Physics-Guided Dual Neural ODEs with Bidirectional Information Exchange for Air Quality Prediction

Jiaxu Feng
Fudan University
Shanghai, China
feng.jiaxu@outlook.com

Jindong Tian
East China Normal University
Shanghai, China
jdtian@stu.ecnu.edu.cn

Yan Cen *
Fudan University
Shanghai, China
cenyan@fudan.edu.cn

Xinyuan Wei
Fudan University
Shanghai, China
xinyuanwei@fudan.edu.cn

Yun Xiong
Fudan University
Shanghai, China
yunx@fudan.edu.cn

Bin Yang
East China Normal University
Shanghai, China
byang@dase.ecnu.edu.cn

Abstract

Air pollution, a critical issue tied to urban life, is governed by complex physical processes that make accurate air quality prediction highly challenging. Recent physics-guided neural networks attempt to address this by modeling physical and data-driven branches independently and fusing their representations at the end. However, these approaches often suffer from error accumulation within each branch and difficulties in the effective fusion of representations. To address these problems, we propose **LIMBO** (**L**inkage **I**nter**M**ediaries **B**etween neural **O**rdinary differential equations), a physics-guided neural network augmented with an information exchange mechanism. LIMBO introduces bidirectional information exchange between the physical and data-driven branches and employs a dedicated LIMBO loss function to mitigate error accumulation and enhance collaboration. We further examine the effect of different exchange intervals on model performance and validate the contribution of the loss function through ablation studies. Experimental results show that LIMBO outperforms the state-of-the-art Air-DualODE model in PM2.5 forecasting, underscoring its practical value for real-world urban air quality prediction. The code is available at https://github.com/jiaxu-feng/LIMBO.

1 Introduction and related work

Urban air pollution poses a severe threat to global health, causing millions of deaths annually and reducing healthy life expectancy [18]. Accurate air quality forecasting is therefore crucial for protecting public health and guiding urban policy. However, predicting pollutant concentrations is challenging due to the complexity of the atmospheric system, where emissions, diffusion, advection, chemical reactions, and deposition interact with meteorological factors such as wind, temperature, humidity, precipitation, and air pressure [14]. Traditional physics-based numerical simulations offer

*Corresponding author

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: UrbanAI: Harnessing Artificial Intelligence for Smart Cities.

limited modeling capacity and are computationally expensive [24, 5, 21, 16]. In parallel, data-driven approaches [9, 17, 15, 11, 13] have gained increasing attention, accompanied by recent advances in time-series modeling [26, 20, 2, 6, 12] and spatiotemporal learning [22, 27, 28]. However, purely data-driven models often neglect underlying physical laws, leading to poor generalization. Therefore, it is difficult for either approach to simultaneously achieve high accuracy and robust predictive performance.

Recent studies have embedded diffusion–advection equations into Neural ODEs [1]. For example, AirPhyNet [8] is the first physics-guided deep learning framework applied to outdoor air quality forecasting. It models diffusion and advection with GNN-based ODE modules that encode physical information into a latent space, solves temporal dynamics via Neural ODEs, and finally decodes back to physical space. This approach substantially reduces errors in PM2.5 forecasting across different prediction horizons and under sparse data scenarios. Building on this, Air-DualODE [23] introduces a dual Neural ODE framework with parallel physical and data-driven branches. The physical branch captures dominant physical processes in the physical space using a diffusion–advection equation tailored to open systems, while the data-driven branch models dynamic information in the latent space that the physical branch cannot describe. The two branches cooperate through temporal alignment and fusion mechanisms to achieve state-of-the-art air quality forecasting performance across multiple scales and regions.

Nevertheless, existing models still face challenges in coordinating the physical and latent spaces. For example, although AirPhyNet incorporates diffusion and advection into its Neural ODE networks, it relies on a high-dimensional latent space to capture spatiotemporal dependencies. These latent variables are often nonlinear combinations or abstract mappings of physical quantities, and thus lack direct correspondence to explicit physical variables (e.g., diffusion coefficients, velocity fields), diminishing their physical interpretability. Similarly, while Air-DualODE decouples physical and latent representations by modeling them as two separate Neural ODEs, the two branches evolve independently and interact only through temporal alignment and representation fusion after each branch has been solved. This design overlooks stepwise interactions between physical space and latent space information, thereby complicating the fusion of the final embedded representations and limiting predictive accuracy.

