Traffic prediction by combining macroscopic models and Gaussian processes

Alexandra Würth, Mickael Binois, Paola Goatin

Traffic prediction by combining macroscopic models and Gaussian processes

Alexandra Würth, Mickaël Binois, Paola Goatin

***Abstract*—We propose a physics informed statistical framework for traffic travel time prediction. On one side, the discrepancy of the considered mathematical model is represented by a Gaussian process. On the other side, the traffic simulator is fed with boundary data predicted by a Gaussian process, forced to satisfy the mathematical equations at virtual points, resulting in a multi-objective optimization problem. This combined approach has the merit to address the shortcomings of the purely model-driven or data-driven approaches, while leveraging their respective advantages. Indeed, models are based on physical laws, but cannot capture all the complexity of real phenomena. On the other hand, pure statistical outputs can violate basic characteristic dynamics. We validate our approach on both synthetic and real world data, showing that it delivers more reliable results compared to other methods.**

***Index Terms*—Macroscopic traffic flow models, Godunov scheme, parameter calibration, Gaussian process modeling, loop detector and trajectory data, travel time prediction.**

I. Introduction

Macroscopic traffic flow models are employed since several decades for state reconstruction and prediction. They consist in partial differential equations (PDEs), whose solutions can be computed by numerical schemes. Their implementation requires information provided by real data, which are typically measured by magnetic loop detectors at fixed locations. Real data are also necessary to identify model parameters. However, mathematical models may fail in capturing involved traffic situations such as congestion, lane closures or accidents, even if the true values of the calibration parameters are known [1]. Aiming to address these shortcomings, in [2] we adopt the statistical framework proposed in [3], [4] by introducing a bias term to better account for possible discrepancies between the mathematical model and reality. Following [4], we model the bias by a Gaussian process (GP), which is a classical choice when dealing with computer simulations, since it provides a flexible non-parametric framework.

Once calibrated, the model can be used for traffic state estimation and prediction. The former consists in reconstructing traffic states for already realized traffic scenarios, whereas the latter deals with the prediction of the unknown future to be used for real time traffic management. In general, we can distinguish between model-driven and data-driven approaches. In the first case, physical knowledge and therefore also the calibrated parameters are used to estimate and predict the traffic state. This purely model-driven approach is often criticized to be an over-simplification of the reality, subject to model choice and calibration limitations [5], [6]. In contrast, data-driven approaches estimate the traffic states from real (historical) data, using statistical or Machine Learning methods. This alternative can deal with irregularities such as noisy data or individual driving behaviors [5]. However, it requires a large amount of data and fails to predict non-recurrent traffic situations, such as accidents [7]. Thus, a natural idea is to consider hybrid approaches for vehicular traffic determination. Currently, the so called physics-informed neural networks (PINNs) are gaining more and more attention in the literature (see e.g. [8]): in PINNs, neural networks are trained to solve PDEs, whose residual is integrated in the training loss function. Analogously to our work, [6] focuses on the reconstruction of vehicular traffic dynamics. However, the authors deal only with traffic estimation based on density measurements and do not consider the prediction part. An extension to flow data is considered in [9].

A. Contribution

In this work, we propose a *hybrid* method for vehicular traffic prediction, which combines the physics with GPs, as in [10]–[12]. More precisely, the observed data are modeled by a GP, which is computationally very efficient, especially when dealing with large amount of data. Moreover, we force the GP to satisfy the model PDE at virtual, i.e. unobserved points. The GP thus calibrated provides predictions for (boundary) loop detector data, which are then given to the simulator to reconstruct the space-time evolution of the traffic speed in order to compute travel times. The reconstructed traffic speeds and travel times are then compared to the ground truth. Due to limited access to both trajectory and average loop detector data, the analysis is not only performed on real world traffic scenarios but also on synthetic data generated by a microscopic simulator.

To the best of our knowledge, this hybrid approach represents an original contribution, since previous works either deal with average loop detector data prediction (see e.g. [13], [14]) or travel time prediction based on purely data-driven methods (see e.g. [15], [16]).

B. Outline

The paper is organized as follows. Section II details the mathematical model and its discretization. The statistical framework for bias correction and traffic data prediction is detailed in Section III. Section IV describes the data sets

Manuscript received , 2023; revised .

This work has been supported by the French government, through the 3IA Côte d'Azur Investments in the Future project managed by the National Research Agency (ANR) with the reference number ANR-19-P3IA-0002.

Alexandra Würth, Mickaël Binois and Paola Goatin are with Université Côte d'Azur, Inria, CNRS, LJAD. E-mail: {alexandra.wuerth, mickael.binois, paola.goatin}@inria.fr

considered for the validation tests, which are presented in Section V. Conclusions and perspectives are commented in Section VI.

II. Description of the mathematical model

We model the traffic dynamics by a macroscopic model consisting in a PDE that guarantees the conservation of the number of cars on the road and describes the spatio-temporal evolution of vehicle density. We refer to the Lighthill-Whitham-Richards (LWR) [17], [18] model, whose key assumption, besides mass conservation, is that the average speed v is a function of the density ρ, i.e. $v = \mathcal{V}(\rho)$, which is referred to as the fundamental diagram. Thus, by the so called hydrodynamic relation $q = \rho v$, the traffic flow q is also a function of the density: $q = Q(\rho) = \rho\mathcal{V}(\rho)$. The model then consists in the scalar conservation law

$$\partial_t \rho + \partial_x \left(\rho \mathcal{V}(\rho)\right) = 0, \tag{1}$$

which is also referred to as *first order* model. In general, the LWR model allows to distinguish between free flow and congested traffic regimes, but it is less suitable for describing more complex situations, such as capacity drops and stop-and-go waves. Additionally, a single fundamental curve is not able to capture complex dynamics observed in congested regimes. Several models were proposed to address these shortcomings, among which the Aw-Rascle-Zhang (ARZ) model [19], [20] and the Generic Second Order traffic flow Model (GSOM) [21]. In these models, consisiting of a system of two PDEs, the speed function depends not only on the density but also on a Lagrangian vehicle property w, which can be interpreted as an empty road velocity, measuring the more or less aggressive behaviour of the drivers. This results in a family of fundamental curves, which can capture better the spread of the data in the congested region. However, it turned out that the increase of the model dimension by the less interpretable and non-measurable variable w does not necessarily lead to better performances when applied to real traffic scenarios (see e.g. [6], [22]–[24]). Therefore, in this work we restrict our analysis to the classical first order LWR model.

Since our aim is traffic dynamics reconstruction and prediction on a road stretch, we consider the initial boundary value problem (IBVP) for (1) on a bounded interval $]x_{in}, x_{out}[\subset \mathbb{R}$, where the variable $\rho = \rho(t,x) \in [0, R]$ is equipped with prescribed initial and boundary data at $t = 0$ and $x = x_{in}$, $x = x_{out}$.

A. Numerical solution

To compute numerically approximate solutions of (1), we use the following scheme. Given a (possibly non-uniform) spatial discretization $\{x_0, \dots, x_M\}$ of the interval $]x_{in}, x_{out}[$ with $x_0 = x_{in}$ and $x_M = x_{out}$, we set the cell sizes $\Delta x_j := x_j - x_{j-1}$ for $j \in \{1, \dots, M\}$ and a time step Δt satisfying the Courant-Friedrichs-Lewy (CFL) condition:

$$\Delta t \cdot \max_{\rho\in[0,R]} Q'(\rho) \leq \min_{j\in\{1,\dots,M\}} \Delta x_j. \tag{2}$$

