

Traffic Flow Modeling With Gradual Physics Regularized Learning

Yun Yuan¹, Qinzhen Wang, and Xianfeng Terry Yang²

Abstract—Traffic flow modeling for traffic state estimation is a vital component in many traffic management and operation systems. To leverage both machine learning (ML) methods and classical traffic flow models, the previous study has developed a hybrid framework for encoding traffic flow into multivariate Gaussian Process. However, the computational efficiency is low due to multiple inputs, outputs and equations. To improve the efficiency of the previous method, this paper presents a new modeling framework, named gradual physics regularized learning, to incrementally encode complex traffic flow models into the ML process. More specifically, the method starts with the involvement of traffic flow models from the lower-order version, such as the fundamental diagram and the kinetic wave models. Then the learned parameters and hyperparameters can be further fine-tuned with the high-order models. A field test based on real-world freeway measurements indicates the proposed model can leverage the additional physical equations to achieve better performance in estimation accuracy and robustness. Meanwhile, the gradual learning method can significantly reduce the computational efforts and further enables its application to scenarios with either larger datasets or more complex traffic flow models.

Index Terms—Second-order traffic flow model, traffic state estimation, gradual physics regularized learning, multivariate Gaussian process.

I. INTRODUCTION

MODELING traffic states (e.g., flow, density, and speed) on highway networks is a key function of many traffic operation tasks, such as travel time prediction, dynamic vehicle routing, etc., in intelligent transportation systems [1]. Due to high installation and maintenance costs, devices such as roadside traffic detectors only deliver real-time traffic measurements at sparse locations. The spatial resolution of available traffic information is typically insufficient for the direct implementation of appropriate control actions. As a result, how to estimate traffic states of road segments between detectors is

an important issue for traffic management and control and has received an increasing amount of research interests.

During the past decades, macroscopic traffic flow models have shown great promise in capturing the features of the traffic patterns. In the literature, they are developed and refined gradually. Based on the complexity of formulations, existing traffic flow models can be categorized into two kinds: (a) the first-order Lighthill-Whitham-Richards (LWR) models [2], [3], and (b) the second-order Payne-Whitham (PW) models [4], [5] and Aw-Rascle-Zhang (ARZ) models [6], [7]. These models are based on the underlying relationship between traffic flow, density, and speed, which is formalized to the Fundamental Diagram (FD). In addition to the FD, the LWR model also involves the mass Conservation Law (CL) to capture the traffic dynamics, such as congestion and shockwave. By involving the momentum CL equation, the second-order models were proposed to replicate more complex phenomena.

Despite wide implementations, these models are often developed under ideal conditions and are difficult to calibrate with real-world data. To handle the inherent randomness of traffic flow and model uncertainties, various methods were proposed to improve the estimation accuracy and robustness, including data assimilation and data imputation methods. Data assimilation methods are model-based, which first estimates the parameters in the models and then predicts the unobserved based on the parameters. Stochastic extensions were proposed to extend the model-based methods [8]–[12]. To address the missing data problem, data imputation methods, such as machine learning (ML) models, can provide prediction without strong assumptions but rely on the quality and quantity of data. This category of research includes Boltzmann-based models [13], [14], Markovian queuing network approaches [15]–[17], cellular automaton based models [18], [19], and Gaussian Process [20]–[26].

However, due to the data-driven nature, those ML-based models fundamentally suffers from three scenarios: (i) training data are scarce and insufficient to reveal the complexity of the system, (ii) training data are noisy and include much incorrect/misleading information, and (iii) test data are far from the training examples, i.e., extrapolation. In these scenarios which are unfortunately very common in the real-world, their performance can drop dramatically along with large and/or biased estimations. To address these issues, researchers have proposed hybrid frameworks to leverage the physical knowledge in the ML framework, such as Physics-guided machine learning (PGML) [27] and Physics-informed machine

Manuscript received June 12, 2020; revised April 14, 2021, October 19, 2021, and November 23, 2021; accepted November 24, 2021. This work was supported by the National Science Foundation (CAREER: Physics Regularized Machine Learning Theory: Modeling Stochastic Traffic Flow Patterns for Smart Mobility Systems) under Grant CMMI 2047268. The Associate Editor for this article was Y. Wang. (Corresponding author: Xianfeng Terry Yang.)

Yun Yuan is with the Department of Civil and Environmental Engineering, The University of Utah, Salt Lake City, UT 84112 USA, and also with the College of Transportation Engineering, Dalian Maritime University, Dalian 116026, China (e-mail: yun.yuan@utah.edu).

Qinzhen Wang and Xianfeng Terry Yang are with the Department of Civil and Environmental Engineering, The University of Utah, Salt Lake City, UT 84112 USA (e-mail: qinzhen.wang@utah.edu; x.yang@utah.edu).

Digital Object Identifier 10.1109/TITS.2021.3131333

1558-0016 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.

See <https://www.ieee.org/publications/rights/index.html> for more information.

learning (PIML) [28]–[30]. PGML is designed to learn the pattern from both real data and the calibrated model predictions. PIML is defined to use the calibrated model error as a part of the loss function of the learning process. As shown in the literature, PGML and PIML were proved to be effective when dealing with a small dataset. However, both of them suffer from the following limitations: i) they often assume the partial differential equations are available to generate the physical inconsistency term; and ii) they are based on the NN family models, which always have over-fitting problems, and their model performance still depends on the quality of the training dataset.

To bridge the existing research gaps, Wang *et al.* [31] introduced the general PRGP concept to extend the conventional GP to incorporate the partial differential equations as the regularizer in the posterior inference algorithm. The physics model-based regularization is conducted by encoding the physics equations into GPs and adding the corresponding log-likelihood into the inference objective function as a penalty term. However, the original PRGP model was developed with a single output variable, and were tested on one single-variable differentiable physics equations. Following the same line, our later study [32] extended the PRGP model to handle the multiple outputs and multiple physics equations simultaneously, and applied the PRGP to the TSE problem. However, this extended model can only employed continues macroscopic traffic model and did not prove the applicability of PRGP on the discretized models where the partial differential equations cannot be obtainable. Also, it should be noted that many traffic flow models in the literature are in discretized forms. Hence, to ensure the applicability of the PRGP model in a broader application domain, the following study [33] aims to further develop a new modeling method to encode discretized models into Gaussian Process. The discretized method study also revised the evident lowerbound formulation to validate the compatibility of the PRGP model and the discretized traffic flow models. It advances PRGP to leverage both classical discrete macroscopic traffic flow models and data-driven methods.

However, the current PRGP theory [32], [33] can only train the GP with a regularized term from the scratch and usually consumes a lot of time for parameter fine-tuning. Due to the integration of complex traffic flow models, which can have multi recursive equations and involve multi-variate inputs/outputs, the low computational efficiency of those PGML, PIML, and PRML models becomes a problem when dealing with a relative larger dataset. Hence, it could create application barrier when users need to deal with a complex estimation/prediction system. To tackle such a critical issue, this study aims to advance the method by providing a new modeling architecture called the gradual learning framework to incrementally fine-tune the ML models with additional physical equations. More specifically, the newly formed modeling concept, called gradual physics regularized learning (GPRL), will start with encoding basic and lower-order models, such as the fundamental diagram and the kinetic wave models. Then the learned parameters and hyperparameters from the base model can be further fine-tuned with the high-order models

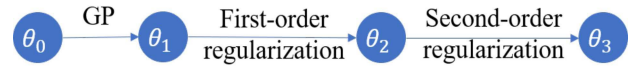


Fig. 1. The gradual regularization mechanism of the proposed model.

so that the ML training process can be gradually regularized and the corresponding computation efficiency would be significantly improved. The proposed gradual learning method is expected to significantly reduce the required training efforts but still achieve the same performance level of PRGP.

In this study, the trained GP model can be further updated with the traffic models, i.e. FD, LWR, PW, and ARZ, to take advantage of the trained models. In view of the fact that PW and ARZ are developed based on LWR and LWR is developed based on FD, the gradual regularization procedure can be implemented by training GP, GPRL-FD, GPRL-LWR, GPRL-PW or GPRL-ARZ incrementally with reduced computational cost. Fig. 1 shows the concept of the GPRL method, where the initial parameter θ_0 is trained by with the base ML model and the resultant parameter θ_1 can be further fine-tuned by involving FD equations. The results yielded by the parameter θ_2 are then improved from the results of parameters θ_1 by involving a first-order traffic flow model (e.g., LWR) and parameter θ_3 is obtained later with the integration of a second-order model (e.g., PW or ARZ). Finally, the PRGP with second-order traffic flow models (such as PW and ARZ) is trained. By gradually adding physics equation-based regularization terms to the objective function, the parameter is fine-tuned. The learning with the gradual structure can effectively narrow down the feasible region in the model's solution algorithm and the required time for training would be consequently reduced.

The rest of this paper is arranged as follows: Section II introduces the modeling fundamentals such as the key formulations of the encoded traffic flow models and the basic theory of multi-variant GP; Section III presents the formulation of the GPRL model along with the solution algorithms; Section IV justifies the proposed method in estimating traffic flow and speed; and Section V summarizes several concluding remarks and future research.

II. FUNDAMENTALS

The physics regularized Gaussian Process (PRGP) is based on two fundamentals including the macroscopic traffic flow model and multi-variant Gaussian Process. The macroscopic traffic flow model is encoded into GPs. The logarithm evidence lowerbound of the posterior probability of the physics model encoded GP is added to the objective function of the original GP as the regularization term. By training PRGP, the hybrid model compromises the fitness of the data-driven model and the discrepancy of the physics model to improve the estimation performance.

