Commun Transp Res

Research Article

GSPINN: A Graph Sequential Physics-Informed Surrogate for Trip Travel Time Prediction

Blessing Itoro Afolayan[1], Arka Ghosh[1,✉], Santhanakrishnan Narayanan[2], Constantinos Antoniou[2], Antonio D. Masegosa[1,3], Jenny F. Calderin[1]

Cite this article: https://doi.org/10.26599/COMMTR.2026.9640044

ABSTRACT: Accurate and computationally efficient traffic prediction remains a fundamental challenge for transportation systems, as microscopic simulators are often too expensive for large-scale applications. This paper addresses this limitation by proposing a physics-consistent surrogate modeling framework, the Graph Sequential Physics-Informed Neural Network (GSPINN). The approach integrates graph-based spatial representation with sequence-aware path aggregation to model trip travel time. It introduces a physics-informed learning formulation that encourages monotonic relationships between travel time and key traffic variables through input-output gradient constraints. To assess robustness across varying traffic conditions, the framework is applied to four heterogeneous road networks, each characterized by distinct topology, demand patterns, and control regimes. The results show consistent predictive performance and stable behavioral properties across all settings. Complementary SHAP-based interpretability further indicates that the model captures network-specific feature dependencies in each case, providing evidence that it adapts to local traffic dynamics rather than overfitting to a single environment. In addition to accuracy and reliability, the proposed surrogate provides substantial computational advantages at inference time, achieving speed-ups of 3x to 55x. This work therefore shows that embedding physically meaningful structure into learning objectives is an effective strategy for traffic surrogate modeling, yielding models that maintain competitive predictive accuracy while substantially improving directional behavioral consistency.

KEYWORDS: graph neural networks, surrogate modeling, travel time prediction, sumo, intelligent transportation systems, simulation acceleration

1 Introduction

Intelligent Transportation Systems (ITS) mostly rely on simulation and prediction tools to support traffic management, infrastructure planning, and operational control. Microscopic traffic simulators such as SUMO (Krajzewicz et al., 2002) provide detailed, agent-based representations of vehicle interactions and are widely used to evaluate operational strategies prior to deployment. However, their computational cost remains a major bottleneck: simulating even a few traffic scenarios can require longer runtimes, which is prohibitive for time-critical applications such as what-if analysis and large-scale scenario exploration.

To mitigate this limitation, surrogate modeling approaches have been proposed to emulate the input-output behaviour of detailed simulators at a fraction of the computational cost. Existing work has explored purely data-driven approaches (Fang et al., 2021; Jiang and Luo, 2022a; Krajzewicz et al., 2002; Lundberg and Lee, 2017; Shi et al., 2020; You et al., 2017) including recurrent neural networks (Yu et al., 2017), convolutional architectures (Belhadi et al., 2020), and graph neural networks (GNNs) (Jiang and Luo, 2022b), for traffic prediction and as surrogates for agent-based transport models. While these methods can achieve strong predictive performance within the training domain, they exhibit three key limitations. First, a number of prior approaches assume access to detailed traffic state variables (e.g., density, queue length, or vehicle trajectories) that are available in simulators but difficult or impossible to measure consistently in the real world. Second, they often neglect physical constraints inherent to traffic flow, leading to implausible predictions such as speeds exceeding speed limits, which reduces robustness under distribution shifts. Third, many approaches operate largely as black boxes, providing limited insight into how network topology and attributes influence predictions.

ssThis study addresses these limitations by developing a simulator independent, physics-aware surrogate for trip travel time prediction on regional road networks. We propose GSPINN (Graph Sequential Physics-Informed Neural Network), which integrates GraphSAGE spatial message passing (Sroczyński and Czyżewski, 2023) with (Kipf and Welling, 2016) GRU-based sequential route aggregation and physics-informed monotonicity constraints to model trip travel time dynamics along road network paths. In this formulation, each node corresponds to a road segment (edge in the physical network), while directed graph edges encode feasible vehicle transitions at junctions. This formulation aligns with standard transportation modeling principles, where traffic states are naturally defined at the road-segment level and junctions govern connectivity. Node features consist exclusively of field-observable and structural attributes, ensuring simulator independence. To motivate this design,

[1] University of Deusto, Bilbao 48007, Spain. [2] Chair of Transportation Systems Engineering, School of Engineering and Design, Technical University of Munich, Munich 80333, Germany. [3] IKERBASQUE, Basque Foundation for Science, Bilbao 48007, Spain.

✉ Corresponding author. E-mail: **arka.ghosh@deusto.es**

Received: March 13, 2026; Revised: April 14, 2026; Accepted: August 5, 2026

清華大学出版社 Tsinghua University Press | SciOpen

Figure 1 presents a conceptual overview that separates the simulator's hidden complexity from the surrogate learning process. The GSPINN architecture integrates two complementary learning mechanisms: (i) a graph neural network that captures spatial dependencies across connected road segments, and (ii) a physics-informed regularization strategy that promotes fundamental traffic-flow constraints to ensure physically plausible predictions. The microscopic simulator is used solely to generate supervisory labels (travel times) during training and is not required during inference. As a result, the surrogate learns a direct mapping from observable network attributes to travel times, enabling millisecond-scale inference and making it well suited for rapid offline analysis, policy screening, and large-scale simulation studies. Although the framework could, in principle, be adapted using field-collected data, such validation is beyond the scope of this work and remains an important direction for future research. The main contributions of this work are as follows:

- We propose a graph-based surrogate modeling framework, GSPINN, for trip travel time prediction that integrates spatial network structure with sequential path-level aggregation. Beyond architectural design, the framework enables modeling of path-dependent traffic dynamics in a way that captures both topological dependencies and temporal propagation effects along trips.
- We formulate a physics-consistent learning objective by explicitly enforcing monotonic relationships between travel time and key roadway attributes through input-output gradient constraints. We demonstrate that this formulation is a principled mechanism that significantly improves the physical plausibility of model predictions by reducing violations of expected traffic behaviour without degrading predictive accuracy.
- We provide a systematic multi-network evaluation across four heterogeneous road networks, varying in topology, demand patterns, and control strategies. The results show that the proposed approach maintains stable predictive performance and consistent behavioural properties across networks, thereby establishing its demonstrated robustness across independently trained networks beyond a single simulation setting and supporting its applicability to diverse ITS scenarios.
- We conduct comprehensive diagnostic and interpretability analyses, including predictive performance evaluation, physics-consistency assessment, and SHAP-based feature attribution. These analyses provide quantitative evidence of how physical constraints influence model behaviour and offer insights into the relative importance of roadway features in shaping travel time predictions.

Together, these elements constitute a fast surrogate for traffic simulation, enabling scalable "what-if" analysis and scenario evaluation on complex urban road networks. The surrogate approximates the simulator's aggregate input-output behaviour without attempting to replicate individual internal processes.

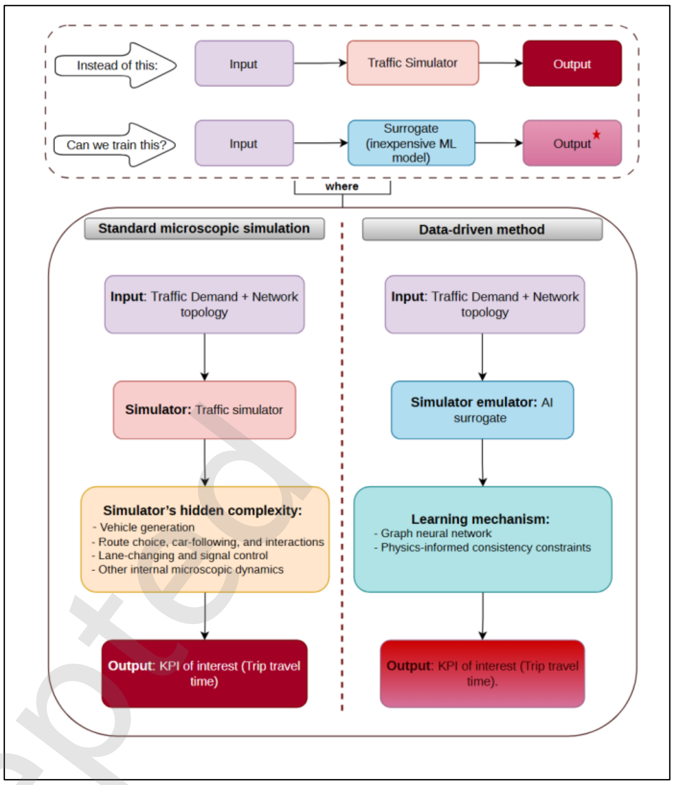


Figure 1. Conceptual overview of the proposed surrogate modeling framework. Instead of repeatedly running computationally expensive traffic simulations, the surrogate learns a mapping function F_θ (X) that directly predicts key performance indicators (KPIs) from simulator-independent input features.

2 Related Work

2.1 Traffic Forecasting

Traffic forecasting has traditionally been formulated as a time-series prediction problem. Early approaches relied on statistical models such as Autoregressive Integrated Moving Average (ARIMA), which capture temporal correlations but assume linear dynamics and stationarity. Classical machine learning methods, including Support Vector Regression (SVR)(Wu et al., 2004a) and k-Nearest Neighbors (k-NN)(Sun et al., 2018), later improved predictive performance by modeling nonlinear relationships, but they require extensive feature engineering and treat road segments largely as independent entities. These limitations motivated the adoption of deep learning models that explicitly encode spatial structure. Graph-based approaches represent road networks as graphs, enabling spatial dependency modeling through message passing between connected road segments and junctions. Graph neural networks (GNNs) and graph attention networks (GATs) (Veličković et al., 2018) have demonstrated substantial improvements by incorporating network topology into traffic prediction. However, these models remain fundamentally local: information propagates only within a limited neighborhood, making it difficult to capture long-range congestion effects arising from network-wide interactions. Moreover, purely data-driven graph models lack explicit physical grounding and may produce predictions that violate basic traffic-flow constraints, particularly under extreme traffic conditions. Because urban road networks inherently follow non-Euclidean structures, standard convolutional or recurrent architectures struggle to capture spatial propagation dynamics. Thereby, GNNs have emerged as the state-of-the-art paradigm for traffic forecasting. Jiang and Luo (2022) established that GNNs outperform traditional models by effectively modeling the complex graph structures of transportation systems and embedding localized contextual information (Jiang and Luo, 2022a). The integration of spatial message passing with sequential modeling has been a major focus from 2023 to 2025. Approaches such as the Dynamic Spatial-Temporal Graph Convolutional Network (DST-GraphSAGE) and hybrid frameworks unifying GNNs with Recurrent Neural Networks (like GRUs) have successfully tackled dynamic travel time predictions by explicitly capturing both exogenous dependencies and sequential trajectory features. Despite these architectural advances, purely data-driven GNNs remain susceptible to learning spurious, physically impossible correlations when trained on heavy-tailed or biased traffic datasets.

2.2 Machine Learning Surrogates for Microscopic Simulation

The concept of replacing agent-based simulators with purely data-driven models has gained substantial traction. Recent literature confirms that deep learning architectures can map macroscopic input parameters to key performance indicators with accuracy rivaling that of exhaustive simulators. For instance, Sroczyński and Czyżewski (2023) illustrated that carefully calibrated machine learning algorithms could predict road traffic metrics as effectively as complex microscopic models (Lundberg and Lee, 2017) . More recently, Natterer et al. (2025) provided a definitive proof of concept by training Graph Neural Networks (GNNs) on extensive MATSim simulation data (Natterer et al., 2025). Their work successfully emulated the impacts of capacity reduction policies across a large-scale urban network in Paris, effectively bypassing the prohibitive runtimes that traditionally bottleneck optimization-based transportation policy analysis. However, while these models successfully replicate simulator outputs, they frequently act as black boxes that fail to guarantee logical consistency under novel edge cases or extreme distribution shifts.