We construct a finite volume [25] approximate solution of (1) of the form $\rho^{\Delta x_j}(t,x) = \rho_j^n$ for $(t,x) \in C_j^n = [t^n, t^{n+1}[\times [x_{j-1}, x_j[$ and $n \in \mathbb{N}$. Since we also want to integrate ramps, which are modeled as junctions (see e.g. [26]) we define by r_j^n (resp. s_j^n) the measured on-ramp (resp. off-ramp) fluxes at position x_j and time $n\Delta t$, which leads us to the formulation of the extended discrete LWR equations:
if $r_j^n \geq 0$ and $s_j^n = 0$ (and $r_{j-1}^n = s_{j-1}^n = 0$):

$$\begin{aligned}
\rho_j^{n+1} &= \rho_j^n + \frac{\Delta t}{\Delta x_j}\Big[F_{j-1}^n \\
&\qquad - \min\left\{D(\rho_j^n), \max\{P_j S(\rho_{j+1}^n), S(\rho_{j+1}^n) - r_j^n\}\right\}\Big], \\
\rho_{j+1}^{n+1} &= \rho_{j+1}^n \\
&\qquad + \frac{\Delta t}{\Delta x_{j+1}}\Big[\min\{D(\rho_j^n) + r_j^n, S(\rho_{j+1}^n)\} - F_{j+1}^n\Big];
\end{aligned}$$

if $s_j^n > 0$ and $r_j^n = 0$ (and $r_{j-1}^n = s_{j-1}^n = 0$):

$$\begin{aligned}
\rho_j^{n+1} &= \rho_j^n + \frac{\Delta t}{\Delta x_j}\Big[F_{j-1}^n - \min\{D(\rho_j^n), s_j^n\} \\
&\qquad - \min\left\{\max\left\{D(\rho_j^n) - s_j^n, 0\right\}, S(\rho_{j+1}^n)\right\}\Big], \\
\rho_{j+1}^{n+1} &= \rho_{j+1}^n - \frac{\Delta t}{\Delta x_{j+1}}\Big[F_{j+1}^n \\
&\qquad - \min\left\{\max\{D(\rho_j^n) - s_j^n, 0\}, S(\rho_{j+1}^n)\right\}\Big],
\end{aligned}$$

for $j \in \{2, \dots, M-1\}$. Above, the priority parameter $P_j \in [0,1]$ is approximated by the number of lanes of cell j divided by the number of lanes of cell j added to the number of lanes of the corresponding on-ramp. Moreover, we choose the discretization in such a way that we cannot have two ramps at subsequent cell interfaces.
The above scheme is based on the widely-used Godunov scheme [27] in its Cell Transmission Model (CTM) version [28], where the fluxes across cell interfaces are given by the minimum of the sending capacity (demand D) of the upstream cell and the receiving capacity (supply S) of the downstream one. Thus, the flux F_j^n is computed for $j \in \{1, \dots, M-1\}$ as

$$F_j^n = \min\left\{D(\rho_j^n), S(\rho_{j+1}^n)\right\},$$

where $D(\rho) = Q(\min\{\rho, \rho_{cr}\})$, $S(\rho) = Q(\max\{\rho, \rho_{cr}\})$ and the critical density ρ_{cr} is given by $\rho_{cr} = \operatorname{argmax}_\rho Q(\rho)$.
For $j \in \{1, M\}$, we consider two different implementations of the boundary conditions:

1) Flow boundary conditions:

$$\begin{aligned}
\rho_1^{n+1} &= \rho_1^n - \frac{\Delta t}{\Delta x_1}\big(F_1^n - \min\{q_{in}^n, S(\rho_1^n)\}\big), \\
\rho_M^{n+1} &= \rho_M^n - \frac{\Delta t}{\Delta x_M}\left(\min\left\{D(\rho_M^n), q_{out}^n\right\} - F_{M-1}^n\right),
\end{aligned}$$

where q_{in}^n (resp. q_{out}^n) denotes the inflow (resp. outflow) measured by the left (resp. right) boundary detector.
2) Density boundary conditions:

$$\rho_1^{n+1} = \rho_{in}^n, \qquad \rho_M^{n+1} = \rho_{out}^n,$$

$\rho_{in}^n, \rho_{out}^n$ being the measured or reconstructed densities at time $t = t^n$.

The choice typically depends on the application: for traffic flow reconstruction, the flow data usually lead to better performances. However, for travel time predictions, the density implementation results to be more favorable (see e.g. [24]).

In the following, we consider the Newell-Franklin [29], [30] speed function

$$\mathcal{V}(\rho) = V\left(1 - \exp\left(\frac{C}{V}\left(1 - \frac{R}{\rho}\right)\right)\right), \tag{3}$$

whose parameters are the maximum speed $V > 0$, the wave propagation speed in congestion $C > 0$ and the maximum density $R > 0$, denoted by $\theta = (V, C, R)$. Note that it holds $\mathcal{V}(\rho) \geq 0$ for $\rho \in [0, R]$, $\mathcal{V}(R) = 0$, $\mathcal{V}(0) = \lim_{\rho \to 0} \mathcal{V}(\rho) = V$ and $Q''(\rho) < 0$ for $\rho \in]0, R]$.

Remark 1: Since in the implementation of our algorithms the measured initial and density boundary data can exceed the parameter value R, we perform a projection algorithm before executing the numerical scheme: given a density $\rho > R$ and a speed $v \geq 0$, the projected densities are computed by the inverse of the speed function (3) at v.

III. Statistical methods for traffic reconstruction and prediction

The model simulator described above can be used for reconstructing and predicting traffic scenarios by integrating the measured data y^F, also called "field" observations. It is generally assumed that the field data are noisy measurements of the real quantity y^P, i.e. $y^F(t,x) = y^P(t,x) + \varepsilon$ at time t and position x. In the absence of any a-priori knowledge, the observation error ε is assumed to be independent and identically normally distributed (iid) with zero mean, i.e. $\epsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$ and $\sigma_\varepsilon^2 > 0$. Kennedy and O'Hagan (KOH) [4] propose to also take into account the discrepancy between the mathematical model with optimal parameter $\theta^* = (V^*, C^*, R^*)$ and the reality, adding a bias term b such that

$$y^P(t,x) = y^M(t,x,\theta^*) + b(t,x,\theta^*),$$

where both the simulation output y^M and b depend on the parameter θ^*. We note that the variables y^k, $k \in \{F, P, M\}$, can stand for any quantity of interest, typically the flow, speed or density in the traffic context. Finally, for all $i \in \{1, \dots, N\}$ it holds:

$$y^F(t_i, x_i) = y^M(t_i, x_i, \theta^*) + b(t_i, x_i, \theta^*) + \varepsilon,$$

where $\mathcal{X}_N = ((t_1, x_1), \dots, (t_N, x_N))$ denotes the set of time-space points where observations have been recorded.

To estimate the bias function, we rely on a GP regression [3], [4], which amounts to consider the discrepancy as a realization of a (zero-mean) multivariate normal distribution, i.e.

$$\mathbf{b}_N \sim \mathcal{N}(\mathbf{0}_N, \mathbf{K}_N) \text{ with } \mathbf{K}_N = \sigma^2(\mathbf{C}_N + g\mathbf{I}_N) \text{ and } g = \frac{\sigma_\varepsilon^2}{\sigma^2},$$

where $\mathbf{b}_N$ denotes the set of observed (noisy) biases, $\mathbf{K}_N$ (resp. $\mathbf{C}_N$) the covariance (resp. correlation) matrix between these observations and l_1, l_2, g, σ^2 the hyper-parameters. We refer to [2]–[4], [31] for more details on GP modeling.

Our aim is to match the real data as well as possible and to predict the system evolution in the future. We present below possible approaches for identifying θ and for forecasting traffic data.

A. Calibration

Parameter identification from measured data is a fundamental step for model validation and real world implementation. Macroscopic traffic flow models are often calibrated by fitting the fundamental diagram to data (see e.g. [32]–[35]). However, in congested regions, traffic data are usually widely spread. Moreover, data are not necessarily measured on the whole diagram and the number of observations are often imbalanced between the free flow and congestion regimes. Thus, we focus instead on a model-driven approach which integrates the numerical solution y^M in the optimization process aiming at minimizing the least squares distance among field and simulated data. The optimal parameter is then given by

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sqrt{\frac{1}{N}\sum_{i=1}^{N} \left|y^F(t_i, x_i) - y^M(t_i, x_i, \theta)\right|^2}. \tag{4}$$