A. Macroscopic Traffic Flow Model

In traffic flow modeling, aggregated measures from traffic detectors, such as flow rates and mean speed, are used

to describe the state of the traffic pattern. In early days, researchers found the existence of the FD to illustrate the relationship among flow q , speed v and density ρ . The FDs are used to predict the capability and the behavior of a road system, as shown in the speed-density relation Eq. 1 and the flow-density relation Eq. 2.

$$v = V\rho \quad (1)$$

$$q = \rho V(\rho) = Q(\rho) \quad (2)$$

where $V(\cdot)$ denotes the density-speed function, and $Q(\cdot)$ represents the density-flow function. The FD formulations can be classified into two categories: (a) single-regime models, such as Greenshields [34], Greenberg [35], Underwood [36], Drake *et al.* [37], Drew [38], Pipes [39] and Munjal [40]; and (b) multi-regime models, such as Edie model [41], the two-regime model [42], modified Greenberg [43], and three-regime model [44]. For example, the widely-used triangular FD is formulated in the following equation.

$$q = \min\{\rho * \beta_0, q_m, (\beta_1 - \rho) * \beta_2\} \quad (3)$$

where β_0, β_2 are the slope of the linear parts, q_m is the maximum traffic flow, and β_1 is the maximum density. To capture aggregated traffic behaviors, macroscopic traffic flow models were investigated inspired by continuum fluid approximation. For example, to describe basic traffic operation phenomenons such as traffic jam and shockwave, the well-known first-order Lighthill-Whitham-Richards (LWR) model [2], [3] is formulated in Eq. 2 and Eq. 4, where Eq. 4 refers to the mass conservation law and Eq. 2 is the FD density-flow function.

$$\partial_t \rho + \partial_x(\rho v) = 0 \quad (4)$$

However, the LWR model has limitations in the reproducibility of more complex phenomena, such as the dynamics of speed. To overcome such limitations, second-order models that use the additional momentum equation were later developed. As a branch of the second-order model, the Payne-Whitham (PW) model [4], [5] is formulated by Eqs. 1,4,5, in which the additional Eq. 5 is the momentum conservation equation.

$$\partial_t v + v \partial_x v = -\frac{V - V(\rho)}{\tau_0} - \frac{c_0^2}{\rho} \partial_x \rho \quad (5)$$

where τ_0 denotes the relaxation time and c_0^2 denotes a parameter related to driver anticipation. Despite the success of the PW model and its variations [45], the PW-like models may produce non-realistic outputs, such as negative speed [46]–[49].

To overcome this limitation, another branch of the second-order model, the Aw-Rascle-Zhang (ARZ) model [6], [7] is formulated in Eqs. 1,4,6, where another formulation of the momentum conservation law is shown in Eq. 6. The original ARZ model was extended extensively in the literature [50]–[53].

$$\partial_t(v - V(\rho)) + v \partial_x(v - V(\rho)) = -\frac{v - V(\rho)}{\tau_0} \quad (6)$$

However, it should be noted that despite of the elegance of differential equation formalization, a traffic flow model is difficult to estimate traffic state uncertainties since the solving the nonlinear differential equations is not a trivial task.

Considering the system and observation noise, data assimilation or inverse modeling techniques were then developed for model estimation and calibration. In the literature, there exist three ways to add randomness in the traffic models: (a) stochastic initial and boundary conditions, (b) stochastic source terms (e.g. inflows), and (c) stochastic speed-density relationship or fundamental diagram [54]. To capture the measurement error in data, a stochastic modeling method is performed by adding Gaussian noise to the traffic state estimates [8]–[12], [21], [54]. For example, given the nonlinearity of the second-order traffic flow model, some studies [10], [11] assumed the error terms on the formula and developed extended Kalman filter (EKF) to estimate a PW-like discrete model [45]. To address these issues, this study proposes to use GP along with gradual physics regularizer to capture the data noise and randomness, and leverages the physical knowledge from existing models, such as FD, LWR, PW, and ARZ, to regularize the training process.

B. Multi-Variant Gaussian Process (GP)

GP is a data-driven method for capturing the similarity between the system states, of which the core idea is to learn the kernel function (i.e. covariance) between variables and to predict the target by the linear combination of the training data [55].

In general, the main task in GP is to learn a mapping $\mathbf{f} : \mathbb{R}^d \rightarrow \mathbb{R}^{d'}$ from a d -dimensional space to a d' -dimensional space from a training set $\mathcal{D} = (\mathbf{X}, \mathbf{Y})$, where $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]^T$ is the input vector, $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_n]^T$ is the output vector, $\mathbf{x}$ is the d dimensional input vector, $\mathbf{y}$ is the d' dimensional output vector, $\mathbf{f} = [f(\mathbf{x}_1), \dots, f(\mathbf{x}_n)]^T$ is the learning function, and n refers to the sample size. Note that $\mathbf{X}, \mathbf{Y}$ may have physical meanings only in their feasible domains. Given the new input $\mathbf{x}^*$, the function value $\mathbf{f}$ can be estimated based on Eq. 7.

$$p(\mathbf{f}(\mathbf{x}^*)|\mathbf{x}^*, \mathbf{X}, \mathbf{Y}) = \mathcal{N}(\mu(\mathbf{x}^*), \sigma(\mathbf{x}^*)) \quad (7)$$

where the mean $\mu(\mathbf{x}^*)$, standard deviation $\sigma(\mathbf{x}^*)$, the kernel matrix $\mathbf{K}$ and the kernel vector $\mathbf{K}_*$ are calculated in Eqs. 8–11, respectively. $\mathbf{I}$ refers to the identity matrix.

$$\mu(\mathbf{x}^*) = \mathbf{K}_*^T(\mathbf{K} + \tau^{-1}\mathbf{I})^{-1}\mathbf{Y} \quad (8)$$

$$\sigma(\mathbf{x}^*) = K(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{K}_*^T(\mathbf{K} + \tau^{-1}\mathbf{I})^{-1}\mathbf{K}_* \quad (9)$$

$$[\mathbf{K}]_{ij} = K(\mathbf{x}_i, \mathbf{x}_j), \forall i, j = 1, \dots, n \quad (10)$$

$$[\mathbf{K}_*]_i = K(\mathbf{x}^*, \mathbf{x}_i), \forall i = 1, \dots, n \quad (11)$$

The kernel K is defined as the non-parametric smooth positive-definite covariance function with parameters $\eta_1, \eta_2, \eta_3, \dots$. The formulation of the kernel function can be a tunable hyperparameter.

The GP method has been applied to the traffic state estimation problems and has shown great performance and potentials in the previous studies. In traffic modeling, GP-based methods have been applied in traffic speed imputation [56], [57], public transport flows [20], traffic volume estimation and prediction [21], travel time prediction [22], driver velocity profiles [23] and traffic congestion [24]. To capture the temporal

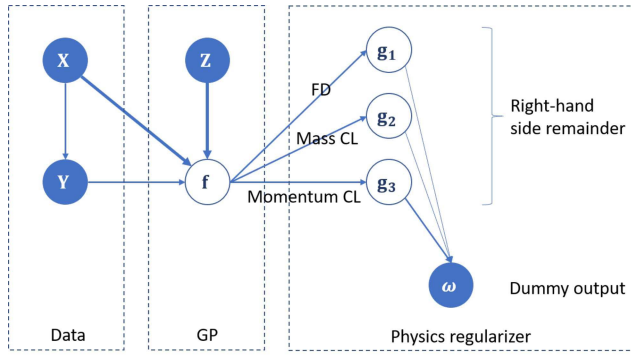


Fig. 2. The framework for Gaussian process learning with three physics regularizers.

correlation, [25] utilized the flows from adjacent road links at the previous four time intervals for traffic flow prediction. To address the missing data, [26] developed local GPs for efficient traffic speed prediction in real-time for clustering of speed in both space and time. However, due to the data-driven nature, those machine learning (ML) models fundamentally suffer from data quality problems. To address these issues, [32] proposed the Physics Regularized Gaussian Process (PRGP) to leverage the physical knowledge in the GP framework. However, the mechanism of efficiently involving complex traffic flow models that includes multi-equations and multi-variables in one framework is under-investigated. This paper aims to fill the gap by proposing a gradual learning method by extending physics regularized Gaussian process.

III. GRADUAL PHYSICS REGULARIZED LEARNING

A. Gaussian Process With Physics Regularizer

Similar to other ML models, the performance of GP also relies heavily on data quality and its results are usually hard to be interpreted with physical meanings. To address this issue, some physical knowledge is employed to regularize the training process for more robust and explainable performance. In the real world, raw data may be biased, noisy, and missing due to system and communication failure, etc. Also note that the flow, density, and speed do not have physical meanings and are only separated isotropic dimensions in the pure GP framework. To repair the data-bared flaw, the GP with a physics regularizer leverages *a priori* dynamics between traffic state measures for improving the estimation accuracy and robustness. In the PRGP framework, physical knowledge from traffic flow models is encoded as additional shadow GPs, which captures both the stochasticity due to flawed and noisy data as well as the unobserved factors, such as missing on-ramp or off-ramp data. To leverage the advantages of both ML models and macroscopic traffic flow models, the design concept of using GP with physics regularizer is illustrated in Fig. 2, where the circled nodes denote the random vectors, the shaded nodes represent known vectors, and the arrows indicate the conditional probabilities [32].

