a ReLU-based expert encourages physically admissible feature extraction and stabilizes learning in regions dominated by advective transport, while the GELU-based experts retain the capacity to model softer transitions and diffusive effects.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

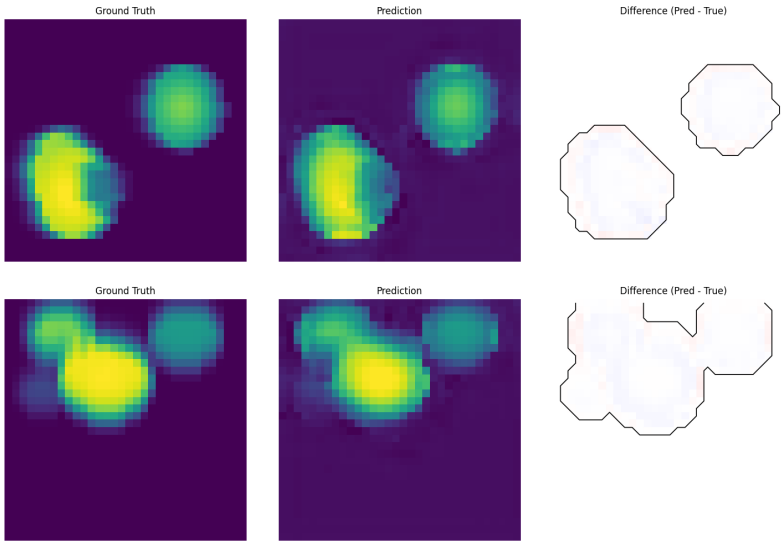

Fig. 2: Representative 2D slices of the scalar field (ground truth, prediction, and difference).

6 Performance and Accuracy Analysis

All the experiments were executed on the ACC partition of MareNostrum 5, using nodes equipped with four NVIDIA H100 GPUs (64 GB HBM) interconnected via NDR200 InfiniBand in a fat-tree topology.

The evaluation addresses two complementary aspects: (i) predictive accuracy of the proposed MoE model, and (ii) computational performance and scalability on the target HPC system.

6.1 Predictive Accuracy

The training objective defined in Eq. (4) decreases from approximately 1200 at initialization to 36.17 on the validation set. The normalized validation loss equals 0.273, closely matching the Mean Absolute Error (0.2734). The average mass conservation error per sample is 7.192, indicating accurate voxel-wise reconstruction together with preservation of global mass consistency (see Table 1).

Qualitative inspection confirms these numerical results. Predicted scalar fields preserve dominant transported structures and spatial coherence, while residual errors remain localized near sharper gradients and do not introduce spurious mass outside the physical support (see Figure 2).

To further assess conservation behavior, we compute the integrated mass within the positive-support region for each validation sample. Predicted and reference masses exhibit near-linear agreement, indicating negligible systematic

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

Table 1: Final validation results.

Metric	Value
Normalized validation loss	0.273
Mean Absolute Error (MAE)	0.2734
Mass Conservation Error (per sample)	7.192

bias. Figure 3 presents a qualitative comparison between the ground–truth simulation fields and the predictions produced by the model.