Since the pure simulation output rarely fits the reality [1], we correct the mathematical model adding the discrepancy term modeled by GPs. In particular, the bias at $\hat{N}$ new locations $\hat{\mathcal{X}}_{\hat{N}}$, given the observations $\mathbf{b}_N$, still follows a GP [36], i.e. $\mathbf{b}(\hat{\mathcal{X}}_{\hat{\mathbf{N}}}) \mid \mathbf{b}_N \sim \mathcal{N}\left(m_N(\hat{\mathcal{X}}_{\hat{N}}), s_N^2(\hat{\mathcal{X}}_{\hat{N}}, \hat{\mathcal{X}}_{\hat{N}})\right)$ with

$$\begin{aligned} m_N(\hat{\mathcal{X}}_{\hat{N}}) &= \mathbb{E}[\mathbf{b}(\hat{\mathcal{X}}_{\hat{\mathbf{N}}})|\mathbf{b}_N] = \sigma^2 \mathbf{c}_N(\hat{\mathcal{X}}_{\hat{N}})^\top \mathbf{K}_N^{-1} \mathbf{b}_N, \\ s_N^2(\hat{\mathcal{X}}_{\hat{N}}, \hat{\mathcal{X}}_{\hat{N}}) &= \mathbb{C}ov[\mathbf{b}(\hat{\mathcal{X}}_{\hat{\mathbf{N}}}), \mathbf{b}(\hat{\mathcal{X}}_{\hat{\mathbf{N}}})|\mathbf{b}_N] \\ &= \sigma^2 c(\hat{\mathcal{X}}_{\hat{N}}, \hat{\mathcal{X}}_{\hat{N}}) - (\sigma^2)^2 \mathbf{c}_N(\hat{\mathcal{X}}_{\hat{N}})^\top \mathbf{K}_N^{-1} \mathbf{c}_N(\hat{\mathcal{X}}_{\hat{N}}), \end{aligned}$$

where we choose $c(\cdot,\cdot)$ as the Gaussian kernel and $\mathbf{c}_N(\hat{\mathcal{X}}_{\hat{n}}) = (c(\hat{\mathcal{X}}_{\hat{N}}^{(j)}, \mathcal{X}_N^{(i)}))_{1 \leq j \leq \hat{N}, 1 \leq i \leq N}$. Finally, the corrected (simulated) data y_c^M at time t and position x are given by

$$y_c^M(\hat{\mathcal{X}}_{\hat{N}}, \theta^*) = y^M(\hat{\mathcal{X}}_{\hat{N}}, \theta^*) + m_N(\hat{\mathcal{X}}_{\hat{N}}). \tag{5}$$

Observe that the hyper-parameters necessary to compute the predictive mean are obtained by maximizing the concentrated log-likelihood function

$$\begin{aligned} \log \tilde{\mathcal{L}}(l_1, l_2, g, \mathbf{b}_N) = &-\frac{N}{2}\log 2\pi - \frac{N}{2}\log \hat{\sigma}^2(l_1, l_2, g, \mathbf{b}_N) \\ &- \frac{1}{2}\log|\mathbf{C}_N + g\mathbf{I}_N| - \frac{N}{2}, \end{aligned} \tag{6}$$

where the process variance is given by

$$\hat{\sigma}^2(l_1, l_2, g, \mathbf{b}_N) = \frac{\mathbf{b}_N^\top (\mathbf{C}_N + g\mathbf{I}_N)^{-1} \mathbf{b}_N}{n}.$$

We emphasize that this bias correction helps also to address the shortcomings of the LWR model (see e.g. [24]), which were mentioned in Section II.

Remark 2: Other approaches can be used for parameter identification, such as a 2-step optimization procedure of the concentrated log-likelihood function or a Markov chain Monte Carlo (MCMC) technique, as done in [3]. However, preliminary experimental tests showed that the computationally less expensive least squares approach provides similar or

even better performances after correcting the simulation by its kriging mean, as indicated in Equation (5).

B. Prediction

In this section, we present several approaches to predict the boundary loop detector data at future times, denoted by $\hat{\mathbf{y}}_B$, which are necessary to run the numerical simulation for traffic forecast. Note that considering the first order LWR model requires the prediction of only one traffic quantity (density or flow). In general, prediction methods can either be purely data-driven, relying on historical data, or consider only data corresponding to a (short) preceding time window and which might be additionally model-driven.

1) Prediction from historical data: In the literature, we can find several prediction methods based on historical traffic data. For example, [15], [16] rely on linear regression. However, we focus on two approaches that can be considered as the most intuitive or frequently used ones. In the following, the historical traffic data form the *train data* set, which is scaled by first subtracting its mean and then dividing by its standard deviation. This normalization is done for each loop detector separately.

DTW: The Dynamic Time Warping (DTW) approach selects a time series in the train data set which behaves similarly to the data which immediately precede the prediction time slot. We will call these data *test data*, also normalized by the mean and standard deviation of the train data. To measure the similarity between two sequences, we follow [37] and use the DTW-metric `DTW` from MATLAB [38], which computes a predefined distance, such as the Euclidean one, between aligned time series. This alignment allows to find similarities between time series with shifted patterns or which evolve differently in time. The reference time series in the train data set is the one with the smallest DTW-distance with respect to the test data. We then perform a linear least square regression to scale and shift the reference data to the test one. The succeeding observations of the reference time series, adapted by the regression parameter, form the predicted (boundary loop detector) data.

LSTM: The long short-term memory (LSTM) recurrent neural network (RNN) is the most frequently used neural network in the context of time series analysis [39]. In particular, it is capable to detect long-term dependencies between time series. To create our LSTM regression network, we make the following specifications[1]: the size of the sequence input layer coincides with the number of considered loop detectors; the number of hidden units of the LSTM layer is set to 128; for the output time series, we consider a fully connected layer with the same size as the input layer and finally we include a regression layer; in the training, we use the `adam`-optimizer with 400 epochs and a learning rate of 0.001; the `SequencePaddingDirection` (resp. `Shuffle`) specification is set to *left* (resp. *every-epoch*). Once the network is trained on the above defined architecture, we use it to predict future time steps iteratively, transmitting non-updated predicted values to the `predictAndUpdateState` function.

[1]https://fr.mathworks.com/help/deeplearning/ug/time-series-forecasting-using-deep-learning.html. Accessed on 06/13/2023.

2) Prediction without historical data: If historical data are not available, we consider the following options to predict the future boundary data $\hat{\mathbf{y}}_B$ based on a (short) preceding time window. The simplest approach is to keep the data constantly equal to the last recorded measurement or an average of the last observed data, as proposed in [5] as a comparative method. However, experimental tests show that this simplistic approach, which disregards traffic evolution information, does not lead to convincing results. This motivates us to develop approaches which exploit better the traffic dynamics of the available past data.

Pure GP: The pure GP approach belongs to the class of data-driven approaches, where traffic data are modeld by a GP. In formulas, this reads

$$y^F(\mathcal{X}_N) \sim \mathcal{N}(\bar{\mathbf{y}}_N, \mathbf{K}_N) \text{ with } \mathbf{K}_N = \sigma^2(\mathbf{C}_N + g\mathbf{I}_N),$$

where the mean $\bar{\mathbf{y}}_N$ is computed by taking the average of all the observed data. The covariance hyper-parameters are obtained by maximizing the concentrated likelihood function, where we replace $\mathbf{b}_N$ by $y^F(\mathcal{X}_N)$ in (6). The predicted data are then given by

$$y^F(\hat{\mathcal{X}}_{\hat{N}}) \mid y^F(\mathcal{X}_N) \sim \mathcal{N}\left(m_N^y(\hat{\mathcal{X}}_{\hat{N}}), (s_N^y)^2(\hat{\mathcal{X}}_{\hat{N}}, \hat{\mathcal{X}}_{\hat{N}})\right),$$

with

$$m_N^y(\hat{\mathcal{X}}_{\hat{N}}) = \bar{\mathbf{y}}_{\hat{N}} + \mathbf{k}_N(\hat{\mathcal{X}}_{\hat{N}})^\top \mathbf{K}_N^{-1}\left(y^F(\mathcal{X}_N) - \bar{\mathbf{y}}_N\right),$$
$$(s_N^y)^2(\hat{\mathcal{X}}_{\hat{N}}, \hat{\mathcal{X}}_{\hat{N}}) := k(\hat{\mathcal{X}}_{\hat{N}}, \hat{\mathcal{X}}_{\hat{N}}) - \mathbf{k}_N(\hat{\mathcal{X}}_{\hat{N}})^\top \mathbf{K}_N^{-1}\mathbf{k}_N(\hat{\mathcal{X}}_{\hat{N}}),$$

where $k(\cdot,\cdot) = \sigma^2 c(\cdot,\cdot)$ and the constant entries of the $\hat{N}$-dimensional vector $\bar{\mathbf{y}}_{\hat{N}}$ coincide with the ones in $\bar{\mathbf{y}}_N$.
Then, denoting by $\hat{\mathcal{X}}_{\hat{N}_B}$ the set of observation points at boundary loop detector positions x_{in} and x_{out} in the future time slot, the boundary data $\hat{\mathbf{y}}_B$ are given by

$$\hat{\mathbf{y}}_B = m_N^y(\hat{\mathcal{X}}_{\hat{N}_B}).$$

Remark 3: The choice of the mean $\bar{\mathbf{y}}_N$ is not evident. Since predicted data reverts typically to its prior mean [40], we believe that $\bar{\mathbf{y}}_N$ is a reasonable and especially simple-to-implement choice. A more advanced suggestion can be found in [40], where they propose the so called Single Nugget Kriging method in order to reduce the influence of the prior mean on the predictions.

