Uncertainty Quantification for Physics-Informed Traffic Graph Networks

Tianshu Bao
tianshu.bao@vanderbilt.edu
Vanderbilt University
Nashville, USA

Xiaoou Liu
xiaoouliu99@gmail.com
Arizona State University
Tempe, USA

Meiyi Ma
meiyi.ma@vanderbilt.edu
Vanderbilt University
Nashville, USA

Taylor T Johnson
taylor.johnson@vanderbilt.edu
Vanderbilt University
Nashville, USA

Hua Wei
hua.wei@asu.edu
Arizona State University
Tempe, USA

Abstract

Predicting spatiotemporal patterns is essential for traffic flow forecasting in Intelligent Transportation Systems (ITS), where accurate predictions can greatly enhance traffic management. Currently, data-driven approaches, such as Graph Neural Networks (GNNs) combined with physics-informed partial differential equations (PDEs), have shown promising performance in capturing complex traffic dynamics. However, these models sometimes make unexpected and incorrect predictions with high certainty which will mislead real-world decision-making, particularly in critical scenarios. This lack of uncertainty quantification (UQ), which estimates the confidence of neural network predictions beyond prediction accuracy, limits the reliability of the predictions. Existing UQ methods in traffic forecasting are typically applied to purely data-driven models, leaving the effects of physics-informed approaches on UQ largely unexplored. In this paper, we address these challenges by combining multiple UQ baselines with physics-informed methods to investigate the impact of physical constraints on UQ for traffic forecasting. Additionally, we introduce a new physics-informed loss function to guide the model learning process, which holds an exponential relation and complements the linear relations of the PDE layer. Through extensive experiments on real-world traffic datasets, we demonstrate that our proposed method outperforms existing approaches, achieving reductions of up to 14.8% in short-term and 8.7% in long-term traffic speed prediction errors. Sensitivity analysis further illustrates the robustness of our approach under perturbations.

CCS Concepts

• **Computing methodologies -> Modeling and simulation; Computer systems organization -> Embedded and cyber-physical systems**;

Keywords

Uncertainty Quantification, Graph Neural Network, Cyber-Physical System

ICCPS '25, May 6–9, 2025, Irvine, CA, USA

ACM ISBN 979-8-4007-1498-6/2025/05
https://doi.org/10.1145/3716550.3722023

ACM Reference Format:
Tianshu Bao, Xiaoou Liu, Meiyi Ma, Taylor T Johnson, and Hua Wei. 2025. Uncertainty Quantification for Physics-Informed Traffic Graph Networks . In *ACM/IEEE 16th International Conference on Cyber-Physical Systems (with CPS-IoT Week 2025) (ICCPS '25), May 6–9, 2025, Irvine, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3716550.3722023

1 Introduction

Traffic flow forecasting plays a critical role in Intelligent Transportation Systems (ITS) by predicting future traffic conditions based on historical data and the spatial structure of road networks [14, 24]. Accurate traffic forecasting enables effective traffic management, route planning, and resource allocation, all of which are essential for mitigating congestion and improving road safety.

Recent advances in data-driven models like Graph Neural Networks (GNNs) are highly effective for processing graph-structured traffic data [5, 18, 27]. Spatiotemporal methods like Diffusion Convolutional Recurrent Neural Networks (DCRNN) [18] and Spatiotemporal Graph Convolutional Networks (STGCN) [27] have demonstrated impressive performance in capturing traffic dynamics. In parallel, partial differential equations (PDEs) are widely used for modeling the spatiotemporal behavior of traffic networks, providing a strong foundation for understanding the flow of traffic [8, 29]. Recent efforts to combine physics-based PDEs with machine learning models have shown promising results in improving the accuracy of traffic forecasting [6].

However, despite these promising advancements, current models still face significant challenges when deployed in real-world traffic systems. These models often produce overconfident predictions, which limits their practical applications in dynamic and uncertain environments. This is especially problematic when dealing with complex, real-world traffic patterns, where the absence of explicit uncertainty measures can hinder decision-making processes.

Uncertainty Quantification (UQ) techniques can help address these limitations by providing confidence intervals for model predictions, allowing a better understanding of the model's reliability and robustness [1]. However, current UQ methods for spatiotemporal traffic forecasting tend to overlook the underlying physics of traffic flow and treat neural networks as purely data-driven models, without integrating any domain-specific knowledge. This creates a gap in the ability to provide reliable and physically consistent predictions in traffic forecasting tasks.

To bridge this gap, we propose a novel approach that integrates UQ techniques within a physics-informed traffic GNN model. By combining the power of GNNs, physics-based constraints, and uncertainty estimation, our method enhances both the accuracy and reliability of traffic flow forecasting, particularly in noisy, real-world traffic environments. Specifically, we introduce a physics-informed loss function that guides the model's learning process and combines it with different UQ baselines. This novel combination not only improves the accuracy of predictions but also quantifies the uncertainty in these predictions, providing a more reliable measure of model confidence.

We validate our approach through extensive experiments on two real-world traffic datasets. Our experimental results show that our model significantly improves both the accuracy and reliability of traffic flow forecasting. Specifically, we demonstrate that our approach reduces prediction errors and provides more reliable uncertainty intervals, making it a more robust solution than existing models. Additionally, our method shows superior performance across various traffic conditions, including both regular and rush-hour traffic. A general illustration of the proposed method is in Fig. 1.

In summary, the main contributions of this work are as follows:

- We integrate physics-informed machine learning (PIML) with UQ methods, enhancing the performance of existing traffic GNN models by incorporating both domain knowledge and uncertainty estimation.
- We propose a novel physics-informed loss function that improves the performance of baseline models while ensuring faster convergence during training.
- We conduct a comprehensive set of experiments, including ablation studies and sensitivity analysis, to validate the superiority of our approach with both lower predictive error and predicted intervals under short-term and long-term traffic forecasting tasks.

2 Related Work

2.1 Traffic Forecasting using GNN and PIML

Traffic flow forecasting consists of temporal and spatial state predictions. To capture temporal dependencies, Recurrent Neural Networks (RNNs) have been widely used, effectively modeling sequential data [28]. For the complex spatial dependencies within traffic networks, traditional methods often apply multivariate time series models [11, 26], where traffic states collected from different locations or regions are combined to extract linear relationships. Some approaches also segment the spatial area into regular grids, using Convolutional Neural Networks (CNNs) to extract spatial features [24, 25, 28, 30]. However, these methods fall short of capturing the topological information inherent to road networks.

The adoption of GNNs, known for their strong capability to process graph-structured data, has been proposed to more effectively learn patterns in traffic networks [16, 19, 31]. Models such as DCRNN [18], STGCN [27], and GraphWaveNet [23] have shown remarkable results in capturing traffic dynamics by using the topological structure of road networks alongside temporal information. However, these models remain purely data-driven and lack the incorporation of physical constraints from real-world scenarios, limiting their interpretability under complex, high-stakes conditions.