To enable Bayesian framework that incorporates the physical knowledge, a generative component $p(\omega|\mathbf{X}, \mathbf{Y})$ is used that acts as a regularizer on the GP model $p(\mathbf{Y}|\mathbf{X})$. To sample

the pseudo observation ω , this method samples the posterior function values at each $\mathbf{z}_j$, $j = 1, \dots, m$ from the Gaussian distribution as shown in Eq. 12.

$$p(f(\mathbf{z}_j)|\mathbf{z}_j, \mathbf{X}, \mathbf{Y}) = \mathcal{N}(\mu(\mathbf{z}_j), \sigma(\mathbf{z}_j)), \quad \forall j = 1, \dots, m \quad (12)$$

The predicted physical function values are obtained at $\mathbf{Z}$, $\mathbf{g} = [g(\mathbf{z}_1), \dots, g(\mathbf{z}_m)]^T$. Given the physical equation remainder value vector $\mathbf{g}$, the pseudo observations ω are sampled from another GP.

$$p(\omega|\mathbf{g}, \mathbf{Z}) = \mathcal{N}(\mathbf{g}, \hat{\mathbf{K}}) \quad (13)$$

where $\hat{\mathbf{K}}$ is the covariance matrix of the shadow GP.

$\mathbf{Z}$ is the set of random pseudo-inputs, and is used to compute $\mathbf{g}$ and test the discrepancy of estimated traffic state and the physics model. If no additional information is given, $\mathbf{Z}$ can be uniformly selected. $\mathbf{g}_1, \mathbf{g}_2, \dots$ is calculated by substituting the estimated output $\mathbf{f}$ to left-hand side of the physics equations $\Psi \mathbf{f} = \mathbf{g}$, and yielding the right-hand side as $\mathbf{g}$. Note that Ψ is a functional operator, and $\Psi[f]$ means to apply the operator on the function f . The auto-differential packages, such as Tensorflow or PyTorch, are used to implement the calculation.

The detailed deduction of the base model of PRGP can be found in the previous studies [31], [32]. This paper proposes to learn the PRGP with multiple physics models where the physics models are gradually loaded as regularizers, and the gradual learning is more efficient than the baseline PRGP.

To infer the PRGP, the posterior regularization is based on optimizing the parameters to maximize the likelihood or the evidence lowerbound (ELBO) of the likelihood [32]. The objective includes the model likelihood on data and a penalty term that encodes the constraints over the posterior of the variables. Via the penalty term, ELBO can incorporate our domain knowledge or constrains outright to the posteriors, rather than through the priors and a complex, intermediate computing procedure.

Regarding multiple outputs and multiple dimensions, the log-likelihood and the ELBO of the traffic flow model can be formulated in Eq. 14.

$$\begin{aligned} \log[p(\mathbf{Y}, \omega|\mathbf{X})] &\geq \mathcal{L} \\ &= \sum_{i=1}^{d'} \log[\mathcal{N}([\mathbf{Y}]_i | \omega, \hat{\mathbf{K}}_i + \tau^{-1} \mathbf{I})] \\ &\quad + \sum_{w=1}^W \gamma_w \mathbb{E}_{p(\mathbf{z})} \mathbb{E}_{p(\hat{\mathbf{f}}_w|\mathbf{Z}, \mathbf{X}, \mathbf{Y})} [\log[\mathcal{N}(\Psi \hat{\mathbf{f}}_w | \omega, \hat{\mathbf{K}}_w)]] \quad (14) \end{aligned}$$

Note that the prefixed positive parameter γ is used to control the strength of regularization effect. The larger the value of γ , the greater regularization effect would be applied on the learning process. The parameter vector θ is defined in Eq. 15.

$$\theta = [\theta_f, \theta_g]^T = [\tau_f, \eta_{f1}, \eta_{f2}, \eta_{f3}, \dots, \tau_{g1}, \eta_{g11}, \eta_{g12}, \eta_{g13}, \dots]^T \quad (15)$$

If the likelihood is maximized for both the original GP and the generative component, the parameters of the original GP are regularized by the equations of the physical model. The ELBO of the log-likelihood is obtained by minimizing the

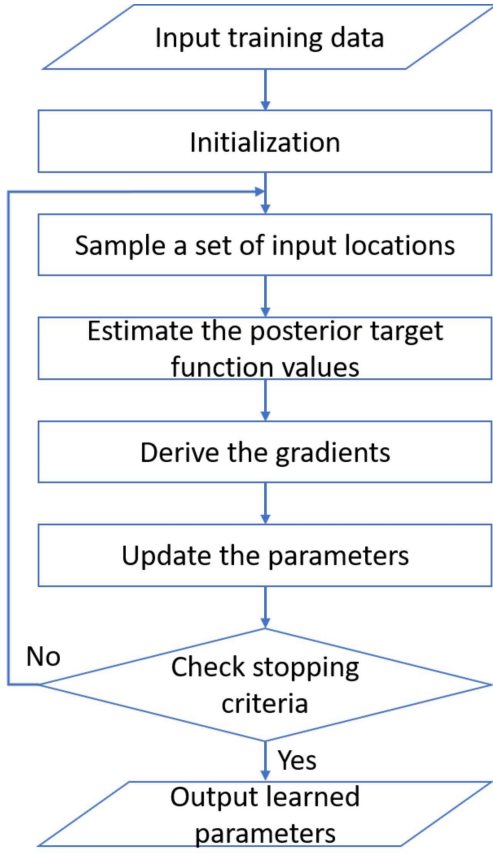


Fig. 3. The flowchart of the solution algorithm.

loss (maximizing the negative loss) via stochastic optimization shown in Fig. 3. Fig. 3 depicts the procedure of the posterior regularization algorithm for the model. In this algorithm, after the initialization step, the parameters and hyper-parameters are iteratively optimized till the prefixed number of loops are reached. In Fig.3, for each loop, sample a set of input locations $\mathbf{Z}$, estimate the posterior target function values $\hat{\mathbf{f}}(\mathbf{Z}) = \mathbf{K}_*^T(\mathbf{K} + \tau^{-1}\mathbf{I})^{-1}\mathbf{Y}$ in Eq. 8, $[\tilde{\mathcal{L}}_f, \tilde{\mathcal{L}}_g]^T$ with samples $(\mathbf{X}, \mathbf{Y})$, $(\mathbf{Z}, \hat{\mathbf{f}})$, then derive the gradients $\nabla_{\theta}\tilde{\mathcal{L}}$ of Eq. 30 and update the parameters $\theta^{(t+1)} = \theta^{(t)} + \phi\nabla_{\theta}\tilde{\mathcal{L}}$ via the unconstrained nonlinear optimization technique. Finally, the learned parameters θ are outputted. In this study, this algorithm is used as a submodule in the proposed gradual learning procedure. The temporal complexity of this algorithm is $\mathcal{O}(n^3 + m^3)$.

B. Learning With Gradual Physics Regularizer

The previous PRGP has great performance, however, it consumes a considerable computational time in the learning process. For example, the time complexity of the inference of the original GP is $\mathcal{O}(n^3)$ and the time complexity of the inference of the shadow GP is $\mathcal{O}(m^3)$. Thus, the total time complexity for the inference of two GPs is $\mathcal{O}(n^3 + m^3)$. Empirically, the computational time is also impacted by the output dimension d' , the number of equations $|g|$, and the number of iterations T . The empirical running time of training the two GPs $\mathcal{O}[d' * T * (n^3 + |g| * m^3)]$. For a traffic state estimation problem, the output dimension is fixed to three,

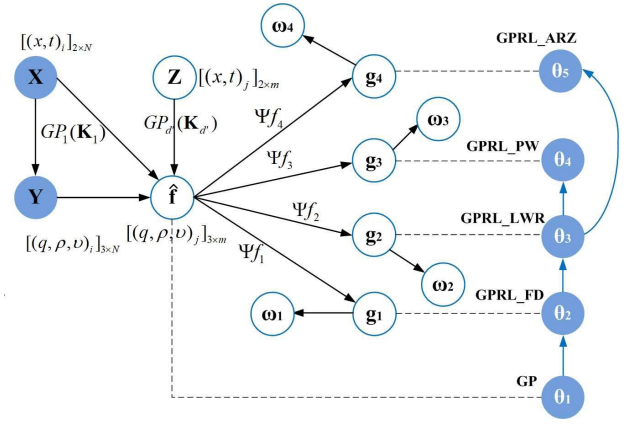


Fig. 4. The structure of gradual regularization process.

the number of iterations T depends on the trade-off between non-underfitting and non-overfitting. Specifically, considering the gradual development of the physical models of traffic state estimation problems, the hyperparameters of GP with the trained basic model (e.g. FD) can be fine-tuned with the high order models (e.g. LWR), then the result of high order models can be further used in higher-order models (e.g. PW and ARZ). Thus, the physical equations can be involved gradually instead of encoding all physical equations concurrently.

To reduce the computational cost, this paper proposes an approximation to learn high-order regularization models with the learned parameters of the lower-order models. The empirical running time of the approximation is reduced to $\mathcal{O}[d' * T/|g| * \sum_{i=1}^{|g|} (n^3 + i * m^3)]$.

Fig 4 shows the proposed GPRL which equips GP with gradual regularizers from different traffic flow models. The initialized parameters, θ_1 , are trained with pure GP, and then fine-tuned with FD models (θ_2), the first-order LWR model (θ_3), and the second-order PW and ARZ models (θ_4 and θ_5). Considering the excessive computational time of the numerical differentiation operation in the model equations, the training processes with less equations have less computational cost. Thus, the pre-training processes by lower order regularization models would yield similar performance with the reduced training duration.