Hybrid GP: Aiming at improving the prediction results using the model information, we consider methods that integrate the PDE into the GP modeling. The method proposed in [10] can be applied only to non-linear PDEs whose non-linear term consists in products of derivatives, which is not our case. A more general approach, without restrictions on the form of the PDE, is suggested in [12]. The idea is to construct two likelihoods, a data and a virtual one. The first likelihood serves to fit the observations and the second one to fulfil the PDE equation at so called virtual points. Since there is no closed form for the posterior distribution available, they

end up with a variational posterior expression. This results in solving a high dimensional optimization problem, where the number of parameters depends on the number of observations and virtual points, making the approach unsuitable for real world scenarios.

This motivates us to propose a new hybrid approach, which applies to all kind of differential equations and whose set of hyper-parameters does not increase compared to the pure GP modeling. The method is based on a multi-objective optimization (MOO) of two cost functions. On one hand, we model the data by a GP, resulting in the minimization of the negative concentrated log-likelihood function:

$$\min_{l_1,l_2,g} f_1^{obj}(l_1,l_2,g) = \min_{l_1,l_2,g} \left(-\log \tilde{\mathcal{L}}\left(l_1,l_2,g,y^F(\mathcal{X}_N)\right)\right).$$

On the other hand, we also require (1) to be satisfied at virtual points, denoted by $\tilde{X}_{\tilde{N}} = \left((\tilde{t}_1,\tilde{x}_1),\ldots,(\tilde{t}_{\tilde{N}},\tilde{x}_{\tilde{N}})\right)$. This leads to the formulation of the second objective, namely the minimization of the PDE residuals at $\tilde{X}_{\tilde{N}}$:

$$\begin{aligned}&\min_{l_1,l_2,g} f_2^{obj}(l_1,l_2,g)\\ &= \min_{l_1,l_2,g} \left|\partial_t\, y^F(\tilde{X}_{\tilde{N}}) + \partial_x\, \left(y^F(\tilde{X}_{\tilde{N}})\mathcal{V}\left(y^F(\tilde{X}_{\tilde{N}})\right)\right)\right|,\end{aligned}$$

where $y^F(\tilde{X}_{\tilde{N}}) = m_N^y(\tilde{X}_{\tilde{N}})$. The derivative expressions are computed by deriving the kernels, thus it holds, for $z \in \{t,x\}$,

$$\partial_z\, y^F(\tilde{X}_{\tilde{N}}) = \partial_z \left(\mathbf{k}_N(\tilde{X}_{\tilde{N}})\right)^\top \mathbf{K}_N^{-1}\left(y^F(\mathcal{X}_N) - \bar{\mathbf{y}}_N\right),$$

and $\partial_z \left(\mathbf{k}_N(\tilde{X}_{\tilde{N}})\right) = \mathbb{C}ov\left(b(t_i,x_i),\ \frac{\mathrm{d}}{\mathrm{dz}}\, b(\tilde{t}_j,\tilde{x}_j)\right)_{1\le i\le N, 1\le j\le \tilde{N}}$.

We observe that the second objective is expressed in terms of the density. Consequently, it is natural to implement density boundary conditions in the numerical scheme. This is why we consider $y^F(\mathcal{X}_N) = \left(\rho(t_1,x_1),\ldots,\rho(t_N,x_N)\right)$ in the MOO approach (and also in the pure GP approach).

Remark 4: The choice of virtual points $\tilde{X}_{\tilde{N}}$ is not evident and has a strong influence on the prediction. In our application, we generate uniformly distributed random points, which seems to deliver reasonable results. However, improvements might probably be achieved by considering more involved methods, as the "active PDE-informed Kriging" (APIK) approach proposed in [10].

For the multi-objective optimization purpose, we compute 100 points on the Pareto front [41], using the MATLAB function `paretosearch`. We then rely on the simple knee-point method to select a solution on the Pareto front without any prior knowledge [42]. The knee-point maximizes the Euclidean distance to the segment connecting the extreme points of the Pareto front, see Figure 1. It is considered as a reasonable solution since moving along the Pareto front would lead to a larger deterioration in one of the objectives.

Once the knee-point is determined, the optimal hyper-parameters l_1, l_2 and g are identified. This enables us to compute the desired boundary loop detector data by exploiting the predictive mean formula, namely $\hat{\mathbf{y}}_B = m_N^y(\hat{\mathcal{X}}_{\hat{N}_B})$.

Remark 5: The second objective function f_2^{obj} reminds of the residual loss function in the PINNs approach [8], where the PDE solution is approximated by a neural network trained on a large amount of so called auxiliary points. In our case, the GP predictive equations are computationally less expensive. Moreover, we keep the two cost functions separated, instead of including them in a single objective [12].

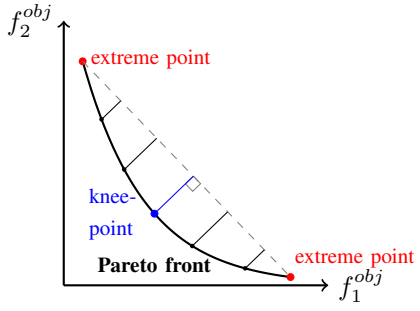


Fig. 1. Illustration of the knee-point method for the MOO approach.

C. Determination of travel times

Aiming to predict travel times in some "future" time interval $[T_{now}, T_F]$, we integrate the previously presented approaches in the following procedure:

1) Calibration of θ by the least squares approach in some preceding time window $[T_I, T_{now}[$.
2) Estimation of predicted boundary data $\hat{\mathbf{y}}_B$ in $[T_{now}, T_F]$.
3) Computation of of the model solution y^M and eventually its bias correction b in $[T_{now}, T_F]\times]x_{in}, x_{out}[$.
4) Estimation of travel times in $[T_{now}, T_F]$.

In particular, the output of the simulation is used to calculate travel times by solving the ordinary differential equation

$$\begin{cases} \frac{\mathrm{d}x}{\mathrm{d}t} = v(t,x(t)), \\ x(t_0) = x_{in}, \end{cases}$$

for $t_0 \in [T_{now}, T_F]$. The travel time is then given by the first time $\hat{\tau} = \hat{\tau}(t_0) > t_0$ such that $x(\hat{\tau}) = x_{out}$. The whole procedure is summarized in Algorithm 1.

Algorithm 1 Travel time computation from simulation output.

Require: Departure time t_0 of a vehicle starting at position x_{in} and going to position x_{out}, simulated speed $v(t,x)$ at time t and position x, time step size Δt of numerical simulation.
Initialize space position by $x = x_{in}$;
Initialize travel time by $\hat{\tau} = 0$;
while $x < x_{out}$ **do**
  Update space position by $x = x + \Delta t \cdot v(t_0 + \hat{\tau}, x)$;
  Update travel time by $\hat{\tau} = \hat{\tau} + \Delta t$;
end while
Return travel time $\hat{\tau}$.

To evaluate the performance of our travel time prediction, we need a comparison to real data, which in the best case are recovered from Global Positioning System (GPS) tacking, probe vehicles or video recordings. However, these measurements are not often available or accessible [15]. In these cases, we will use aggregated loop detector measurements

as an approximation to the ground truth. As an error metric, we consider a relative version of the root mean square error (rRMSE), comparing the true and reconstructed travel times, τ and $\hat{\tau}$, for the same road stretch at N_τ different departure instants. Thus, the total travel time error $\mathbf{E}^\tau$ reads as

$$\mathbf{E}^\tau = \sqrt{\sum_{i=1}^{N_\tau} (\tau_i - \hat{\tau}_i)^2} \Big/ \sum_{i=1}^{N_\tau} \tau_i^2 \,.$$

In the following, we detail the computation of the reference travel times in the two mentioned cases:

a) Trajectory data: Real travel time data may fluctuate a lot due to different traffic situations such as congestion and free flow regimes, different driving behaviors and vehicle types or due to external factors such as weather conditions. Instead, the mathematical model provides average speed values and therefore only average travel times can be obtained by Algorithm 1. For a better comparison, we also average the real recorded data: to compute the reference travel time for a vehicle starting at time t, we take the mean over the travel times of vehicles which started their trip between $t - \varepsilon$ and $t + \varepsilon$, where $\varepsilon > 0$ has to be chosen.

b) Aggregated data: If no or only few trajectory data are available, it is possible to derive travel times approximations from aggregated data using the N-curves method [43], obtained by summing up the aggregated flow data measured by loop detectors. To estimate the travel time between two loop detectors A and B for a vehicle starting at time t_A at loop A, we calculate the number of accumulated vehicle counts for loop A at time t_A and we intersect this number with the N-curve of loop B, obtaining the arrival time t_B. The travel time is thus given by $\tau = t_B - t_A$. Due to the usage of aggregated data, this approach leads directly to average travel times. However, a major source of error for the N-curve method is the presence of on- of off-ramps, lane changes and overtaking maneuvers [43].