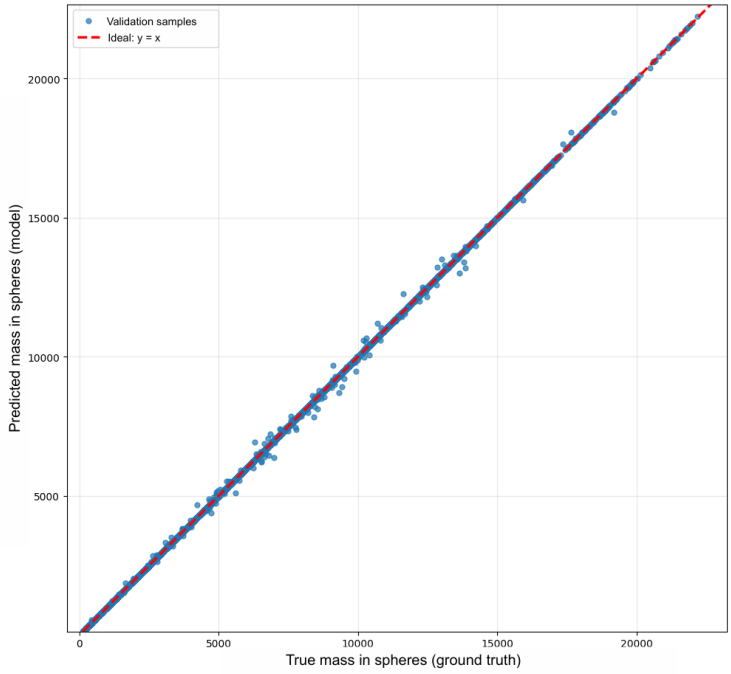

Fig. 3: Predicted versus true mass in the positive-support region (validation set). The dashed diagonal indicates perfect agreement.

6.2 Performance analysis

Before multi-node experiments, a single-GPU throughput sweep was conducted to determine an efficient numerical precision and batch size configuration. Epoch time was measured for FP32, FP16, and BF16 arithmetic across batch sizes from 2 to 128 (see Table 2).

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

Table 2: Single-GPU epoch time (in seconds) for different numerical precision and batch sizes.

	Batch size						
Precision	**2**	**4**	**8**	**16**	**32**	**64**	**128**
FP32	235	203	187	177	172	169	173
FP16	150	133	120	107	122	107	**105**
BF16	153	132	114	108	109	108	109

FP16 with batch size 128 achieves the shortest epoch time (105 s) without memory overflow and is therefore used in subsequent scaling experiments. For scalability tests, the per-GPU batch size is fixed at 128, implying linear growth of the global batch size with GPU count. We evaluate two placement strategies: (i) dense allocation using four GPUs per node, and (ii) distributed allocation using one GPU per node (see Table 3).

Dense placement achieves near-linear scaling up to 16 GPUs (13.6 s/epoch). In contrast, distributing the same number of GPUs across more nodes reduces efficiency (19.8 s/epoch at 16 GPUs), highlighting the impact of inter-node synchronization and communication overhead in multi-node data-parallel training. Keeping GPUs within fewer nodes yields higher training throughput on this system. While the total computational capacity is identical, the multi-node configuration introduces additional inter-node communication overhead, which is reflected in increased latency and reduced parallel efficiency compared to the single-node setup, where communication remains intra-node (see Figure 4).

Table 3: Scalability results for dense (4 GPUs per node) and distributed (1 GPU per node) placement strategies.

Total GPUs	Nodes	Time per epoch [s]	Speedup
1	1	105	1.00
Dense placement (4 GPUs per node)			
4	1	52.55	2.00
8	2	26.64	3.94
12	3	18.07	5.81
16	4	13.60	7.72
Distributed placement (1 GPU per node)			
4	4	54.20	1.94
8	8	29.80	3.52
12	12	22.50	4.67
16	16	19.80	5.30