To overcome some limitations of data-driven models, physics-based approaches have emerged, combining physical principles with machine learning models to improve predictive performance and enhance generalizability, particularly in scientific and engineering domains [4, 9, 20]. For instance, [6] integrates PDEs with neural networks to better capture the spatiotemporal dynamics of traffic flow. This physics-informed approach adds valuable structure to the model, allowing it to respect real-world traffic dynamics and avoid overfitting. While these physics-informed models have successfully improved prediction accuracy, they still lack mechanisms to quantify uncertainty, a critical feature for interpretability and reliability in traffic management applications.

2.2 Uncertainty Quantification

UQ is a process for measuring and analyzing the uncertainty in model predictions, especially prevalent in data-driven models. Typically, uncertainty in the models is categorized into two types [1]: aleatoric uncertainty, which arises from inherent noise in the data and cannot be reduced by adding more data; and epistemic uncertainty, which stems from the model's limited knowledge or insufficient data and can potentially be reduced with more data or a more complex model. For example, Quantile Regression [17] estimates prediction intervals by modeling different quantiles, while the Mean Interval Score (MIS) Regression [12] method incorporates confidence intervals into the loss function to optimize both prediction accuracy and interval width [22].

However, both methods often require extensive resampling and model retraining to generate interval predictions, which can be computationally costly. The Evidential Deep Learning (EDL) [2] method introduces evidence terms into the loss function, reducing the need for resampling and improving computational efficiency.

In this paper, we use various UQ methods as baselines to assess the improvements in confidence intervals and prediction accuracy achieved by our proposed method. Further details on UQ methods are provided in Section 4, showcasing different UQ methods as well as our proposed approach.

3 Preliminaries

This section covers the definition of traffic spatial-temporal forecasting and the Graph PDE Network.

3.1 Problem Definition

In traffic flow forecasting, a road network can naturally be modeled as a graph, where each node and edge represent a road segment and its connections, respectively. This graph representation allows us to capture spatial dependencies and the topological relationships between road segments. We represent the traffic network as a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{A}\}$, where:

- $\mathcal{V}$ denotes the set of N nodes, each representing a road segment.
- $\mathcal{E}$ represents the edges, indicating connections between road segments.
- $\mathbf{A} \in \mathbb{R}^{N \times N}$ is the adjacency matrix, encoding the spatial relationships among nodes.

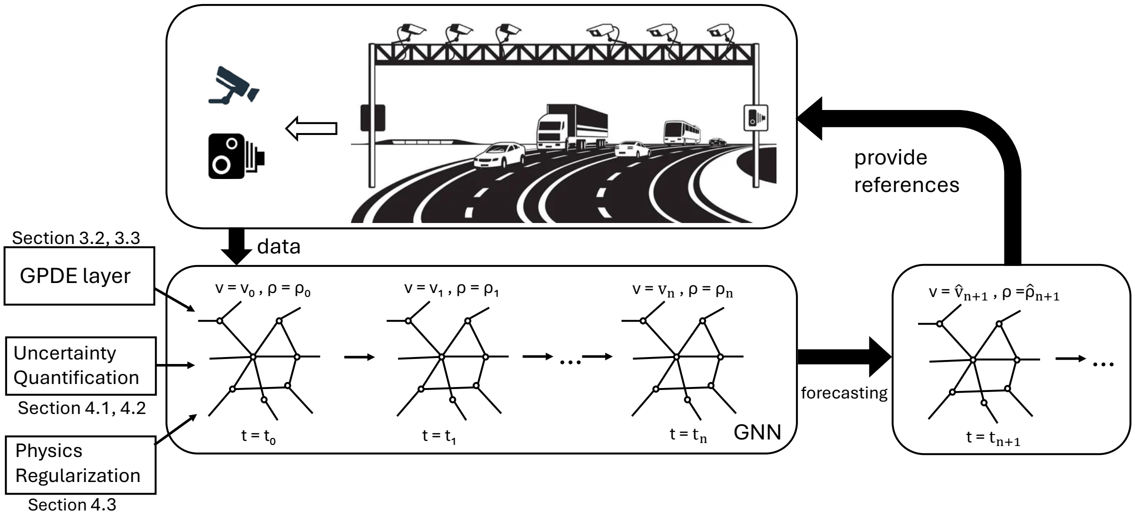


Figure 1: Diagram for the Framework.

At time t, each node i has an observation $\boldsymbol{x}_i^t \in \mathbb{R}^F$, where F is the feature dimension. The full observation at time t is represented as $X^t \in \mathbb{R}^{N\times F}$, and the sequence of observations over time as $\mathcal{X} = (X^1, X^2, ..., X^T) \in \mathbb{R}^{T\times N\times F}$. The forecasting problem is to use historical data T to predict future traffic observations over a time horizon T', formally defined as:

$$f : (X^{t-T+1}, X^{t-T+2}, ..., X^t; \mathcal{G}) \rightarrow (X^{t+1}, X^{t+2}, ..., X^{t+T'}). \quad (1)$$

This is the traditional definition of traffic forecasting, where the output prediction is a deterministic value. However, to measure the uncertainty in forecasts, we extend this definition to include interval outputs. The next section provides a detailed explanation of our approach.

3.2 Traffic Graph PDE Network

3.2.1 DCRNN model. The Diffusion Convolutional Recurrent Neural Network (DCRNN) [18] combines diffusion convolution with RNN mechanisms to capture both spatial and temporal dynamics in traffic flow. In DCRNN, traffic flow is modeled as a diffusion process over a network graph, where diffusion convolution helps capture spatial dependencies between nodes, while RNN structures capture temporal patterns over time. The diffusion convolution operation at each layer l is defined as:

$$\mathbf{h}^{(l+1)} = \sigma\left(\sum_{k=0}^{K-1} \mathbf{W}_k^{(l)} (\mathbf{D}^{-1}\mathbf{A})^k \mathbf{h}^{(l)} + \mathbf{b}^{(l)}\right), \quad (2)$$

where $\mathbf{h}^{(l)}$ is the hidden state matrix at layer l, $\mathbf{A}$ is the adjacency matrix, $\mathbf{D}$ is the degree matrix, $\mathbf{W}_k^{(l)}$ is the weight matrix, $\mathbf{b}^{(l)}$ is the bias term, σ represents the activation function, and K is the maximum diffusion step.

3.2.2 Physics-Informed Graph PDE Layer. To enhance DCRNN with physical constraints, [6] incorporates a Physics-Informed PDE layer based on the Aw-Rascle-Zhang (ARZ) model. The ARZ model is commonly used in traffic flow theory to represent non-linear traffic dynamics, accounting for vehicle conservation and momentum. Here, we use the simplified ARZ model to focus on predicting traffic speed by assuming constant traffic density over short intervals:

$$\frac{\partial v}{\partial t} + \frac{\partial f(v)}{\partial s} = g, \quad (3)$$

where $f(v) = \frac{v^2}{2} - v_f \frac{\rho}{\rho_{max}} v$ represents the traffic flow dynamics, g is an external influence term, $v \in \boldsymbol{x}$ is the traffic speed, $\rho \in \boldsymbol{x}$ is the traffic density, and ρ_{max} is the maximum allowable density. For detailed discussions, please refer to [6].