An alternative for travel times reconstruction can be the application of Algorithm 1, where the entries of the field $v(t, x)$ are given by the piece-wise constant average speed measurements. We refer to this as the *baseline method*.

IV. Description of traffic data sets

To validate the proposed approaches, we refer to traffic data recorded by loop detectors, providing aggregated information over time, such as the traffic flow q and the occupancy O (the percentage of time a detector is occupied by a vehicle [14]). From the occupancy measurement, one can derive the traffic density ρ from the relation $\rho = O/l$, where l denotes the average vehicle length. If two detectors are installed in very close succession, the average traffic speed v can be also directly computed. Otherwise, it can be derived from the fundamental relation $v = q/\rho$, which leads to a spatial and not a temporal average value. We point out that contrary to the flow (resp. density) which belongs to the class of temporal (resp. spatial) traffic data, the speed can be defined as both a temporal and a spatial quantity [43]. However, these two definitions differ from each other, thus they naturally lead to different results in applications such as travel time predictions.

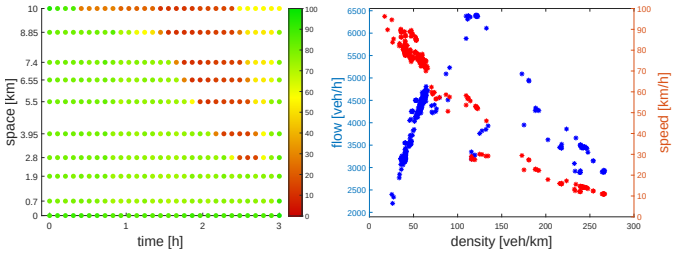
A. Synthetic microscopic traffic data

Since real data often suffer serious limitations, such as non-functioning sensors or measurement errors, we first validate the proposed approaches on synthetic data. To this aim, we rely on data generated by *Simulation of Urban MObility* (in short SUMO) [44]. Unlike data derived by simulations based on a macroscopic traffic flow model [23], the microscopic simulator tracks each vehicle, giving direct access to trajectory data and the corresponding travel times.

We consider two traffic scenarios, both simulating a highway traffic situation for a rampless 10km road stretch with three lanes and a constant speed limit of 100km/h, equipped with 3×10 loop detectors, one for each lane at ten different, non-equidistant locations. Moreover, the sensor data are aggregated over 6 minutes. The traffic flow consists of three different vehicle types differing in their desired maximum speed. The length of all vehicles is set to 5m. SUMO provides all measurable traffic quantities: the flow, occupancy and speed per lane. To compute the average speed over the three lanes, we use the space mean formula $v = \frac{1}{\rho} \sum_{i=1}^{3} \rho_i v_i$, v_i given by the fundamental equation. We note that other choices for v_i are possible, such as the arithmetic or harmonic mean speed; however, for these alternatives, we observed worse traffic speed reconstructions.

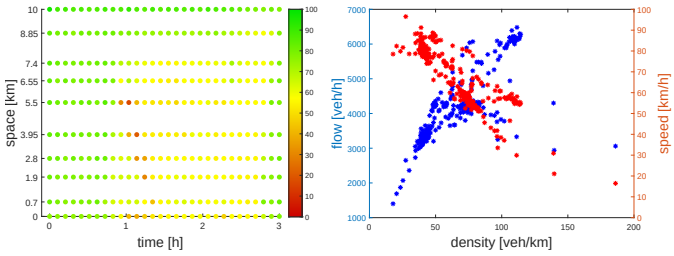
SUMO-1: The first synthetic scenario reproduces a highly congested traffic situation, generated by gradually reducing the speed limit on the last 0.01km of the road, thus inducing a backward moving congestion wave, which dissipates after the considered time window of 3 hours. In Figure 2a, we depict the speed values measured by the loop detectors for the 6 minute aggregated time intervals. The distribution of all the data points is visualized by the fundamental diagrams in Figure 2b. The non-monotonicity of the flow data indicates the presence of different traffic regimes: the free flow and congestion. In particular, the congested phase presents more widely spread data than the free flow, which is typical of real traffic scenarios. For computing reference travel times, we refer to Section III-C a), where we choose $\varepsilon = 10$s . The first (resp. last) considered departure time is after 6 minutes (resp. 2 hours and 30 minutes), resulting in $N_\tau = 865$ vehicle trajectories. Considering Figure 5a, we observe a steadily increasing travel time up to $\tau \simeq 27$min due to congestion. The results for the method of N-curves (resp. baseline method) is depicted in black (resp. blue). It appears that the N-curve method reflects better the ground truth than the baseline one, which is confirmed by the rRMSEs: $\mathbf{E}^\tau_{\text{N-curve}} = 0.027 < \mathbf{E}^\tau_{\text{baseline}} = 0.052$. This can be explained by the more precise information on flow variations exploited by the N-curves, compared to the piece-wise constant speed data employed by the baseline approach.

SUMO-2: The second scenario simulates an accident occurring at time $t = 1$h. We implement this in SUMO by closing the rightmost lane between kilometer 6 and 9 for a duration of 80 minutes. This creates less congestion than in **SUMO-1**, see Figure 3a and the few data points in the congested region in Figure 3b. However, we observe a sudden change in the traffic regime, especially for the loop detector located at $x = 5.5$km,



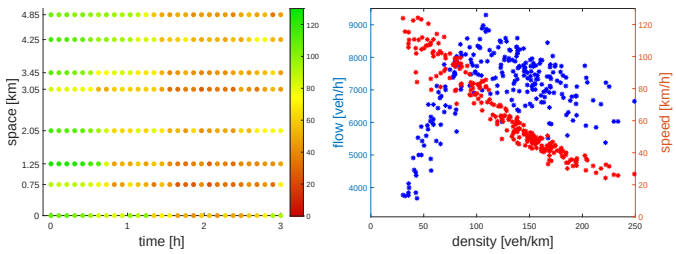
(a) Speed evolution. (b) Fundamental diagrams.

Fig. 2. Traffic scenario **SUMO-1**.



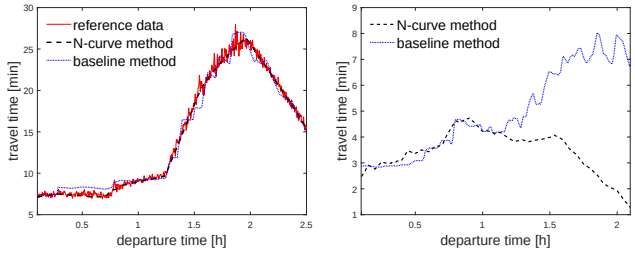
(a) Speed evolution. (b) Fundamental diagrams.

Fig. 3. Traffic scenario **SUMO-2**.



(a) Speed evolution. (b) Fundamental diagrams.

Fig. 4. Traffic scenario **RTMC**.



(a) **SUMO-1**. (b) **RTMC**.

Fig. 5. Illustration of average travel times.

right before the lane closure. This scenario is interesting to analyze since the sudden change in the dynamics cannot be captured by the model without ad-hoc changes. Moreover, the last boundary detector is not affected by this lane closure. In this case, the role of the bias becomes crucial.