2.3 Physics-Informed Neural Networks

Physics-Informed Neural Networks (PINNs), introduced by Raissi et al. (Raissi et al., 2019, 2017), incorporate governing physical laws into neural networks by penalizing violations of partial differential equations (PDEs) during training. This paradigm has been successfully applied across a range of scientific computing tasks, enabling physically consistent learning from sparse or noisy data (Lu et al., 2021). However, classical PINNs are primarily designed for continuous domains with relatively simple geometries and do not scale naturally to large, irregular, and topologically complex systems such as urban traffic networks (Cuomo et al., 2022). Urban traffic systems differ substantially from the domains typically addressed by classical PINNs. Traffic networks are inherently graph-structured, consisting of road segments and intersections connected through a complex topology. They also evolve temporally and depend on interactions among neighboring links rather than continuous spatial fields. Monotonicity constraints have also been explored in machine learning models to enforce known relationships between inputs and outputs. Approaches such as deep lattice networks (You et al., 2017) incorporate architectural constraints to guarantee monotonic behaviour with respect to selected input features. Other approaches employ constrained optimization techniques to enforce monotonic relationships. In contrast, our work encourages monotonicity through a physics-informed loss term that penalizes violations of expected traffic flow behaviour, allowing the graph-based surrogate model to learn flexible representations while still respecting key transportation principles.

2.4 Interpretability

As traffic surrogate models continue to advance, algorithmic transparency and reliability have become increasingly important. Between 2024 and 2026, researchers have actively integrated SHapley Additive exPlanations (SHAP) (Lundberg and Lee, 2017) into graph architectures, adapting feature attribution to multi-hop computational subgraphs to provide domain experts with transparent, local, and global explanations of the network's predictive logic. Simultaneously, addressing the stochasticity of urban traffic has led to the integration of conformal prediction frameworks into GNNs. Patil et al. (2025) proposed an adaptive conformal prediction (ACP) framework for GNNs, providing statistically reliable uncertainty bounds around travel time predictions that dynamically adjust to varying environmental conditions (Patil et al., 2025).

The above literature highlights a critical gap: existing physics-informed models either ignore graph topology, while graph-based traffic models typically lack physical consistency, this work bridges these directions by proposing a unified framework that integrates topology-aware graph encoding and physics-informed regularization. The GSPINN architecture builds directly upon these progressive foundations by synthesizing the spatial power of GraphSAGE, the sequential logic of GRUs, the theoretical grounding of monotonic physics-informed constraints, and SHAP-based interpretability, it directly addresses the critical gaps identified in the current state-of-the-art.

3 Methodology

This section presents the traffic-state generation procedure, graph-based representation, and the surrogate learning formulation for the GNN framework.

3.1 Traffic-State Generation

To capture interactions between network topology, traffic demand, and signal control, we generate a multi-state dataset using the SUMO microscopic simulator. The physical network $\mathcal{G}$ is fixed, while traffic states vary through systematic perturbations. Each *traffic state* is defined as

$$\mathcal{T}^{(k)} = \left(\mathcal{G}, d^{(k)}, u^{(k)}\right), \tag{1}$$

where $d^{(k)}$ is a uniform demand scaling factor applied to all OD pairs and $u^{(k)}$ represents adjusted signal timings. Perturbations applied per state include:

1. **Demand scaling** modifies baseline OD flows via $d^{(k)}$.
2. **Signal control adjustment:** proportional changes to green phase durations while preserving operational feasibility.

This dual perturbation strategy induces both demand-driven and control-driven congestion regimes, enabling the surrogate model to learn under heterogeneous network-wide conditions. The responses corresponding to all completed trips within each traffic state in the network are given as:

$$y_{\text{trip}}^{(k)} = \mathcal{S}\left(\mathcal{T}^{(k)}\right), \tag{2}$$

where $\mathcal{S}$ is the microscopic simulator. Collecting N traffic states yield

$$\{\mathcal{T}^{(1)}, \dots, \mathcal{T}^{(N)}\}. \tag{3}$$

3.2 Graph Representation: Road Segment Graph Formulation

The surrogate model operates on a road-segment graph.

Node definition: The surrogate model operates on a road network graph, capturing all road segments within a traffic state. Trip travel time predictions are obtained by indexing into the global node embeddings along each vehicle path.

$$v_e \in \mathcal{V}_{\text{state}}. \tag{4}$$

Line-graph Connectivity: Two road segments (nodes) are connected if they share a common junction, i.e., if vehicles can legally transition from segment e_i to segment e_j via a junction. This produces a directed connectivity graph encoding feasible vehicle movements, as illustrated in Figure 2 from the Ingolstadt road network.

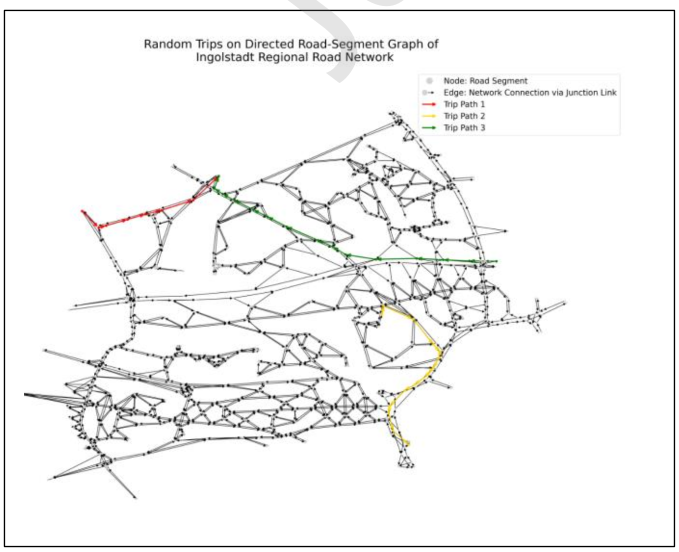


Figure 2. Graph Representation: Each road segment is treated as the graph node, while the road junctions are the graph edges.

Node and edge features: Node features $\mathbf{x}_v$ describe road-segment geometry, traffic attributes, and control parameters (Table 2). Edge features $\mathbf{x}_{uv}$ encode upstream junction characteristics (Table 1). Each trip graph maintains all edges and nodes along the vehicle's path, enabling end-to-end trip travel time prediction.

Trip travel time prediction: Once the global graph is encoded, each trip k is represented as an ordered sequence of node embeddings along its path. The model refines these embeddings using a sequential GRU that aggregates segment embeddings into a trip representation, which is then passed through a readout MLP for travel time prediction. Details of the architecture are given in Architecture, Section 3.5.

3.3 Dataset Partitioning

To preserve topological and traffic-state consistency, splits are performed at the traffic-state level. 50 distinct traffic states, each containing multiple trips. Of which we have:

- Training set: 70% of traffic states,
- Validation set: 15% of traffic states,
- Test set: 15% of traffic states.

This strategy ensures that the surrogate is evaluated on unseen global operating conditions while preserving complete, topologically consistent graphs in every split.

Table 1. Edge (junction) features.

Feature category	Description
Topological structure	Number of incoming and outgoing edges
Junction geometry	Junction type (priority, traffic light, etc.)
Turning behaviour	Fractions of left, right, and straight maneuvers

Table 2. Node (road-segment) features.

Feature category	Description
Route geometry	Road length (m); number of lanes; road curvature
Speed and demand	Speed limit (m/s); demand flow (veh/s) obtained from OD; demand class
Signal influence	Green split fraction; cycle length; number of phases

Table 3. Trip-level feature.

Feature category	Description
Trip descriptor	Trip sequence length ℓ_k: number of road segments along the trip path (normalised)

Feature category	Description
Target variable	Trip travel time (s)

3.4 Surrogate Model Formulation

The surrogate model approximates this mapping:

$$\hat{\mathcal{S}}_\theta \approx \mathcal{S}, \qquad \hat{y}^{(k)}_{\text{trip}} = \hat{\mathcal{S}}_\theta\left(\mathbf{x}^{\text{state}}_v, \mathbf{x}^{\text{state}}_{uv}, \text{trip path}_k\right) \tag{5}$$

where $\mathbf{x}^{\text{state}}_v$ and $\mathbf{x}^{\text{state}}_{uv}$ are node and edge features of the global state graph, and trip path_k selects the subset of nodes corresponding to trip k.

3.5 The Architecture

The GSPINN surrogate operates on a state graph where nodes represent road segments and edges encode junction connectivity. Each node and edge carries a feature vector describing traffic, geometry, and control attributes. The architecture is composed of three main stages: node feature preprocessing, graph encoding, and trip prediction.

Node feature preprocessing: Node features $\mathbf{x}_v$ are first standardized using the mean and standard deviation computed over all training states:

$$\tilde{\mathbf{x}}_v = \frac{\mathbf{x}_v - \boldsymbol{\mu}_{\text{train}}}{\boldsymbol{\sigma}_{\text{train}}} \tag{6}$$

Standardized features are then projected to a hidden dimension d_h via a learnable linear layer:

$$\mathbf{h}^{(0)}_v = \text{ReLU}(\mathbf{W}_{\text{in}}\tilde{\mathbf{x}}_v) \tag{7}$$

Graph message passing: Node embeddings are updated through a stack of GraphSAGE layers that propagate information across the road network. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be the graph where $\mathcal{V}$ is the set of nodes (road segments) and $\mathcal{E}$ is the set of edges (junctions). Each node $v \in \mathcal{V}$ has an embedding $\mathbf{h}^{(l)}_v$ at layer l.

GraphSAGE performs neighborhood aggregation followed by feature transformation, allowing each road segment to incorporate contextual information from its adjacent segments. Node representations are updated as

$$\mathbf{h}^{(l+1)}_v = \text{ReLU}\left(\mathbf{W}^{(l)}\left[\mathbf{h}^{(l)}_v \parallel \text{AGG}\left(\{\mathbf{h}^{(l)}_u : u \in \mathcal{N}(v)\}\right)\right]\right), \tag{8}$$

where:

- $\mathbf{h}^{(l)}_v \in \mathbb{R}^{d_h}$ is the embedding of node v at layer l,
- $\mathbf{h}^{(l+1)}_v$ is the updated embedding,
- $\mathcal{N}(v)$ denotes the set of neighboring nodes of v,
- $\text{AGG}(\cdot)$ represents the neighborhood aggregation operator (mean aggregation in our implementation),
- $\parallel$ denotes vector concatenation,
- $\mathbf{W}^{(l)}$ is a learnable transformation matrix at layer l,
- $\text{ReLU}(\cdot)$ is the element-wise Rectified Linear Unit activation.

In our architecture, three GraphSAGE layers are stacked to progressively propagate spatial traffic information across the network, producing the final node embeddings

$$\mathbf{h}^{\text{final}}_v \tag{9}$$

Route Sequential Aggregator and Trip prediction: Once the global graph is encoded, each trip k is represented by the ordered sequence of nodes along its path, then the sequence of node embeddings corresponding to the trip is processed using a gated recurrent unit (GRU), which sequentially aggregates information along the route. Let the ordered node embeddings of trip k be

$$\{\mathbf{h}^{\text{final}}_{v_1}, \mathbf{h}^{\text{final}}_{v_2}, \dots, \mathbf{h}^{\text{final}}_{v_T}\} \tag{10}$$

The GRU iteratively updates a hidden state to summarize the cumulative traffic conditions encountered along the route:

$$\mathbf{s}_t = \text{GRU}\left(\mathbf{h}^{\text{final}}_{v_t}, \mathbf{s}_{t-1}\right) \tag{11}$$

where $\mathbf{s}_t$ denotes the hidden state at step t. The final hidden state provides the trip representation

$$\mathbf{h}^{(k)}_{\text{trip}} = \mathbf{s}_T \tag{12}$$

This representation captures the sequential accumulation of traffic interactions along the trip. The trip embedding is then concatenated with a normalized trip sequence length ℓ_k (the number of road segments along the trip path) and passed through a multi-layer perceptron (MLP) for travel time prediction:

$$\hat{y}^{(k)}_{\text{trip}} = f_{\text{MLP}}\left(\mathbf{h}^{(k)}_{\text{trip}}, \ell_k\right) \tag{13}$$

Alternative Route Pooling Strategies (Ablation Study): To evaluate the influence of different route aggregation mechanisms, we also consider two alternative pooling strategies: mean pooling and long short-term memory (LSTM) pooling. These variants are used exclusively for the ablation analysis to examine how different methods of aggregating node embeddings along a route affect travel time prediction performance.