3.2.3 Integrating GPDE with DCRNN. [6] integrates the GPDE layer into the DCRNN's sequence-to-sequence framework by placing it before the recurrent layers in both the encoder and decoder components, as shown in Fig. 2. This setup allows the DCRNN model to incorporate physically grounded traffic flow patterns from the GPDE layer. The hidden states $\mathbf{h}_{\text{enc/dec}}^{t+1}$ within the encoder and decoder are updated as follows:

$$\mathbf{h}_{\text{enc/dec}}^{t+1} = \text{DCGRU}_{\text{enc/dec}}(\text{GPDE}(\mathbf{h}^t, \rho^t, \rho_{max}, v_f, \mathbf{A}, g)), \quad (4)$$

The current hidden states now incorporate input information processed through the GPDE layer.

4 Physics-Informed Uncertainty Quantification

So far, we've covered how the physics-informed GPDE layer helps guide the DCRNN model to better align with real-world traffic dynamics. In this section, we introduce our proposed method for integrating UQ along with a new physics-informed component.

4.1 Overall Framework

In practical applications of traffic flow forecasting, UQ is crucial for making reliable decisions. For example, in traffic light control and

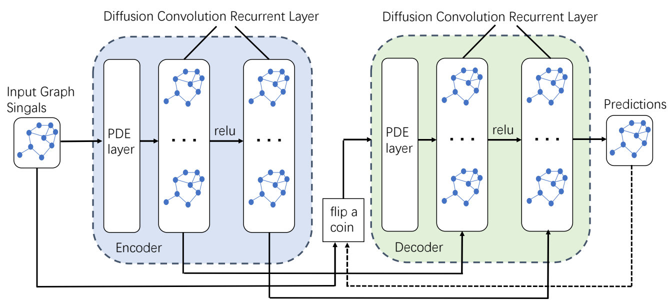


Figure 2: Diagram of DCRNN model with GPDE layer.

congestion management, understanding the confidence of predictions helps in better planning and response. While physics-informed methods have demonstrated strong accuracy in forecasting tasks, their impact on UQ remains underexplored.

In our framework, we use the GPDE layer to generate physics-guided predictions, which are then combined with various UQ baselines to estimate confidence intervals. Additionally, the physics-informed loss function guides the model toward solutions that satisfy fundamental traffic flow principles. Illustrated in Fig. 1, these components enable the model to produce more reliable and robust predictions, with each forecast accompanied by a confidence interval.

4.2 Uncertainty Quantification Methods

We start by describing the UQ methods we use in this work. The general idea behind UQ methods is to either capture variations in the dataset or incorporate evidential distributions in the output to calculate the uncertainty.

4.2.1 Quantile Regression. The Quantile Regression [17] aims to predict different quantiles of the target distribution rather than a single point estimate prediction. For instance, predicting the 5th and 95th quantiles provides a 90% prediction interval, indicating that the true value is likely to fall within this range with 90% confidence. The confidence level is 90%.

We use the one-sided quantile loss function with the output from the GPDE network to generate predictions at a fixed confidence level α. Given an input x, the ground truth y and the output $f_{\text{GPDE}}(x)$ from a GPDE network, parameterized by θ, the quantile loss is defined as follows:

$$\begin{aligned} &L_{\text{Quantile}}(y, f_{\text{GPDE}}(x); \theta, \alpha) = \\ &\min_{\theta}\{\mathbb{E}\left[(y - f_{\text{GPDE}}(x))\left(\alpha - \mathbb{I}\{y < f_{\text{GPDE}}(x)\}\right)\right]\}, \end{aligned} \tag{5}$$

where $\mathbb{I}$ is an indicator function and $\mathbb{E}$ is the expectation.

By minimizing this quantile loss function for different quantile levels, the model generates prediction intervals that capture uncertainty with the physics-informed structure of the traffic flow model.

4.2.2 MIS Regression. The Mean Interval Score (MIS) [3] is a robust metric for evaluating the quality of uncertainty intervals. MIS evaluates both the accuracy of predicted intervals and their coverage probability by applying penalties to intervals that are either too narrow, which underestimates the uncertainty, or too wide, which overestimates it.

For large samples, MIS is defined by expectations in Eq. 6, where α denotes confidence level and the u and l are the upper and lower boundary of the confidence interval, respectively. The score function contains three penalty terms. The first penalty term is the gap between the two bounds; The second term is for the situation that predicted values higher than the upper bound; The third term is for predictions under the confidence lower bound. These three are mainly estimations of uncertainty.

$$MIS_{\infty}(u, l; \alpha) = (u - l) + \frac{2}{\alpha}(\mathbb{E}[y - u | y > u] + \mathbb{E}[l - y | y < l]) \tag{6}$$

MIS regression [12] directly minimizes MIS to obtain estimates of the fixed confidence intervals. Specifically, the MIS regression loss in the GPDE network is defined as follows:

$$\begin{aligned} &L_{MIS}(y, u_{\text{GPDE}}(x), l_{\text{GPDE}}(x), f_{\text{GPDE}}(x); \theta, \alpha) = \\ &\min_{\theta}\{\mathbb{E}[(u_{\text{GPDE}}(x) - l_{\text{GPDE}}(x)) + \frac{2}{\alpha}(y - u_{\text{GPDE}}(x))\mathbb{I}\{y > u_{\text{GPDE}}(x)\} \\ &+ \frac{2}{\alpha}(l_{\text{GPDE}}(x) - y)\mathbb{I}\{y < l_{\text{GPDE}}(x)\} + |y - f_{\text{GPDE}}(x)|]\} \end{aligned} \tag{7}$$

4.2.3 Evidential Deep Learning Network. The EDL network [2] considers the target value y, a sample drawn from a normal distribution but does not know the parameters of the normal distribution. Two parameters, μ and σ^2 are drawn from a higher-order evidential distribution, specifically a normal-inverse-gamma (NIG) distribution, which serves as the conjugate prior to the normal distribution. EDL network uses the negative log-likelihood (NLL) loss function to optimize both the prediction accuracy and the uncertainty estimation of the model. The NLL loss function is defined as follows:

$$\begin{aligned} L_{\text{NLL}}(y, m) = &\frac{1}{2}\log\left(\frac{\pi}{\nu}\right) - \zeta\log(\Omega) + \left(\zeta + \frac{1}{2}\right)\log\left((y - \gamma)^2\nu + \Omega\right) \\ &+ \log\left(\frac{\Gamma(\zeta)}{\Gamma\left(\zeta + \frac{1}{2}\right)}\right), \end{aligned} \tag{8}$$

where $\Omega = 2\beta(1 + \nu)$, and m = $(\gamma, \nu, \zeta, \beta)$ are the parameters of the NIG distribution.