B. Real traffic data

Aiming to test our approach in real world scenarios, we consider the RTMC data set [45], provided by the Minnesota Department of Transportation (MnDOT). As the synthetic SUMO data, the RTMC data are 6 minute averages obtained by single loop detectors measuring the traffic flow and the occupancy. For the tests, we consider a 4.85km long road stretch on the northbound direction of the interstate highway I-35W, which is equipped with 8 sensors on the mainlane, 2 at on-ramps and 3 at off-ramps. The road stretch counts five lanes and the speed limit is 55miles/hour ($\approx$ 90km/h). We refer to the year 2013 measures, serving as the historical train data, and some selected days in 2014 as test data. The data were pre-processed to remove abnormalities: we discarded eight full days (due to the presence of negative traffic quantities) and we replaced all measured flow (resp. density) values by zero if the corresponding density (resp. flow) was zero. In these cases, we set the average speed to 200km/h. We remark that a zero flow could also correspond to a fully congested road, however this is never the case for our extracted RTMC data, which can be easily verified by comparing the data measured by nearby loop detectors.

RTMC: The selected real data scenario covers the morning time slot from 6am to 9am of Wednesday, November 5th, 2014. Figure 4a show free flow conditions in the first 30 minutes. Then, the traffic gets denser until almost the end of the considered time period, see also the flow fundamental diagram in Figure 4b. Again, the congested part show a larger spread of the data for densities higher than 110veh/km.
Since in this case we do not have access to trajectory data, the reference travel times can be only estimated by using the aggregated loop detector data. We remark that the initialization of the N-curves belonging to the first and last loop detector are based on a free flow assumption at 6am. In Figure 5b, we illustrate the results of the two approaches for vehicles departing during the first 2 hours. Unlike the SUMO scenario, here the N-curve method fails since the travel time decreases constantly, resulting finally in negative values. The baseline approach however captures the increasing of travel times, due to lower measured speeds in the second part of the time slot.

V. Model and Statistical Methods Validation

Since we are interested int travel time prediction, we consider the speed as our quantity of interest. In the numerical scheme, we focus on a density boundary implementation, which turned out to perform better in our context. Moreover, the initial datum is approximated by piece-wise constant density values, which are given by the first recorded aggregated measurements.

A. Traffic calibration and reconstruction

The calibration is run on the first 2 out of 3 hours for each scenario, using the MATLAB nonlinear optimization function `fmincon`. The last hour is left for the prediction tests in Section V-B.
In Table I, we list the parameters and the rRMSE $\mathbf{E}$ (resp. $\mathbf{E_c}$)

at observation points between the field and simulated (resp. corrected) speed data. For comparison, and to underline the benefit of the physical model, we consider the error $\mathbf{E}_{GP}$ between the kriging mean of $y^F(\mathcal{X}_N) \sim \mathcal{N}(\bar{\mathbf{y}}_N, \mathbf{K}_N)$ and the field data. We observe that the bias corrected version $\mathbf{E}_c$ always outperforms $\mathbf{E}$ and $\mathbf{E}_{GP}$. Moreover, the errors for the

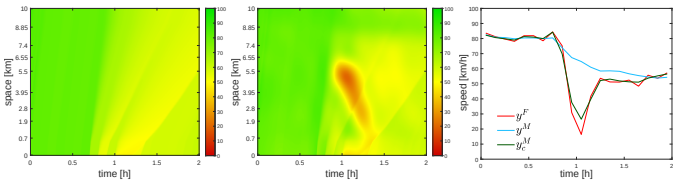
TABLE I
CALIBRATION RESULTS AND ERROR METRIC (km/h)

	V	C	R	$\mathbf{E}$	$\mathbf{E_c}$	$\mathbf{E_{GP}}$
SUMO-1	96	24	344	0.076	**0.034**	0.063
SUMO-2	93	31	300	0.117	**0.034**	0.083
RTMC	120	54	291	0.227	**0.069**	0.128

RTMC data are sensibly higher, probably due to measurement errors.

Remark 6: We recall that the **RTMC** scenario includes ramps; however, we set $r_j^n = 0 = s_j^n$ in the simulation code, since this choice leads to slightly lower rRMSEs, compared to $\mathbf{E} = 0.228$, $\mathbf{E_c} = 0.071$ and $\theta^* = (107, 86, 290)$ including ramp information. We will proceed analogously in the prediction section, thus avoiding to forecast ramp data, whose impact is questionable.

We also point out the strong improvement of the error metric after the bias correction in **SUMO-2**. As previously discussed, the pure simulation cannot reflect the traffic dynamics properly, because the model is not designed to capture the lane closure. This scenario emphasizes the benefit of the bias modeling, which is also underlined by the space-time speed plots in Figure 6 (a) and (b). Indeed, the corrected version is able to detect the traffic jam. This can also be observed in the speed profile for the sixth sensor, see Figure 6 (c).



(a) Simulation. (b) Corrected simulation. (c) Speed profile.

Fig. 6. **SUMO-2**. Illustration of space-time speed and speed profile.

B. Traffic prediction

For the prediction of the density boundary data needed to run the numerical scheme, we introduce the density boundary prediction rRMSE $\hat{\mathbf{E}}_B^\rho$ between the measured aggregated $y_\rho^F(\hat{\mathcal{X}}_{\hat{N}_B})$ and predicted $\hat{\mathbf{y}}_\mathbf{B}$ density data at $2\hat{n}_{\hat{t}}$ future boundary observation points. We also compare the simulated (resp. corrected) speed with the available coarse field data at loop detector positions and aggregated future times by the rRMSE, denoted by $\hat{\mathbf{E}}$ (resp. $\hat{\mathbf{E}}_c$). These metrics are used as indicators for travel time prediction quality, lower values corresponding to better forecasts. Additionally, we compute the two speed metrics using the real boundary data $y_\rho^F(\hat{\mathcal{X}}_{\hat{N}_B})$ in the simulation. This will be referred to as the *oracle case*. Finally, in the SUMO data case, we also compare the predicted relative travel time error metrics $\hat{\mathbf{E}}^\tau$ and $\hat{\mathbf{E}}_c^\tau$ with the reference data, where we consider the vehicles departing between $t = 90$min and $t = 150$min, corresponding to 361 trajectories.

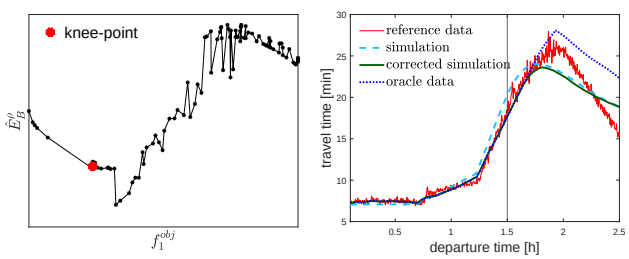
Remark 7: For the determination of the number of virtual time points which are used in the second objective function, we differentiate between the time and space dimension: for the first one, we double the size of the 6 minute aggregated data in the 3 hours, thus sampling $\frac{180}{6} \cdot 2 = 60$ points from the uniform distribution. The virtual space points however are not sampled, instead they match exactly the position of the main loop detectors. We observe that sampling the space points lead to worse results in our case.

In Table II, we highlight in bold the lowest value for each considered error metric. First of all, we observe that, compared to the previous estimation section, the difference between the predicted speed errors $\hat{\mathbf{E}}$ and $\hat{\mathbf{E}}_c$ are negligible, meaning that the bias correction in the future time slot has almost no or even a negative impact. Second, a good density boundary prediction does not necessarily lead to the lowest speed error $\hat{\mathbf{E}}$, as in the oracle boundary case where it holds $\hat{\mathbf{E}}_B^\rho = 0$ but $\hat{\mathbf{E}}$ can be bigger than in other approaches. This underlines the difficulty when dealing with different traffic quantities: since the model describes density dynamics, a good reconstruction of the density does not necessarily result in a good speed estimation. In this regard, the independent speed reconstruction provided by second order models do not lead generally to improvements, as shown in [24]. Third, against expectations, the lowest $\hat{\mathbf{E}}$ value does not correspond to the best travel time prediction result: in **SUMO-1** the MOO approach outperforms the other cases although its coarse speed reconstruction is worse.