To apply the proposed method, the traffic flow models need to be reformulated in the proposed framework. The pseudo observation pair $\mathbf{Z}, \omega$ has the same structure with the data observation pair $\mathbf{X}, \mathbf{Y}$, and is designed to encode the physical equations into GP.

Given the vector of remainder values, $\mathbf{g}$, of the physical equations, this method samples the pseudo observations ω from another GP. The predicted output $f(z)$ is substituted to the physical model to yield the right-hand side values $\mathbf{g}$. The physical equations are supposed to be in form of Eq. 16, where $\hat{\mathbf{f}}(\mathbf{Z})$ is the predicted outputs upon the input $\mathbf{Z}$ and $\Psi[\cdot]$ is a physical model function of the output. The physical equations are converted into the desired function forms by moving terms to one side of equation and let the other side be zero. Then, when the data perfectly meets with the physical model function, the remaining error $\mathbf{g}$ is supposed to be close to zero. Considering the unobserved latent value and the data

error, $\mathbf{g}$ is assumed to be a GP and the following equation is satisfied:

$$\Psi\hat{\mathbf{f}}(\mathbf{Z}) = \mathbf{g} \quad (16)$$

Then the differential equations $\Psi[\cdot]$ shown in Fig. 4 can be obtained from the four traffic flow models. For example, the fundamental diagrams, Ψf_1 , are reformulated in Eqs. 17-18.

$$v - V(\rho) = g_{11} \quad (17)$$

$$q - \rho V(\rho) = g_{12} \quad (18)$$

The stochastic mass conservation law is formulated in Eq. 19.

$$\Psi f_2(q, \rho, v) = \partial_t \rho + \partial_x q = g_2 \quad (19)$$

The stochastic PW momentum conservation law is formulated in Eq. 20.

$$\Psi f_3(q, \rho, v) = \partial_t v + v \partial_x v + \frac{V - V(\rho)}{\tau_0} + \frac{c_0^2}{\rho} \partial_x \rho = g_3 \quad (20)$$

And the stochastic ARZ momentum conservation law is formulated in Eq. 21.

$$\Psi f_4(q, \rho, v) = \partial_t (v - V(\rho)) + v \partial_x (v - V(\rho)) + \frac{v - V(\rho)}{\tau_0} = g_4 \quad (21)$$

where $g_1, \dots, g_4$ are right-hand side remainders. Note that one or more of these reformulated physical models can be used in the framework. More complex models can be learned based on the pre-learned base model.

The proposed procedure of gradual learning method for PRGP is described as follows.

Step 1. Learn the parameter θ_0 of the pure GP model. The ELBOs are defined as follows.

$$\mathcal{L}_{\mathbf{f}} = \inf \sum_i^{d'} \log[\mathcal{N}([\mathbf{Y}]_i | \omega, \hat{\mathbf{K}}_i + \tau^{-1} \mathbf{I})] \quad (22)$$

$$\mathcal{L}_{\mathbf{g}} = \inf[\mathcal{N}(\Psi\hat{\mathbf{f}}_w | \omega, \hat{\mathbf{K}}_w)] \quad (23)$$

where $\inf$ means infimum.

$$\theta_0 = [\tau_f, \eta_{f1}, \eta_{f2}, \eta_{f3}, \dots]^T \quad (24)$$

where the parameters $\tau_f, \eta_{f1}, \eta_{f2}, \eta_{f3}$ are initialized randomly or reused from a pre-trained model.

$$\theta_1 \leftarrow \theta_0 \quad (25)$$

$$\theta_1 \leftarrow \theta_1 + \phi \nabla \mathcal{L}_{\mathbf{f}} \quad (26)$$

The initial parameters are denoted as θ_0 . By θ_0 , the objective function log evidence lowerbound is calculated. The minimization is iteratively conducted by the general-purpose optimizers, such as ADAM [58].

Step 2. Gradual learning by fine-tuning the parameters

In the gradual learning procedure, the optimal value of $\theta_0, \theta_1, \theta_2, \theta_3, \theta_4$ is the partial initial value of $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$. The additional parameters for each step are initialized randomly if no further information is given.

Step 2.1 Reuse the same parameters with the learning parameters, and initialize the additional parameter randomly if no *a priori* information is given.

$$\theta_2 = [\theta_1, \tau_{g11}, \eta_{g111}, \eta_{g112}, \dots, \tau_{g12}, \eta_{g121}, \eta_{g122}, \dots]^T \quad (27)$$

where θ_1 is the learned parameter, and $\tau_{g11}, \eta_{g111}, \eta_{g112}, \tau_{g12}, \eta_{g121}, \eta_{g122}$ are additional parameters.

Step 2.2 Fine-tune the parameters with the inference algorithm presented in Section III-A. After the initialization, the following equations are used to update the parameters.

$$\theta_2 \leftarrow \theta_2 + \phi \nabla [\zeta \mathcal{L}_{\mathbf{f}} + \kappa \mathcal{L}_{\mathbf{g}_{11}} + \kappa \mathcal{L}_{\mathbf{g}_{12}}] \quad (28)$$

where $\mathcal{L}_{\mathbf{g}_{11}}, \mathcal{L}_{\mathbf{g}_{12}}$ are the ELBOs of the corresponding GPs $\mathbf{g}_{11}, \mathbf{g}_{12}$ on right-hand sides of Eqs. 17-18.

Step 3. If a higher level regularizer exists, initialize the additional parameters and fine-tune the parameters with the new regularizer. Else stop and output the final parameter.

The following equations are used in the iterations.

$$\theta_3 = [\theta_2, \tau_{g2}, \eta_{g21}, \eta_{g22}, \dots] \quad (29)$$

where θ_2 is the learned parameter, and $\tau_{g2}, \eta_{g21}, \eta_{g22}$ are additional parameters.

$$\theta_3 \leftarrow \theta_3 + \phi \nabla [\zeta \mathcal{L}_{\mathbf{f}} + \zeta \mathcal{L}_{\mathbf{g}_{11}} + \zeta \mathcal{L}_{\mathbf{g}_{12}} + \kappa \mathcal{L}_{\mathbf{g}_2}] \quad (30)$$

where $\mathcal{L}_{\mathbf{g}_2}$ corresponds to the GP $\mathbf{g}_2$ on the right-hand sides of Eq. 19.

$$\theta_4 = [\theta_3, \tau_{g3}, \eta_{g31}, \eta_{g32}, \dots] \quad (31)$$

where θ_3 is the learned parameter, and $\tau_{g3}, \eta_{g31}, \eta_{g32}$ are additional parameters.

$$\theta_4 \leftarrow \theta_4 + \phi \nabla [\zeta \mathcal{L}_{\mathbf{f}} + \zeta \mathcal{L}_{\mathbf{g}_{11}} + \zeta \mathcal{L}_{\mathbf{g}_{12}} + \kappa \mathcal{L}_{\mathbf{g}_2} + \kappa \mathcal{L}_{\mathbf{g}_3}] \quad (32)$$

where $\mathcal{L}_{\mathbf{g}_3}$ is related to the GP $\mathbf{g}_3$ on the right-hand sides of Eq. 20.

$$\theta_5 = [\theta_4, \tau_{g4}, \eta_{g41}, \eta_{g42}, \dots] \quad (33)$$

where θ_4 is the learned parameter, and $\tau_{g4}, \eta_{g41}, \eta_{g42}$ are additional parameters.

$$\theta_5 \leftarrow \theta_5 + \phi \nabla [\zeta \mathcal{L}_{\mathbf{f}} + \zeta \mathcal{L}_{\mathbf{g}_{11}} + \zeta \mathcal{L}_{\mathbf{g}_{12}} + \zeta \mathcal{L}_{\mathbf{g}_2} + \kappa \mathcal{L}_{\mathbf{g}_4}] \quad (34)$$

where $\mathcal{L}_{\mathbf{g}_4}$ is related to the GP $\mathbf{g}_4$ on the right-hand sides of Eq. 21.

Assuming the posterior distribution $M(\theta_{old}) = p(\mathbf{f}(\mathbf{x}^*) | \mathbf{x}^*, \mathcal{E}_{old})$ is inferred from the equation $\mathcal{E}_{old}$. The model can be updated every time step based on the previous model M_{old} and the supplemental equation $\mathcal{E}_{new}$ as shown in Eq. 35.

$$\theta_{new} \leftarrow \theta_{old} + \alpha \nabla \mathcal{L}(\mathcal{E}_{new}) \quad (35)$$

The gradual learning procedure is summarized as follows. In the initial learning stage, the minimal set of behaviors can be involved in inferring the posterior $M(\theta_{old})$. After the initial model is trained, the additional behavior $\Psi \mathbf{f}_{new}(\mathbf{x}) = \mathbf{g}_{new}(\mathbf{x})$ is supplemented to enhance the existing model. The new model $M(\theta_{new})$ can be yield by maximizing the updated evidence

lowerbound of the log-likelihood $\theta_{new} = \operatorname{argmax} \mathcal{L}_{new} \approx \operatorname{argmax} \Delta \mathcal{L}_{new}$, which is shown in Eq. 36.

$$\Delta \mathcal{L}_{new} = \gamma_{new} \mathbb{E}_{p(\mathbf{z})} \mathbb{E}_{p(\hat{\mathbf{f}}_{new}|\mathbf{z}, E)} [\log[\mathcal{N}(\Psi \hat{\mathbf{f}}_{new} | \omega, \hat{\mathbf{K}}_{new})]] \quad (36)$$

To reinforce new physical knowledge in multiple stages, the parameters are updated based on the gradient of the weighted sum of the likelihood from new equation and additional regularization term.

$$\theta_{new} \leftarrow \theta_{old} + \phi \nabla [\zeta \mathcal{L}_{old}(\mathcal{E}_{new}) + \kappa \Delta \mathcal{L}_{new}(\mathcal{E}_{old} \cup \mathcal{E}_{new})] \quad (37)$$

where ϕ is the iterative step length, κ, ζ are weights for balancing the old physical equations with the new equation and using the new physical knowledge with both new and old equations.