In addition to the NLL loss, the EDL network also uses a regularization term that penalizes incorrect evidence when the model's prediction is inaccurate:

$$L_{\text{R}}(y, m) = |y - \gamma| \cdot (2\nu + \zeta) \tag{9}$$

where $\gamma = f_{\text{GPDE}}(x)$. Mean square error evidential network [10], used in this paper, combines the traditional MSE loss with the evidential loss used in EDL. The total loss function for the MSE Evidential Network is defined as:

$$L_{EDL} = L_{\text{NLL}}(y, m) + L_{\text{MSE}}(y, \gamma) + \lambda L_{\text{R}}(y, m), \tag{10}$$

where $L_{\text{MSE}}(y, \gamma)$ is the mean square error of (y, γ) and λ is the regularization coefficient.

4.3 Underwood Traffic Equation Loss

After discussing how to get the UQ in the GPDE network, we introduce an additional traffic equation for guiding model training with physics-informed principles. The Underwood traffic equation [21] is $v = v_f \cdot exp(-k/k_m)$, where v_f is the free flow speed and k_m is the maximum density. The parameters v_f and k_m can be estimated using linear regression.

The model can be transformed into a linear equation by taking the logarithm of both sides of the equation: $log(v) = log(v_f) - k/k_m$. The Underwood equation was proposed as an exponential relationship to resolve a flaw in Greenberg's logarithmic relationship between speed and density [13]. Greenberg's model was an improvement in some cases, but the log function explodes as the density approaches zero, making the velocity of the vehicles appear limitless. We manually set the $v = 0$ in the dataset to a small number to avoid $log(v)$ becoming negative infinity.

Meanwhile, since we have a bunch of sensors in the traffic network, running a general regression model to describe the traffic property is not precise enough to form the regularization. Instead, we run an individual regression model at each sensor and measure their coefficients. The Underwood physics loss function is defined as:

$$L_{phy} = \sum_{i}^{N} (y_{pred,i} - v_{f,i} \cdot exp^{-\frac{k_i}{k_{m,i}}}), \tag{11}$$

where N is the number of nodes, $y_{pred,i} = f_{\text{GPDE}}(x)_i$ is the prediction at node i, $v_{f,i}$ is the free flow speed at node i, k_i is the density at node i, $k_{m,i}$ is the maximum density at node i. $k_{m,i}, v_{f,i}$ are measured in the training set. The final loss function is:

$$L = L_{uq} + \lambda \cdot L_{phy}, \tag{12}$$

where L_{uq} is the loss function from UQ baselines and λ is the regularization parameter. We show how to pick an ideal λ in the experiments section.

5 Experiments

In this section, we describe the experiment details and results. Some experiments are conducted using PyTorch on a computer with the following configuration: Intel Core i7-8750H CPU @2.20GHz × 6 Processor, 16 GB Memory, GeForce GTX 1060, 64-bit Win10 OS. Other parts of the experiment ran on Google Colab T4 GPU. [1]

5.1 Datasets

5.1.1 Datasets. We evaluate the proposed method on two different traffic datasets, the METR-LA [15] and the Caltrans Performance Measurement System (PEMSD8) [7].

• The METR-LA dataset consists of four months of vehicle speed data, collected from March to June 2012, via loop detector sensors across different road segments in the Los Angeles County highway system. This traffic forecasting task involves predicting the maximum traffic speed simultaneously for 207 sensors. We split the data into 70% for training, 10% for validation, and the remaining 20% for testing. Due to significant noise within this dataset, it is challenging for models to learn the underlying patterns, often resulting in higher prediction errors.

• PEMSD8 dataset [7] contains the traffic data in San Bernardino from July to August in 2016, with 170 detectors on 8 roads with a time interval of 5 minutes. Three kinds of traffic measurements were considered, including total flow, average speed, and average occupancy. We split the data in the same way as METR-LA.

5.2 Performance Evaluation

We report the short-term and long-term testing performance of different methods for traffic speed prediction across the two datasets. We fine-tuned each baseline model to its best performance. For evaluation metrics, each experiment is conducted five times with random model initialization, and the mean values of the following metrics are reported:

• Mean Absolute Error (MAE): MAE measures the average magnitude of the errors between the predicted and true traffic speeds. Lower values indicate better model accuracy.

• Mean Interval Score (MIS): MIS assesses the quality of predictive intervals. A lower MIS indicates that the model generates more reliable and calibrated uncertainty estimates, balancing interval width and prediction accuracy.

• Mean Squared Error (MSE): MSE is a standard regression metric that squares the prediction error to give more weight to larger errors. Lower MSE values indicate better performance, especially in avoiding large prediction errors.

• Root Mean Squared Error (RMSE): RMSE is the square root of MSE.

5.2.1 Baselines. We set 2 RNN layers for encoder and decoder in the DCRNN base model with 16 hidden states. The learning rate is 0.01. We pick $\lambda = 0.0005$ for METR-LA and $\lambda = 0.002, 0.001$ for PEMSD8.

• **Quantile** We apply the quantile loss function with the corresponding quantile (0.025,0.5,0.975). Another baseline similar to quantile is spline quantile. We were trying to implement it but could not get ideal results as [22] claimed.

• **MAEMIS** We combine MAE with MIS and directly minimize this loss function.

• **EDL** DCRNN outputs four parameters needed to calculate EDL loss: $m = (\gamma, v, \zeta, \beta)$, and the loss function is changed to EDL loss. An additional MSE loss is combined with the EDL loss in Eq. 10.

5.2.2 Short-term prediction. In Table 1, the results for the 15-minute prediction demonstrate the performance for short-term prediction on two datasets. We have the following observations:

• The GPDE-enhanced methods steadily overperform the baselines without GPDE layers since the PDE layers simulate the traffic in-and-out flows in the graph. The free flow speed is measured through a Greenshield regression and serves as an input to the GPDE layer. The improvement is more considerable in PEMSD8 than in METR-LA because PEMSD8 dataset has less noise and the negative relation between traffic speed and density is significant.

Furthermore, the regularization of physics loss can help improve performance further. The reason for using the Underwood equation loss is making complementation to the Greenshield equation in the

[1] The repository is at https://github.com/tianshubao1/uq4phy_traffic_network

GPDE layer, as the Underwood equation describes an exponentially speed-density relation in contrast with a linear relation described by the Greenshield equation.

The EDL-based methods exhibit larger MIS compared to the other two baselines since EDL does not use a standard loss function like those in Quantile-based or MAEMIS-based methods. The first work using EDL in the UQ domain [10] demonstrates that EDL networks generally produce wider confidence intervals, which in turn leads to a higher MIS. In our experiments, the addition of the GPDE layer and the physics-informed loss function improves the MAE, MIS, and MSE performance compared to the vanilla EDL baseline.