TABLE II
PREDICTION RESULTS

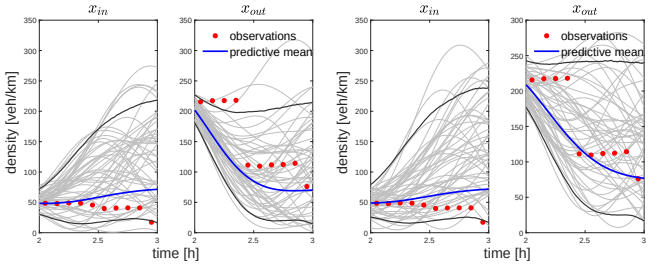
		pure GP	MOO	DTW	LSTM	oracle
SUMO-1	$\hat{\mathbf{E}}_B^\rho$	0.357	**0.268**	-	-	0
	$\hat{\mathbf{E}}$	0.348	0.402	-	-	0.373
	$\hat{\mathbf{E}}_c$	**0.341**	0.394	-	-	0.358
	$\hat{\mathbf{E}}^\tau$	0.103	**0.079**	-	-	0.157
	$\hat{\mathbf{E}}_c^\tau$	0.112	0.082	-	-	0.147
SUMO-2	$\hat{\mathbf{E}}_B^\rho$	0.321	**0.316**	-	-	0
	$\hat{\mathbf{E}}$	0.135	0.124	-	-	0.155
	$\hat{\mathbf{E}}_c$	0.130	**0.118**	-	-	0.147
	$\hat{\mathbf{E}}^\tau$	0.056	0.054	-	-	0.052
	$\hat{\mathbf{E}}_c^\tau$	0.037	0.034	-	-	**0.031**
RTMC	$\hat{\mathbf{E}}_B^\rho$	0.257	0.210	**0.158**	0.172	0
	$\hat{\mathbf{E}}$	0.383	**0.245**	0.445	0.281	0.371
	$\hat{\mathbf{E}}_c$	0.388	0.255	0.441	0.318	0.359

In **SUMO-2** all proposed methods perform very similarly in terms of travel time prediction: the oracle boundary case shows a slight outperformance, although its speed rRMSEs are worse than the others. Looking at the traffic volume in the prediction hour in Figure 3a, the good results are easily explainable: the congestion induced by the lane closure has almost no impact on the last hour, consequently it is easier to predict the boundary data. For **RTMC**, we observe that the DTW



(a) $\hat{\mathbf{E}}_B^\rho$ vs f_1^{obj} along the Pareto front. (b) Predicted travel time.

Fig. 7. **SUMO-1**. Travel time prediction results for MOO approach.



(a) Pure GP approach. (b) MOO approach.

Fig. 8. **SUMO-1**. Uncertainty quantification for pure GP and MOO approach.

approach, which provides clearly the lowest $\hat{\mathbf{E}}_B^\rho$, performs the worst in terms of speed rRMSE. In contrast, although the boundary predictions obtained by the LSTM network are disappointing, its speed prediction power is convincing.

In Figure 7, we illustrate the results for **SUMO-1** and the MOO approach. Although the knee-point heuristic does not lead to the lowest possible density boundary prediction error $\hat{\mathbf{E}}_B^\rho$ (see Figure 7a), the performance is acceptable since only a few points on the Pareto front undercut this point. Moreover, in order to understand the rather disappointing performance of the oracle case, we illustrate in Figure 7b the travel time profiles for the simulations and oracle cases. By the black vertical line, we highlight the starting point of the 361 considered trajectories, used in the travel time error computation. Naturally, the oracle boundary curve and corrected simulated curve coincide in the past because the simulation is executed with the same boundary data. In the last hour, the oracle boundary case starts to overestimate the travel times. In contrast, the (corrected) simulation both under- and overestimates the speed which leads therefore to a lower rRMSE in average. Due to the increasing uncertainty in the future, it is clear that the correction tends to go back to the pure simulation, which underestimates the speeds when using real boundary data. This in turn explains the behavior of the oracle curve and the similar performances of $\hat{\mathbf{E}}$ and $\hat{\mathbf{E}}_c$.

As a final analysis, we compare in Figure 8 the uncertainty of the pure GP and our proposed MOO approach. Together with the real future boundary observations (red stars) and its predictions $\hat{\mathbf{y}}_B = m_N^y(\hat{\mathcal{X}}_{\hat{N}_B})$ (blue line), we depict 100 realizations of $\mathcal{N}\left(m_N^y(\hat{\mathcal{X}}_{\hat{N}_B}), (s_N^y)^2(\hat{\mathcal{X}}_{\hat{N}_B}, \hat{\mathcal{X}}_{\hat{N}_B})\right)$ and the corresponding 90% predictive intervals. We observe that in the hybrid MOO approach, all observations are lying inside the predictive intervals, whereas in the pure GP approach the uncertainty is higher for the right boundary detector. This last analysis reinforces the evidence of the advantage of integrating the physics in the prediction approach.

VI. Conclusion

In this paper, we focused on traffic flow reconstruction and prediction by using a first order macroscopic traffic flow model and statistical approaches. We have shown the benefit of correcting the simulation output by a bias in order to compensate model limitations in reconstructing real data. For prediction, the bias correction can be neglected due to its convergence towards the pure simulation.

Moreover, we proposed a hybrid MOO approach for boundary data forecast, combining the physical knowledge, given by the mathematical model, and GPs. This method is based on multi-objective optimization and it addresses the shortcomings of the existing ones: it applies to all kind of differential equations and the set of hyperparameters does not increase compared to the pure GP modeling. The results are robust, providing competitive error metric values in all tested scenarios. Additionally, we observed a non-logical anti-correlated behavior of the metrics $\hat{\mathbf{E}}$ and $\hat{\mathbf{E}}^\tau$, which might be partly due to the choice between time or space mean speeds. Indeed, the difficulty of predicting the speed is a frequently mentioned problem (see e.g. [46], [47]).

This work opens several perspectives for future research, as the choice of the virtual points in the MOO approach, for example considering the so called APIK approach [10]. Moreover, it would be interesting to compare the performance of our proposed MOO approach with the PINNs method.

References

[1] R. Tuo and C. F. J. Wu, "Efficient calibration for imperfect computer models," *The Annals of Statistics*, vol. 43, no. 6, pp. 2331–2352, 2015.

[2] A. Würth, M. Binois, P. Goatin, and S. Göttlich, "Data-driven uncertainty quantification in macroscopic traffic flow models," *Advances in Computational Mathematics*, vol. 48, no. 6, pp. 1–26, 2022.

[3] D. Higdon, M. Kennedy, J. C. Cavendish, J. A. Cafeo, and R. D. Ryne, "Combining field data and computer simulations for calibration and prediction," *SIAM Journal on Scientific Computing*, vol. 26, no. 2, pp. 448–466, 2004.

[4] M. C. Kennedy and A. O'Hagan, "Bayesian calibration of computer models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 3, pp. 425–464, 2001.

[5] M. Briani, E. Cristiani, and E. Onofri, "Inverting the fundamental diagram and forecasting boundary conditions: How machine learning can improve macroscopic models for traffic flow," *arXiv preprint arXiv:2303.12740*, 2023.

[6] 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 Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 11 688–11 698, 2022.

[7] A. Allström, J. Ekström, D. Gundlegård, R. Ringdahl, C. Rydergren, A. M. Bayen, and A. D. Patire, "Hybrid approach for short-term traffic state and travel time prediction on highways," *Transportation Research Record*, vol. 2554, no. 1, pp. 60–68, 2016.

[8] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics informed deep learning (part I): Data-driven solutions of nonlinear partial differential equations," *arXiv preprint arXiv:1711.10561*, 2017.

[9] P. Goatin and D. Inzunza, "A PINN approach for traffic state estimation and model calibration based on loop detector flow data," in *MT-ITS 2023 - 8th International Conference on Models and Technologies for*

Intelligent Transportation Systems, Saint-Laurent-Du-Var, France, Jun. 2023. [Online]. Available: https://hal.science/hal-04206224

[10] J. Chen, Z. Chen, C. Zhang, and C. F. J. Wu, "APIK: Active physics-informed kriging model with partial differential equations," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 10, no. 1, pp. 481–506, 2022.

[11] Y. Chen, B. Hosseini, H. Owhadi, and A. M. Stuart, "Solving and learning nonlinear PDEs with Gaussian processes," *Journal of Computational Physics*, vol. 447, p. 110668, 2021.

[12] D. Long, Z. Wang, A. Krishnapriyan, R. Kirby, S. Zhe, and M. Mahoney, "AutoIP: A united framework to integrate physics into Gaussian processes," in *International Conference on Machine Learning*. PMLR, 2022, pp. 14 210–14 222.

[13] O. Mohammed and J. Kianfar, "A machine learning approach to short-term traffic flow prediction: A case study of interstate 64 in Missouri," in *2018 IEEE International Smart Cities Conference (ISC2)*. IEEE, 2018, pp. 1–7.