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

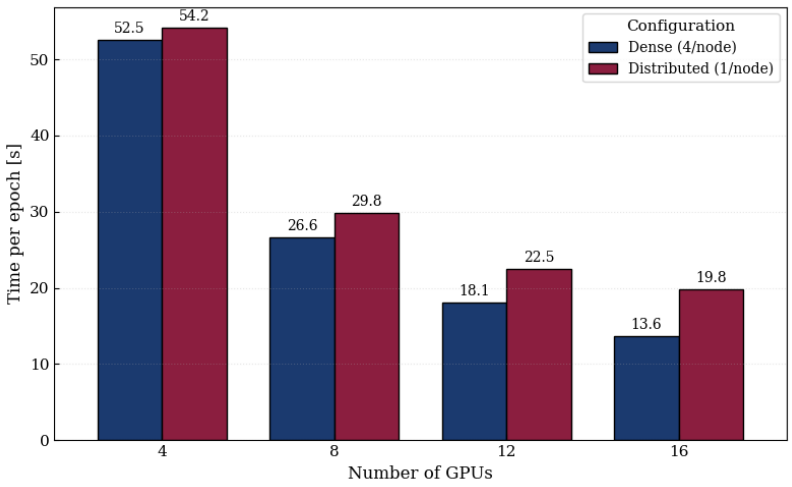

Fig. 4: Epoch time comparison for dense (4 GPUs per node) and distributed (1 GPU per node) configurations.

7 Related Work

High-resolution advection algorithms that enforce positivity and monotonicity are central to geophysical and atmospheric simulations, where spurious oscillations or excessive diffusion degrade physical fidelity. A prominent representative is MPDATA, which attains high-order accuracy through iterative corrective passes that reduce truncation errors in multidimensional transport while preserving sign and limiting numerical diffusion, even in the presence of sharp gradients and complex flows [17]. However, the repeated correction steps that underpin its robustness also impose substantial computational cost, particularly in large three-dimensional domains, motivating surrogate strategies that approximate its outputs at reduced expense.

Recent progress in scientific ML has enabled data-driven surrogates for computationally intensive CFD components. Early demonstrations showed that neural networks can augment or partially replace numerical solvers while retaining strong generalization properties [7]. Operator-learning frameworks, including Fourier Neural Operators (FNO), further extended this paradigm to parametric PDE families and large-scale flow prediction tasks [11, 8]. More recently, pretrained foundation models for PDEs have leveraged multiscale transformer architectures and large simulation corpora to achieve cross-operator generalization [5]. Complementary directions include hybrid U-Net/FNO architectures for turbulent flows [19], 3D FNO variants for real-time airflow modeling [2], graph-based surrogates for complex geometries [9], and attention-enhanced encoder–decoder

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

networks for accelerated flow-field prediction [1]. Surveys categorize these approaches into purely data-driven, physics-informed, and hybrid methodologies for accelerating advection-dominated and turbulent simulations [18, 12].

Beyond monolithic convolutional backbones, MoE architectures have emerged as an effective means of scaling model capacity while controlling per-sample computational cost. Sparse expert routing, originally introduced to expand neural capacity without proportional increases in inference time [16], has since been refined in large-scale MoE transformer systems that demonstrate strong scalability and efficiency across distributed hardware [10, 3]. These advances motivate the use of expert-based decomposition for physics surrogates, where heterogeneous flow structures may benefit from specialized subnetworks coordinated through learned gating mechanisms.

In contrast to prior work emphasizing convolutional ResNet-style surrogates or lightweight CNN accelerators for specific CFD tasks [4, 20], the present study adopts a 3D MoE architecture tailored to emulate MPDATA-generated scalar density fields and associated force vectors. Rather than focusing on heterogeneous data modalities, we concentrate on faithful reproduction of a well-defined numerical scheme across volumetric domains. The model distributes representation learning across multiple experts, enabling specialization for distinct flow regimes while maintaining tractable inference cost through sparse routing.

Training is conducted in a distributed setting on a multi-GPU cluster (up to 16 GPUs), leveraging data parallelism and expert parallelism to scale both dataset throughput and model capacity. This configuration aligns with contemporary large-scale MoE training practices [10, 3] and contrasts with CPU-oriented optimizations. Our objective is not solely runtime benchmarking, but rigorous quantitative and qualitative agreement with MPDATA outputs—including preservation of sharp fronts, localized structures, and full dynamic range—on unseen three-dimensional samples. By combining MPDATA-consistent supervision with scalable MoE training on GPU clusters, the proposed framework advances surrogate modeling for high-fidelity advection schemes in computational fluid dynamics.

