FLOWGMAE: PHYSICS-CONSTRAINED VIRTUAL SENSING FOR ANOMALY DETECTION BEYOND SENSOR COVERAGE

Anonymous authors
Paper under double-blind review

ABSTRACT

Real-world flow networks are only partially observed. Sensors are typically installed on only a subset of edges, leaving much of the network without direct measurements. As a result, the network state is only partially known, and anomalous events outside sensor coverage can easily go undetected. We introduce **FlowGMAE**, a physics-constrained graph learning framework for learning virtual sensors on partially observed networks. We make three main contributions: (1) a physics-guided graph masked autoencoder operating on the full physical network, with incidence-aware message passing and adaptive masking; (2) a decoder in which conservation directly parameterizes the reconstruction space, analytically determining the constrained component of missing flow while learning only the remaining degrees of freedom; and (3) transferable virtual sensing that combines a time-series foundation model (TSFM) with graph learning for zero-shot transfer and anomaly detection beyond sensor coverage. Experiments on physical flow networks show that FlowGMAE substantially improves flow reconstruction over diverse baselines, with particularly strong gains as the distance from observed sensors increases. Moreover, anomalies on unobserved edges remain detectable from the reconstructed flows, demonstrating the potential of virtual sensing to extend monitoring beyond the physical sensor network.

1 INTRODUCTION

Critical infrastructure such as road and power networks rely on monitoring flows to detect disruptions, such as traffic incidents or line faults. Yet high sensor deployment and maintenance costs leave large portions of these networks unmonitored (Kaiser, 2026; Peddiraju et al., 2024). This *observability gap* limits both network state estimation and anomaly detection on edges without sensors (Peddiraju et al., 2024; Zhou et al., 2025). Figure 1 illustrates this challenge: approximately 57.3% of incidents in PeMSD4 occur outside sensors. Our goal is therefore to reconstruct flows across the full physical network and detect anomalies beyond monitored locations.

Existing approaches to network state estimation are broadly model-driven or data-driven. Model-driven methods use governing equations but often require detailed system parameters and struggle with real-world nonlinear dynamics (Huang & Agarwal, 2023; Koch et al., 2022). Data-driven methods, particularly spatio-temporal graph neural networks (ST-GNNs), learn complex dependencies from observations (Yu et al., 2017; Li et al., 2017; Mallick et al., 2020), but may produce physically inconsistent estimates under sparse sensing (Hou et al., 2023; Zhang et al., 2023; 2024). Moreover, many graph-based methods operate on sensor networks rather than the full physical topology. Physics-informed methods bridge these approaches by incorporating physical laws through regularization or constrained reconstruction (da Silva et al., 2020; Goswami et al., 2023; Li et al., 2025). However, combining physical consistency with full-network reconstruction and transfer across different network topologies remains challenging.

We introduce **FlowGMAE**, a physics-constrained graph masked autoencoder that operates on the full physical network, including unmonitored edges. FlowGMAE uses conservation to define the feasible reconstruction space: it analytically determines the conservation-consistent component of missing flow, while a GNN learns only the remaining degree of free-

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.

dom. A pretrained time-series foundation model (TSFM) provides transferrable temporal representtions, enabling reconstruction on unseen networks without retraining. The reconstructed flows act as virtual sensors, supporting anomaly detection beyond physical sensor coverage. Our main contributions are:

- **Physics-guided graph masked learning.** We develop a masked autoencoder over the full physical network, with incidence-aware message passing and adaptive masking that emphasizes edges difficult to reconstruct from conservation alone.
- **Conservation-preserving reconstruction.** We analytically determine the conservation-constrained component of missing flows and learn only the remaining degrees of freedom through null-space projection.
- **Transferable virtual sensing.** We integrate pretrained temporal representations with graph learning to enable zero-shot transfer across different network topologies and anomaly detection beyond physical sensor coverage.

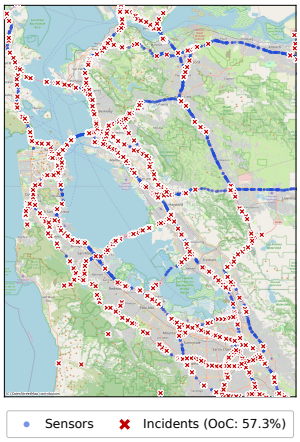

Figure 1: Significant number (∼57.3%) of incidents occur outside of sensor coverage in PeMSD4.

2 Problem Setup

We first introduce the problem setup for a partially observed flow network. We define the network and its edge-level variables, describe the flow conservation constraint and sparse sensing setting, and then formulate the flow imputation and anomaly scoring tasks.

Network Topology. We consider a flow network as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the set of $N = |\mathcal{V}|$ nodes and $\mathcal{E}$ denotes the set of $M = |\mathcal{E}|$ edges. We assume that $\mathcal{G}$ represents the full physical network, including both instrumented and uninstrumented edges, rather than a graph defined only over sensor locations. Each edge represents a physical connection along which flow occurs, while nodes represent the locations where these connections meet. In a road network, for example, edges correspond to road segments and nodes to junctions; in a power grid, edges correspond to transmission lines and nodes to buses. To represent flow conservation, we define the *node-edge incidence matrix* $\mathbf{B} \in \{-1, 0, 1\}^{N \times M}$. For a node v and edge e, $B_{v,e} = 1$ if e enters v, $B_{v,e} = -1$ if e leaves v, and $B_{v,e} = 0$ otherwise. We call a node *internal* if there is no external source or sink at that node, and denote the corresponding rows of $\mathbf{B}$ by $\mathbf{B}_{\text{int}}$.