[14] N. Polson and V. Sokolov, "Bayesian analysis of traffic flow on interstate I-55: The LWR model," *The Annals of Applied Statistics*, vol. 9, no. 4, pp. 1864 – 1888, 2015. [Online]. Available: https://doi.org/10.1214/15-AOAS853

[15] J. Kwon, B. Coifman, and P. Bickel, "Day-to-day travel-time trends and travel-time prediction from loop-detector data," *Transportation Research Record*, vol. 1717, no. 1, pp. 120–129, 2000.

[16] J. Rice and E. Van Zwet, "A simple and effective method for predicting travel times on freeways," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 3, pp. 200–207, 2004.

[17] M. J. Lighthill and G. B. Whitham, "On kinematic waves. II. A theory of traffic flow on long crowded roads," *Proc. Roy. Soc. London Ser. A*, vol. 229, pp. 317–345, 1955. [Online]. Available: https://doi.org/10.1098/rspa.1955.0089

[18] P. I. Richards, "Shock waves on the highway," *Operations Res.*, vol. 4, pp. 42–51, 1956. [Online]. Available: https://doi.org/10.1287/opre.4.1.42

[19] 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. [Online]. Available: https://doi.org/10.1137/S0036139997332099

[20] H. M. Zhang, "A non-equilibrium traffic model devoid of gas-like behavior," *Transportation Res. Part B*, vol. 36, no. 3, pp. 275–290, 2002.

[21] J.-P. Lebacque, S. Mammar, and H. Haj-Salem, "Generic second order traffic flow modelling," in *Transportation and Traffic Theory 2007*, 2007.

[22] B. Piccoli, K. Han, T. L. Friesz, T. Yao, and J. Tang, "Second-order models and traffic data from mobile sensors," *Transportation Research Part C: Emerging Technologies*, vol. 52, pp. 32–56, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0968090X14003635

[23] A. Würth, M. Binois, and P. Goatin, "Validation of calibration strategies for macroscopic traffic flow models on synthetic data," *MT-ITS 2023 Proceedings*. Weblink: https://hal.science/hal-04197769v1, 2023, accessed on 09/13/2023.

[24] A. Würth, "Road traffic flow reconstruction and prediction with macroscopic models enhanced by data-based statistical approaches," Theses, Université Cote d'Azur, Dec. 2023. [Online]. Available: https://hal.science/tel-04334753

[25] R. J. LeVeque, *Finite volume methods for hyperbolic problems*. Cambridge University press, 2002, vol. 31.

[26] M. Garavello, K. Han, and B. Piccoli, *Models for vehicular traffic on networks*, ser. AIMS Series on Applied Mathematics. American Institute of Mathematical Sciences (AIMS), Springfield, MO, 2016, vol. 9.

[27] S. K. Godunov, "A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics," *Mat. Sb. (N.S.)*, vol. 47 (89), pp. 271–306, 1959.

[28] C. F. Daganzo, "The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory," *Transportation Research Part B: Methodological*, vol. 28, no. 4, pp. 269–287, 1994. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0191261594900027

[29] R. Franklin, "The structure of a traffic shock wave," *Civil Engineering Pulb. Wks. Rev*, vol. 56, pp. 1186–1188, 1961.

[30] G. Newell, "A theory of traffic flow in tunnels," *Theory of Traffic Flow*, pp. 193–206, 1961.

[31] C. E. Rasmussen and C. Williams, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2.

[32] G. Dervisoglu, G. Gomes, J. Kwon, R. Horowitz, and P. Varaiya, "Automatic calibration of the fundamental diagram and empirical observations on capacity," in *Transportation Research Board 88th Annual Meeting*, vol. 15. Citeseer, 2009, pp. 31–59.

[33] S. Fan, "*Data-fitted generic second order macroscopic traffic flow models*," Phd thesis, ProQuest LLC, Ann Arbor, MI, 2013.

[34] S. Fan, M. Herty, and B. Seibold, "Comparative model accuracy of a data-fitted generalized Aw-Rascle-Zhang model," *Netw. Heterog. Media*, vol. 9, no. 2, pp. 239–268, 2014. [Online]. Available: https://doi.org/10.3934/nhm.2014.9.239

[35] S. Fan, Y. Sun, B. Piccoli, B. Seibold, and D. B. Work, "A collapsed generalized Aw-Rascle-Zhang model and its model accuracy," eprint 1702.03624, physics.soc-ph, 2017.

[36] R. B. Gramacy, *Surrogates: Gaussian process modeling, design, and optimization for the applied sciences*. CRC press, 2020.

[37] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE transactions on acoustics, speech, and signal processing*, vol. 26, no. 1, pp. 43–49, 1978.

[38] MATLAB, The MathWorks Inc., Natick, Massachusetts, United States, version: 9.12.0 (R2022a). Website: https://www.mathworks.com, 2022, accessed on 09/13/2023.

[39] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[40] M. R. Lee and A. B. Owen, "Single nugget kriging," *Statistica Sinica*, pp. 649–669, 2018.

[41] K. Miettinen, *Nonlinear multiobjective optimization*. Springer Science & Business Media, 1999, vol. 12.

[42] Y. Setoguchi, K. Narukawa, and H. Ishibuchi, "A knee-based EMO algorithm with an efficient method to update mobile reference points." in *EMO (1)*, 2015, pp. 202–217.

[43] M. Treiber and A. Kesting, "Traffic flow dynamics: data, models and simulation," *Physics Today*, vol. Vol. 67, no. 3, 2014.

[44] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using SUMO," in *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018. [Online]. Available: https://elib.dlr.de/124092/

[45] Minnesota Departement of Transportation, "Mn/Dot Traffic Data." Website: http://data.dot.state.mn.us/datatools/, accessed on 09/13/2023.

[46] Z. Liu, C. Lyu, Z. Wang, S. Wang, P. Liu, and Q. Meng, "A Gaussian-process-based data-driven traffic flow model and its application in road capacity analysis," *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[47] Y. Wang, X. Yu, J. Guo, I. Papamichail, M. Papageorgiou, L. Zhang, S. Hu, Y. Li, and J. Sun, "Macroscopic traffic flow modelling of large-scale freeway networks with field data verification: State-of-the-art review, benchmarking framework, and case studies using metanet," *Transportation Research Part C: Emerging Technologies*, vol. 145, p. 103904, 2022.

Alexandra Würth is a PhD student at Université Côte d'Azur, supervised by P. Goatin and M. Binois, at Inria in the ACUMES Project Team since February 2021. Her thesis, titled by "Road traffic flow reconstruction and prediction with macroscopic models enhanced by data-based statistical approaches", aims to analyse information derived from traffic data using different statistical methods and exploiting them within deterministic PDE models.

Mickaël Binois is a researcher at the Inria Centre at Université Côte d'Azur, in the Acumes project-team. Prior to joining Inria, he was a postdoctoral researcher at Argonne National Laboratory and before at the University of Chicago Booth School of Business. His research interests include surrogate models (especially Gaussian processes), computer experiments, Bayesian optimization (constrained, single or multi-objective), and uncertainty quantification. He has led and contributed to multiple open source R-packages for opti-mization including GPareto, hetGP, DiceOptim, GPGame, and activegp.

Paola Goatin received the M.Sc. degree in mathematics from University of Padua (Italy), the Ph.D. in Applied Mathematics from SISSA-ISAS (Trieste, Italy) and the Habilitation in Mathematics from Toulon University. She is currently Senior Researcher at the Inria Centre of Université Côte d'Azur (France) and leader of the project-team ACUMES (Analysis and Control of Unsteady Models for Engineering Sciences), joint with the mathematics department at Université Côte d'Azur. Before joining Inria in 2010, she held an Applied Mathematics Associate Professorship at Toulon University. Her research interests include: hyperbolic systems of conservation laws, finite volume numerical schemes, macroscopic traffic flow models and PDE-constrained optimization. From 2010 to 2016 she held an ERC Starting Grant on "Traffic Management by Macroscopic models". In 2014, she was awarded the Inria - French Science Academy prize for young researchers. She is author of a hundred publications, including more than 70 journal papers and 1 monograph. She is also member of the Editorial Boards of *SIAM Journal on Applied Mathematics*, *ESAIM: Mathematical Modelling and Numerical Analysis* and *Networks and Heterogeneous Media*.