8 Conclusion

This work presented a distributed Mixture-of-Experts surrogate model for learning the temporal evolution of MPDATA simulations in three-dimensional domains. The proposed architecture leverages multiple specialized experts coordinated through a gating mechanism, enabling scalable representational capacity while maintaining efficient training on modern GPU-based HPC systems.

From the modeling perspective, the MoE network achieves strong quantitative agreement with MPDATA outputs. The normalized validation loss (0.273) and MAE (0.2734) confirm accurate voxel-wise reconstruction, while the low mass conservation error and near-linear agreement between predicted and reference integrated mass demonstrate that the surrogate preserves key physical properties of the underlying transport process. Qualitative inspection further

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

shows that dominant transported structures and spatial coherence are retained, with errors localized primarily near sharper gradients.

From the computational perspective, the study highlights the importance of hardware-aware scaling strategies. Mixed-precision FP16 training with large batch sizes provides the best single-GPU performance. In multi-GPU experiments, dense placement (4 GPUs per node) achieves near-linear scaling up to 16 GPUs, reaching 13.6 s per epoch. In contrast, distributing the same number of GPUs across more nodes introduces measurable inter-node communication overhead, reducing parallel efficiency (19.8 s per epoch at 16 GPUs). These results demonstrate that communication patterns and node locality significantly influence training throughput in distributed data-parallel MoE setups.

Acknolwledgement

The researcher from BSC is involved in the Barcelona Zettascale Laboratory, backed by the Ministry for Digital Transformation and of Public Services, within the framework of the Recovery, Transformation, and Resilience Plan – funded by the European Union – NextGenerationEU.

References

1. Chen, X., Fu, C., Tie, M., Sham, C.W., Ma, H.: AFFNet: An Attention-Based Feature-Fused Network for Surface Defect Segmentation. Applied Sciences **13**(11) (2023). https://doi.org/10.3390/app13116428
2. Ding, X., Zhang, H., Zhang, W., et al.: A fourier neural operator-based method for rapid prediction of 3d indoor airflow dynamics. Building Simulation **18**(5), 1435–1451 (2025). https://doi.org/10.1007/s12273-025-1263-5, published online 2025
3. Fedus, W., Zoph, B., Shazeer, N.: Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. Journal of Machine Learning Research **23**(120), 1–39 (2022), http://jmlr.org/papers/v23/21-0998.html
4. Han, J.M., Malkawi, A.: Airvox: efficient computational fluid dynamics prediction using 3D convolutional neural networks for building design. Journal of Building Performance Simulation **18**(1), 1–16 (2025). https://doi.org/10.1080/19401493.2024.2410744
5. Herde, M., Raonic, B., Rohner, T., Käppeli, R., Molinaro, R., de Bezenac, E., Mishra, S.: Poseidon: Efficient Foundation Models for PDEs. In: The Thirty-eighth Annual Conference on Neural Information Processing Systems (2024)
6. Iserte, S., Martín-Álvarez, I., Rojek, K., Aliaga, J.I., Castillo, M., Folwarska, W., Peña, A.J.: Resource optimization with MPI process malleability for dynamic workloads in HPC clusters. Future Generation Computer Systems **174**, 107949 (2026). https://doi.org/https://doi.org/10.1016/j.future.2025.107949, https://www.sciencedirect.com/science/article/pii/S0167739X25002444
7. Kochkov, D., Smith, J.A., Alieva, A., Wang, Q., Brenner, M.P., Hoyer, S.: Machine learning–accelerated computational fluid dynamics. Proceedings of the National Academy of Sciences **118**(21), e2101784118 (2021). https://doi.org/10.1073/pnas.2101784118, https://www.pnas.org/doi/abs/10.1073/pnas.2101784118