Network Variables. We denote the flow on each edge $e \in \mathcal{E}$ at time step t by $q_{e,t} \in \mathbb{R}$. For example, flow may represent vehicles per hour on a road segment or active power on a transmission line. In addition to flow, the network may provide other edge-level measurements, which we denote by $\mathbf{a}_{e,t} \in \mathbb{R}^{p_a}$, such as speed and occupancy in traffic networks. We further denote static edge-level features by $\mathbf{f}_e \in \mathbb{R}^{p_f}$, which may include information such as road length, number of lanes, or capacity for road networks. We use flow to impose the physical constraints of the network, while the auxiliary measurements and edge features provide complementary information for recovering flow on unobserved edges. Let $\mathbf{Q} \in \mathbb{R}^{M \times T}$ denote the flow matrix over T time steps, with $\mathbf{q}_t \in \mathbb{R}^M$ denoting the flow vector at time t. Similarly, let $\mathcal{A} \in \mathbb{R}^{M \times T \times p_a}$ denote the auxiliary measurements and $\mathbf{F} \in \mathbb{R}^{M \times p_f}$ the static edge-level features.

Physical Constraint. A valid flow field satisfies flow conservation at internal nodes:

$$\mathbf{B}_{\text{int}} \mathbf{q}_t = \mathbf{0}. \tag{1}$$

The equation above indicates the total flow entering an internal node equals the total flow leaving it. For example, this corresponds to vehicle conservation at a road junction or power balance at a bus with no generation or load.

Sparse Observability. In this work, we assume that flow and auxiliary measurements are available only on a subset of edges $\mathcal{E}_{\text{obs}} \subset \mathcal{E}$, while static edge features are available independently of the sensing pattern. We define a binary mask $\mathbf{m} \in \{0,1\}^M$, where $m_e = 1$ for a monitored edge and $m_e = 0$ for an unmonitored edge, and let $\mathcal{E}_{\text{mis}} = \mathcal{E} \setminus \mathcal{E}_{\text{obs}}$ and $d = |\mathcal{E}_{\text{mis}}|$. The observed flow and auxiliary measurements are given by

$$\mathbf{Q}^{\text{obs}} = \mathbf{Q} \odot \mathbf{m}, \qquad \mathcal{A}^{\text{obs}} = \mathcal{A} \odot \mathbf{m},$$

where $\mathbf{m}$ is broadcast over time and, for $\mathcal{A}^{\text{obs}}$, over the auxiliary measurement channels.

Task Formulation. We consider anomaly detection under sparse observability as two related subtasks:

1. **Conservation-Guided Flow Imputation:** Given the observed flow and auxiliary measurements, together with the network structure and static edge features, we learn a mapping
$$f_\phi\big(\mathbf{Q}^{\text{obs}}, \mathcal{A}^{\text{obs}}, \mathbf{F}, \mathcal{G}, \mathbf{B}\big) \to \hat{Q},$$
where $\hat{Q}$ denotes the reconstructed flow over full network subject to the conservation constraint in Eq. 1.
2. **Anomaly Scoring:** Given the reconstructed flow, we compute an edge-level anomaly score
$$\hat{s}_{e,t} = g(\widehat{q}_{e,t})\,,$$
where larger values indicate a stronger deviation from the expected flow behavior at edge e and time t.

3 METHODOLOGY

Observed flows and the conservation constraints are generally not sufficient to uniquely recover the flows on edges with missing measurements, $\mathcal{E}_{\text{mis}}$. Instead, they restrict the missing flows to a set of feasible solutions that are consistent with the observed network state. In this section, we describe how we use this structure to separate the reconstruction problem into two parts: the component that can be obtained directly from conservation, and the remaining ambiguity, which we seek to resolve from data through a GNN-based architecture that operates within the conservation-defined feasible space.

3.1 PHYSICS-CONSTRAINED FLOW SPACE

Let $\mathbf{x}_t \in \mathbb{R}^d$ collect the unknown flows on the unobserved edges, and let $\mathbf{H} \in \{0,1\}^{M \times d}$ denote the insertion matrix that maps these flows to their corresponding edges in the full network. The complete flow vector can then be written as

$$\mathbf{q}_t = \mathbf{q}_t^{\text{obs}} + \mathbf{H}\mathbf{x}_t, \tag{2}$$

where $\mathbf{q}_t^{\text{obs}}$ contains the observed flows on monitored edges and zeros on unobserved edges. Substituting Eq. 2 into the conservation constraint gives the following linear system for the missing flows:

$$\mathbf{A}\mathbf{x}_t = \mathbf{c}_t, \qquad \mathbf{A} = \mathbf{B}_{\text{int}}\mathbf{H}, \qquad \mathbf{c}_t = -\mathbf{B}_{\text{int}}\mathbf{q}_t^{\text{obs}}. \tag{3}$$

Under the conservation assumption, the missing flows satisfy this system. In practice, measurement noise may make the observed system only approximately consistent. We therefore use the minimum-norm least-squares solution

$\mathbf{x}_t^p = \mathbf{A}^\dagger \mathbf{c}_t$, where $\mathbf{A}^\dagger$ denotes the Moore–Penrose pseudoinverse (Penrose, 1955). This particular solution, however, does not fully characterize the missing flows. Let $\mathbf{P} = \mathbf{I} - \mathbf{A}^\dagger\mathbf{A}$ denote the orthogonal projection onto the null space of $\mathbf{A}$ (Lu et al., 2021). Since $\mathbf{A}\mathbf{P} = \mathbf{0}$, the set of minimum-residual solutions can be expressed as

$$\mathbf{x}_t = \mathbf{x}_t^p + \mathbf{r}_t, \qquad \mathbf{r}_t \in \text{null}(\mathbf{A}) \tag{4}$$

This decomposition separates the part of the missing flow obtained directly from the observations and conservation constraints, $\mathbf{x}_t^p$, from the remaining unresolved variation $\mathbf{r}_t$. The dimension of this unresolved space is $d - \text{rank}(\mathbf{A})$, and thus depends on both sensor coverage and sensor placement in the network.