• The GPDE + physics regularization method performs the best over the other 2 approaches and preserves the accuracy in multistep time predictions. For instance, in terms of the PEMSD8 dataset, the GPDE+Quantile+Phy method achieves MAE score of 1.024, compared to 1.202 in the baseline, representing a 14.8% reduction.

As discussed above, the Underwood equation serves as an additional explainer for speed-density relation and thus further improves the model performance. Meanwhile, the METR-LA dataset contains noise and zero values, making it difficult to find a speed-density relation. To alleviate this issue, we pick a small regularization parameter in METR-LA dataset. However, this issue does not affect PEMSD8 datasets much.

5.2.3 Long-term prediction. For 30-minute and 60-minute long-term predictions in Table 1, we have the following observations:

• The GPDE enhanced methods preserve the advantage observed in short-term predictions in both MSE and UQ metrics. This improvement proves that the PDE regularization can model complex spatial-temporal dynamics more accurately in long horizons.

• For GPDE + regularization models, there are observable error reductions when transitioning from short-term to long-term forecasts. Specifically, the MAE reduction is 0.02 in short-term predictions and approximately 0.12 in long-term predictions in MAEMIS baseline, METR-LA dataset, and the MAE reduction is 0.04 in short-term predictions and 0.14 in long-term predictions in Quantile baseline, METR-LA dataset. The error reduction indicates the effectiveness of physics loss and GPDE in enhancing the model's predictive capability over longer horizons.

5.2.4 UQ Analysis. For short-term predictions, the physics-informed methods (GPDE + physics regularization methods) consistently outperform their respective baselines (Quantile, MAEMIS, EDL) in terms of MIS across both the METR-LA and PEMSD8 datasets. This demonstrates that physics-informed models not only produce narrower confidence intervals but also improve accuracy, as reflected in other accuracy metrics. Moreover, the physics-informed methods demonstrate consistently lower MIS values than their baseline counterparts in the long-term prediction. This reduction in MIS is critical for ensuring reliable UQ in multi-step forecasting scenarios, where predictive intervals tend to widen, increasing the potential for uncertainty. The stability provided by our physics-informed approach mitigates this tendency, effectively managing uncertainty even in longer forecasts. This is particularly valuable for applications in real-world traffic management, where consistent reliability over both short-term and long-term horizons is essential.

Notably, the improvements in UQ are more pronounced in the PEMSD8 dataset, which contains fewer noise elements and exhibits clearer traffic patterns. This suggests that physics-informed models, when applied to cleaner datasets, can more effectively capture inherent traffic patterns. On the other hand, the METR-LA dataset, which includes greater variability and noise, still benefits from the physics-informed approach, although to a lesser extent. This indicates that the physics-based framework retains its robustness even under noisy conditions.

Comparing different UQ methods, both the Quantile and MAEMIS approaches exhibit similar performance in MIS, with physics-informed enhancements leading to notable reductions in MIS. While EDL achieves UQ without requiring retraining for each quantile, it tends to yield higher MIS values, suggesting a trade-off between computational efficiency and interval precision.

5.3 Model Analysis

5.3.1 Ablation Study. This section summarizes the results of the ablation study. We plot the results in various metrics for the Quantile-based models and the MAEMIS-based models.

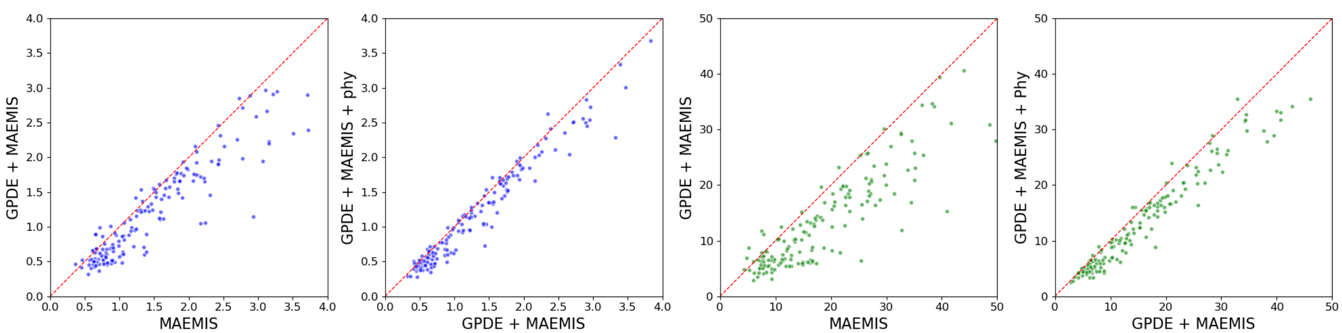
Fig. 3 presents the first ablation study for short-term prediction using the MAEMIS-based models. The left two plots compare MAE metrics, while the right two plots compare MIS metrics across different MAEMIS-based models. Each point represents the prediction error for a specific sensor in the PEMSD8 dataset. With most points positioned below the red diagonal line, it is visually evident that the models on the y-axis generally yield lower prediction errors than those on the x-axis. This finding confirms that incorporating GPDE and physics-informed components into the baseline MAEMIS model helps reduce prediction errors, which is particularly valuable for real-time short-term forecasting.

Fig. 4 presents the second ablation study, focused on long-term prediction using the Quantile-based models. Similar to Fig. 3, the left two plots compare MAE metrics, and the right two plots compare MIS metrics across different Quantile-based models. A similar conclusion can be drawn that the majority of points below the diagonal line indicate that adding the GPDE and physics-informed components improves model performance. However, the effect of the physics-informed component is less pronounced in the Quantile-based models. Nonetheless, the results suggest that the additional components contribute to reducing long-term prediction errors across various sensors, enhancing the model's effectiveness in capturing the dynamics of the traffic network.

5.3.2 Regularization Sensitivity Analysis. Since the METR-LA dataset contains much noise that restricts the usage of regularization, we test the model's robustness in PEMSD8 datasets by using various λ in Table 2 and plot the resulting error distribution in Fig. 5 and Fig. 6. Table 2 demonstrates that our method is stable against the parameter perturbation. The EDL baseline uses a different set of values, since its loss function is quite different from others. Fig. 5 and Fig. 6 indicate that most errors concentrate on small values, which are probably caused by data noise and inaccuracies. By varying the regularization, the performance does not differ too much, demonstrating that our method is stable against various perturbations.