Mean pooling: The simplest aggregation strategy computes the average of all node embeddings along the trip path. Let the ordered node embeddings of the trip k be $\{\mathbf{h}^{\text{final}}_{v_1}, \mathbf{h}^{\text{final}}_{v_2}, \dots, \mathbf{h}^{\text{final}}_{v_T}\}$. Trip representation is

$$\mathbf{h}^{(k)}_{\text{trip}} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{h}^{\text{final}}_{v_t} \tag{14}$$

This operation treats all nodes independently and ignores the sequential order of road segments along the route.

LSTM pooling: To capture sequential dependencies along the route, we also evaluate an LSTM-based aggregator. The node embeddings are processed sequentially using an LSTM network:

$$\mathbf{s}_t = \text{LSTM}\left(\mathbf{h}^{\text{final}}_{v_t}, \mathbf{s}_{t-1}\right) \tag{15}$$

where $\mathbf{s}_t$ denotes the hidden state at step t. Similar to the GRU formulation, the final hidden state summarizes the trip:

$$\mathbf{h}_{\text{trip}}^{(k)} = \mathbf{s}_T \tag{16}$$

The resulting trip embedding is then concatenated with the normalized trip length and passed to the same MLP predictor used in the GRU-based model.

The combination of GraphSAGE and GRU allows the model to capture both spatial and sequential dependencies inherent in traffic systems. The GraphSAGE layers first encode topology-aware interactions between adjacent road segments, enabling each node embedding $\mathbf{h}_v^{\text{final}}$ to reflect congestion propagation, geometric characteristics, and signal control effects from neighboring segments. However, travel time depends not only on local conditions but also on the cumulative conditions encountered along a route. The GRU therefore processes the ordered sequence of node embeddings along each trip, sequentially integrating traffic conditions encountered across successive segments. This design allows the surrogate to model how delays accumulate along routes while preserving the spatial context encoded by the graph, leading to improved representations of network dynamics and trip travel time prediction. While this architecture captures the structural and sequential characteristics of traffic flow, it does not guarantee by itself that the learned relationships remain physically consistent. In particular, purely data-driven models may learn counterintuitive sensitivities, such as predicting shorter travel times when roadway length or signal cycle length increases. To mitigate such inconsistencies, the training objective is augmented with a physics-informed regularization that substantially reduces violations of directional monotonicity with respect to selected traffic related attributes, as described in the following section.

Positive Monotone Physics-Informed Loss: Classical models such as the Lighthill Whitham Richards (LWR) conservation law (Lighthill and Whitham, 1955) describe traffic as a continuous spatio-temporal density field governed by $\rho + \partial_x q(\rho) = 0$.

Such formulations require explicit density and flow variables, calibrated fundamental diagrams, and continuous spatial discretization. In contrast, the proposed formulation should not be interpreted as a full traffic-flow physics model in the sense of explicitly modeling conservation laws, queue spillback, shockwave propagation, or microscopic vehicle interactions. Instead, the proposed regularization introduces lightweight directional consistency constraints that encode expected first-order behavioral relationships between selected roadway attributes and trip travel time. The objective of these constraints is to improve behavioral plausibility and directional consistency while preserving the flexibility and scalability of graph-based surrogate learning. Hence, we adopt a lighter form of physics consistency based on first-order directional monotonicity. This encourages fundamental macroscopic principles without imposing a full dynamical PDE structure, preserving scalability while maintaining physically coherent predictions. From the surrogate operator equation (5), let $\mathcal{F}_+$ denote the subset of input feature indices corresponding to roadway attributes that are known to exhibit an increasing relationship with trip travel time (e.g., road length, signal cycle length and curvature). For each $f \in \mathcal{F}_+$, physical consistency requires

$$\frac{\partial \hat{y}_{\text{trip}}^{(k)}}{\partial x_f} \geq 0 \tag{17}$$

To encourage this directional condition during training, we introduce the monotonicity regularization term,

$$\mathcal{L}_{\text{PINN}} = \frac{1}{|\mathcal{F}_+|} \sum_{f \in \mathcal{F}_+} \text{ReLU}\left(-\frac{\partial \hat{y}_{\text{trip}}^{(k)}}{\partial x_f}\right) \tag{18}$$

which penalizes violations of non-negativity in the corresponding partial derivatives. Hence, the learned surrogate $\hat{\mathcal{S}}_\theta$ is encouraged to exhibit directionally monotone behavior with respect to the selected feature subset $\mathcal{F}_+$. Increasing physically monotone attributes, therefore, is discouraged from reducing predicted travel time. For example, increasing road length cannot decrease predicted delay, and increasing signal cycle length cannot reduce predicted trip time. The constraint thus restricts the hypothesis space $\hat{\mathcal{S}}_\theta$ to functions satisfying directional consistency with macroscopic traffic principles, acting as a mechanism that improves robustness and prevents counter-physical sensitivities under distributional shift. Gradients are taken with respect to node features of all nodes visited along the route. Thus, the monotonic constraint is encouraged at the path level. The imposed monotonic assumptions are intended to capture dominant macroscopic operational trends rather than strict universal relationships. Under highly coordinated signal systems or locally oversaturated conditions, variables such as cycle length or phase allocation may exhibit non-monotonic effects due to queue spillback, progression offsets, demand redistribution, or route adaptation dynamics. The proposed regularization therefore encourages globally consistent traffic trends while still permitting localized deviations through soft physics-informed constraints.

Loss functions: To train the surrogate model, we employ the Smooth L1 loss (also known as the Huber loss (Huber, 1964)) between the predicted and real travel times. Formally, trip prediction loss is defined as:

$$\mathcal{L}_{\text{trip}} = \frac{1}{N} \sum_{i=1}^{N} \ell_\beta\left(\hat{y}_i - y_i\right) \tag{19}$$

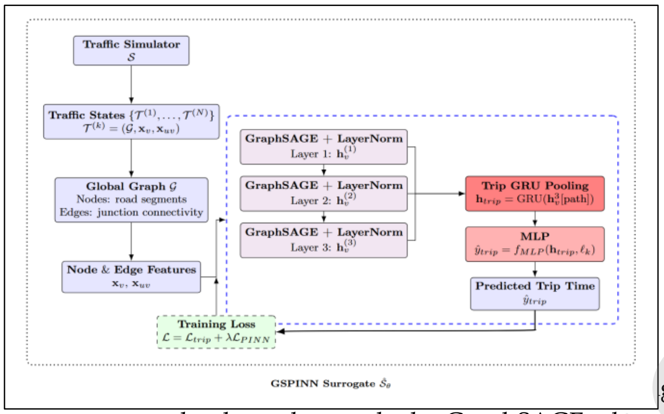
where $r_i = \hat{y}_i - y_i$ denotes the prediction residual and $\ell_\beta(\cdot)$ is defined as

$$\ell_\beta(r) = \begin{cases} \frac{1}{2\beta} r^2, & |r| < \beta \\ |r| - \frac{\beta}{2}, & \text{otherwise.} \end{cases} \tag{20}$$

The Smooth L1 loss behaves quadratically for small residuals and transitions to a linear penalty for larger deviations. This formulation preserves the stable optimization properties of mean squared error while reducing sensitivity to large errors. In traffic networks, the distribution of trip travel times is typically heavy-tailed: while most trips have moderate travel times, a small number of trips may experience very large delays due to congestion or long network paths. These rare events can produce extremely large regression residuals that influence optimization when using a purely quadratic loss such as mean squared error.

To mitigate this effect, we use the Smooth L1 loss with transition parameter $\beta = 50$. This choice allows the loss to behave approximately as a squared error for small prediction deviations while switching to a linear penalty for large residuals, thereby limiting the influence of the few trips with exceptionally long travel times. As a result, the surrogate model focuses on accurately capturing the dominant travel time patterns in the network while remaining robust to extreme delays. Training optimizes a combination of L1 and a positive monotone physics-informed loss:

$$\mathcal{L} = \mathcal{L}_{\text{trip}} + \lambda \mathcal{L}_{\text{PINN}} \tag{21}$$



gs are computed through stacked GraphSAGE layers, aggregated along trip paths using a GRU, and decoded with an MLP to predict trip time. Training minimizes a combined regression and physics-informed monotonicity loss.

4 Experiments

This section evaluates the proposed framework in terms of predictive accuracy and compares it against a comprehensive set of baseline models. The experimental analysis is organized along four complementary dimensions: (i) predictive performance, (ii) ablation analysis, (iii) speed-up calculation, and (iv) interpretability, which comprises the model behaviour as well as the feature explainability.

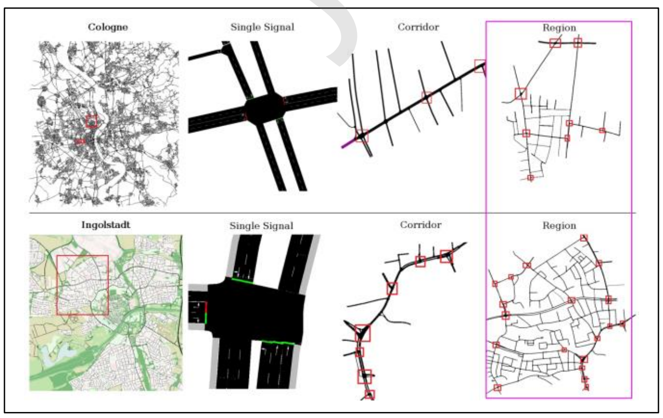


Figure 4. The Ingolstadt and Cologne regional sub-network from the Reinforcement Signal Control (RESCO) benchmark(Ault and Sharon, 2021)

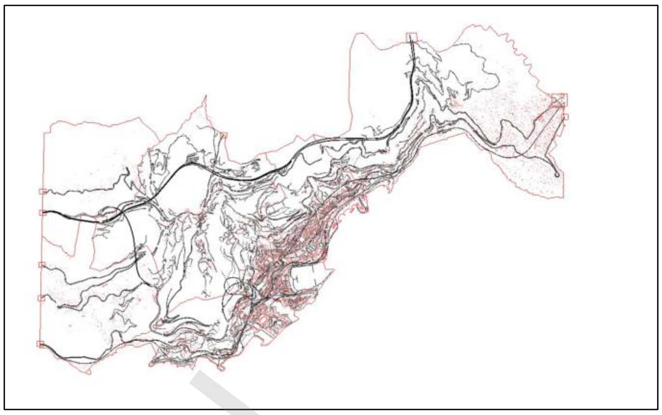


Figure 5. The MoST Road Network(Codeca and Härri, 2018)

Table 4. Test-set predictive accuracy of baseline and graph-based models across multiple datasets.