IV. EXPERIMENT TEST

To evaluate the performance of the proposed method, this study selects a stretch of the interstate freeway I-15 across Utah, U.S. (see Fig. 5), as the study site, for traffic flow and speed estimations. Along the I-15 corridor, the Utah Department of Transportation (UDOT) has installed sensors every a few miles along the freeway. Each sensor counts the number of vehicles passed, measures the speed of each vehicle, and sends the data back to a central database, called Performance Measurement System (PeMS). The collected real-time data and road conditions are available online and can be accessed by users with granted access. The data used for evaluating the proposed models were collected from August 5, 2019 to August 19, 2019. Since both flow and speed data are aggregated with an interval of 5 min, there are 288 observations, per detector per day, for both flow and speed. The input variables include the location mileage of each sensor and the time of each read. In the literature, the data index representation $(\mathbf{X}, \mathbf{Y})$ has three major variations: (road segment, time interval), (road segment, day, time interval), and (road segment, week, day-of-week, time interval). In the experiments, the compatible representation (road segment, time interval), namely (i, k) is used for consistent purpose. The traffic measures, flow q and speed v , are employed in the training and testing because the density is directly related to these two measures and is not recorded in the original data source.

The model parameters are initialized randomly if no additional information is given. The physics-model parameters can be initialized by calibrated values, and are updated during the training of the model. The GP-related parameters can be initialized by the previous training procedure. In the experiments, the gradual learning is robust regarding the specific iteration for shifting from a lower-level regularier to a higher-level regularier. In this paper, an empirical scheduler for regularizer shifting is used in the experiments. This scheduler shifts the learning regularier from lower-level to higher-level by the rule-of-thumb. The rule is to shift the regularizer when the relative improvement in a few iteration is smaller than a threshold.

The training and testing data are presented in Fig. 5. Data from three sensors, S412, S414, and S420, are selected for

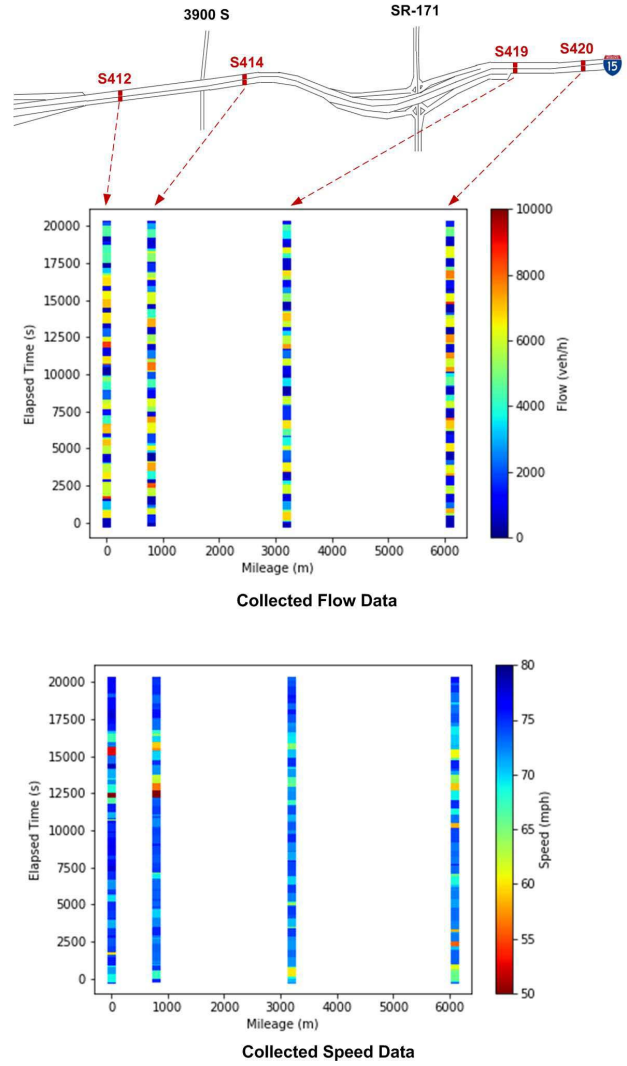


Fig. 5. The stretch of the I-15 and collected data.

training, and data from the other sensor, S419, are used for testing.

A. Evaluation of the Gradual Physics Regularizer

To evaluate the effectiveness of the proposed gradual physics regularizer, this subsection will test the performance of two models:

- **GPRL_PW**, where the GPRL framework is integrated PW and the base model, GP, is sequentially regularized by FD, LWR, and PW; and
- **GPRL_ARZ**, where the GPRL framework is integrated ARZ and the base model, GP, is sequentially regularized by FD, LWR, and ARZ.

The training process of both models, indicated by the number of solution iterations (see Fig. 3) and the objective value of Log ELBO (see Eq. 14), are shown in Fig. 6. Herein, it took about 140 iterations to complete the training of the base model, GP, and the corresponding objective value has been reduced to -1.80×10^3 . Then, by applying the gradual physics regularizers, from FD, LWR, PW, and ARZ, the training objective value was further minimized and both GPRL_PW and GPRL_ARZ

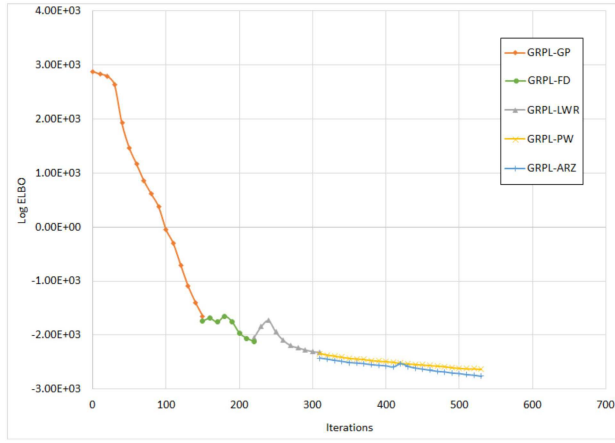


Fig. 6. Illustration of the training process of each model.

TABLE I
MODEL PERFORMANCE IMPROVEMENT BY THE GRADUAL
PHYSICS REGULARIZERS

Method	Flow RMSE (veh/5min)	Flow MAPE	Speed RMSE (mph)	Speed MAPE
GP	43.39	15.78%	3.76	3.64%
GPRL_FD	41.16	14.67%	3.32	2.70%
GPRL_LWR	39.36	13.60%	3.00	2.56%
GPRL_PW	38.91	13.49%	3.08	2.64%
GPRL_ARZ	38.37	13.30%	2.91	2.58%

converged after 530 iterations. Moreover, Fig. 6 also indicated that the integration of higher-order traffic flow models is based on the completion of encoding lower-order models. Such GRML process can greatly reduce the required solution iterations for both GPRL_PW and GPRL_ARZ and corresponding computational times are consequently decreased, considering that one iteration of GPRL_PW or GPRL_ARZ could consume much longer time than that of GPRL with lower-order models.

Besides the analysis of the training process, it is also important to study how the physics regularizers can gradually improve the model performance on the testing dataset. In this study, Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) are selected as the performance evaluation indicators:

$$RMSE_j = \sqrt{\frac{1}{N} \sum_{i=1}^N ([y_j]_i - [\hat{f}_j]_i)^2}, \quad \forall j \in 1, \dots, d' \quad (38)$$

$$MAPE_j = \frac{100\%}{N} \sum_{i=1}^N \left| \frac{[y_j]_i - [\hat{f}_j]_i}{[y_j]_i} \right|, \quad \forall j \in 1, \dots, d' \quad (39)$$

Table I shows the obtained RMSE and MAPE of both flow and speed estimations by each model. Based on the results, it can be observed that the model performance has been gradually improved after encoding a higher-order traffic flow model at each stage. Compared with GPRL_PW, GPRL_ARZ can yield more accurate estimations on both flows and speeds as the ARZ model can outperform the PW model, as shown in the literature, while both of them are classified as the second-order traffic flow models. Moreover, the shown results also highlight one main research contribution of the proposed

TABLE II
MODEL PERFORMANCE UNDER STATIONARY AND
NON-STATIONARY CONDITIONS

Scenario	Flow RMSE (veh/5min)	Flow MAPE	Speed RMSE (mph)	Speed MAPE
stationary without noise	36.91	14.70%	1.47	1.48 %
non-stationary without noise	60.12	14.47%	7.86	9.76 %
stationary with noise	42.69	12.68%	1.78	1.58%
non-stationary with noise	67.42	17.34%	8.78	11.54 %

ML framework, making the results more explainable and the model itself is no longer a “black box”. However, one may note that the improvement from GP to GPRL_PW or GPRL_ARZ is not significant. The main reason is that the collected dataset is sufficiently large and the pure ML model, GP, can already achieve high estimation accuracy. Hence, to further prove the benefit of encoding physical knowledge into ML, a robustness study will be conducted in subsection C.