Methods	METR-LA					PEMSD8				
		MAE	MIS	MSE	RMSE		MAE	MIS	MSE	RMSE
Quantile	15min	2.615	25.749	24.443	4.944	15min	1.202	13.738	6.541	2.557
	30min	2.941	34.898	33.824	5.815	30min	1.386	19.878	9.453	3.074
	60min	3.448	51.243	49.509	7.036	60min	1.666	30.333	14.622	3.823
GPDE+Quantile	15min	2.596	25.574	24.364	4.936	15min	1.041	12.633	4.732	2.175
	30min	2.907	34.211	33.648	5.801	30min	1.251	19.200	7.730	2.780
	60min	3.375	49.384	48.641	6.974	60min	1.540	29.760	12.714	3.565
GPDE+Quantile+Phy	15min	**2.577**	**25.137**	**23.553**	**4.853**	15min	**1.024**	**12.063**	**4.676**	**2.162**
	30min	**2.885**	**33.596**	**32.301**	**5.683**	30min	**1.236**	**18.497**	**7.696**	**2.774**
	60min	**3.347**	**47.992**	**46.506**	**6.819**	60min	**1.521**	**28.812**	**12.524**	**3.539**
MAEMIS	15min	2.722	25.129	24.677	4.967	15min	1.554	22.314	9.059	3.009
	30min	3.066	34.199	33.791	5.813	30min	1.721	31.362	12.064	3.473
	60min	3.625	51.486	49.936	7.066	60min	1.957	42.781	16.747	4.092
GPDE+MAEMIS	15min	2.711	24.664	24.614	4.961	15min	1.269	15.406	6.191	2.488
	30min	3.042	33.184	33.841	5.817	30min	1.454	22.196	9.231	3.038
	60min	3.538	48.508	49.399	7.028	60min	1.736	33.147	14.398	3.794
GPDE+MAEMIS+Phy	15min	**2.703**	**24.545**	**24.420**	**4.941**	15min	**1.142**	**12.853**	**5.432**	**2.330**
	30min	**3.018**	**32.371**	**33.043**	**5.748**	30min	**1.369**	**19.678**	**8.714**	**2.950**
	60min	**3.499**	**46.560**	**47.737**	**6.909**	60min	**1.686**	**30.957**	**14.204**	**3.768**
EDL	15min	2.990	78.510	51.670	7.188	15min	1.921	**19.919**	16.968	4.119
	30min	3.523	90.311	76.110	8.724	30min	2.326	**24.996**	22.897	4.785
	60min	4.328	106.981	114.534	10.702	60min	3.133	42.023	35.304	5.941
GPDE+EDL	15min	**2.979**	74.798	52.253	7.228	15min	1.366	31.500	8.189	2.861
	30min	**3.484**	86.069	76.140	8.725	30min	1.521	36.273	10.740	3.277
	60min	**4.233**	104.476	113.142	10.636	60min	1.748	43.461	15.118	3.888
GPDE+EDL+Phy	15min	3.048	**71.317**	**51.190**	**7.154**	15min	**1.214**	30.675	**6.749**	**2.698**
	30min	3.539	**82.696**	**76.093**	**8.723**	30min	**1.404**	34.463	**9.681**	**3.111**
	60min	4.240	**99.762**	**110.188**	**10.497**	60min	**1.685**	**39.687**	**14.626**	**3.824**

Table 1: Experiments of speed prediction on METR-LA and PEMSD8 datasets

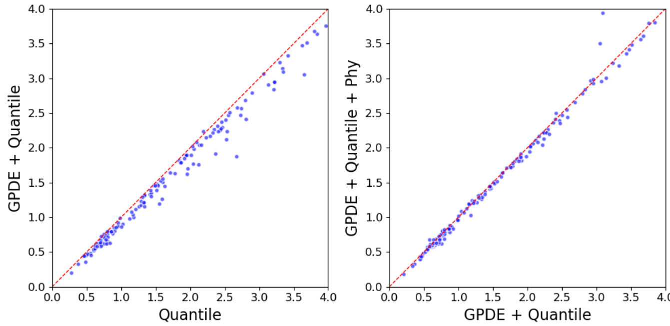


(a) Short-term prediction MAE of MAEMIS-based models.

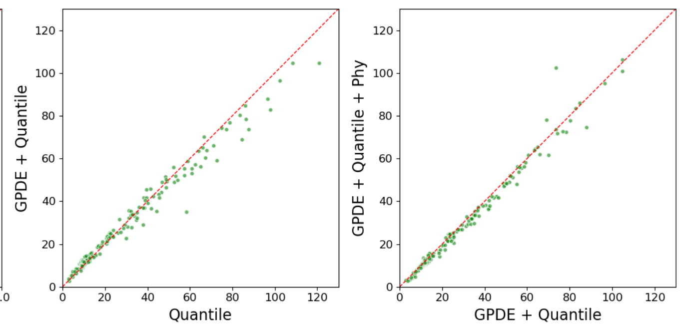
(b) Short-term prediction MIS of MAEMIS-based models.

Figure 3: Ablation study for MAEMIS-based methods in PEMSD8 datasets. Every node in the plots denotes the corresponding short-term prediction errors of the given sensors in the traffic network. It is easy to tell that most of the nodes are below the diagonal, indicating that the y-axis methods overperform the x-axis methods.

5.3.3 Regularization Convergence Analysis. Another advantage of using physics regularization is accelerating the convergence, especially in the PEMSD8 dataset. With regularization, the model can roughly double the convergence speed of the original model. In PEMSD8, regularization helps the model to reach a similar performance of base models using 20 epochs rather than 40 epochs. Meanwhile, the convergence speed acceleration is observed, as mentioned in the short-term prediction. One example is in Fig. 7.



(a) Long-term prediction MAE of Quantile-based models.



(b) Long-term prediction MIS of Quantile-based models.

Figure 4: Ablation study for Quantile methods in PEMSD8 datasets. Every node in the plots denotes the corresponding long-term prediction errors of the given sensors in the traffic network. It is easy to tell that most of the nodes are below the diagonal, indicating that the y-axis methods overperform the x-axis methods.

PEMSD8	λ	MAE	MIS	MSE	RMSE
GPDE+Quantile+Phy	0.01	1.626	31.431	14.537	3.812
	0.005	1.562	29.982	12.997	3.605
	0.002	**1.521**	**28.812**	**12.524**	**3.539**
GPDE+MAEMIS+Phy	0.01	1.681	31.624	14.701	3.834
	0.005	**1.680**	31.238	**13.918**	**3.731**
	0.002	1.686	**30.957**	14.204	3.768
GPDE+EDL+Phy	0.01	1.776	**28.329**	15.319	3.914
	0.002	1.794	37.12	15.099	3.885
	0.001	**1.685**	39.687	**14.626**	**3.824**

Table 2: Regularization sensitivity analysis using various λ values in PEMSD8 dataset. The optimal outcomes do not come from a single setting because the loss function contains multiple sources instead of a single metric. Changing λ in a reasonable range does not alter the outcome too much.