Dataset	Model	MAE	RMSE	R^2	Ps. (r)	Sp. (ρ)
Ingolstadt	SVR	63.044	125.267	0.538	0.751	0.853
Ingolstadt	Ridge	78.125	132.966	0.480	0.693	0.778
Ingolstadt	MLP	60.692	108.534	0.654	0.809	0.861
Ingolstadt	GCN	55.956	98.908	0.644	0.807	0.860
Ingolstadt	GATv2	57.437	102.224	0.619	0.794	0.854
Ingolstadt	EGATv2	57.521	101.491	0.625	0.798	0.854
Ingolstadt	Transformer	58.095	105.120	0.597	0.788	0.846
Ingolstadt	ETransformer	58.049	103.235	0.612	0.791	0.848
Ingolstadt	GSPINN (baseline)	55.102	97.989	0.650	0.814	0.865
Ingolstadt	**GSPINN**	**53.815**	**94.476**	**0.675**	**0.827**	**0.872**
Cologne	SVR	23.999	37.014	0.738	0.860	0.892
Cologne	Ridge	28.176	40.700	0.683	0.826	0.861
Cologne	MLP	22.730	33.215	0.789	0.889	0.908
Cologne	GCN	22.538	33.260	0.782	0.884	0.905
Cologne	GATv2	22.991	33.960	0.772	0.882	0.900
Cologne	EGATv2	24.277	37.023	0.730	0.856	0.868
Cologne	Transformer	23.038	33.633	0.777	0.883	0.895
Cologne	ETransformer	22.901	33.143	0.783	0.887	0.898
Cologne	GSPINN (baseline)	21.037	29.322	0.830	0.912	0.916
Cologne	**GSPINN**	21.074	29.464	0.829	0.911	0.916
MoST	SVR	48.449	69.000	0.920	0.960	0.958
MoST	Ridge	51.173	70.043	0.918	0.958	0.958
MoST	MLP	36.939	52.997	0.953	0.976	0.973
MoST	GCN	47.729	68.029	0.920	0.959	0.956
MoST	GATv2	45.813	65.556	0.926	0.962	0.961
MoST	EGATv2	47.909	67.834	0.920	0.960	0.958
MoST	Transformer	42.762	60.446	0.937	0.968	0.966

Dataset	Model	MAE	RMSE	R^2	Ps. (r)	Sp. (ρ)
MoST	ETransformer	43.075	61.261	0.935	0.967	0.966
MoST	GSPINN (baseline)	35.529	54.686	0.948	0.975	0.975
MoST	**GSPINN**	35.535	**53.551**	**0.950**	**0.975**	**0.975**
LuST	SVR	104.075	191.728	0.646	0.811	0.879
LuST	Ridge	128.335	205.160	0.595	0.772	0.830
LuST	MLP	84.640	137.886	0.817	0.904	0.924
LuST	GCN	81.798	158.425	0.765	0.876	0.933
LuST	GATv2	85.127	160.229	0.759	0.874	0.928
LuST	EGATv2	81.375	152.110	0.783	0.886	0.931
LuST	Transformer	81.910	159.328	0.760	0.875	0.933
LuST	ETransformer	82.898	160.648	0.758	0.875	0.932
LuST	GSPINN (baseline)	68.636	137.475	0.823	0.909	0.947
LuST	**GSPINN**	**66.870**	**134.433**	**0.831**	**0.914**	**0.948**

We benchmark the proposed approach across multiple model categories, including standard machine learning methods, graph-based architectures, and non-graph neural networks such as fully connected networks and Transformer-based models. The standard baselines include SVR (Wu et al., 2004b) and Ridge regression (Hoerl and Kennard, 1970). For graph-based approaches, we evaluate Graph Convolutional Networks (GCN) (Kipf and Welling, 2016), Graph Attention Networks (GATv2) (Brody et al., 2021), and Transformer Convolutions (TransConv) (Shi et al., 2020), a form of graph Transformer. These models differ in how they propagate and aggregate information over the graph, enabling diverse forms of spatial reasoning. As a non-graph baseline, we include a multilayer perceptron (MLP) (Rumelhart et al., 1986), which processes input features independently without explicitly leveraging graph structure.

Table 5. Summary of dataset characteristics and data splits.

Dataset	Time	Train	Val	Test
Ingolstadt	16:00–17:00	138,057	25,446	30,891
Cologne	07:00–08:00	54,985	11,427	12,388
MoST	07:00–09:00	19,017	3,479	4,420
LuST	07:00–08:00	115,538	33,027	64,839

All experiments are conducted on realistic urban traffic networks spanning four distinct scenarios: Ingolstadt and Cologne (derived from the Reinforcement Signal Control (RESCO) benchmark (Ault and Sharon, 2021)), as well as MoST (Codeca and Härri, 2018) and LuST (Codeca et al., 2015). These datasets vary significantly in network topology, demand levels, and traffic dynamics, enabling evaluation across diverse spatial and temporal scales. Each network is represented as a graph where road segments correspond to nodes and junctions define connectivity relationships.

To ensure unbiased evaluation, each dataset is partitioned into mutually exclusive training, validation, and test subsets at the state level. This guarantees that no trip instance appears across splits, preventing information leakage and enabling reliable generalization assessment within networks. Across all datasets, the splits preserve demand diversity while maintaining balanced distributions.

Table 4 reports predictive performance across all evaluated models and Table 5 summarizes the key characteristics of each dataset, including network size, temporal coverage, and data splits. The results in Table 4 provide a comprehensive evaluation of model performance across four heterogeneous traffic datasets: Ingolstadt, Cologne, MoST, and LuST. These datasets differ substantially in scale, temporal coverage, and traffic dynamics, enabling a robust assessment of model stability across multiple independently trained network environments.

Across all datasets, a consistent trend emerges. Linear models such as ridge regression exhibit the weakest performance, confirming that travel time prediction is inherently nonlinear. This is particularly evident in networks such as Ingolstadt and LuST, where R^2 values remain comparatively low. Although SVR improves performance through nonlinear kernel mappings, it still treats trips independently and fails to explicitly capture spatial dependencies within the road network.

Neural models without structural awareness (MLP) consistently outperform classical methods, demonstrating the importance of nonlinear function approximation. However, their inability to model spatial relationships limits further gains, particularly in datasets where congestion propagation and network interactions are dominant.

Graph-based architecture addresses this limitation by explicitly incorporating network topology. Models such as GCN and GATv2 leverage message passing to encode spatial dependencies between connected road segments, resulting in improved predictive accuracy across most datasets. However, the relative gains vary with dataset characteristics. On the other hand, attention-based and Transformer variants show competitive performance but do not consistently outperform simpler graph convolution approaches. This suggests that, for this task, localized spatial interactions play a more critical role than long-range dependencies. Incorporating junction-level features (EGATv2 and ETransformer) yields only marginal improvements, indicating that most predictive signal is already captured through road-segment attributes and graph connectivity.

Across all datasets, the proposed GSPINN model consistently achieves comparable performance. In Ingolstadt, it attains the lowest MAE (53.815) and highest R^2 (0.675), indicating marginally improved modeling of complex urban dynamics. In MoST, it achieves R^2 (0.950), demonstrating comparable performance even in highly irregular traffic regimes. In LuST, which represents the largest dataset, GSPINN also maintains comparable accuracy and correlation, demonstrating potential scalability. In Cologne, performance remains marginally consistent with the best baseline, indicating robustness across medium-scale networks. These results highlight that combining spatial message passing with sequential route modeling enables consistent predictive performance across diverse traffic scenarios.

4.2 Ablation Analysis

To evaluate the impact of the trip-path pooling mechanism, we compare three variants: simple mean pooling, LSTM-based pooling, and GRU-based pooling across all datasets.

Table 6. Ablation study evaluating the impact of trip-path pooling mechanisms across all datasets. Best results per dataset are highlighted in bold.

Dataset	Pooling	MAE	RMSE	R^2	Ps. (r)	Sp. (ρ)
Ingolstadt	Mean	56.552	102.013	0.621	0.799	0.859
Ingolstadt	LSTM	54.504	97.332	0.655	0.820	0.869
Ingolstadt	GRU	**53.815**	**94.476**	**0.675**	**0.827**	**0.872**
Cologne	Mean	22.121	32.033	0.797	0.893	0.909
Cologne	LSTM	**20.851**	**29.348**	**0.830**	0.911	0.915
Cologne	GRU	21.074	29.464	0.829	0.911	**0.916**
MoST	Mean	42.715	60.958	0.936	0.967	0.967
MoST	LSTM	**35.013**	53.582	0.950	0.975	0.975
MoST	GRU	35.535	**53.551**	0.950	0.975	0.975
LuST	Mean	74.682	144.753	0.804	0.899	0.940
LuST	LSTM	67.217	**133.822**	0.832	0.913	0.948
LuST	GRU	**66.870**	134.433	0.831	0.914	0.948

Table 6 summarizes the results across all datasets. Across all datasets, the choice of pooling mechanism has a consistent and significant impact on performance. Mean pooling, while computationally simple, yields the weakest results across all metrics, indicating that ignoring sequential dependencies along the trip path leads to a loss of critical structural information.

In contrast, recurrent pooling mechanisms (LSTM and GRU) consistently improve performance by modeling the ordered dependencies between road segments. In Ingolstadt, GRU pooling achieves the best overall performance, improving R^2 from 0.621 to 0.675 while reducing RMSE substantially. A similar trend is observed in LuST, where GRU pooling provides a notable improvement over mean pooling, highlighting its effectiveness in large-scale and highly complex networks.

For Cologne and MoST, both LSTM and GRU pooling achieve comparable performance, with LSTM slightly outperforming in MAE and R^2 in Cologne, while both methods are nearly identical in MoST. This suggests that the choice between LSTM and GRU is less critical, as both are capable of capturing the underlying sequential dynamics effectively. In all, these results reveal that incorporating sequential modeling into the pooling operation is essential for accurate travel time prediction.

4.3 Computational and Speed-up Analysis

We evaluate the computational efficiency of the proposed surrogate model against microscopic traffic simulation by comparing wall-clock runtimes under aligned workloads. Specifically, both the simulator and the surrogate are evaluated on the same set of network states and trips, ensuring a direct and fair comparison. This avoids a common limitation in prior work, where reported speed-ups are inflated due to mismatched evaluation settings. While the surrogate model requires an upfront training phase, this cost is incurred offline and amortized over repeated inference queries. Once trained, the model enables rapid predictions that are orders of magnitude faster than simulation, making it well-suited for real-time and iterative ITS applications such as dynamic routing and signal control.

All experiments were conducted in CPU-only mode to eliminate any advantage from hardware acceleration. The experiments were executed on an Intel(R) Xeon(R) Silver 4210R CPU @ 2.40 GHz (40 cores, 2 threads per core), providing a reproducible and hardware-consistent benchmark. For this, we measure the total simulation time $T_{\text{simulator}}$ and the total surrogate inference time $T_{\text{surrogate}}$ over all test states. To account for variations in demand levels across datasets, we additionally report per-trip runtimes:

$$t_{\text{simulator}}^{\text{per-trip}} = \frac{T_{\text{simulator}}}{N_{\text{trips}}}, \qquad t_{\text{surrogate}}^{\text{per-trip}} = \frac{T_{\text{surrogate}}}{N_{\text{trips}}},$$

where N_{trips} denotes the total number of evaluated trips. This normalization ensures that comparisons are independent of workload size and reflect computational efficiency. The speed-up is therefore defined as:

$$\text{Speed-up} = \frac{T_{\text{simulator}}}{T_{\text{surrogate}}}.$$

Table 7. Computational efficiency comparison across datasets. Values are reported as total and per-trip runtime.

Dataset	Trips	Simulator time (total/per-trip)	Surrogate time (total/per-trip)	Speed-up
Ingolstadt	30,891	2732.71 / 0.0885	499.43 / 0.0162	5.5×
Cologne	12,388	415.04 / 0.0335	138.90 / 0.0112	3.0×
MoST	4,420	6687.56 / 1.5130	121.23 / 0.0274	55.2×
LuST	64,839	10441.02 / 0.1610	2991.66 / 0.0461	3.5×

Table 7 summarizes the results across all datasets. The surrogate consistently achieves lower runtime both in aggregate and on a per-trip basis. For moderate-sized networks (e.g., Cologne, Ingolstadt, and LuST), the model provides speed-ups in the range of 3 × to 5.5 ×, while more computationally complex scenarios (e.g., MoST) exhibit substantially higher gains due to the increased burden of microscopic simulation, despite involving fewer trips. This indicates that computational gains are primarily driven by simulation complexity rather than dataset size. The reported computational gains correspond specifically to inference-stage acceleration after offline dataset generation and model training. The reported runtimes do not include the costs associated with data collection, graph construction, or model training. Consequently, the proposed framework should not be interpreted as a complete replacement for microscopic traffic simulation. Instead, it is more appropriately viewed as a complementary acceleration tool for repeated inference queries, rapid scenario screening, and iterative traffic analysis once the offline training stage has been completed.

4.4 Explainability

This section examines both the model and the feature explainability. First, the impact of the physics constraints on the model is assessed. Secondly, the beeswarm, as well as the bar plot, is used to reveal the internal reasoning of the surrogate model and to verify whether the learned relationships align with established traffic principles.

4.4.1 Model Behaviour

From the evaluation plots in Figures 7, 8, 9, and 10, corresponding to Ingolstadt, Cologne, MoST, and LuST respectively, the top-left panels show the relationship between predicted and true trip travel times on the test datasets. Across all networks, predictions closely follow the diagonal reference line, indicating strong agreement with ground truth values. The surrogate consistently achieves high coefficients of determination, along with strong Pearson and Spearman correlations, demonstrating its ability to accurately capture both the magnitude and relative ordering of travel times under diverse traffic conditions. The top-right panels present the empirical distributions of true trip travel times for each dataset. All four datasets exhibit heavy-tailed distributions, a well-known characteristic of urban traffic systems, where the majority of trips are relatively short while a smaller proportion experience significantly longer delays due to congestion, signal timing, and network effects.