To address these challenges, we propose **LIMBO** (**L**inkage **I**nter**M**ediaries **B**etween neural **O**DEs), a framework that introduces a bidirectional information exchange mechanism to enhance representation communication and fusion. The main contributions of this paper are as follows:

- We present a novel physics-guided dual Neural ODE architecture that incorporates a bidirectional information exchange mechanism between the physical and data-driven branches, complemented by a LIMBO loss function to mitigate error accumulation and strengthen inter-branch coordination.
- We systematically investigate the effect of different information exchange intervals on model performance, identify the optimal interval hyperparameter, and provide a detailed analysis of its influence.
- We conduct extensive experiments to evaluate the effectiveness of LIMBO and demonstrate that it outperforms the state-of-the-art Air-DualODE model under identical settings on the KnowAir dataset.

2 Methodology

2.1 Problem definition

The problem is formulated as follows [8, 23]: let there be N air-quality monitoring stations in the target region and D meteorological variables. Historical PM2.5 concentrations over the past T time steps are arranged as a tensor $X_{1:T} \in \mathbb{R}^{T\times N\times 1}$, and auxiliary meteorological and environmental covariates (e.g., temperature, wind speed) as $A_{1:T} \in \mathbb{R}^{T\times N\times (D-1)}$. Based on station locations we construct an undirected graph $G = (\mathbb{V}, \mathbb{E})$ with $|\mathbb{V}| = N$, and the forecasting task is to learn a mapping $f(\cdot)$ that, given $X_{1:T}$, $A_{1:T}$ and G, predicts concentrations for the next τ time steps, formally, $\tilde{X}_{T+1:T+\tau} \in \mathbb{R}^{\tau\times N\times 1}$ with

$$\tilde{X}_{T+1:T+\tau} = f\big(X_{1:T},\, A_{1:T},\, G\big).$$

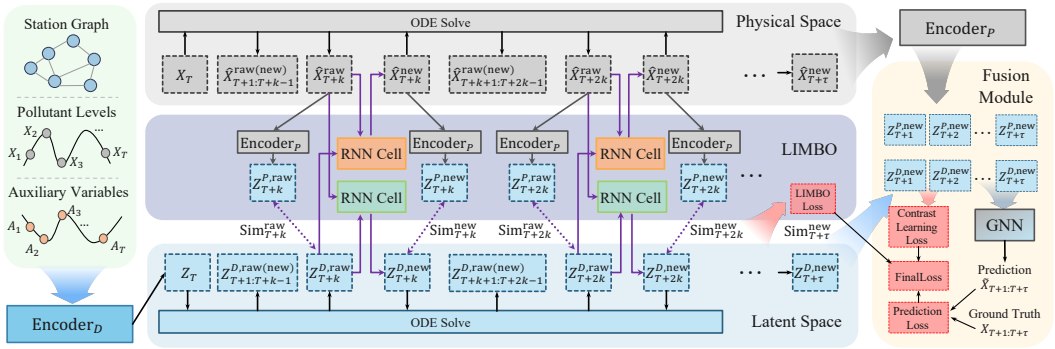

Figure 1: Our proposed architecture, which consists of the physical branch, the data-driven branch, the fusion module, and the information exchange mechanism (LIMBO)

2.2 Model overview

To address the lack of timely interactions between the physical and data-driven branches in the baseline Air-DualODE framework [23], we propose the LIMBO architecture, as shown in Figure 1. The architecture comprises the following three basic components from the baseline architecture (more details in Appendix B) and the information exchange mechanism described in Section 2.3.

Physical branch. The physical branch models atmospheric pollutant transport using an advection–diffusion equation (see Appendix A) for an open system. A Neural ODE captures spatiotemporal dependencies in physical space, and the ODE solver produces an initial multi-step forecast $\hat{X}_{T+1:T+\tau}$.

Data-driven branch. The data-driven branch encodes observations (pollutants and meteorological covariates) into a latent initial state Z_T and evolves it using a Neural ODE with spatial masked self-attention. This latent trajectory $Z^D_{T+1:T+\tau}$ captures complex effects not described by the advection–diffusion equation.