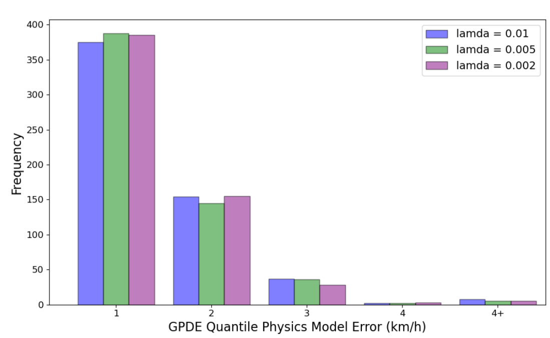


Figure 5: GPDE+Quantile+Phy model error distribution with various λ values at Location 5 for 48 hours. The error does not vary much as we change λ, indicating the Quantile-based model's robustness against regularization perturbation.

5.4 Case Study

To better illustrate the advantages of our physics-informed models , we visualized the traffic speed predictions of different models at

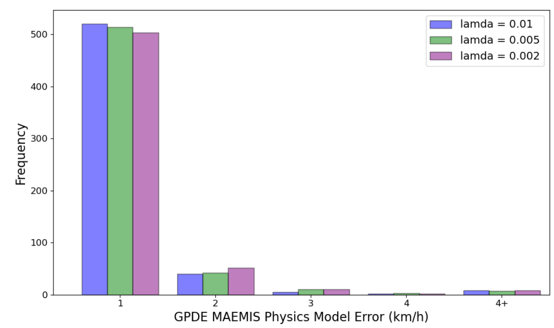


Figure 6: GPDE+MAEMIS+Phy Error Distribution with various λ values at Location 107 for 48 hours. The error does not vary much as we change λ, indicating the MAEMIS-based model's robustness against regularization perturbation.

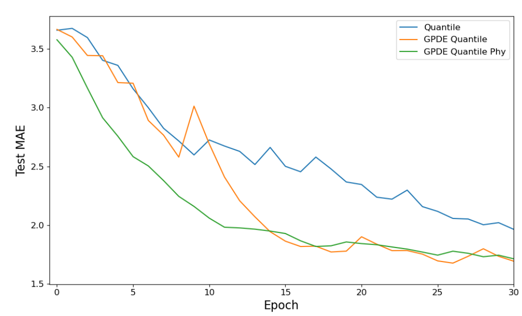


Figure 7: The test loss for quantile-based method. The physics model converges quickly within a few epochs.

various locations in different datasets, along with their performance and prediction uncertainty during regular hours and rush hours.

Fig. 8 shows the predictions for Location 9 over a 48-hour period using the three Quantile-based models' loss compared to the ground truth. In the full 48-hour plot, significant fluctuations in traffic speed are observed during rush hours, indicating periods of

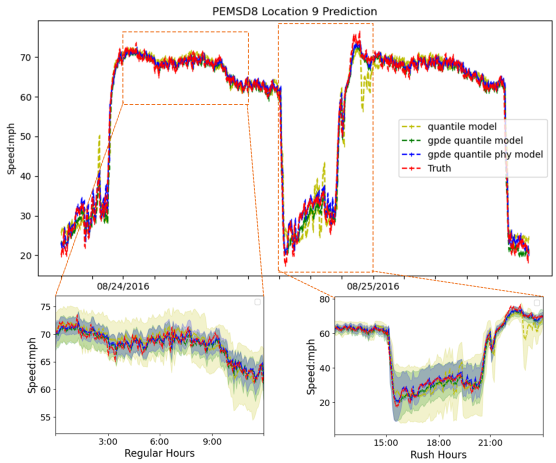


Figure 8: The predictions of three Quantile-based models compared to the ground truth for Location 9 from the PEMSD8 dataset, over the period from August 23, 2016, to August 25, 2016. The top plot shows the full 48-hour prediction, while the two bottom plots provide zoomed-in views of the regular hours (left) and rush hours (right) for more detailed inspection. The shaded regions represent the confidence intervals for each model's predictions, highlighting the uncertainty in each method's estimates.

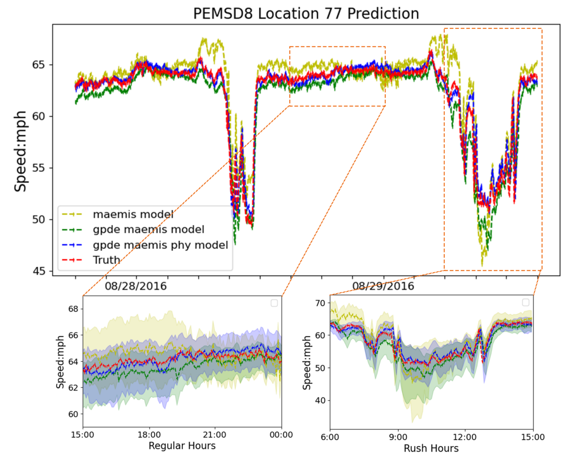


Figure 9: The traffic speed predictions of three MAEMIS-based models compared to the ground truth for Location 77 from the PEMSD8 dataset, covering the period from August 28, 2016, to August 29, 2016. The top plot provides a full 48-hour prediction, while the two bottom plots zoom in on rush hours (left) and regular hours (right) for more detailed inspection. The shaded regions represent the confidence intervals for each model's predictions, highlighting the uncertainty in each method's estimates.

heavy traffic. The zoomed-in views on regular hours and rush hours highlight how the different models handle various time periods. The confidence intervals (shaded regions) show that the Quantile model tends to have larger uncertainty, particularly during periods of higher traffic variation. In contrast, the GPDE models, especially the physics model, produce more accurate predictions that align better with the ground truth, especially during rush hours when traffic speeds change rapidly. The physics model captures the speed changes better, showing more consistency and smaller confidence intervals.

Fig. 9 presents predictions for Location 77 over a similar 48-hour time frame using the MAEMIS-based models. Again, all three models are compared against the ground truth. During periods of high traffic variation, especially in the rush hours, there are clear differences in model performance. The zoomed-in view of rush hours reveals that the MAEMIS model shows larger deviations from the true values, and its confidence intervals are wider, particularly during steep drops in traffic speed. On the other hand, the GPDE+MAEMIS+Phy model maintains smaller confidence intervals and aligns more closely with the true values, demonstrating its ability to model traffic patterns more accurately during high-variability periods. In normal traffic hours, the performance of all models is relatively stable, but the physical model still maintains better performance.