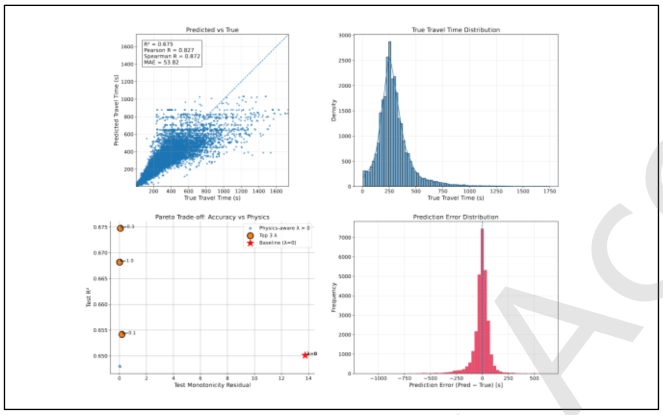


Figure 7. Ingolstadt diagnostic evaluation of the proposed model, illustrating predictive accuracy, dataset characteristics, physics consistency, and error behaviour.

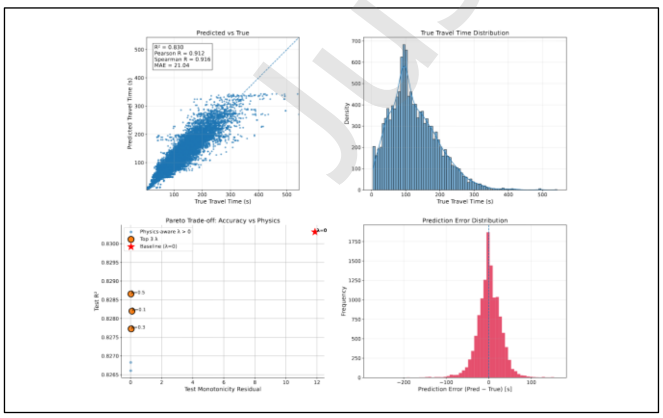


Figure 8. Cologne network diagnostic evaluation of the proposed model, illustrating predictive accuracy, dataset characteristics, physics consistency, and error behaviour.

The bottom-left panels illustrate the trade-off between predictive accuracy and physics consistency across different values of the monotonicity regularization parameter λ. The horizontal axis represents the monotonicity residual, quantifying violations of the imposed physical constraints, while the vertical axis shows the corresponding test R^2. Across all datasets, the unconstrained baseline ($\lambda = 0$) consistently exhibits the highest residuals, indicating substantial violations of expected monotonic relationships. Introducing monotonicity constraints ($\lambda > 0$) significantly reduces these violations while maintaining, and in some cases improving, predictive accuracy. This Pareto behaviour is consistent across small (Cologne), medium (Ingolstadt), and large-scale (LuST) networks, as well as highly dynamic scenarios such as MoST, demonstrating that physics-informed regularization effectively improves model consistency without compromising performance.

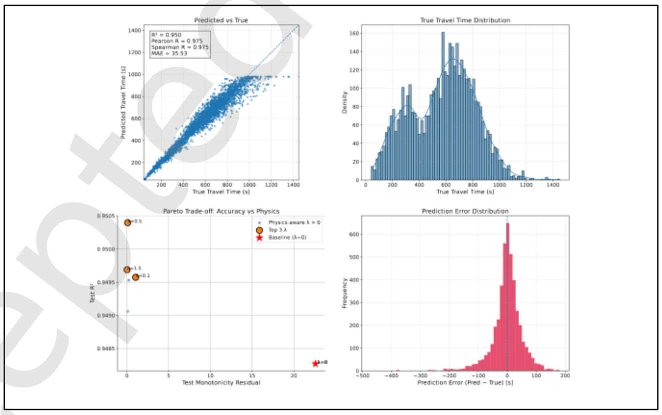


Figure 9. MoST network diagnostic evaluation of the proposed model, illustrating predictive accuracy, dataset characteristics, physics consistency, and error behaviour.

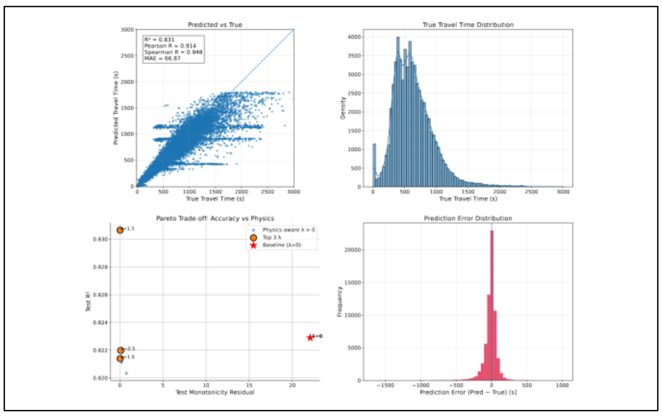


Figure10. LuST network diagnostic evaluation of the proposed model, illustrating predictive accuracy, dataset characteristics, physics consistency, and error behaviour.

Finally, the bottom-right panels show the distributions of prediction errors (predicted minus true travel time). Across all datasets, the error distributions are centered close to zero, indicating minimal systematic bias. While variability increases with network scale and demand complexity, the majority of errors remain concentrated within a moderate range, confirming stable and robust predictive behaviour across heterogeneous traffic regimes.

4.4.2 Monotonicity Constraint Impact

The results in Figure 11 provide evidence of the effectiveness of the proposed monotonicity-constrained formulation. As shown in Figure 11(A), the introduction of monotonicity constraints has a negligible impact on predictive accuracy across all datasets. The mean absolute error (MAE) remains largely unchanged, with differences that are marginal (e.g., 21.037 vs. 21.074 in Cologne, 35.529 vs. 35.535 in MoST) and, in some cases, even slightly improved (e.g., 68.636 to 66.870 in LuST and 55.102 to 53.815 in Ingolstadt). This confirms that enforcing domain constraints does not compromise the model's ability to capture the underlying data distribution. In contrast, Figure 11(B) reveals a consistent reduction in monotonicity violations across all scenarios. The unconstrained baseline exhibits substantial violations, ranging from 11.873 to 22.584, indicating frequent inconsistencies with known physical relationships. However, the constrained model reduces these violations by several orders of magnitude, bringing them close to zero (e.g., 11.873 to 0.025 in Cologne, 13.758 to 0.071 in Ingolstadt, 22.584 to 0.090 in MoST, and 22.068 to 0.032 in LuST). The logarithmic scale further highlights the magnitude of this improvement, demonstrating reductions exceeding 99% in all cases. Monotonicity violations are quantified using gradient-based sensitivity analysis. For selected physically constrained input features (e.g., road length, cycle length, and curvature), we compute the partial derivative of the model output with respect to each feature using automatic differentiation. Any negative derivative indicates a violation of the monotonicity constraint. The final metric is obtained by averaging this violation magnitude across all sampled nodes, edges, and test batches. These findings indicate a key advantage of the proposed approach, as it achieves strong predictive performance while promoting adherence to fundamental domain principles. Unlike conventional models that may produce physically implausible predictions despite good aggregate accuracy, the monotonicity-constrained model encourages consistent and reliable behavior. This is critical for traffic systems, where physical validity is as important as predictive accuracy. The primary benefit of the proposed monotonic regularization is improved directional behavioral consistency rather than uniformly superior predictive accuracy. Across several datasets, the constrained formulation achieves predictive performance comparable to the unconstrained variant while substantially reducing physically implausible feature-response relationships. The proposed regularization is therefore a mechanism for improving behavioral robustness and interpretability while preserving competitive predictive performance.

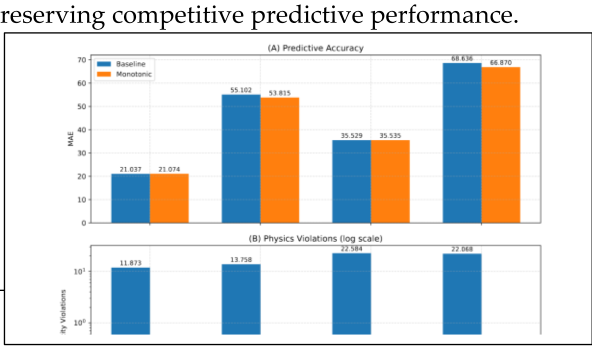


Figure 11. Comparison of predictive performance between the unconstrained baseline and the monotonicity-constrained GSPINN across the four traffic networks. (A) MAE shows that the monotonicity penalty has negligible impact on predictive performance, with comparable or slightly improved accuracy on every network. (B) Monotonicity violations show a dramatic reduction, by two to three orders of magnitude, once the penalty is applied.

4.4.3 Feature-Level Explainability

To further understand the internal behaviour of the proposed surrogate model, we perform a SHAP analysis. SHAP provides a framework for attributing the contribution of each input feature to the final prediction. This allows us to quantify which roadway and traffic characteristics most influence predicted trip travel times.

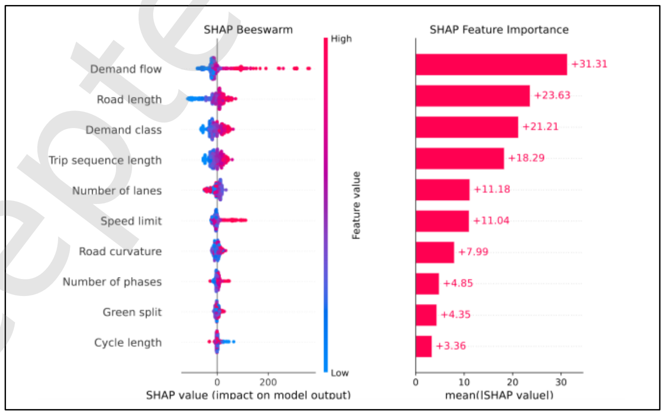


Figure 12. Ingolstadt SHAP bee-swarm summary plot (left) and mean absolute values (right) indicating feature importance

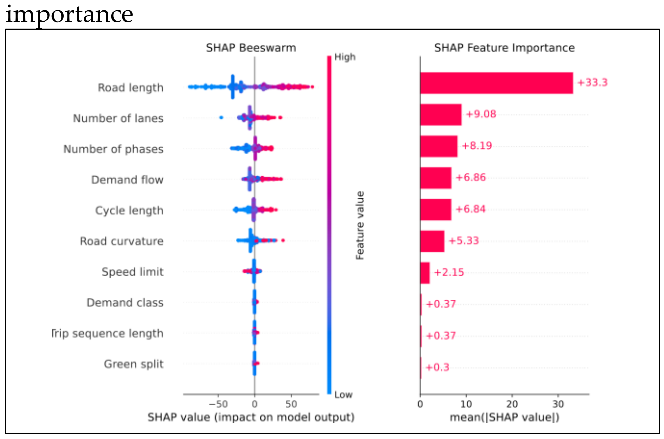


Figure 13. The Cologne SHAP bee-swarm summary plot (left) and mean absolute values (right) indicating feature importance

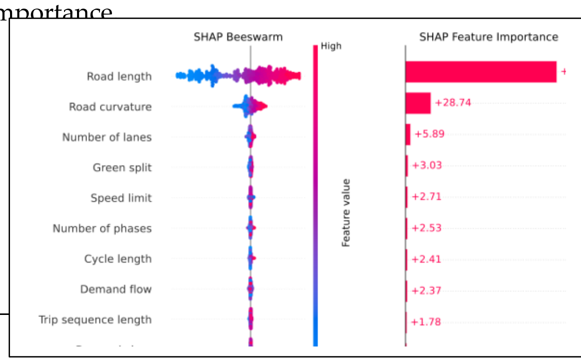


Figure 14. MoST SHAP bee-swarm summary plot (left) and mean absolute values (right) indicating feature importance.