3.2 Graph Learning in the Feasible Flow Space

We learn the unresolved component from data by conditioning on

$$\mathcal{C}_t = \left(\mathcal{G}, \mathbf{q}_t^{\text{obs}}, \mathbf{a}_t^{\text{obs}}, \mathbf{F}, \mathbf{m}, \mathbf{x}_t^p\right). \tag{5}$$

We next use a graph-based encoder to capture how the remaining feasible flow variation depends on the observed network state.

Physics-Guided Learnable Masking. Conservation constrains some missing flows more strongly than others. We group candidate sensor edges according to how much freedom remains after applying conservation, from strongly constrained to highly underdetermined, and partition them into K equal-sized groups. Rather than masking all groups uniformly, we learn a sampling probability p_k for each group, parameterized by learnable logits α_k through a softmax:

$$p_{\boldsymbol{\alpha},k} = \frac{\exp(\alpha_k)}{\sum_{j=1}^{K} \exp(\alpha_j)}. \tag{6}$$

The masking policy emphasizes groups with larger reconstruction errors, while the reconstruction model learns to recover their missing flows:

$$\min_{\theta} \max_{\boldsymbol{\alpha}} \; \mathbb{E}_{\tilde{\mathbf{m}} \sim p_{\boldsymbol{\alpha}}} \left[\mathcal{L}_{\text{rec}}(\theta; \tilde{\mathbf{m}})\right], \tag{7}$$

where $\tilde{\mathbf{m}}$ denotes the sampled training mask and $p_{\boldsymbol{\alpha}}$ is the learned sampling distribution over the K groups. This encourages FlowGMAE to learn from physically challenging masking patterns while reconstructing flows within the conservation-constrained space.

Physics-Aware Message Passing (Encoder). We initialize the representation of each edge using the available edge information, the conservation-derived particular solution, and a pretrained temporal representation. Since unobserved edges do not have measured flow histories, we construct a proxy flow at each time step τ using the measurement when available and the particular solution otherwise:

$$q_{e,\tau}^{\text{proxy}} = \begin{cases} q_{e,\tau}^{\text{obs}}, & m_e = 1, \\ (\mathbf{H}\mathbf{x}_\tau^p)_e, & m_e = 0. \end{cases} \tag{8}$$

We then encode the causal history of each edge using a frozen time-series foundation model (TSFM):

$$\mathbf{z}_{e,t}^{\text{temp}} = \psi\Big(f_{\text{TSFM}}\left(q_{e,t-K:t-1}^{\text{proxy}}\right)\Big), \tag{9}$$

where K denotes the history length, f_{TSFM} is a frozen pretrained time-series foundation model, and ψ is a trainable projection layer. Measurements from masked or unmonitored edges are never provided to the TSFM.

The initial edge representation is then

$$\mathbf{h}_{e,t}^{(0)} = \mathbf{W}_{\text{in}}\left[q_{e,t}^{\text{obs}},\ \mathbf{a}_{e,t}^{\text{obs}},\ \mathbf{f}_e,\ m_e,\ (\mathbf{H}\mathbf{x}_t^p)_e,\ \mathbf{z}_{e,t}^{\text{temp}}\right]. \tag{10}$$

where the flow and auxiliary measurements are set to zero on unobserved edges. Inspired by graph-based physical solvers (Brandstetter et al., 2022), we use the incidence structure to exchange information through shared junctions. At each layer ℓ, we first aggregate the signed edge representations at each node, then form an edge message from the representations of its two endpoints, and finally update the edge representation:

$$\mathbf{z}^{(\ell)} = \mathbf{B}\,\mathbf{h}^{(\ell)}, \tag{11a}$$

$$\mathbf{g}_e^{(\ell)} = \phi\Big(\mathbf{W}^{(\ell)}\mathbf{z}_v^{(\ell)}\Big) + \phi\Big(\mathbf{W}^{(\ell)}(-\mathbf{z}_u^{(\ell)})\Big), \tag{11b}$$

$$\mathbf{h}_e^{(\ell+1)} = \text{LayerNorm}\Big(\mathbf{h}_e^{(\ell)} + \text{MLP}\Big([\mathbf{h}_e^{(\ell)} \,\|\, \mathbf{g}_e^{(\ell)}]\Big)\Big), \tag{11c}$$

where u and v denote the source and destination nodes of edge e, respectively. Specifically, Eq. 11a aggregates the signed edge representations at each node, Eq. 11b combines the information from the

two endpoints of each edge, and Eq. 11c updates the edge representation through a residual MLP block. After L layers, the encoder generates

$$\mathbf{h}_t = f_{\text{enc}}(\mathcal{C}_t), \tag{12}$$

which summarizes the observed network state, its temporal context, and the information propagated through the network topology. This representation provides the data-driven signal used to resolve the remaining degrees of freedom that are not determined by the conservation constraints alone.

Feasible Flow Completion (Decoder). Given the learned edge representations $\mathbf{h}_t$, we first predict an unconstrained residual for the unobserved edges:

$$\widetilde{\mathbf{r}}_t = \mathbf{H}^\top f_{\text{dec}}(\mathbf{h}_t) \in \mathbb{R}^d, \tag{13}$$

where f_{dec} is a decoder head that produces residual flow predictions, and $\mathbf{H}^\top$ restricts these predictions to the unobserved edges. We then project the predicted residual and combine it with the particular solution:

$$\widehat{\mathbf{x}}_t = \mathbf{x}_t^p + \mathbf{P}\widetilde{\mathbf{r}}_t. \tag{14}$$

The complete flow field is then obtained by inserting the recovered missing flows into the observed network state as in Eq. 2. The projection enforces that the learned correction does not alter the conservation residual of the solution, while allowing the decoder to resolve the remaining ambiguity from the observed network state.