Fusion module. Physical predictions $\hat{X}_{T+1:T+\tau}$ are encoded into latent space via an encoder to obtain $Z^P_{T+1:T+\tau}$. It is fused with $Z^D_{T+1:T+\tau}$ using a graph neural network that aggregates spatial information and produces the final spatiotemporal prediction.

2.3 Information exchange mechanism

To enable dynamic cooperation and corrective feedback between the physical and data-driven branches, the model employs an explicit bidirectional information exchange mechanism implemented with two gated recurrent units (GRUs) [3, 4]. Time is partitioned into intervals of length k; within each interval, both branches evolve independently via their Neural ODE solvers. The physical branch solver produces a raw multi-step output $\hat{X}^{\text{raw}}_{T+mk+1:T+mk+k}$, and the data-driven branch solver produces a raw latent trajectory $Z^{D,\text{raw}}_{T+mk+1:T+mk+k}$. At the discrete boundary $t = T + mk + k$, information is exchanged in both directions through GRU updates that act as cross-branch messengers.

Specifically, one GRU takes the physical-space terminal forecast $\hat{X}^{\text{raw}}_{T+mk+k}$ as its input and treats the data-driven branch's raw latent $Z^{D,\text{raw}}_{T+mk+k}$ as the hidden state to be updated; the GRU output yields an updated latent $Z^{D,\text{new}}_{T+mk+k}$ which becomes the data-driven branch's initial status for the next ODE window. Symmetrically, the second GRU ingests $Z^{D,\text{raw}}_{T+mk+k}$ and updates the physical branch state $\hat{X}^{\text{raw}}_{T+mk+k}$ to produce $\hat{X}^{\text{new}}_{T+mk+k}$ for the ensuing physical integration. This bidirectional exchange thus provides mutual corrective signals: the physical branch injects physically plausible states into the data-driven latent dynamics, while the data-driven branch encodes unmodeled complex effects into physical updates.

The exchange interval k is a key hyperparameter controlling the trade-off between fidelity of continuous-time modeling and correction timeliness. Smaller k introduces more frequent updates, incurring higher computational costs and yielding evolution dynamics closer to a discrete-time

RNN. Larger k lowers overhead and preserves smoother ODE behavior but may delay inter-branch corrections. In practice, k is tuned to balance efficiency, continuity, and timely mutual correction.

2.4 LIMBO loss

To ensure that cross-branch exchanges lead to constructive alignment rather than destructive perturbation, we introduce the LIMBO loss: a regularizer that requires the similarity between the two branches' latent representations to increase after an exchange. This design allows each branch to maintain its unique expertise prior to the exchange, followed by a controlled fusion of information that intentionally drives the system toward consensus, thereby enhancing collaborative effectiveness. Denote by $\mathrm{Encoder}_P(\cdot)$ the encoder that maps a sequence of physical forecasts into latent space. At exchange time $T + mk$ we form the *raw* and *new* latent representations from the physical branch:

$$Z^{P,\mathrm{raw}}_{T+mk} = \mathrm{Encoder}_P\big(\hat{X}^{\mathrm{new}}_{T:T+mk-1},\, \hat{X}^{\mathrm{raw}}_{T+mk}\big), \qquad Z^{P,\mathrm{new}}_{T+mk} = \mathrm{Encoder}_P\big(\hat{X}^{\mathrm{new}}_{T:T+mk}\big),$$

and the data-driven branch produces corresponding $Z^{D,\mathrm{raw}}_{T+mk}$ and $Z^{D,\mathrm{new}}_{T+mk}$ before and after GRU updates. Define

$$\mathrm{Sim}^{\mathrm{raw}}_{T+mk} = \cos\big(Z^{P,\mathrm{raw}}_{T+mk},\, Z^{D,\mathrm{raw}}_{T+mk}\big), \qquad \mathrm{Sim}^{\mathrm{new}}_{T+mk} = \cos\big(Z^{P,\mathrm{new}}_{T+mk},\, Z^{D,\mathrm{new}}_{T+mk}\big).$$