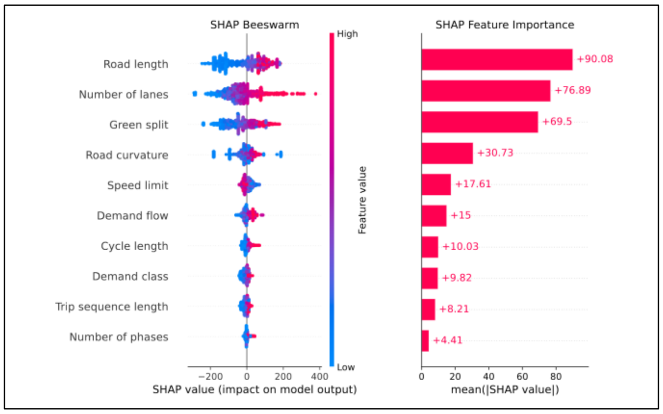


Figure 15. LuST SHAP bee-swarm summary plot (left) and mean absolute values (right) indicating feature importance.

The bar plots in Figures 12–15 present the global feature importance based on the mean absolute SHAP values across the test dataset. Larger values indicate that a feature has a stronger average influence on the model output. The SHAP bee-swarm summary plot for the surrogate model: each point represents a test trip instance. The horizontal axis shows the SHAP value, i.e., the marginal contribution of a feature to the predicted travel time relative to the model's expected output. Positive SHAP values (right side) indicate that the feature increases the predicted travel time, whereas negative values (left side) indicate a reduction in predicted travel time. Color encodes the feature magnitude for each trip (red = high feature value, blue = low feature value). Thus, red points on the positive side indicate that large values of that feature increase travel time, while red points on the negative side indicate that large values decrease travel time. The SHAP analysis is intended to explain the learned behavior of the surrogate model rather than establish causal traffic mechanisms or operational policy conclusions. The reported feature attributions therefore reflect model-specific learned associations within the evaluated datasets. The analysis reveals that the determinants of trip travel time are fundamentally network-dependent, reflecting distinct underlying traffic regimes, with suggestions for ITS design. In the Ingolstadt network, travel time is primarily governed by demand-driven dynamics, with demand flow, demand class, and trip sequence length emerging as the dominant contributors. The prominence of these variables indicates nonlinear and path-dependent delay accumulation, consistent with congestion propagation and queue spillback effects, where upstream conditions significantly influence downstream performance. Therefore, strategies in such environments may prioritize demand-responsive mechanisms and perimeter control traffic management.

In the Cologne network, a hybrid regime is observed in which travel time is jointly influenced by structural, operational, and control-related factors. While road length remains a primary determinant, the substantial contributions of lane count, number of signal phases, and cycle length indicate that both infrastructure capacity and intersection control play a critical role in shaping traffic performance. This reflects a signalized urban environment where delays are strongly mediated by intersection dynamics and capacity constraints. As a result, interventions in this context can achieve significant improvements through coordinated and adaptive signal control, phase optimization, and infrastructure-aware traffic management strategies that account for lane-level capacity and turning movements.

In contrast, the MoST network exhibits a structurally dominated regime, where travel time is explained by road length, with minimal contribution from demand and signal control variables. This suggests a system in which delays are weakly coupled to intersection operations and traffic fluctuations. Under such circumstances, management efforts might be better directed toward routing optimization, demand redistribution, and efficient path assignment.

Finally, the LuST network exhibits a structurally and control-dominated regime with broadly distributed feature influence. Road length emerges as the leading predictor, followed by the number of lanes and the green-split fraction, indicating that segment geometry, roadway capacity, and signal timing jointly govern travel time; road curvature and speed limit contribute at an intermediate level. Demand-related variables (demand flow and demand class), cycle length, trip sequence length, and the number of phases carry comparatively lower but non-negligible weight. This pattern indicates a large network in which travel time is shaped primarily by structural capacity and signal control, with demand acting as a secondary modulator rather than the dominant driver. The joint prominence of geometric, capacity, and control attributes suggests that delays accumulate through the interaction of road structure and intersection operation along long routes. From an ITS perspective, this points to interventions that combine capacity and geometry-aware management with coordinated signal-timing control, particularly green-time allocation, while treating demand-side measures such as localized routing adjustments and flow balancing as complementary levers.

4.4.4 Discussion on Cross-Network Generation

One limitation of the present study is that the surrogate models were trained and evaluated independently within each traffic network rather than through cross-network transfer learning experiments. Although cross-network generalization is an important research direction, direct transfer across heterogeneous urban traffic networks remains challenging due to substantial differences in topology, control strategies, demand patterns, and congestion dynamics. This behavior is consistent with the structural heterogeneity across the evaluated traffic systems. As summarized in Table 5, the networks differ considerably in scale, temporal coverage, and traffic dynamics. These differences imply that the relationship between local traffic states and resulting trip travel times is strongly network-dependent. Consequently, a surrogate trained on one network may learn topology-specific traffic interactions that do not directly transfer to another urban environment.

The SHAP analyses presented in Figures 12–15 further support this observation. Distinct feature importance patterns were observed across the four evaluated networks, indicating that the dominant drivers of travel time prediction vary substantially between traffic systems. For example, some networks exhibited stronger sensitivity to signal-related features, whereas others were more strongly influenced by geometric or congestion-related variables.

Therefore, the objective of the proposed framework is not to learn a universal traffic representation across arbitrary cities, but rather to construct computationally efficient, physically consistent surrogate models tailored to specific network environments. This formulation is particularly relevant for operational transportation planning, where surrogates are typically calibrated for a specific urban region under known infrastructure and control conditions. Future work may investigate domain adaptation, transfer learning, or meta-learning strategies to improve cross-network generalization across heterogeneous traffic topologies.

4.5 Monotonic Architecture Baseline Comparison

While the proposed GSPINN surrogate incorporates monotonicity through gradient-based physical consistency constraints, an important question is whether the observed behavior simply arises from generic monotonic learning mechanisms rather than from the proposed physics-regularized formulation itself. Therefore, an additional constrained learning baseline was constructed to isolate the effect of architectural monotonicity from the proposed soft regularization strategy. To achieve this, a Monotonic Graph Neural Network (Monotonic GNN) baseline was implemented and compared against both the standard unconstrained Graph Neural Network (Unconstrained ($\lambda = 0$) GNN) and the proposed GSPINN model.

4.5.1 Monotonic GNN Baseline

The Monotonic GNN baseline preserves the same graph representation and trajectory aggregation framework used by the proposed model, including the following:

- graph-based spatial message passing,
- identical graph construction,
- identical node and edge features,
- identical trajectory representations,
- and identical train/validation/test state splits.

However, unlike the proposed physics-regularized approach, monotonicity is enforced directly through architectural constraints rather than through gradient-based regularization. Specifically, the Monotonic GNN incorporates the following:

- non-negative parameterization of monotonic layers,
- monotonic activation functions,
- positive additive message aggregation,
- monotonic trajectory pooling,
- and constrained positive readout operators.

To preserve monotonic behavior more consistently, the original GRU-based sequence aggregation mechanism was replaced with monotonic mean pooling over trajectory embeddings. This modification reduces the influence of potentially non-monotonic recurrent dynamics while maintaining consistent trajectory aggregation. Also, unlike the proposed GSPINN formulation, no monotonicity loss term was used for the Monotonic GNN baseline. Instead, monotonic behavior was enforced primarily through the architectural design itself.

4.5.2 Results and Discussions

All models were evaluated using identical training, validation, and testing state partitions to ensure a fair comparison across heterogeneous traffic conditions. Performance was evaluated using : MAE, RMSE, Coefficient of Determination (R^2), Pearson Correlation, Spearman Rank Correlation, and a monotonicity violation metric. The monotonicity metric measures the degree to which the predicted travel time violates expected physically consistent relationships with respect to positively monotonic traffic features. Lower values therefore indicate fewer monotonicity violations and improved physical consistency.

Table 8 reports predictive accuracy and the monotonicity-violation score for four models on each network: the unconstrained model ($\lambda = 0$), the proposed GSPINN, a mean-pooling variant of GSPINN, and the architecture-constrained Monotonic GNN. The comparison is designed to separate two questions: whether physical consistency is better obtained through a soft gradient penalty or through hard architectural constraints, and how much of any performance difference is attributable to the constraint itself rather than to a change in the route-aggregation method.

This separation is necessary because the Monotonic GNN differs from GSPINN in two respects. First, it imposes monotonicity architecturally, through positive weights, monotone activations, and positive aggregation, and secondly, it also replaces the GRU route aggregation with mean pooling. The second change is not a separate design choice but a consequence of the first: a gated recurrent unit is not monotonicity-preserving, since its multiplicative gates can invert the sign of an input's effect, so retaining the GRU would defeat the architectural guarantee. To attribute the resulting differences correctly, the table therefore includes a mean-pooling variant of GSPINN, in which the GRU is removed and replaced by mean pooling while the soft-loss formulation is retained. This variant isolates the effect of removing sequential aggregation from that of the hard architectural constraint.

Table 8. Comparison of predictive performance and physical consistency across the unconstrained model, the proposed GSPINN, a mean-pooling variant of GSPINN that removes the GRU, and the architecture-constrained Monotonic GNN. The mean-pooling variant isolates the effect of removing sequential aggregation. Lower monotonicity-violation scores indicate stronger physical consistency, and the lowest score per network is shown in bold.

Network	Model	MAE	RMSE	R^2	Pearson	Spearman	Monotonicity
Ingolstadt	Unconstrained ($\lambda = 0$)	55.102	97.989	0.650	0.814	0.865	13.758
Ingolstadt	GSPINN	53.815	94.476	0.675	0.827	0.872	0.071
Ingolstadt	GSPINN (mean pooling)	56.552	102.013	0.621	0.799	0.859	**0.061**
Ingolstadt	Monotonic GNN	66.505	116.851	0.578	0.768	0.835	31.768
Cologne	Unconstrained ($\lambda = 0$)	21.037	29.322	0.830	0.912	0.916	11.873
Cologne	GSPINN	21.074	29.464	0.829	0.911	0.916	0.025
Cologne	GSPINN (mean pooling)	22.121	32.033	0.797	0.893	0.909	**0.010**
Cologne	Monotonic GNN	25.194	36.306	0.740	0.861	0.881	12.812
MoST	Unconstrained ($\lambda = 0$)	35.529	54.686	0.948	0.975	0.975	22.584
MoST	GSPINN	35.535	53.551	0.950	0.975	0.975	**0.090**
MoST	GSPINN (mean pooling)	42.715	60.958	0.936	0.967	0.967	0.103
MoST	Monotonic GNN	57.743	76.992	0.898	0.948	0.948	14.725
LuST	Unconstrained ($\lambda = 0$)	68.636	137.475	0.823	0.909	0.947	22.068
LuST	GSPINN	66.870	134.433	0.831	0.914	0.948	**0.032**
LuST	GSPINN (mean pooling)	74.682	144.753	0.804	0.899	0.940	0.112
LuST	Monotonic GNN	88.749	164.021	0.748	0.870	0.924	12.749

Reading the rows in order, from GSPINN to the Monotonic GNN, makes the decomposition explicit. On Ingolstadt, the coefficient of determination falls from 0.675 for GSPINN with GRU pooling to 0.621 when the GRU is replaced by mean pooling and then further to 0.578 for the fully constrained Monotonic GNN. The same ordering holds across the other networks: the accuracy lost by removing the GRU is consistently small relative to the additional accuracy lost when hard architectural constraints are imposed. The degradation of the Monotonic GNN is therefore driven mainly by the architectural constraint, not by the change of aggregator. A complementary pattern appears in the monotonicity column: both soft-penalty models, GSPINN and its mean-pooling variant, reduce violations to the same low range (for example, 0.071 and 0.061 on Ingolstadt, against 13.758 for the unconstrained model), and both remain one to three orders of magnitude more consistent than the hard-constrained Monotonic GNN. The two design choices therefore act on different properties: in these experiments the recurrent aggregator (the GRU) drives predictive accuracy, while the gradient penalty lowers violations by a similar margin under either aggregator, indicating that its effect on physical consistency is largely independent of the pooling choice. Confirming this separation would require a fuller factorial study across aggregators, penalty weights, and seeds, which is beyond the scope of this work. This also exposes a practical advantage of the soft penalty, namely that it attains near-monotone behaviour while retaining the expressive, order-aware GRU aggregator that the architectural route is forced to discard.