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9

8. Kurth, T., Subramanian, S., Harrington, P., Pathak, J., Mardani, M., Hall, D., Miele, A., Kashinath, K., Anandkumar, A.: FourCastNet: Accelerating Global High-Resolution Weather Forecasting Using Adaptive Fourier Neural Operators. In: Proceedings of the Platform for Advanced Scientific Computing Conference. PASC '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3592979.3593412
9. Lei, B., Fu, Y.H., Cadena, J., Saini, A., Hu, Y., Bao, J., Xu, Z., Ng, B., Nguyen, P.: Accelerating computational fluid dynamics simulation of post-combustion carbon capture modeling with MeshGraphNets. Frontiers in Artificial Intelligence **7**, 1441985 (Jan 2025). https://doi.org/10.3389/frai.2024.1441985, published online January 7, 2025
10. Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., Krikun, M., Shazeer, N., Chen, Z.: GShard: Scaling Giant Models with Conditional Computation and Automatic Sharding. In: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net (2021)
11. Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., Anandkumar, A.: Fourier Neural Operator for Parametric Partial Differential Equations (2021), https://arxiv.org/abs/2010.08895
12. Mao, R., Lan, Y., Liang, L., Yu, T., Mu, M., Leng, W., Long, Z.: Rapid CFD Prediction Based on Machine Learning Surrogate Model in Built Environment: A Review. Fluids **10**(8) (2025). https://doi.org/10.3390/fluids10080193
13. Rojek, K., Wyrzykowski, R.: Parallelization of 3D MPDATA Algorithm Using Many Graphics Processors. In: Malyshkin, V. (ed.) Parallel Computing Technologies. pp. 445–457. Springer International Publishing, Cham (2015)
14. Rojek, K., Wyrzykowski, R.: Performance Modeling of 3D MPDATA Simulations on GPU Cluster. The Journal of Supercomputing **73**(2), 664–675 (2017)
15. Rojek, K., Wyrzykowski, R., Kuczynski, L.: Systematic Adaptation of Stencil-based 3D MPDATA to GPU Architectures. Concurrency and Computation: Practice and Experience **29**(9), e3970 (2017)
16. Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q.V., Hinton, G.E., Dean, J.: Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. ArXiv **abs/1701.06538** (2017), https://api.semanticscholar.org/CorpusID:12462234
17. Smolarkiewicz, P.K.: Multidimensional positive definite advection transport algorithm: an overview. International Journal for Numerical Methods in Fluids **50**(10), 1123–1144 (2006)
18. Wang, H., Cao, Y., Huang, Z., Liu, Y., Hu, P., Luo, X., Song, Z., Zhao, W., Liu, J., Sun, J., Zhang, S., Wei, L., Wang, Y., Wu, T., Ma, Z.M., Sun, Y.: Recent Advances on Machine Learning for Computational Fluid Dynamics: A Survey. ArXiv **abs/2408.12171** (2024), https://api.semanticscholar.org/CorpusID:271924513
19. Wang, Y., Yang, H., Zelong, Y., Li, Z., Peng, W., Wang, J.: A hybrid U-Net and Fourier neural operator framework for the large-eddy simulation of turbulent flows over periodic hills (04 2025). https://doi.org/10.48550/arXiv.2504.13126
20. Yaqoob, M., Ansari, M.Y., Ishaq, M., Ashraf, U., Pavuluri, S., Rabbani, A., Rabbani, H.S., Seers, T.D.: FluidNet-Lite: Lightweight convolutional neural network for pore-scale modeling of multiphase flow in heterogeneous porous media. Advances in Water Resources **200**, 104952 (2025). https://doi.org/https://doi.org/10.1016/j.advwatres.2025.104952
21. Zou, X., Tong, S., Peng, W., Huang, Q., Wang, J.: Residual U-Net for accurate and efficient prediction of hemodynamics in two-dimensional asymmetric stenosis. Physics of Fluids **37**(7) (07 2025). https://doi.org/10.1063/5.0274672

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_9