When using as FD the physics model, the GPRL_FD is trained with partial parameters of GPRL_LWR. Thereafter, the GPRL_LWR with random initialized additional parameters can converge faster than training PRGP_LWR from scratch (all randomly initialized parameters). In the PRGP method, the FD can be used as a soft constrain on the relation between the flow, speed, and density at a specific input (time and location), and is not required to capture the equilibrium state of traffic.

Table II presents the performance metrics of FD results in non-stationary and stationary situations, where the 15% percentile of the speed is the threshold for the stationary and non-stationary conditions. The results show the GPRL FD method performs worse in the non-stationary than in the stationary conditions, and the data noise has more negative impacts on the non-stationary estimations than the stationary conditions.

Fig 7 shows the estimation of whole freeway segment, and the wave propagation and congestion growths are observed.

B. Model Performance Comparisons

Recall that one main motivation of introducing the gradual physics regularizer is to reduce the required computational time when dealing with complex physical models. Hence, in this subsection, the performance of the two GPRL models are further compared with two baseline models that directly encode the PW and ARZ models into GP:

CPU *PRML_PW*, where the PW model is encoded without the gradual regularizing process; and *PRML_ARZ*, where the ARZ model is encoded without the gradual regularizing process.

Also, besides the RMSE and MAPE, the consumed computational time is listed as the third evaluation indicator. Note that all models were carried out on a workstation with a 3.9GHz 8-core CPU 16GB RAM and RTX2080Ti GPU accelerator.

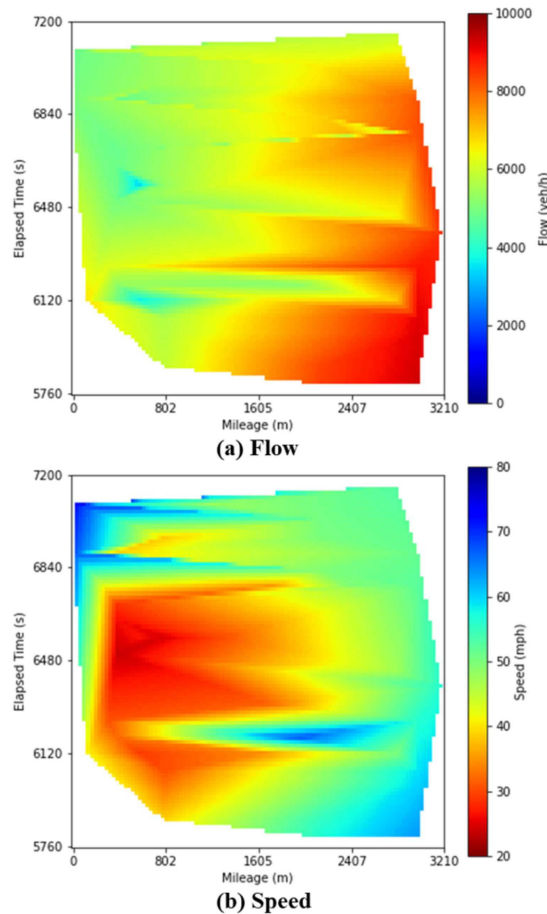


Fig. 7. One-day estimation plot.

Based on the model results summarized in Table III, it can be observed that the proposed GPRL_PW model took 5,225 seconds of computational time when the baseline PRML_PW consumed 25,075 seconds. The reduction is about 78.2%. Meanwhile, GPRL_PW's performance on the RMSE and MAPE of flow and speed estimations are quite closed to that of PRML_PW. A similar trend can be observed when comparing GPRL_ARZ (6,817 seconds) with PRML_ARZ (34,085 seconds) and the reduction in computation time is about 80%. In summary, it can be concluded that the proposed GPRL framework can significantly improve the computational efficiency, when the GP is dealing with complex physical models, and still guarantee a satisfying model performance. The piled up structure can reduce the computational cost by reusing trained parameters in the higher level regularizers. For example, when encoding FD as the physics regularizer, the GPRL_FD is trained with partial parameters of GPRL_LWR. Thereafter, the GPRL_LWR with random initialized additional parameters can converge faster than training PRGP_LWR from scratch (i.e. all randomly initialized parameters).

C. Model Robustness Analysis

As shown in the Subsections A and B, the improvement of the proposed models, compared with the GP, on both flow

TABLE III
COMPARISON OF THE RESULTS OF THE PROPOSED METHOD
AND THE BASELINE METHODS

Method	Flow RMSE	Flow MAPE	Speed RMSE	Speed MAPE	Duration (s)
GPRL_PW	38.91	13.90%	3.08	2.64%	5,225
PRML_PW	38.19	12.70%	2.96	2.61%	25,075
GPRL_ARZ	38.37	13.30%	2.91	2.58%	6,817
PRML_ARZ	38.23	12.72%	2.89	2.57%	34,085

TABLE IV
MODEL PERFORMANCE IMPROVEMENT BY THE GRADUAL PHYSICS
REGULARIZERS WITH NOISY TRAINING DATA

Method	RMSE flow (veh/5min)	MAPE flow	RMSE speed (mph)	MAPE speed
GP	64.93	21.64%	4.76	3.64%
GPRL_FD	47.26	13.38%	3.79	3.09%
GPRL_LWR	46.03	12.08%	4.45	3.27%
GPRL_PW	42.65	10.80%	4.43	3.34%
GPRL_ARZ	42.60	10.70%	4.55	3.40%

and speed estimation is not significant. The main reason is that the quality of the collected data is at the acceptable level and the data size is sufficiently large for training pure ML models. However, it should be noted that data noises, e.g., caused by malfunctioning traffic sensors, are commonly existed in practice and many noisy data are hard to identify using conventional data screening algorithms. Therefore, pure ML models, such as the GP, would have very limited resistance to such high flawed dataset.

By encoding valuable physical knowledge from traffic flow models into the ML process, it is expected that the proposed models would be much more robust to noises in the training dataset as those flawed data would violate the physical laws defined in the traffic flow models. Hence, this subsection aims to further evaluate the model robustness performance. To such needs, the noisy data scenarios are created by artificially adding high measure errors, with the mean of 100 *veh/5min*, to 25% of traffic flows the training data to mimic the common device malfunction situations. The testing set is not polluted original data. Then all models were developed based on the created noisy training dataset and tested on the original testing dataset.

Table IV summarizes the resulting RMSE and MAPE of both flow and speed estimates by different models. Similar to the results shown in Table I, it can be observed that the applied physics regularizers, obtained from different orders of traffic flow models, can gradually improve the GPRL's estimation accuracy. Moreover, it should be noted that both GPRL_PW and GPRL_ARZ can greatly outperform GP by about 22 *veh/5min* of RMSE and over 11% of MAPE in flow estimations. This is due to GPRL's capability of adopting physical knowledge to regularized the ML training process. The difference in speed estimation is not quite obvious since no noise is added to the speed data. For better illustration, Fig. 8(a) and (b) present the time-dependent distribution of the estimation absolute percentage error (APE) during day time (6 AM-6 PM) and nighttime (6 PM-6 AM), respectively, in one of the studied days. As shown in Fig. 8(a), in the

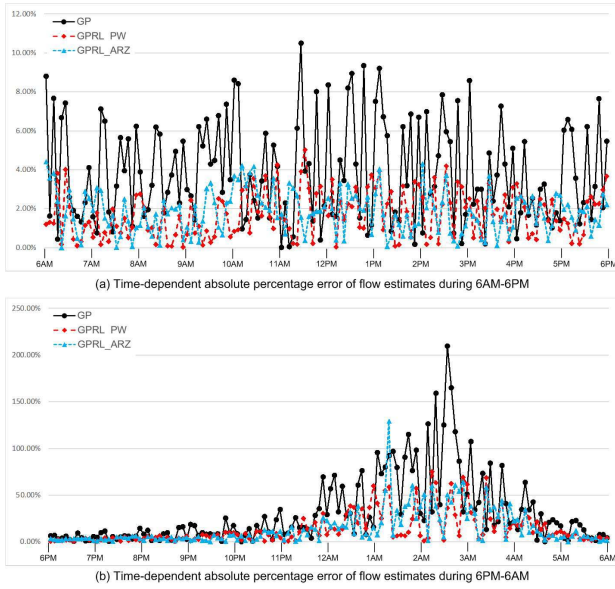


Fig. 8. Distribution of flow estimation errors.

TABLE V

MODEL PERFORMANCE COMPARISON WITH NOISY TRAINING DATASET

Method	RMSE flow	MAPE flow	RMSE speed (mph)	MAPE speed	Duration (s)
GPRL_PW	42.65	10.70%	4.43	3.34%	5,930
PRML_PW	42.61	10.69%	4.42	3.33%	30,200
GPRL_ARZ	42.60	10.80%	4.55	3.40%	7,805
PRML_ARZ	42.35	10.77%	4.46	3.40%	39,025

day time period, most APEs yield by the GP are between 4% and 10% while the APEs produced by both GPRL_PW and GPRL_ARZ below 4%. During the midnight period (11 AM-5 AM), a major group of APEs generated by the GP can go up to 80%-100%. This is due to the relatively low flow level during that time period. In Fig. 8(b), similar trends can be observed but the APEs by the GP is much smaller. Hence, it can be concluded that the proposed modeling framework are much more robust than the pure ML models when the input data is subject to unobserved random noise.