Therefore, the proposed GSPINN achieves the most favourable balance of predictive accuracy and physical consistency on every network. Relative to the unconstrained model, it matches or slightly improves predictive performance while reducing the monotonicity-violation score by two to three orders of magnitude. On Ingolstadt, for example, GSPINN lowers the RMSE from 97.989 to 94.476 while reducing the violation score from 13.758 to 0.071; on MoST it lowers the RMSE from 54.686 to 53.551 while reducing violations from 22.584 to 0.090 . Comparable behaviour is observed in Cologne and LuST, where predictive accuracy is essentially unchanged while physical consistency improves substantially. It should be emphasised that the soft penalty encourages monotonic behaviour and sharply reduces violations, but being a finite-weight gradient penalty, it does not guarantee pointwise monotonicity; the reported scores are residual violations rather than a guarantee of global monotonicity.

The architecture-constrained Monotonic GNN does reduce monotonicity violations relative to the unconstrained model on LuST and MoST, confirming that positive aggregation and monotone activations encourage physically consistent trends. However, it remains substantially less physically consistent than either soft-penalty model on every network, often by two to three orders of magnitude; on Cologne, for instance, it attains a violation score of 12.812, against 0.025 for GSPINN and 0.010 for the mean-pooling variant. At the same time, it incurs a substantial loss of predictive accuracy, with consistently higher MAE and RMSE and lower correlation scores, most pronounced on the large and heterogeneous LuST and MoST networks.

The Ingolstadt network illustrates the central limitation of purely architectural monotonicity. Despite its positive operators and monotone activations, the Monotonic GNN there produces the largest violation score of any model evaluated, indicating that making each component monotonic did not, in this case, make the network monotonic as a whole. A plausible explanation is the interaction between normalization layers, deep graph propagation, and trajectory aggregation: although each operator may be locally monotone, the composition of repeated propagation and normalization stages can introduce nonlocal feature interactions under heterogeneous traffic states and long-range dependencies. Similar concerns over the expressiveness and stability of constrained monotonic architecture have been raised in the monotonic-network literature (Chen et al., 2022; Runje and Shankaranarayana, 2023; Wehenkel and Louppe, 2019).

By contrast, GSPINN regularizes the gradients of the output with respect to physically meaningful inputs rather than constraining the representational capacity of the network. This steers learning toward operationally consistent behaviour while preserving the flexibility needed to model complex nonlinear traffic dynamics. Taken together, the results show that the proposed soft-penalty formulation achieves a more favourable trade-off between predictive performance and physical consistency than both the unconstrained model and the purely architecture-constrained Monotonic GNN, and that it does so while keeping the order-aware GRU aggregator, which the architectural approach cannot use.

5 Operational Case Study: Surrogate-Assisted Signal Policy Optimization

To evaluate the operational applicability of the proposed surrogate beyond offline prediction, we conducted an adaptive signal control experiment in which the surrogate model was integrated into a simulation-assisted optimization framework. While predictive metrics such as MAE and RMSE quantify regression accuracy, operational traffic control requires a surrogate that can approximate the network-level response to localized signal interventions while substantially reducing computational cost relative to microscopic simulation.

The objective of this experiment was therefore not to completely replace the microscopic simulator, but rather to investigate whether the surrogate can reliably assist policy exploration and rapid evaluation during optimization. In particular, the experiment evaluates whether a surrogate trained primarily on perturbation-based traffic states can preserve the directional and relative effects of signal interventions observed in full microscopic simulation.

The optimization experiment focused on a single signalized intersection identified by a traffic light controller. A localized intervention was applied by modifying the effective green allocation of the controlled phases while preserving the overall cycle structure. Although the intervention is local, the resulting congestion propagation affects vehicle trajectories throughout the surrounding network.

5.1 Traffic Signal Intervention Representation

A key challenge in coupling the microscopic simulator and the surrogate model is ensuring that both systems receive an equivalent representation of the signal intervention. To achieve this, we constructed an edge-level green allocation mapping directly from the simulator signal logic. For each controlled movement, the green exposure was computed as:

$$g_e = \frac{\sum_{p\in\mathcal{P}_e} d_p}{\sum_{p\in\mathcal{P}} d_p} \tag{22}$$

where:

- g_e denotes the green fraction associated with edge e,
- $\mathcal{P}$ is the set of all signal phases,
- $\mathcal{P}_e$ is the subset of phases providing green access to movement e,
- d_p is the duration of phase p.

The intervention modifies the green allocation through:

$$g'_e = g_e + \Delta g \tag{23}$$

where Δg is the optimization variable representing the green adjustment applied to the controlled intersection. The resulting green fractions were then mapped onto the graph representation used by the surrogate model. Specifically, the simulator-controlled link transitions were converted into directed edge movements:

$$(u, v) \to \{e_u, e_v\} \tag{24}$$

where:

- u denotes the incoming edge,
- v denotes the outgoing edge,
- e_u and e_v are the corresponding graph edges used in the surrogate representation.

This step establishes a direct correspondence between the microscopic signal intervention executed inside the simulator and the graph-level feature observed by the surrogate. Therefore, both the simulator and the surrogate are evaluated under aligned operational conditions.

5.2 Microscopic Simulation and Surrogate Coupling

For each optimization iteration, multiple traffic states were sampled to ensure robustness across heterogeneous operating conditions. For each sampled state and candidate signal policy:

1. the traffic demand was scaled within SUMO to generate heterogeneous operating scenarios,
2. the signal timing plan was modified using the proposed green adjustment,
3. a new edge-level green map was extracted from the updated signal logic,
4. microscopic simulation was executed,

5. vehicle travel times were collected,
6. the identical green map was injected into the surrogate graph representation,
7. surrogate-based travel time estimates were generated.

The surrogate objective is, therefore, defined as

$$\hat{\mathcal{S}}_{\theta}(s_i, \Delta g) \approx \mathcal{S}(s_i, \Delta g) \tag{25}$$

where:

- $\mathcal{S}$ is the microscopic simulator response,
- $\hat{\mathcal{S}}_{\theta}$ is the surrogate prediction,
- Δg is the green-time intervention.

5.3 Bayesian Optimization Framework

Bayesian Optimization (BO) (Agriesti et al., 2023) was employed to search the continuous signal adjustment space:

$$\Delta g \in [-0.1,\ 0.2] \tag{26}$$

At each BO iteration:

1. a candidate green adjustment was proposed,
2. multiple latent traffic states were sampled,
3. simulator and surrogate evaluations were executed under aligned conditions,
4. mean travel time was computed across sampled states,
5. and optimization statistics were recorded.

The optimization objective was defined as:

$$J(\Delta g) = \frac{1}{K}\sum_{i=1}^{K} \mathrm{TT}\,(s_i, \Delta g) \tag{27}$$

where:

- K is the number of sampled states,
- TT denotes mean travel time for vehicles traversing the controlled intersection.

Table 9 summarizes the resulting optimization metrics between the simulator and the surrogate.

Table 9. Performance of the proposed surrogate for signal policy optimization across the LuST and Ingolstadt networks.

Metric	LuST	Ingolstadt
MAE	77.44	59.86
RMSE	81.65	60.75
MAPE (%)	20.24	7.43
Pearson Correlation (r)	0.8879	0.8071
Spearman Correlation (ρ)	0.9408	0.7192
Directional Agreement	0.8205	0.7949
Opt-time average speedup	1793 ×	23412 ×

The correlation-based metrics demonstrate that the surrogate preserves the relative ordering and directional response of traffic states under signal interventions across both the LuST and Ingolstadt networks. Particularly, the Spearman correlations ($\rho = 0.9408$ for LuST and $\rho = 0.7192$ for Ingolstadt) indicate that the surrogate captures monotonic operational trends across candidate signal policies, even under heterogeneous network structures and demand conditions. Similarly, the directional agreement values of 82.05% for LuST and 79.49% for Ingolstadt show that the surrogate consistently predicts whether a signal intervention improves or worsens traffic conditions relative to the microscopic simulator. This property is especially important for Bayesian optimization, where preserving the relative ordering between candidate policies is often more critical than achieving exact numerical equivalence. Although absolute prediction errors remain non-negligible, the relatively low MAPE values, particularly for Ingolstadt (7.43%), indicate that the surrogate maintains useful operational accuracy during control evaluation. The larger absolute errors observed in LuST are expected given the substantially larger network scale and higher trajectory diversity within the scenario. More importantly, the consistently strong correlation and directional metrics across both datasets suggest that the surrogate remains effective for identifying beneficial control policies.

Computational speedups further demonstrate the operational value of the proposed framework. During signal optimization, the surrogate achieved average evaluation accelerations of approximately 1793 × on LuST and 23412 × on Ingolstadt relative to repeated microscopic simulator rollouts. The computational speedup is computed as the ratio between the total runtime of the simulator and the end-to-end surrogate evaluation pipeline. The surrogate runtime includes all components required for a single policy evaluation, namely the construction of the intervened traffic graph, trip-level feature extraction, and forward inference through the graph neural surrogate model. Formally, the speedup is defined as:

$$\text{Speedup} = \frac{T_{\text{simulator}}}{T_{\text{graph}} + T_{\text{inference}}} \tag{28}$$

where T_{graph} denotes the time required to construct the intervention-aware graph representation under a given policy, and $T_{\text{inference}}$ corresponds to batched forward inference and aggregation over trip trajectories.

These values are substantially larger than the end-to-end surrogate inference speedups reported for the main travel-time prediction task because the optimization framework evaluates only localized policy-sensitive trajectories under inference, whereas the simulator must still execute full microscopic simulation dynamics for each candidate policy. Therefore, the reported speedups should be interpreted as optimization-time policy evaluation accelerations rather than strict simulator factors.

5.4 Operational Relevance

The operational relevance analysis compares the surrogate predictions against microscopic simulation during optimization. Unlike the original training procedure, which was constructed primarily from perturbation-generated route data, the optimization experiment evaluates the surrogate against dynamically simulated travel times produced directly by the simulator under active signal interventions. As a result, the experiment represents a substantially more difficult generalization setting. The surrogate must not only preserve static perturbation relationships learned during training but also approximate the emergent traffic dynamics generated during online simulation.

Furthermore, surrogate predictions can improve policy evaluation during optimization. Since Bayesian optimization relies on repeated comparative assessment of candidate signal policies, accurate surrogate estimates can improve policy ranking consistency, reduce the likelihood of selecting suboptimal interventions, and accelerate convergence toward effective signal control strategies. Within the simulation-based optimization framework considered in this work, this enables faster signal tuning and more computationally efficient exploration of large-scale traffic management scenarios. Importantly, the proposed framework is not intended to replace microscopic simulation. Instead, the surrogate serves as a fast, physically consistent policy-screening mechanism that identifies plausible near-optimal signal control strategies across a large search space. The microscopic simulator can then be selectively executed only on a small subset of promising surrogate-proposed candidate policies for final validation and refinement. Consequently, the framework substantially reduces the number of expensive simulator evaluations required during optimization while preserving the reliability of simulation-based policy evaluation.

6 Conclusion

The computational bottleneck inherent in microscopic traffic simulation has long hindered the real-time operation of dynamic traffic management systems and the scalable evaluation of complex urban mobility scenarios. While deep learning surrogates offer a promising pathway to circumvent these computational constraints, their widespread adoption has been hindered by their inherent black-box nature, their heavy reliance on inaccessible microscopic state data, and their propensity to generate physically implausible predictions under novel, out-of-distribution traffic conditions. This research addresses these challenges by proposing GSPINN, a Graph Sequential Physics-Informed Neural Network that emulates the macroscopic input-output behavior of simulators using exclusively field-observable network attributes. By unifying GraphSAGE-based spatial message passing with GRU-driven sequential route aggregation, the proposed architecture captures both the topological propagation of urban congestion and the cumulative, path-dependent delay experienced along distinct trip trajectories.

The most significant contribution of this work is the integration of a directional monotonicity constraint: by penalizing negative input-output gradients for physically monotonic variables, such as road length, signal cycle length, and road curvature (computed from the change in direction between consecutive segments), the model is encouraged toward, though not strictly guaranteed to satisfy, macroscopic traffic-consistent trends, bridging the gap between purely data-driven flexibility and physically plausible behaviour. This approach bypasses the mathematical difficulty of mapping continuous partial differential equations onto highly discrete, discontinuous graph structures.