The LIMBO loss aggregates a smooth hinge penalty over exchange points, which penalizes cases where post-exchange similarity does not exceed pre-exchange similarity.

$$\mathrm{LIMBOLoss} = \frac{k}{\tau}\sum_{m=1}^{\tau/k} \log\Big(1 + \exp\big(\mathrm{Sim}^{\mathrm{raw}}_{T+mk} - \mathrm{Sim}^{\mathrm{new}}_{T+mk}\big)\Big).$$

The model is trained using three complementary loss functions: the prediction loss, which measures the discrepancy between predictions and ground truth; the contrastive learning loss, which aligns latent representations prior to fusion; and the LIMBO loss, which promotes constructive information exchange between branches. The pseudocode of the LIMBO architecture is shown below.

Algorithm 1 LIMBO

Require: Past pollutant levels $X_{1:T}$, auxiliary covariates $A_{1:T}$, and graph structure G

$Z_T \leftarrow \mathrm{Encoder}_D\,(X_{1:T},\, A_{1:T})$
for $m = 0$ to $\tau/k - 1$ **do**
  $\hat{X}^{\mathrm{raw}}_{T+mk+1:T+mk+k} \leftarrow \mathrm{ODESolve}\left(\hat{X}^{\mathrm{new}}_{T+mk},\, \mathbf{F^P},\, [T+mk+1, \ldots, T+mk+k];\, \Theta\right)$
  $Z^{D,\mathrm{raw}}_{T+mk+1:T+mk+k} \leftarrow \mathrm{ODESolve}\left(Z^{D,\mathrm{new}}_{T+mk},\, \mathbf{F^D},\, [T+mk+1, \ldots, T+mk+k];\, \Phi\right)$
  $\hat{X}^{\mathrm{new}}_{T+mk+k} \leftarrow \mathrm{GRU}_{D\to P}\left(\mathrm{input} = Z^{D,\mathrm{raw}}_{T+mk+k},\, \mathrm{hidden} = \hat{X}^{\mathrm{raw}}_{T+mk+k}\right)$
  $Z^{D,\mathrm{new}}_{T+mk+k} \leftarrow \mathrm{GRU}_{P\to D}\left(\mathrm{input} = \hat{X}^{\mathrm{raw}}_{T+mk+k},\, \mathrm{hidden} = Z^{D,\mathrm{raw}}_{T+mk+k}\right)$
  $\hat{X}^{\mathrm{new}}_{T+mk+1:T+mk+k-1} \leftarrow \hat{X}^{\mathrm{raw}}_{T+mk+1:T+mk+k-1}$
  $Z^{D,\mathrm{new}}_{T+mk+1:T+mk+k-1} \leftarrow Z^{D,\mathrm{raw}}_{T+mk+1:T+mk+k-1}$
  $Z^{P,\mathrm{raw}}_{T+mk+k} \leftarrow \mathrm{Encoder}_P\left(\hat{X}^{\mathrm{new}}_{T:T+mk+k-1},\, \hat{X}^{\mathrm{raw}}_{T+mk+k}\right)$
end for
$Z^{P,\mathrm{new}}_{T+1:T+\tau} \leftarrow \mathrm{Encoder}_P\left(\hat{X}^{\mathrm{new}}_{T+1:T+\tau}\right)$
$\tilde{X}_{T+1:T+\tau} \leftarrow \mathrm{GNN}\left(Z^{P,\mathrm{new}}_{T+1:T+\tau}, Z^{D,\mathrm{new}}_{T+1:T+\tau}\right)$
$\mathrm{PredictionLoss} \leftarrow \mathrm{MAE}\left(\tilde{X}_{T+1:T+\tau},\, X^{GT}_{T+1:T+\tau}\right)$
$\mathrm{ContrastLearningLoss} \leftarrow \mathrm{ContrastLearning}\left(Z^{P,\mathrm{new}}_{T+1:T+\tau},\, Z^{D,\mathrm{new}}_{T+1:T+\tau}\right)$
$\mathrm{LIMBOLoss} \leftarrow \mathrm{LIMBOLoss}\left(Z^{P,\mathrm{new}}_{T+1:T+\tau},\, Z^{D,\mathrm{new}}_{T+1:T+\tau}, Z^{P,\mathrm{raw}}_{T+1:T+\tau}, Z^{D,\mathrm{raw}}_{T+1:T+\tau}\right)$
$\mathrm{FinalLoss} \leftarrow \mathrm{PredictionLoss} + \alpha \times \mathrm{ContrastLearningLoss} + \beta \times \mathrm{LIMBOLoss}$
return $\tilde{X}_{T+1:T+\tau}$