Similarly, in the METR-LA dataset, we selected one location to visualize the traffic speed predictions. Due to the higher noise levels in METR-LA (such as periods when the ground truth drops to zero), we only selected a set of predictions for visualization. Fig. 10 presents predictions for Location 130 from the METR-LA

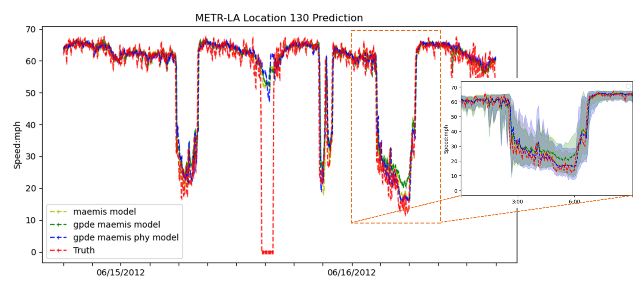


Figure 10: The predictions of three MAEMIS-based models compared to the ground truth for Location 130 from the METR-LA dataset, covering the period from June 14, 2012, to June 16, 2012. The main plot shows a full 48-hour prediction, while the inset zooms in on a specific section of the day to highlight more detailed variations. The shaded regions represent the confidence intervals for each model.

dataset. The inset zooms in on one of the steep drops, showing how each model handles extreme conditions. The GPDE+MAEMIS+Phy model again performs better in these situations, with its predictions closely following the ground truth and its confidence intervals remaining relatively narrow. This indicates a better ability to capture sharp drops in speed, a common characteristic in urban traffic patterns. The MAEMIS model, however, shows more uncertainty with larger confidence intervals, particularly during the periods of rapid speed fluctuations, thus making it less reliable in these extreme cases.

6 Conclusion

In this study, we present a novel traffic forecasting network that combines the GPDE with UQ and physics loss. To the best of our knowledge, this is cutting-edge work to enhance the model's safety utilizing physics equations to the node of road networks in the area of traffic, which enables us to construct deeper networks and leverage wider-range dependencies. Furthermore, the UQ loss function enables us to statistically evaluate the output intervals and the physics loss function significantly accelerates the model convergence speed and improves accuracy. Extensive experiments prove the effectiveness of proposed methods over existing methods on different time spans and road networks. Since this work relies on numerical experiments, future work can include verifying the theoretical correctness of the model.

Acknowledgments

The work was partially supported by NSF awards #2421839. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

References

[1] Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U Rajendra Acharya, et al. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information fusion*, 76:243–297, 2021.

[2] Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. Deep evidential regression. *Advances in neural information processing systems*, 33:14927–14937, 2020.

[3] Ross Askanazi, Francis X Diebold, Frank Schorfheide, and Minchul Shin. On the comparison of interval forecasts. *Journal of Time Series Analysis*, 39(6):953–965, 2018.

[4] Tianshu Bao, Shengyu Chen, Taylor T Johnson, Peyman Givi, Shervin Sammak, and Xiaowei Jia. Physics guided neural networks for spatio-temporal super-resolution of turbulent flows. In *Uncertainty in Artificial Intelligence*, pages 118–128. PMLR, 2022.

[5] Tianshu Bao, Xiaowei Jia, Jacob Zwart, Jeffrey Sadler, Alison Appling, Samantha Oliver, and Taylor T Johnson. Partial differential equation driven dynamic graph networks for predicting stream water temperature. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 11–20. IEEE, 2021.

[6] Tianshu Bao, Hua Wei, Junyi Ji, Daniel Work, and Taylor Thomas Johnson. Spatial-temporal pde networks for traffic flow forecasting. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 166–182. Springer, 2024.

[7] Chao Chen, Karl Petty, Alexander Skabardonis, Pravin Varaiya, and Zhanfeng Jia. Freeway performance measurement system: mining loop detector data. *Transportation Research Record*, 1748(1):96–102, 2001.

[8] Christian G Claudel and Alexandre M Bayen. Guaranteed bounds for traffic flow parameters estimation using mixed lagrangian-eulerian sensing. In *2008 46th Annual Allerton Conference on Communication, Control, and Computing*, pages 636–645. IEEE, 2008.

[9] Xuan Di, Rongye Shi, Zhaobin Mo, and Yongjie Fu. Physics-informed deep learning for traffic state estimation: A survey and the outlook. *Algorithms*, 16(6):305, 2023.

[10] Zhiyuan Feng, Kai Qi, Bin Shi, Hao Mei, Qinghua Zheng, and Hua Wei. Deep evidential learning in diffusion convolutional recurrent neural network. *Electronic Research Archive*, 31(4):2252–2264, 2023.

[11] Bidisha Ghosh, Biswajit Basu, and Margaret O'Mahony. Multivariate short-term traffic flow forecasting using time-series analysis. *IEEE transactions on intelligent transportation systems*, 10(2):246–254, 2009.

[12] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

[13] Harold Greenberg. An analysis of traffic flow. *Operations research*, 7(1):79–85, 1959.

[14] Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 922–929, 2019.

[15] Hosagrahar V Jagadish, Johannes Gehrke, Alexandros Labrinidis, Yannis Papakonstantinou, Jignesh M Patel, Raghu Ramakrishnan, and Cyrus Shahabi. Big data and its technical challenges. *Communications of the ACM*, 57(7):86–94, 2014.

[16] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[17] Roger Koenker and Kevin F Hallock. Quantile regression. *Journal of economic perspectives*, 15(4):143–156, 2001.

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

[19] Qingqing Long, Yilun Jin, Yi Wu, and Guojie Song. Theoretically improving graph neural networks via anonymous walk graph kernels. In *Proceedings of the Web Conference 2021*, pages 1204–1214, 2021.

[20] Yuyan Annie Pan, Jifu Guo, Yanyan Chen, Siyang Li, Wenhao Li, et al. Incorporating traffic flow model into a deep learning method for traffic state estimation: A hybrid stepwise modeling framework. *Journal of Advanced Transportation*, 2022, 2022.

[21] Robin T Underwood. Speed, volume, and density relationships. *Transportation Research Board*, 1960.

[22] Dongxia Wu, Liyao Gao, Matteo Chinazzi, Xinyue Xiong, Alessandro Vespignani, Yi-An Ma, and Rose Yu. Quantifying uncertainty in deep spatiotemporal forecasting. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1841–1851, 2021.

[23] Zonghan Wu, Shirui Pan, Guodong Long, et al. Graph wavenet for deep spatial-temporal graph modeling. *arXiv:1906.00121*, 2019.

[24] Huaxiu Yao, Xianfeng Tang, Hua Wei, Guanjie Zheng, and Zhenhui Li. Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 5668–5675, 2019.

[25] Huaxiu Yao, Fei Wu, Jintao Ke, et al. Deep multi-view spatial-temporal network for taxi demand prediction. In *AAAI*, 2018.

[26] Yi Yin and Pengjian Shang. Forecasting traffic time series with multivariate predicting method. *Applied Mathematics and Computation*, 291:266–278, 2016.

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

[28] Rose Yu, Yaguang Li, Cyrus Shahabi, Ugur Demiryurek, and Yan Liu. Deep learning: A generic approach for extreme condition traffic forecasting. In *SDM'17*, 2017.

[29] H Michael Zhang. A non-equilibrium traffic model devoid of gas-like behavior. *Transportation Research Part B: Methodological*, 36(3):275–290, 2002.

[30] Junbo Zhang, Yu Zheng, and Dekang Qi. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

[31] Xiyue Zhang, Chao Huang, et al. Spatial-temporal convolutional graph attention networks for citywide traffic flow forecasting. In *CIKM'18*, 2020.