Empirical validation across four road networks indicated the advantages of this hybrid approach. GSPINN achieved marginally improved predictive accuracy compared to state-of-the-art spatial, attention-based, and graph baselines, while substantially reducing physically implausible feature-response relationships. A controlled comparison against an architecturally constrained monotonic baseline further indicated that the soft-penalty formulation attains a more favourable accuracy-consistency trade-off while retaining the sequential aggregation that the architectural alternative cannot use. More importantly, the surrogate yielded a substantial acceleration in computational runtime relative to the microscopic simulator, and SHAP-based interpretability analyses confirmed that its internal logic aligns with expected drivers of travel-time predictions. By enabling near-instantaneous travel-time estimates, GSPINN has the potential to support offline what-if analyses, such as screening candidate signal-timing changes and capacity-reduction policies prior to simulator validation, rather than replacing the simulator itself.

The present study, however, evaluates the framework exclusively on simulator-generated datasets derived from controlled traffic environments. While this enables reproducible experimentation, controlled perturbation analysis, and evaluation across multiple traffic states, it does not establish direct real-world deployability. Future work will investigate transfer and adaptation of the framework using real-world traffic measurements, including loop-detector and sensor data, alongside domain adaptation, online fine-tuning, and uncertainty-aware deployment under evolving traffic conditions.

Author contributions

B.I.Afolayan: Writing – original draft, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **A.Ghosh:** Conceptualization, Methodology, Writing – Final draft preparation, review & editing, Supervision,. **S.Narayanan**: Supervision, Methodology, Writing – review. **C.Antoniou**: Supervision. **A.D.Masegosa**: Supervision, Funding Acquisition. **J.F.Calderin**: Supervision.

Replication and data sharing

The source codes and replication package are available on ETS-data at https://ets-data.sciopen.com/ETSD.2026.9190033.

Acknowledgements

This work was supported by the Spanish Ministry of Science and Innovation through the REsearch on New Artificial Intelligence techniques to Improve Sustainability, SAfety and resilieNCE of mobility (RENAISSANCE) project under Grant PID2022-140612OB-I00.

Declaration of Competing Interest

The authors have no competing interests to declare that are relevant to the content of this article

References

Agriesti, S., Kuzmanovski, V., Hollmén, J., Roncoli, C., Nahmias-Biran, B.-H., 2023. A Bayesian optimization approach for calibrating large-scale activity-based transport models. IEEE Open J. Intell. Transp. Syst. 4, 740–754.

Ault, J., Sharon, G., 2021. Reinforcement learning benchmarks for traffic signal control, in: Thirty-Fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1).

Belhadi, A., Djenouri, Y., Djenouri, D., Lin, J.C.-W., 2020. A recurrent neural network for urban long-term traffic flow forecasting. Appl. Intell. 50, 3252–3265. https://doi.org/10.1007/s10489-020-01716-1

Brody, S., Alon, U., Yahav, E., 2021. How attentive are graph attention networks? ArXiv Prepr. ArXiv210514491.

Chen, Y., Tang, X., Qi, X., Li, C.-G., Xiao, R., 2022. Learning graph normalization for graph neural networks. Neurocomputing 493, 613–625.

Codeca, L., Frank, R., Engel, T., 2015. Luxembourg sumo traffic (lust) scenario: 24 hours of mobility for vehicular networking research, in: 2015 Ieee Vehicular Networking Conference (Vnc). IEEE, pp. 1–8.

Codeca, L., Härri, J., 2018. Monaco SUMO traffic (MoST) scenario: A 3D mobility scenario for cooperative ITS. EPiC Ser. Eng. 2, 43–55.

Cuomo, S., Di Cola, V.S., Giampaolo, F., Rozza, G., Raissi, M., Piccialli, F., 2022. Scientific machine learning through physics–informed neural networks: Where we are and what's next. J. Sci. Comput. 92, 88.

Fang, Z., Long, Q., Song, G., Xie, K., 2021. Spatial-temporal graph ode networks for traffic flow forecasting, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. pp. 364–373.

Hoerl, A.E., Kennard, R.W., 1970. Ridge regression: Biased estimation for nonorthogonal problems. Technometrics 12, 55–67.

Huber, P.J., 1964. Robust Estimation of a Location Parameter. Ann. Math. Stat. 35, 73–101. https://doi.org/10.1214/aoms/1177703732

Jiang, W., Luo, J., 2022a. Graph neural network for traffic forecasting: A survey. Expert Syst. Appl. 207, 117921.

Jiang, W., Luo, J., 2022b. Graph neural network for traffic forecasting: A survey. Expert Syst Appl 207. https://doi.org/10.1016/j.eswa.2022.117921

Kipf, T.N., Welling, M., 2016. Semi-supervised classification with graph convolutional networks. ArXiv Prepr. ArXiv160902907.

Krajzewicz, D., Hertkorn, G., Rössel, C., Wagner, P., 2002. SUMO (Simulation of Urban MObility)–an open-source traffic simulation, in: Proceedings of the 4th Middle East Symposium on Simulation and Modelling (MESM2002). pp. 183–187.

Lighthill, M.J., Whitham, G.B., 1955. On kinematic waves II. A theory of traffic flow on long crowded roads. Proc. R. Soc. Lond. Ser. Math. Phys. Sci. 229, 317–345.

Lu, L., Meng, X., Mao, Z., Karniadakis, G.E., 2021. DeepXDE: A deep learning library for solving differential equations. SIAM Rev. 63, 208–228.

Lundberg, S.M., Lee, S.-I., 2017. A unified approach to interpreting model predictions. Adv. Neural Inf. Process. Syst. 30.

Natterer, E., Engelhardt, R., Hörl, S., Bogenberger, K., 2025. Machine Learning Surrogates for Optimizing Transportation Policies with Agent-Based Models. ArXiv Prepr. ArXiv250111057.

Patil, M., Ahmed, Q., Midlam-Mohler, S., 2025. Travel Time and Weather-Aware Traffic Forecasting in a Conformal Graph Neural Network Framework. IEEE Trans. Intell. Transp. Syst.

Raissi, M., Perdikaris, P., Karniadakis, G.E., 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J. Comput. Phys. 378, 686–707.

Raissi, M., Perdikaris, P., Karniadakis, G.E., 2017. Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. ArXiv Prepr. ArXiv171110561.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning representations by back-propagating errors. nature 323, 533–536.

Runje, D., Shankaranarayana, S.M., 2023. Constrained Monotonic Neural Networks, in: Proceedings of the 40th International Conference on Machine Learning, Proceedings of Machine Learning Research. PMLR, pp. 29338–29353.

Shi, Y., Huang, Z., Feng, S., Zhong, H., Wang, W., Sun, Y., 2020. Masked label prediction: Unified message passing model for semi-supervised classification. ArXiv Prepr. ArXiv200903509.

Sun, B., Cheng, W., Goswami, P., Bai, G., 2018. Short-term traffic forecasting using self-adjusting k-nearest neighbours. IET Intell. Transp. Syst. 12, 41–48. https://doi.org/10.1049/iet-its.2016.0263

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y., 2018. Graph Attention Networks. https://doi.org/10.48550/arXiv.1710.10903

Wehenkel, A., Louppe, G., 2019. Unconstrained monotonic neural networks. Adv. Neural Inf. Process. Syst. 32.

Wu, C.-H., Ho, J.-M., Lee, D.T., 2004a. Travel-time prediction with support vector regression. IEEE Trans. Intell. Transp. Syst. 5, 276–281. https://doi.org/10.1109/TITS.2004.837813

Wu, C.-H., Ho, J.-M., Lee, D.-T., 2004b. Travel-time prediction with support vector regression. IEEE Trans. Intell. Transp. Syst. 5, 276–281.

You, S., Ding, D., Canini, K., Pfeifer, J., Gupta, M., 2017. Deep lattice networks and partial monotonic functions. Adv. Neural Inf. Process. Syst. 30.

Yu, H., Wu, Z., Wang, S., Wang, Y., Ma, X., 2017. Spatiotemporal Recurrent Convolutional Networks for Traffic Prediction in Transportation Networks. Sensors 17, 1501. https://doi.org/10.3390/s17071501

Author biography

Blessing Itoro Afolayan received her master's degree in mathematical sciences from AIMS, Cameroon, and the African Masters in Machine Intelligence (AMMI) degree, specialized in machine learning from Ghana. She is a Researcher with a strong foundation in applied mathematics. Her academic journey was driven by a passion for solving real-world problems, exemplified by her work as an Intern in South Korea, where she designed a machine learning-based sensor clustering system for farmers. In the USA, she further honed her skills as an Intern with Carnegie Mellon University, focusing on image and text data analysis. Currently, she applies her expertise in AI with DeustoTech, Spain, where she is focusing on graph machine learning, probabilistic modeling, explainable AI and AI-driven decision systems applied to large-scale spatial networks model optimization.

Arka Ghosh received the Ph.D. degree in computer science and technology from the Indian Institute of Engineering Science and Technology, Shibpur, in 2021. He is an Associate Researcher with DeustoTech, University of Deusto, Bilbao, Spain. He has authored several high-impact publications in premier journals. He has contributed to several EU Horizon projects, such as TANGENT and CONDUCTOR. His research focuses on smart mobility, multi-modal transport optimization, differential evolution, black-box optimization, and evolutionary adversarial learning. He serves as a reviewer for leading journals in his field.

Santhanakrishnan Narayanan is a postdoctoral researcher at the Technical University of Munich (TUM), Germany. He earned his Bachelor's degree in Civil Engineering from SASTRA University in 2014 and his M.Sc. and Dr.-Ing. in Transportation Systems from TUM in 2019 and 2023, respectively. Before his Master's, he worked at Larsen & Toubro Construction, contributing to the Chennai Metro project. He is a recipient of best paper award from TRB Urban Freight Committee in 2019. His expertise lies in travel behaviour analysis and modelling emerging mobility services, integrating statistical and data-driven methodologies.

Prof. Constantinos Antoniou holds a degree in civil engineering from the National Technical University of Athens in Greece (1995) as well as a master's degree in transportation (1997) and a PhD in transportation systems (2004) from MIT (USA). He has authored more than 500 scientific publications, including more than 210 papers in international, peer-reviewed journals, 270 published in international conference proceedings, three books and 30 book chapters. He is a member of several scientific organizations and committees, including the Transport Research Board and the Road and Transportation Research Association (FGSV). Professor Antoniou is Co-Editor-in-Chief of the journal Transportation Research Part A: Policy and Practice, and sits on the editorial boards of various international journals such as Transportation Research - Part C: Emerging Technologies, Accident Analysis and Prevention and Journal of ITS.

Antonio D. Masegosa received the university degree in computer engineering and the Ph.D. degree in computer sciences from the University of Granada, Spain, in 2005 and 2010, respectively. From June 2010 to November 2014, he was a Postdoctoral Researcher with the Research Center for ICT, University of Granada. In 2014, he received an IKERBASQUE Research Fellowship to work in the Mobility Unit, Deusto Institute of Technology, in Bilbao, Spain. In 2019, he received the IKERBASQUE Research Associate Mention and currently coordinates the Artificial Intelligence area with DeustoTech. His main research interests include artificial intelligence, intelligent systems, soft computing, hybrid metaheuristics, machine learning, deep learning, intelligent transportation systems, logistic networks, travel behaviour analysis, and traffic forecasting.

Jenny Fajardo Calderin received the university degree in computer engineering and the master's degree in applied informatics from the University of CUJAE (Universidad Tecnologica de la Habana), Cuba, in 2008 and 2010, respectively, and the Ph.D. degree in information and communication technologies in Granada, in 2015. Since then, she has worked in both teaching and research. She has more than ten years of teaching experience in both undergraduate and graduate lessons with CUJAE. Since 2019, she has been as a Researcher with Deusto Institute of Technology, Bilbao, Spain. In research, she has focused on the field of artificial intelligence, the main research topics are focused on intelligent systems based on soft computing techniques for decision and optimization problems. In particular, her main research interests include artificial intelligence, intelligent systems, soft computing, hybrid metaheuristics, machine learning, and intelligent transportation systems.