3 Experiments

3.1 Model evaluation

We conduct experiments on the KnowAir dataset [25], which provides 3-hour interval observations of PM2.5 and 17 meteorological factors across 184 Chinese cities over three years. Our study uses PM2.5 and five meteorological variables (temperature, surface pressure, relative humidity, and U/V wind components) as inputs. Using the past 24 time steps (three days) to predict the PM2.5 level over the next 24 steps (three days), we evaluate the baseline Air-DualODE and our proposed LIMBO model under different exchange interval k, with results reported in Table 1.

Table 1: Performance of Air-DualODE and LIMBO at different interval k. Table entries are results from three repeated experiments, presented as "mean (standard deviation)".

Metrics	Air-DualODE	LIMBO $k=1$	LIMBO $k=2$	LIMBO $k=4$	LIMBO $k=8$	LIMBO $k=12$	LIMBO $k=24$
MAE	18.8747 (0.1088)	18.9071 (0.1445)	18.9200 (0.1158)	**18.6979** (0.1071)	18.8673 (0.0768)	18.9988 (0.1801)	18.8692 (0.1053)
SMAPE	0.4241 (0.0016)	0.4242 (0.0026)	0.4247 (0.0023)	**0.4205** (0.0019)	0.4238 (0.0021)	0.4247 (0.0011)	0.4229 (0.0016)
RMSE	30.3111 (0.2008)	30.4519 (0.2457)	30.2540 (0.3589)	**29.9283** (0.1413)	30.4661 (0.3800)	30.5770 (0.2338)	30.2525 (0.1688)

Experimental results show that when $k = 4$, the three-day forecasting performance outperforms the baseline across all metrics with the lowest error, while errors increase at extremes ($k = 1$ or $k = 24$). Small k disrupts the Neural ODE evolution with frequent GRU corrections, reducing the model to an RNN-like structure that struggles to capture continuous dynamics, leading to lower accuracy and higher computational cost. While a large k maintains continuous modeling, the lack of timely corrections causes accumulated deviations. Thus, $k = 4$ strikes the best balance in our experiment settings, combining ODE continuity with timely GRU corrections.

3.2 Ablation studies

To assess the constraining effect of the LIMBO loss on information exchange, we conduct ablation studies (Table 2). For LIMBO ($k = 4$), adding LIMBO loss significantly improves all evaluation metrics, increases prediction accuracy, and lowers standard deviation, thereby making the information exchange mechanism's contribution to model performance more consistent and reliable.

Table 2: Ablation studies on LIMBOLoss ($k = 4$). Table entries are results from three repeated experiments, presented as "mean (standard deviation)"

Metrics	w/ LIMBOLoss	w/o LIMBOLoss
MAE	18.6979 (0.1071)	18.8015 (0.2643)
SMAPE	0.4205 (0.0019)	0.4221 (0.0040)
RMSE	29.9283 (0.1413)	30.2614 (0.3511)

4 Conclusion and future work

We address atmospheric complexity in air quality forecasting by proposing LIMBO, which introduces a GRU-based bidirectional information exchange between the physical and data-driven branches to mitigate error accumulation and improve cross-branch fusion. By mutually correcting branch states at regular ODE integration intervals, LIMBO achieves tighter collaboration between branches. On the KnowAir dataset, LIMBO (exchange interval $k = 4$) outperforms the state-of-the-art model for three-day PM2.5 forecasting, reducing the prediction error by 1%. Considering the costs of tuning hyperparameter k, future work will focus on developing an adaptive step-size information exchange strategy to dynamically detect error accumulation and trigger exchanges, further improving efficiency and predictive accuracy.