Training Objective. During training, the instrumented edges are split into three disjoint sets, $\mathcal{E}_{\text{obs}} = \mathcal{E}_{\text{input}} \cup \mathcal{E}_{\text{sup}} \cup \mathcal{E}_{\text{eval}}$, and the edges in $\mathcal{E}_{\text{sup}} \cup \mathcal{E}_{\text{eval}}$ are treated as virtual sensors. For every such edge, its flow and auxiliary measurements are removed from $\mathbf{q}_t^{\text{obs}}$ and $\mathbf{a}_t^{\text{obs}}$, respectively, for all time steps. The corresponding observation mask and conservation-defined feasible space are therefore constructed using only the edges in $\mathcal{E}_{\text{input}}$. The recorded flows on $\mathcal{E}_{\text{sup}}$ are retained only as reconstruction targets, and those on $\mathcal{E}_{\text{eval}}$ are never used during training. Let Ω denote the pairs (e, t) with $e \in \mathcal{E}_{\text{sup}}$, t in the training period and a recorded flow. We optimize

$$\mathcal{L}_{\text{rec}} = \frac{1}{|\Omega|} \sum_{(e,t)\in\Omega} \left(\hat{q}_{e,t} - q_{e,t}\right)^2. \tag{15}$$

This masking procedure trains the model to recover completely unobserved edges from measurements on the remaining observed edges, network structure, and conservation constraints. Edges that are never instrumented, and the held-out edges in $\mathcal{E}_{\text{eval}}$, do not contribute reconstruction targets, but are inferred through the same conservation-constrained completion mechanism. At inference, the observation mask reflects actual sensor availability rather than the training mask. We reconstruct flows on all unobserved edges using Eq. (14) and combine them with available measurements to obtain the completed flow field.

3.3 Anomaly Detection

Our primary objective is to reconstruct the flow field beyond the available sensor coverage. We use the resulting completed flow field for downstream anomaly detection, with the goal of identifying abnormal behavior on both observed and unobserved edges. In traffic networks, for example, an incident on an uninstrumented road segment may alter the surrounding traffic state and thereby induce an abnormal reconstructed flow on that segment.

We treat anomaly detection separately from the reconstruction model. The reconstruction network is trained only to recover missing flows, while anomaly scores are computed from the completed flow field in an unsupervised manner. This separation allows the proposed reconstruction method to be paired with different downstream anomaly detectors.

Flow Anomaly Score. Normal flow behavior can vary systematically across recurring regimes. Following deviation-based time-series anomaly detection (Hundman et al., 2018; Blázquez-García et al., 2021), we compare each reconstructed flow against its context-dependent reference distribution. We therefore partition the training data into a set of context bins and compare each test-time flow against the reference distribution associated with its corresponding bin. Let τ_t denote the context bin assigned to time t, and let $\bar{q}_e(\tau)$ and $\sigma_e(\tau)$ denote the mean and standard deviation of the

completed flow on edge e over normal training samples assigned to bin τ. The choice of binning function can reflect the sampling frequency and dominant periodicities of the application. For example, in road or power networks, bins may encode combinations of time of day and day of week, with finer or coarser temporal resolution depending on the data frequency. For unobserved edges, these reference statistics are estimated from their reconstructed flows during normal training periods. Given the completed flow $\widehat{q}_{e,t}$ at test time, we define the anomaly score as follows. Higher scores indicate greater deviations from expected behavior.

$$\hat{s}_{e,t} = \frac{|\widehat{q}_{e,t} - \bar{q}_e(\tau_t)|}{\sigma_e(\tau_t) + \epsilon}, \tag{16}$$

4 EXPERIMENTS

Traffic. We build directed road graphs of **PeMS D4** (San Francisco Bay Area) and **PeMS D7** (Los Angeles) from OpenStreetMap (OpenStreetMap contributors, 2017), retaining motorway and motorway-link segments. Unlike most traffic benchmarks, which operate on graphs of sensor locations, we model the full physical network, including segments without sensors. We impose flow conservation at internal junctions, excluding sources and sinks. Matching loop detectors to corresponding road segments yields $1{,}863$ instrumented edges in D4 (19.9%) and $1{,}834$ in D7 (20.3%), with 5-minute flow measurements.

Power. **ACTIVSg2000** is a 2,000-bus transmission grid on the footprint of Texas (Birchfield et al., 2017) with hourly load, wind and solar profiles for 2016 (Li et al., 2021). Edges are the $3{,}206$ lines and transformers plus one injection edge per bus with load or generation, connected to a common ground node, so that active power is conserved at every bus. Line flows follow from a DC power flow. A random 20% of edges carry a meter that records active power with Gaussian noise, hourly over 2016, matching the traffic coverage. Dataset statistics are summarized in Table 4 in Appendix A.

Metrics. We report normalized MAE and R^2 on the held-out validation detectors and mean and standard deviation over 5 folds.

Baselines. We compare against EDGE MEAN (road-type mean flow), NEIGHBOUR INTERPOLATION (average of nearby sensors), an MLP (feature-based prediction without message passing), GNN + PHYSICS (our encoder with a soft conservation penalty instead of projection), GRAPHMAE2 (Hou et al., 2023) (masked graph feature reconstruction), IGNNK (Wu et al., 2021) (inductive graph-based spatiotemporal kriging), and KITS (Xu et al., 2025) (knowledge transfer for spatiotemporal imputation). All methods use the same folds, and their hyperparameters are selected on a validation subset of $\mathcal{E}_{\text{sup}}$. Implementation details are given in Appendix B.