To quantify the benefit of the proposed GPRL process on reducing computational time when the training dataset is noisy, Table V further compares the two GPRL models' performance with the two PRML models. The results show that the proposed GPRL_PW model can reduce about 81.4% of computational time and achieve a similar accuracy level on flow and speed estimations, compared with the baseline model, PRML_PW. Meanwhile, GPRL_ARZ can contribute an 80% computational time reduction compared with the PRML_ARZ. Hence, it can be proved that the proposed GPRL framework can also improve the computational efficiency greatly when the training data contains noises and achieve similar model performance as the PRML that directly encode the complex traffic flow models.

To test the impact of the noise level on the performance, the experiment of the flaw levels 0.3, 0.5, 0.7 are tested in Table VI. The higher level noise has more negative impacts on the performance. The results show the proposed method is resistant to various noise levels.

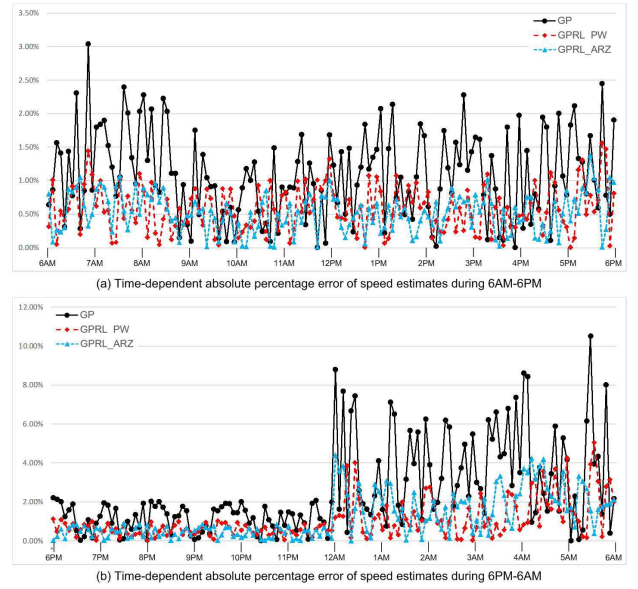


Fig. 9. Distribution of speed estimation errors.

TABLE VI

COMPARISON OF THE RESULTS OF THE PROPOSED GPRL_ARZ AND THE BASELINE METHODS UNDER VARIOUS NOISE LEVELS

Noise level	Noise intensity	Flow RMSE (veh/5min)	Flow MAPE	Speed RMSE	Speed MAPE (mph)
0.3	100	41.32	10.45%	4.13	3.24 %
0.5	100	42.60	10.70%	4.55	3.40 %
0.7	100	44.01	13.14%	5.30	4.69 %
0.5	50	40.01	10.01%	4.32	3.55 %
0.5	150	43.60	11.38%	4.83	3.61 %

V. CONCLUSION AND FUTURE RESEARCH

In the literature, few studies have discussed how to leverage multiple traffic models to improve the performance of data-driven approaches for traffic state estimation. By encoding the traffic flow knowledge into the ML framework, the proposed physics regularized machine learning framework is expected to outperform pure ML methods but may require more computational efforts at the same time. To address this issue, this paper aims to find low computational cost method to improve estimation accuracy of highway traffic. This paper presents an gradual learning method by leveraging Gaussian Process (GP) to capture the randomness and correlation of outputs. A posterior regularization framework is used to estimate GP parameters by appropriately fusing multiple macroscopic traffic flow models. The macroscopic traffic flow model equations are encoded as GPs and are capable to handle the unobserved variables and randomness. Considering the limit of memory and the constantly arriving data stream, the gradual learning method is proposed to leverage additional equations and datasets.

A case study is conducted in the real-world detector data in I-15 in Utah. The macroscopic traffic models, including the triangular FD, the LWR model, the PW model, and the ARZ model are tested in the proposed framework. The results show the gradual model outputs similar performance metrics and much shorter computational time in comparison to the

non-gradual models, and the more physical equations are used, the better performance is yielded in the proposed method. Thus, the macroscopic traffic models can be regularized gradually to create an efficient hybrid model. The effectiveness of the proposed gradual method is justified.

Future research directions along the line of the proposed models include: 1) extend the GPRL framework to encode different types of physical models so as to conduct the ensemble learning; and 2) modify the proposed model to address the streaming dataset for incremental learning.