References

[1] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

[2] Yuxuan Chen, Shanshan Huang, Yunyao Cheng, Peng Chen, Zhongwen Rao, Yang Shu, Bin Yang, Lujia Pan, and Chenjuan Guo. Aimts: Augmented series and image contrastive learning for time series classification. In *41st IEEE International Conference on Data Engineering, ICDE 2025, Hong Kong, May 19-23, 2025*, pages 1952–1965. IEEE, 2025. doi: 10.1109/ICDE65448.2025.00149. URL `https://doi.org/10.1109/ICDE65448.2025.00149`.

[3] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[4] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[5] Aaron Daly and Paolo Zannetti. Air pollution modeling–an overview. *Ambient air pollution*, pages 15–28, 2007.

[6] Hongfan Gao, Wangmeng Shen, Xiangfei Qiu, Ronghui Xu, Bin Yang, and Jilin Hu. Ssd-ts: Exploring the potential of linear state space models for diffusion models in time series imputation. In *SIGKDD*, 2025.

[7] Leo J Grady and Jonathan R Polimeni. *Discrete calculus: Applied analysis on graphs for computational science*, volume 3. Springer, 2010.

[8] Kethmi Hirushini Hettige, Jiahao Ji, Shili Xiang, Cheng Long, Gao Cong, and Jingyuan Wang. Airphynet: Harnessing physics-guided neural networks for air quality prediction. *arXiv preprint arXiv:2402.03784*, 2024.

[9] Gaganjot Kaur Kang, Jerry Zeyu Gao, Sen Chiao, Shengqiang Lu, and Gang Xie. Air quality prediction: Big data and machine learning approaches. *Int. J. Environ. Sci. Dev*, 9(1):8–16, 2018.

[10] Pijush K Kundu, Ira M Cohen, David R Dowling, and Jesse Capecelatro. *Fluid mechanics*. Elsevier, 2024.

[11] Xiang Li, Ling Peng, Yuan Hu, Jing Shao, and Tianhe Chi. Deep learning architecture for air quality predictions. *Environmental Science and Pollution Research*, 23:22408–22417, 2016.

[12] Zhe Li, Xiangfei Qiu, Peng Chen, Yihang Wang, Hanyin Cheng, Yang Shu, Jilin Hu, Chenjuan Guo, Aoying Zhou, Christian S Jensen, et al. Tsfm-bench: A comprehensive and unified benchmark of foundation models for time series forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 5595–5606, 2025.

[13] Yuxuan Liang, Yutong Xia, Songyu Ke, Yiwei Wang, Qingsong Wen, Junbo Zhang, Yu Zheng, and Roger Zimmermann. Airformer: Predicting nationwide air quality in china with transformers. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 14329–14337, 2023.

[14] Yansui Liu, Yang Zhou, and Jiaxin Lu. Exploring the relationship between air pollution and meteorological conditions in china under environmental governance. *Scientific reports*, 10(1): 14518, 2020.

[15] Zhipeng Luo, Jianqiang Huang, Ke Hu, Xue Li, and Peng Zhang. Accuair: Winning solution to air quality prediction for kdd cup 2018. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1842–1850, 2019.

[16] Björn Maronga, Sabine Banzhaf, Cornelia Burmeister, Thomas Esch, Renate Forkel, Dominik Fröhlich, Vladimir Fuka, Katrin Frieda Gehrke, Jan Geletič, Sebastian Giersch, et al. Overview of the palm model system 6.0. *Geoscientific Model Development*, 13(3):1335–1372, 2020.

[17] Manuel Méndez, Mercedes G Merayo, and Manuel Núñez. Machine learning algorithms to forecast air quality: a survey. *Artificial Intelligence Review*, 56(9):10031–10066, 2023.

[18] World Health Organization et al. *WHO global air quality guidelines: particulate matter (PM2. 5 and PM10), ozone, nitrogen dioxide, sulfur dioxide and carbon monoxide*. World Health Organization, 2021.

[19] Aloke Paul, Tomi Laurila, Vesa Vuorinen, Sergiy V Divinski, Aloke Paul, Tomi Laurila, Vesa Vuorinen, and Sergiy V Divinski. Fick's laws of diffusion. *Thermodynamics, diffusion and the kirkendall effect in solids*, pages 115–139, 2014.