4.1 IMPUTATION PERFORMANCE

Table 1 reports flow imputation on held-out sensors (E_{eval}), averaged over the five folds. FlowGMAE achieves the best performance on PeMS D4 and on the power network, with nMAE of 4.33% and 1.94% against 7.53% and 3.61% for the strongest baseline, IGNNK. The gain over the graph masked autoencoder is of the same size (8.64% for GraphMAE2 on D4), and the corresponding R^2 of 0.839 and 0.690 indicates that the reconstructed flows follow the observed variation closely. The method outperforms a GNN with physics as a loss regularization (GNN+physics) because a penalty does not guarantee satisfaction of the constraint, whereas the projection decoder cannot leave the conservation-consistent set. We provide an ablation on physics contribution in Table 5.

Figure 3 provides a representative example on a held-out virtual sensor. FlowGMAE closely follows the true flow, including changes in both the magnitude and temporal shape of the signal, whereas the competing methods show larger deviations. Figure 2 further examines performance as the sensing problem becomes more difficult. FlowGMAE remains the strongest method as the fraction of masked edges increases and consistently performs best across different distances from the nearest working sensor. The advantage is particularly clear for edges several hops away from an observed sensor, where methods that rely primarily on nearby measurements degrade more substantially. These results suggest that combining the physical network structure with conservation-guided

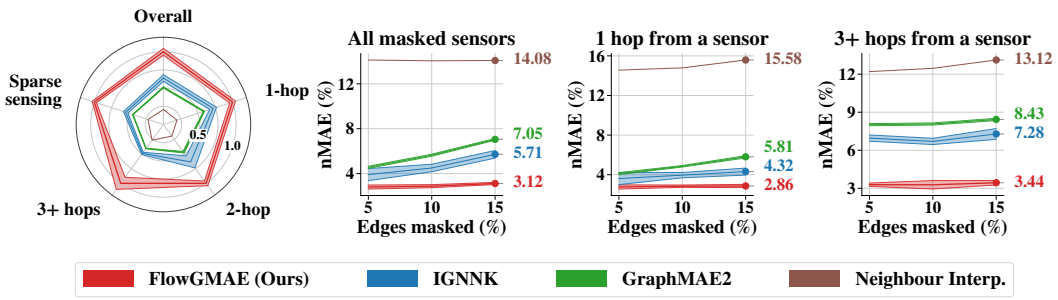

Figure 2: Robustness to sensor coverage and distance from an observed sensor (monitored segment), averaged over PeMSD4 and PeMSD7. **Left:** relative accuracy $\min_m \mathrm{nMAE}_m/\mathrm{nMAE}$ (1 = best model on that axis), with 10% of edges masked, or 15% for the *Sparse sensing* axis. **Right:** nMAE (%) as the share of masked edges grows, for all masked sensors, masked sensors 1 hop from a working sensor, and masked sensors 3+ hops away. Lines show the mean over five seeds.

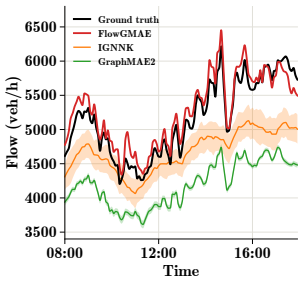

Figure 3: Flow imputation performance on a held-out masked sensor for FlowGMAE vs. baselines.

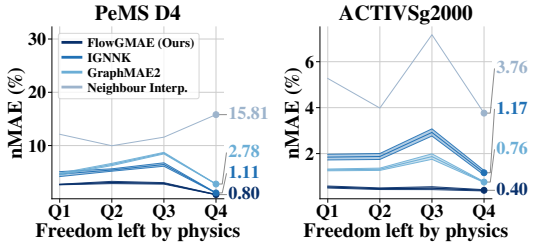

Figure 4: Imputation performance of masked sensors grouped by how much freedom conservation leaves for their flow (Q1: almost determined; Q4: unconstrained).

graph learning allows information to propagate beyond the immediate sensor neighborhood, making virtual sensing effective even under sparse and spatially distant observations.

Where the physics gain comes from. Figure 4 shows that FlowGMAE is the most stable across Q1-Q4. Also, it gains most over data-driven baselines when conservation partially constrains missing flows. In these cases, physics reduces the solution space while graph learning resolves the remaining uncertainty. The smaller gains at either extreme suggest that combining physics and learning is particularly useful when neither alone is sufficient.

4.2 Anomaly Detection

A key application of virtual sensing is detecting anomalies beyond physical sensor coverage. We therefore evaluate whether the imputed flows are accurate enough to support anomaly detection on unmonitored edges. We test whether reconstructed flow reveals real disruptions, using California Highway Patrol incidents from California Department of Transportation in the test period on the held-out detectors (E_{eval}) for PeMSD4 and PeMSD7. Each incident is scored by the largest deviation of Eq. 16 over $[t-2, t+6]$ and paired with a window on the same edge and time of day on a nearby day. for power dataset we inject controlled anomalies into the $\mathcal{E}_{\text{eval}}$ edges of each fold, which are never used as model input or reconstruction targets. Results are averaged over the five folds. As shown in Table 2, FlowGMAE achieves the best AUROC and AUPRC on all domains. The results demonstrate the potential of reconstructed flows to serve as virtual sensors, enabling anomaly detection beyond physical sensor coverage.

Table 1: **Flow imputation on masked detectors.** nMAE is mean absolute error as a percentage. Learned methods report mean ± standard deviation over five training folds. Best in bold.