REFERENCES

- [1] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: Where we are and where we're going," *Transp. Res. C, Emerg. Technol.*, vol. 43, pp. 3–19, Jun. 2014.
- [2] M. J. Lighthill and G. B. Whitham, "On kinematic waves II. A theory of traffic flow on long crowded roads," *Proc. Roy. Soc. London A. Math. Phys. Sci.*, vol. 229, no. 1178, pp. 317–345, 1955.
- [3] P. I. Richards, "Shock waves on the highway," *Oper. Res.*, vol. 4, no. 1, pp. 42–51, 1956.
- [4] H. J. Payne, "Models of freeway traffic and control," *Math. Models Public Syst.*, vol. 1, no. 1, pp. 51–61, 1971.
- [5] G. B. Whitham, *Linear and Nonlinear Waves*. Hoboken, NJ, USA: Wiley, 1974.
- [6] A. Aw and M. Rascle, "Resurrection of 'second order' models of traffic flow," *SIAM J. Appl. Math.*, vol. 60, no. 3, pp. 916–938, 2000.
- [7] H. M. Zhang, "A non-equilibrium traffic model devoid of gas-like behavior," *Transp. Res. B, Methodol.*, vol. 36, no. 3, pp. 275–290, 2002.
- [8] D. C. Gazis and C. H. Knapp, "On-line estimation of traffic densities from time-series of flow and speed data," *Transp. Sci.*, vol. 5, no. 3, pp. 283–301, Aug. 1971.
- [9] M. W. Szeto and D. C. Gazis, "Application of Kalman filtering to the surveillance and control of traffic systems," *Transp. Sci.*, vol. 6, no. 4, pp. 419–439, Nov. 1972.
- [10] D. Gazis and C. Liu, "Kalman filtering estimation of traffic counts for two network links in tandem," *Transp. Res. B, Methodol.*, vol. 37, no. 8, pp. 737–745, Sep. 2003.
- [11] Y. Wang and M. Papageorgiou, "Real-time freeway traffic state estimation based on extended Kalman filter: A general approach," *Transp. Res. B, Methodol.*, vol. 39, no. 2, pp. 141–167, 2005.
- [12] Y. Wang, M. Papageorgiou, and A. Messmer, "Real-time freeway traffic state estimation based on extended Kalman filter: A case study," *Transp. Sci.*, vol. 41, no. 2, pp. 167–181, May 2007.
- [13] I. Prigogine and R. Herman, *Kinetic Theory of Vehicular Traffic*. Cambridge, MA, USA: Amer. Elsevier Pub. Co, 1971.
- [14] S. L. Paveri-Fontana, "On Boltzmann-like treatments for traffic flow: A critical review of the basic model and an alternative proposal for dilute traffic analysis," *Transp. Res.*, vol. 9, no. 4, pp. 225–235, Aug. 1975.
- [15] X. Di, H. X. Liu, and G. A. Davis, "Hybrid extended Kalman filtering approach for traffic density estimation along signalized arterials: Use of global positioning system data," *Transp. Res. Rec.*, vol. 2188, no. 1, pp. 165–173, 2010.
- [16] C. Osorio, G. Flötteröd, and M. Bierlaire, "Dynamic network loading: A stochastic differentiable model that derives link state distributions," *Proc. Social Behav. Sci.*, vol. 17, pp. 364–381, Jan. 2011.
- [17] S. Jabari and H. X. Liu, "A stochastic model of traffic flow: Theoretical foundations," *Transp. Res. B, Methodol.*, vol. 46, no. 1, pp. 156–174, Jan. 2012.
- [18] A. Sopasakis and M. Katsoulakis, "Stochastic modeling and simulation of traffic flow: Asymmetric single exclusion process with Arrhenius look-ahead dynamics," *SIAM J. Appl. Math.*, vol. 66, no. 3, pp. 921–944, 2006.
- [19] A. Sopasakis, "Lattice free stochastic dynamics," *Commun. Comput. Phys.*, vol. 12, no. 3, pp. 691–702, Sep. 2012.
- [20] M. Neumann, K. Kersting, Z. Xu, and D. Schulz, "Stacked Gaussian process learning," in *Proc. 9th IEEE Int. Conf. Data Mining*, Dec. 2009, pp. 387–396.
- [21] Y. Xie, K. Zhao, Y. Sun, and D. Chen, "Gaussian processes for short-term traffic volume forecasting," *Transp. Res. Rec.*, vol. 2165, no. 1, pp. 69–78, 2010.
- [22] T. Idé and S. Kato, "Travel-time prediction using Gaussian process regression: A trajectory-based approach," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, Apr. 2009, pp. 1185–1196.
- [23] A. Armand, D. Filliat, and J. Ibanez-Guzman, "Modelling stop intersection approaches using Gaussian processes," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2013, pp. 1650–1655.
- [24] S. Liu, Y. Yue, and R. Krishnan, "Adaptive collective routing using Gaussian process dynamic congestion models," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2013, pp. 704–712.
- [25] J. Zhao and S. Sun, "High-order Gaussian process dynamical models for traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 7, pp. 2014–2019, Jul. 2016.
- [26] T. V. Le, R. Oentaryo, S. Liu, and H. C. Lau, "Local Gaussian processes for efficient fine-grained traffic speed prediction," *IEEE Trans. Big Data*, vol. 3, no. 2, pp. 194–207, Jun. 2017.
- [27] X. Jia *et al.*, "Physics guided recurrent neural networks for modeling dynamical systems: Application to monitoring water temperature and quality in lakes," 2018, *arXiv:1810.02880*.
- [28] J. Wang, J. Wu, and H. Xiao, "A physics-informed machine learning approach of improving RANS predicted Reynolds stresses," in *Proc. 55th AIAA Aerosp. Sci. Meeting*, Jan. 2017, p. 1712.
- [29] R. Shi, Z. Mo, K. Huang, X. Di, and Q. Du, "A physics-informed deep learning paradigm for traffic state and fundamental diagram estimation," *IEEE Trans. Intell. Transp. Syst.*, early access, Sep. 8, 2021, doi: [10.1109/TITS.2021.3106259](https://doi.org/10.1109/TITS.2021.3106259).
- [30] M. Barreau, J. Liu, and K. H. Johansson, "Learning-based state reconstruction for a scalar hyperbolic PDE under noisy Lagrangian sensing," in *Proc. 3rd Conf. Learn. Dyn. Control*, vol. 144. PMLR, 2021, pp. 34–46.
- [31] Z. Wang, W. Xing, R. Kirby, and S. Zhe, "Physics regularized Gaussian processes," 2020, *arXiv:2006.04976*.
- [32] Y. Yuan, Z. Zhang, X. T. Yang, and S. Zhe, "Macroscopic traffic flow modeling with physics regularized Gaussian process: A new insight into machine learning applications in transportation," *Transp. Res. B, Methodol.*, vol. 146, pp. 88–110, Apr. 2021.
- [33] Y. Yuan, Z. Zhang, and X. Terry Yang, "Highway traffic state estimation using physics regularized Gaussian process: Discretized formulation," 2020, *arXiv:2007.07762*.
- [34] B. D. Greenshields, J. R. Bibbins, W. S. Channing, and H. H. Miller, "A study of traffic capacity," in *Highway Research Board Proceedings*, vol. 1935. USA: National Research Council, Highway Research Board, 1935.
- [35] H. Greenberg, "An analysis of traffic flow," *Oper. Res.*, vol. 7, no. 1, pp. 79–85, Feb. 1959.
- [36] R. T. Underwood, *Speed, Volume, and Density Relationships*. State College, PA, USA: Pennsylvania State Univ., 1960.
- [37] J. Drake and J. L. Schofer, "A statistical analysis of speed-density hypotheses," *Highway Res. Rec.*, vol. 154, pp. 53–87, Jul. 1966.
- [38] D. R. Drew, *Deterministic Aspects Freeway Operations Control*. Austin, TX, USA: Texas Transportation Institute College Station, 1965.
- [39] L. A. Pipes, "Car following models and the fundamental diagram of road traffic," *Transp. Res.*, vol. 1, pp. 21–29, 1967, doi: [10.1016/0041-1647\(67\)90092-5](https://doi.org/10.1016/0041-1647(67)90092-5).
- [40] P. K. Munjal and L. A. Pipes, "Propagation of on-ramp density perturbations on unidirectional two- and three-lane freeways," *Transp. Res.*, vol. 5, no. 4, pp. 241–255, 1971, doi: [10.1016/0041-1647\(71\)90036-0](https://doi.org/10.1016/0041-1647(71)90036-0).
- [41] L. C. Edie, "Car-following and steady-state theory for noncongested traffic," *Oper. Res.*, vol. 9, no. 1, pp. 66–76, Feb. 1961.
- [42] A. D. May, *Traffic Flow Fundamentals*. Eaglewood Cliffs, NJ, USA: Prentice-Hall, 1990.
- [43] H. Wang, J. Li, Q.-Y. Chen, and D. Ni, "Logistic modeling of the equilibrium speed-density relationship," *Transp. Res. A, Policy Pract.*, vol. 45, no. 6, pp. 554–566, Jul. 2011.
- [44] H. Wang, D. Ni, Q.-Y. Chen, and J. Li, "Stochastic modeling of the equilibrium speed-density relationship," *J. Adv. Transp.*, vol. 47, no. 1, pp. 126–150, Jan. 2013.
- [45] M. Papageorgiou, J.-M. Blosseville, and H. Hadj-Salem, "Macroscopic modelling of traffic flow on the Boulevard Périphérique in Paris," *Transp. Res. B, Methodol.*, vol. 23, no. 1, pp. 29–47, 1989.
- [46] J. M. Del Castillo, P. Pintado, and F. G. Benitez, "The reaction time of drivers and the stability of traffic flow," *Transp. Res. B, Methodol.*, vol. 28, no. 1, pp. 35–60, Feb. 1994.
- [47] C. F. Daganzo, "Requiem for second-order fluid approximations of traffic flow," *Transp. Res. B, Methodol.*, vol. 29, no. 4, pp. 277–286, Aug. 1995.
- [48] M. Papageorgiou, "Some remarks on macroscopic traffic flow modelling," *Transp. Res. A, Policy Pract.*, vol. 32, no. 5, pp. 323–329, 1998.

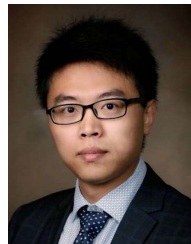
- [49] S. P. Hoogendoorn and P. H. L. Bovy, "State-of-the-art of vehicular traffic flow modelling," *Proc. Inst. Mech. Eng., I, J. Syst. Control Eng.*, vol. 215, no. 4, pp. 283–303, 2001.
- [50] R. M. Colombo, "Hyperbolic phase transitions in traffic flow," *SIAM J. Appl. Math.*, vol. 63, no. 2, pp. 708–721, Jan. 2003.
- [51] J.-P. Lebacque, S. Mammar, and H. H. Salem, "Generic second order traffic flow modelling," presented at the Transp. Traffic Theory Papers Sel. (ISTTT), London, U.K., Jul. 2007, pp. 755–776.
- [52] S. Blandin, J. Argote, A. M. Bayen, and D. B. Work, "Phase transition model of non-stationary traffic flow: Definition, properties and solution method," *Transp. Res. B, Methodol.*, vol. 52, pp. 31–55, Jun. 2013.
- [53] S. Fan, M. Herty, and B. Seibold, "Comparative model accuracy of a data-fitted generalized Aw-Rascle-Zhang model," 2013, *arXiv:1310.8219*.
- [54] A. Sumalee, R. X. Zhong, T. L. Pan, and W. Y. Szeto, "Stochastic cell transmission model (SCTM): A stochastic dynamic traffic model for traffic state surveillance and assignment," *Transp. Res. B, Methodol.*, vol. 45, no. 3, pp. 507–533, 2011.
- [55] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer School on Machine Learning*. Springer, 2003, pp. 63–71.
- [56] F. Rodrigues and F. C. Pereira, "Heteroscedastic Gaussian processes for uncertainty modeling in large-scale crowdsourced traffic data," *Transp. Res. C, Emerg. Technol.*, vol. 95, pp. 636–651, Oct. 2018.
- [57] F. Rodrigues, K. Henrickson, and F. C. Pereira, "Multi-output Gaussian processes for crowdsourced traffic data imputation," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 2, pp. 594–603, Feb. 2019.
- [58] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.



Yun Yuan received the B.S. and M.S. degrees in transportation engineering from Nanjing Agricultural University, Nanjing, China, in 2012 and 2015, respectively, and the Ph.D. degree from the Department of Civil and Environmental Engineering, University of Wisconsin–Milwaukee. He is currently an Associate Professor with the College of Transportation Engineering, Dalian Maritime University. Previously, he is a Post-Doctoral Research Associate at The University of Utah, USA. His research interests include transportation system modeling and machine learning techniques, deep reinforcement learning, and applications of optimization and simulation methods.



Qinzhen Wang received the B.S. degree in transportation engineering from Fujian Agriculture and Forestry University, Fuzhou, China, in 2014, and the M.S. degree in transportation planning and management from Beijing Jiaotong University, Beijing, China, in 2018. He is currently pursuing the Ph.D. degree with the Department of Civil and Environmental Engineering, The University of Utah, Salt Lake City, UT, USA. His research interest includes traffic control under a connected automated vehicle environment.



Xianfeng Terry Yang received the B.S. degree in civil engineering from Tsinghua University in 2009, and the M.S. and Ph.D. degrees in civil engineering from the University of Maryland, College Park, MD, USA, in 2011 and 2015, respectively.

His research has been funded by multiple agencies, such as the National Science Foundation, the U.S. Department of Transportation, the Department of Energy, Federal Highway Administration, and the Utah Department of Transportation. He has published over 100 peer-reviewed research articles in journals and conferences. His current research interests include machine learning for transportation modeling, traffic operations with connected automated vehicles, traffic safety, transportation equity, and transportation planning. He is the appointed member of two TRB standing committees. He was a recipient of the prestigious NSF CAREER Award in 2021. He is currently the Editorial Board Member of *Transportation Research Part C*, the Associate Editor of *ASCE Journal of Urban Planning and Development* and IEEE OPEN JOURNAL INTELLIGENT TRANSPORTATION SYSTEMS, and the Handling Editor of *TRB Transportation Research Record*. He is also the Chair of INFORMS JST ITS Committee and the Secretary of the ASCE Artificial Intelligence Committee.