[20] Tianyu Qiu, Yi Xie, Hao Niu, Yun Xiong, and Xiaofeng Gao. Enhancing masked time-series modeling via dropping patches. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 20077–20085, 2025.

[21] Jaroslav Resler, Petra Bauerová, Michal Belda, Martin Bureš, Kryštof Eben, Vladimír Fuka, Jan Geletič, Radek Jareš, Jan Karel, Josef Keder, et al. Challenges of high-fidelity air quality modeling in urban environments–palm sensitivity study during stable conditions. *Geoscientific Model Development*, 17(20):7513–7537, 2024.

[22] Jindong Tian, Yifei Ding, Ronghui Xu, Hao Miao, Chenjuan Guo, and Bin Yang. Arrow: An adaptive rollout and routing method for global weather forecasting. *arXiv preprint arXiv:2510.09734*, 2025.

[23] Jindong Tian, Yuxuan Liang, Ronghui Xu, Peng Chen, Chenjuan Guo, Aoying Zhou, Lujia Pan, Zhongwen Rao, and Bin Yang. Air quality prediction with physics-guided dual neural odes in open systems. *ICLR*, 2025.

[24] Sotiris Vardoulakis, Bernard EA Fisher, Koulis Pericleous, and Norbert Gonzalez-Flesca. Modelling air quality in street canyons: a review. *Atmospheric environment*, 37(2):155–182, 2003.

[25] Shuo Wang, Yanran Li, Jiang Zhang, Qingye Meng, Lingwei Meng, and Fei Gao. Pm2. 5-gnn: A domain knowledge enhanced graph neural network for pm2. 5 forecasting. In *Proceedings of the 28th international conference on advances in geographic information systems*, pages 163–166, 2020.

[26] Mi Wen, Zhehui Chen, Yun Xiong, and Yichuan Zhang. Lgat: A novel model for multivariate time series anomaly detection with improved anomaly transformer and learning graph structures. *Neurocomputing*, 617:129024, 2025.

[27] Yi Xie, Tianyu Qiu, Yun Xiong, Xiuqi Huang, Xiaofeng Gao, Chao Chen, Qiang Wang, and Haihong Li. Weather knows what will occur: Urban public nuisance events prediction and control with meteorological assistance. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6037–6048, 2024.

[28] Zijian Zhang, Xiao Han, Xiangyu Zhao, Chenjuan Guo, and Bin Yang. Learning from spatio-temporal data in the llm era: Foundations, models, and emerging trends. In *Proceedings of the 19th International Symposium on Spatial and Temporal Data*, pages 107–110, 2025.

A Diffusion–Advection Equation

The transport of atmospheric pollutants is governed by the continuity equation from fluid mechanics, which states that local concentration changes arise from net flux divergence, assuming source and sink processes are absent [10]. Two primary mechanisms contribute to this flux: diffusion and advection. Diffusion models the spontaneous spread of particles from high to low concentration regions. By Fick's first law, it yields a diffusive flux $\vec{F}_{\text{diff}} = -k\nabla X$ (with diffusion coefficient k) [19] and hence a Laplacian term in the governing equation. Advection represents transport by bulk flow (e.g., wind) and is expressed by the advective flux $\vec{F}_{\text{adv}} = \vec{v}X$ where $\vec{v}(p,t)$ denotes the velocity field [7]. Combining these contributions under the continuity constraint produces the diffusion–advection

equation used to model spatiotemporal pollutant dynamics, where the Laplace term $\nabla^2 = \nabla \cdot \nabla$ captures spatial diffusion and the divergence term captures advection by the velocity field.

$$\frac{\partial X}{\partial t} = k\nabla^2 X - \nabla \cdot (X\vec{v})$$

B Baseline architecture

Physical branch. This branch encodes a diffusion–advection formulation on a spatial graph and integrates it within a Neural ODE. Concretely, the instantaneous physical vector field is parameterized to combine two graph Laplacian terms for diffusion and advection, respectively, and an open-system correction:

$$\mathbf{F}^P(X_t;\Theta) \;=\; \frac{dX}{dt} \;=\; \boldsymbol{\alpha} \odot \big(-k \cdot L_{\text{diff}} X\big) \;+\; (1-\boldsymbol{\alpha}) \odot \big(L_{\text{adv}} X\big) \;+\; \boldsymbol{\beta} \odot X,$$

where $\boldsymbol{\alpha}$ is a site-wise gating vector that adaptively weights diffusion versus advection, k is the diffusion coefficient, and $\boldsymbol{\beta}$ is an open-system correction that models net local loss and generation of pollutant mass. Learnable Coefficient Estimators produce k and $\boldsymbol{\beta}$ from historical local signals. The physical branch is unrolled by an ODE solver to yield a multi-step physical forecast $\hat{X}_{T+1:T+\tau}$.