	PeMS D4		PeMS D7		ACTIVSg2000	
Method	nMAE (%)	R^2	nMAE (%)	R^2	nMAE (%)	R^2
Neighbour Interpolation	12.46 ± 0.46	0.212 ± 0.025	15.78 ± 0.61	-0.082 ± 0.067	5.05 ± 0.25	-0.415 ± 0.186
Edge Mean	10.91 ± 0.35	0.362 ± 0.005	9.82 ± 0.28	0.480 ± 0.015	3.97 ± 0.13	0.091 ± 0.042
MLP	10.87 ± 0.35	0.363 ± 0.005	9.80 ± 0.27	0.476 ± 0.017	3.84 ± 0.18	0.098 ± 0.027
GraphMAE2 (Hou et al., 2023)	8.64 ± 0.23	0.577 ± 0.009	8.40 ± 0.43	0.591 ± 0.038	3.93 ± 0.18	0.076 ± 0.018
IGNNK (Wu et al., 2021)	7.53 ± 0.35	0.605 ± 0.108	7.30 ± 0.11	0.629 ± 0.048	3.61 ± 0.19	0.168 ± 0.046
KITS (Xu et al., 2025)	7.63 ± 0.35	0.600 ± 0.036	7.30 ± 0.18	0.629 ± 0.022	4.34 ± 0.20	-0.091 ± 0.080
GNN + physics	7.55 ± 0.30	0.595 ± 0.063	7.54 ± 0.51	0.598 ± 0.039	4.88 ± 0.74	-0.334 ± 0.336
FlowGMAE	**4.33** ± 0.34	**0.839** ± 0.053	**6.06** ± 0.51	**0.659** ± 0.066	**1.94** ± 0.13	**0.690** ± 0.062

Table 2: **Anomaly detection from each imputer's reconstruction.** Reconstructions from every imputer in Table 1 are scored with the same detector (Eq. 16) on the same targets.

	PeMS D4 (real)		PeMS D7 (real)		ACTIVSg2000 (injected)	
Method	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
MLP	0.500 ± 0.000	0.500 ± 0.000	0.500 ± 0.000	0.500 ± 0.000	0.499 ± 0.007	0.003 ± 0.000
GraphMAE2 (Hou et al., 2023)	0.752 ± 0.050	0.838 ± 0.037	0.562 ± 0.021	0.664 ± 0.029	0.500 ± 0.014	0.004 ± 0.000
IGNNK (Wu et al., 2021)	0.701 ± 0.048	0.654 ± 0.050	0.625 ± 0.034	0.588 ± 0.030	0.555 ± 0.011	0.003 ± 0.000
KITS (Xu et al., 2025)	0.738 ± 0.054	0.777 ± 0.046	0.604 ± 0.051	0.572 ± 0.049	0.540 ± 0.012	0.004 ± 0.001
GNN + physics	**0.877** ± 0.029	0.887 ± 0.023	0.725 ± 0.037	0.766 ± 0.036	0.581 ± 0.013	0.008 ± 0.001
FlowGMAE	0.863 ± 0.042	**0.898** ± 0.031	**0.778** ± 0.039	**0.827** ± 0.034	**0.624** ± 0.013	**0.020** ± 0.005

4.3 Transfer learning

We evaluate whether FlowGMAE can generalize to unseen road networks without retraining on target. As shown in Table 3, zero-shot FlowGMAE achieves an nMAE of 4.47% on D4 and 6.38% on D7, close to the target-trained results of 4.33% and 6.06%, respectively, while outperforming the baselines in both directions.

The ablations highlight the roles of pretrained time-series foundation models and physics in transfer. Replacing the frozen TSFM with an RNN increases the error, particularly on D7, while removing conservation increases the error on both datasets. These results suggest that the frozen TSFM provides transferable temporal features, while conservation offers complementary physical structure that can improve generalization across networks.

5 Related Work

Graph-Based Imputation. Graph-based imputation exploits relational dependencies to recover missing observations. GRAPE (You et al., 2020) uses graph representation learning for missing-data imputation, while Feature Propagation(Rossi et al., 2022) recovers missing nodes features through graph diffusion. For temporal data, BRITS (Cao et al., 2018) and SAITS (Du et al., 2023) model temporal dependencies, while IGNNK (Wu et al., 2021) and GRIN (Cini et al., 2021) incorporate graph structure into spatiotemporal imputation. Masked graph autoencoders, such as GraphMAE2 (Hou et al., 2023; 2022), further learn representations by reconstructing masked features. Unlike these approaches, FlowGMAE operates on full physical network and explicitly constrains edge-flow reconstruction through conservation.

Physics-Informed Learning for Flow Networks. Physics-informed learning incorporates governing equations through training losses (Raissi et al., 2019; Karniadakis et al., 2021; Jagtap et al., 2020) or architectural constraints (Beucler et al., 2021). Recent approaches embed physical structure into graph-based solvers (Brandstetter et al., 2022; Sanchez-Gonzalez et al., 2020; Gao et al., 2022) and combine conservation with learned regularization (da Silva et al., 2020; Smith et al., 2022). FlowSymm (Demirci et al., 2026) uses group action theory to develop graph attention for flow completion. FlowGMAE instead combines conservation-based null-space reconstruction with masked graph learning and pretrained temporal representations for transferable virtual sensing.