Data-driven branch. It captures residual and contextual effects that the diffusion–advection equations cannot fully represent (for example, nonlinear chemical interactions or meteorological modulation). Observations (pollutant fields and auxiliary covariates) are encoded into a latent initial state Z_T, and the branch models its continuous-time evolution via a Neural ODE whose vector field is implemented with masked self-attention blocks in latent space:

$$\mathbf{F}^D(Z_t^D;\Phi) \;=\; \frac{dZ^D}{dt} \;=\; \text{Attention}\big(Z_t^D, G\big),$$

producing a parallel latent trajectory $Z^D_{T+1:T+\tau}$ that complements the physically derived forecast.

C Implementation details

Hardware and Software: The model is implemented in PyTorch 2.5.0 and executed on a server equipped with an NVIDIA H20 GPU and an Intel Xeon Platinum 8469C CPU.

Training Parameters: The Adam optimizer is used with a batch size of 64 and an initial learning rate of 0.005, which is gradually decayed during training. The number of training epochs is determined by early stopping: training is terminated if the validation loss fails to improve for 10 consecutive epochs.

ODE Solvers: The physical branch employs the *dopri5* numerical integration method with relative and absolute tolerances set to 1×10^{-2}. The data-driven branch uses the *rk4* method with relative and absolute tolerances set to 1×10^{-4}. Both branches adopt the adjoint method [1] for backpropagation.

Model Parameters: Both the physical branch and the fusion module contain 3 GNN layers. The hidden state dimension of the data-driven branch is set to 64, and the outputs from the physical branch are encoded into a latent space of the same dimension.

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: The abstract and introduction clearly state the main contributions: proposing LIMBO with bidirectional information exchange between physical and data-driven branches, investigating optimal exchange intervals, and outperforming the state-of-the-art Air-DualODE model on the KnowAir dataset.

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: In the section "Conclusion and future work", the paper mentions future work on adaptive step-size exchange strategies, considering the tuning of hyperparameter k could be computationally expensive.

   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: The paper does not include formal theoretical results or proofs. It focuses on empirical methodology and experimental validation.

   Guidelines:

   - 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: Appendix C provides the necessary implementation details to reproduce the experimental results. In addition, the code has been released alongside the 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: The code has been released alongside the paper.

   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: Appendix C provides the necessary experiment settings to understand the experimental results.

   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: [Yes]

   Justification: The paper reports results with standard deviations from three repeated experiments across all evaluation metrics (MAE, SMAPE, RMSE), clearly indicating the statistical variability.

   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: The paper specifies hardware requirements in Appendix C, including NVIDIA H20 GPU and Intel Xeon Platinum 8469C CPU specifications, along with software versions (PyTorch 2.5.0).

   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: The research focuses on air quality prediction for public health benefit and does not involve any ethically problematic methods, data collection, or applications.

   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: [Yes]

    Justification: The paper addresses air quality prediction, which has a clear positive societal impact for public health, yet might yield a negative impact if the prediction is not accurate.

    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: The paper focuses on air quality prediction methodology and does not involve releasing models or datasets that pose significant misuse risks.

    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: The paper cites the KnowAir dataset, which uses the MIT 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: [NA]

    Justification: The paper does not introduce new datasets, pretrained models, or other assets that would require separate documentation.

    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: The paper does not involve crowdsourcing or research with human subjects. It uses existing air quality monitoring data.

    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: The paper does not involve research with human subjects, as it uses existing air quality monitoring datasets.

    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: The core methodology does not involve large language models. The research focuses on physics-guided neural networks for air quality prediction.

    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.