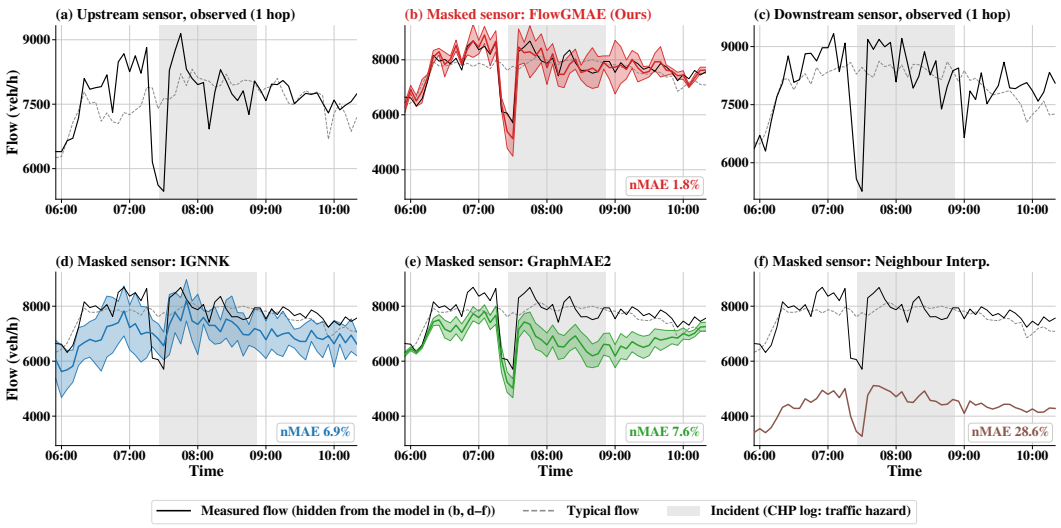

Figure 5: Real traffic incident at a masked detector (PeMS D4, CHP-reported traffic hazard, 27 March 2017, 07:26; grey). (a, c) Observed upstream and downstream sensors show the flow drop. (b) The hidden detector: FlowGMAE recovers the drop more accurately compared to baselines.

Table 3: **Zero-shot transfer to unseen network.** All methods are trained on the source data (PeMSD7 → PeMSD4, PeMSD4 → PeMSD7) and applied without any training on the target data.

	PeMSD7 → PeMSD4		PeMSD4 → PeMSD7	
Method	nMAE (%)	R^2	nMAE (%)	R^2
MLP	10.87 ± 0.35	0.363 ± 0.005	9.80 ± 0.27	0.476 ± 0.017
GraphMAE2 (Hou et al., 2023)	8.94 ± 0.19	0.546 ± 0.016	8.37 ± 0.31	0.597 ± 0.026
IGNNK (Wu et al., 2021)	7.81 ± 0.25	0.602 ± 0.057	7.19 ± 0.13	**0.642** ± 0.034
KITS (Xu et al., 2025)	7.81 ± 0.23	0.589 ± 0.043	7.26 ± 0.19	0.623 ± 0.018
GNN + physics	7.60 ± 0.50	0.602 ± 0.045	9.58 ± 0.66	0.349 ± 0.123
FlowGMAE (w/ RNN)	5.01 ± 0.45	0.816 ± 0.050	7.54 ± 1.33	0.416 ± 0.267
FlowGMAE (w/o conservation)	5.49 ± 0.20	0.793 ± 0.022	6.81 ± 0.40	0.603 ± 0.040
FlowGMAE	**4.47** ± 0.42	**0.852** ± 0.035	**6.38** ± 0.66	0.629 ± 0.091
FlowGMAE, trained on target	4.33 ± 0.34	0.839 ± 0.053	6.06 ± 0.51	0.659 ± 0.066

Time-Series Foundation Models. Pretrained time-series foundation models, including Chronos (Ansari et al., 2024), TimesFM (Das et al., 2023), Lag-Llama (Rasul et al., 2023), Moirai (Woo et al., 2024), have demonstrated the potential of large-scale temporal pretraining for forecasting and representation learning with zero-shot transferability. FlowGMAE combines frozen pretrained TSFM with topology-aware graph learning and conservation-based reconstruction to support zero-shot virtual sensing across different networks.

6 Conclusion

We introduced FlowGMAE, a framework for reconstructing flows and detecting anomalies beyond physical sensor coverage. By combining graph learning with flow conservation, FlowGMAE recovers missing flows across the full network while preserving physical consistency. Experiments on traffic and power networks show improved reconstruction, particularly when sensors are sparse, and demonstrate how these estimates can help detect anomalies on unmonitored edges. We also show that FlowGMAE can transfer between networks without retraining. A frozen time-series foundation model provides transferable temporal features, while the graph encoder and conservation projection adapt to each network's structure. Together, these results represent a step forward toward general-purpose foundation models for virtual sensing across physical networks with different topologies.

AI USE STATEMENT

In this work, we used generative AI tools for editing the manuscript. We have not used generative AI tools for developing research ideas, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to improve code efficiency. We have reviewed all AI-assisted work. AI-generated code was verified and tested for correctness by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

Tom Beucler, Michael Pritchard, Stephan Rasp, Jordan Ott, Pierre Baldi, and Pierre Gentine. Enforcing analytic constraints in neural networks emulating physical systems. *Physical review letters*, 126(9):098302, 2021.

Adam B. Birchfield, Ti Xu, Kathleen M. Gegner, Komal S. Shetye, and Thomas J. Overbye. Grid structural characteristics as validation criteria for synthetic networks. *IEEE Transactions on Power Systems*, 32(4):3258–3265, 2017.

Ane Blázquez-García, Angel Conde, Usue Mori, and Jose A Lozano. A review on outlier/anomaly detection in time series data. *ACM computing surveys (CSUR)*, 54(3):1–33, 2021.

Johannes Brandstetter, Daniel Worrall, and Max Welling. Message passing neural pde solvers. *arXiv preprint arXiv:2202.03376*, 2022.

California Department of Transportation. Performance Measurement System (PeMS). URL `https://pems.dot.ca.gov/`. Accessed: 2026.

Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. Brits: Bidirectional recurrent imputation for time series. *Advances in neural information processing systems*, 31, 2018.

Andrea Cini, Ivan Marisca, and Cesare Alippi. Filling the g_ap_s: Multivariate time series imputation by graph neural networks. *arXiv preprint arXiv:2108.00298*, 2021.

Arlei Lopes da Silva, Furkan Kocayusufoglu, Saber Jafarpour, Francesco Bullo, Ananthram Swami, and Ambuj Singh. Combining physics and machine learning for network flow estimation. In *International Conference on Learning Representations*, 2020.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. *arXiv preprint arXiv:2310.10688*, 2023.

Ege Demirci, Francesco Bullo, Ananthram Swami, and Ambuj Singh. Flowsymm: Physics aware, symmetry preserving graph attention for network flow completion. *arXiv preprint arXiv:2601.22317*, 2026.

Wenjie Du, David Côté, and Yan Liu. Saits: Self-attention-based imputation for time series. *Expert Systems with Applications*, 219:119619, 2023.

Han Gao, Matthew J Zahr, and Jian-Xun Wang. Physics-informed graph neural galerkin networks: A unified framework for solving pde-governed forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering*, 390:114502, 2022.

Somdatta Goswami, Aniruddha Bora, Yue Yu, and George Em Karniadakis. Physics-informed deep neural operator networks. In *Machine learning in modeling and simulation: methods and applications*, pp. 219–254. Springer, 2023.

Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Hongxia Yang, Chunjie Wang, and Jie Tang. Graphmae: Self-supervised masked graph autoencoders. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 594–604, 2022.

Zhenyu Hou, Yufei He, Yukuo Cen, Xiao Liu, Yuxiao Dong, Evgeny Kharlamov, and Jie Tang. Graphmae2: A decoding-enhanced masked self-supervised graph learner. In *Proceedings of the ACM web conference 2023*, pp. 737–746, 2023.

Archie J Huang and Shaurya Agarwal. On the limitations of physics-informed deep learning: Illustrations using first-order hyperbolic conservation law-based traffic flow models. *IEEE Open Journal of Intelligent Transportation Systems*, 4:279–293, 2023.

Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 387–395, 2018.

Ameya D Jagtap, Ehsan Kharazmi, and George Em Karniadakis. Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering*, 365:113028, 2020.

Silke K Kaiser. Sensor placement for urban traffic interpolation: A data-driven evaluation to inform policy. *arXiv preprint arXiv:2601.10747*, 2026.

George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

James Koch, Thomas Maxner, Vinay Amatya, Andisheh Ranjbari, and Chase Dowling. Physics-informed machine learning of parameterized fundamental diagrams. *arXiv preprint arXiv:2208.00880*, 2022.

Hanyue Li, Ju Hee Yeo, Ashly L. Bornsheuer, and Thomas J. Overbye. The creation and validation of load time series for synthetic electric power systems. *IEEE Transactions on Power Systems*, 36 (2):961–969, 2021.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*, 2017.

Zhihao Li, Ting Wang, Guojian Zou, Ruofei Wang, and Ye Li. Physics-informed deep operator network for traffic state estimation. *Transportmetrica B: Transport Dynamics*, 13(1):2595549, 2025.

Lu Lu, Raphael Pestourie, Wenjie Yao, Zhicheng Wang, Francesc Verdugo, and Steven G Johnson. Physics-informed neural networks with hard constraints for inverse design. *SIAM Journal on Scientific Computing*, 43(6):B1105–B1132, 2021.

Tanwi Mallick, Prasanna Balaprakash, Eric Rask, and Jane Macfarlane. Graph-partitioning-based diffusion convolutional recurrent neural network for large-scale traffic forecasting. *Transportation Research Record*, 2674(9):473–488, 2020.

OpenStreetMap contributors. Planet dump retrieved from https://planet.osm.org . `https://www.openstreetmap.org`, 2017.

Sai Shashank Peddiraju, Kaustubh Harapanahalli, Edward Andert, and Aviral Shrivastava. Incidentnet: Traffic incident detection, localization and severity estimation with sparse sensing. In *2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1470–1475. IEEE, 2024.

Roger Penrose. A generalized inverse for matrices. In *Mathematical proceedings of the Cambridge philosophical society*, volume 51, pp. 406–413. Cambridge University Press, 1955.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Hena Ghonia, Rishika Bhagwatkar, Arian Khorasani, Mohammad Javad Darvishi Bayazi, George Adamopoulos, Roland Riachi, Nadhir Hassen, et al. Lag-llama: Towards foundation models for probabilistic time series forecasting. *arXiv preprint arXiv:2310.08278*, 2023.

Emanuele Rossi, Henry Kenlay, Maria I Gorinova, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M Bronstein. On the unreasonable effectiveness of feature propagation in learning on graphs with missing node features. In *Learning on graphs conference*, pp. 11–1. PMLR, 2022.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to simulate complex physics with graph networks. In *International conference on machine learning*, pp. 8459–8468. Pmlr, 2020.

Kevin D Smith, Francesco Seccamonte, Ananthram Swami, and Francesco Bullo. Physics-informed implicit representations of equilibrium network flows. *Advances in Neural Information Processing Systems*, 35:7211–7221, 2022.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. *arXiv preprint arXiv:2402.02592*, 2024.

Yuankai Wu, Dingyi Zhuang, Aurelie Labbe, and Lijun Sun. Inductive graph neural networks for spatiotemporal kriging. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 4478–4485, 2021.

Qianxiong Xu, Cheng Long, Ziyue Li, Sijie Ruan, Rui Zhao, and Zhishuai Li. Kits: Inductive spatio-temporal kriging with increment training strategy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12945–12953, 2025.

Jiaxuan You, Xiaobai Ma, Yi Ding, Mykel J Kochenderfer, and Jure Leskovec. Handling missing data with graph representation learning. *Advances in Neural Information Processing Systems*, 33: 19075–19087, 2020.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017.

Qianru Zhang, Haixin Wang, Siu-Ming Yiu, and Hongzhi Yin. Graph masked autoencoder for spatio-temporal graph learning. *arXiv preprint arXiv:2410.10915*, 2024.

Xu Zhang, Yongshun Gong, Xinxin Zhang, Xiaoming Wu, Chengqi Zhang, and Xiangjun Dong. Mask-and contrast-enhanced spatio-temporal learning for urban flow prediction. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, pp. 3298–3307, 2023.

Qishen Zhou, Yifan Zhang, Michail A Makridis, Anastasios Kouvelas, Yibing Wang, and Simon Hu. Network-wide freeway traffic estimation using sparse sensor data: A dirichlet graph auto-encoder approach. *arXiv preprint arXiv:2503.15845*, 2025.