

# Physics-informed Learning for Identification and State Reconstruction of Traffic Density

Matthieu Barreau<sup>1</sup>, Miguel Aguiar<sup>1</sup>, John Liu<sup>1</sup> and Karl Henrik Johansson<sup>1</sup>

**Abstract**—We consider the problem of traffic density reconstruction using measurements from probe vehicles (PVs) with a low penetration rate. In other words, the number of sensors is small compared to the number of vehicles on the road. The model used assumes noisy measurements and a partially unknown first-order model. All these considerations make the use of machine learning to reconstruct the state the only applicable solution. We first investigate how the identification and reconstruction processes can be merged and how a sparse dataset can still enable a good identification. Secondly, we propose a pre-training procedure that aids the hyperparameter tuning, preventing the gradient descent algorithm from getting stuck at saddle points. Examples using numerical simulations and the SUMO traffic simulator show that the reconstructions are close to the real density in all cases.

## I. INTRODUCTION

Traffic state control has recently attracted a lot of attention [1]. The possibility, in a near future, of using automated vehicles within the flow of vehicles opened many new control and observation perspectives.

There are two classical methods for traffic state reconstruction. The most used one relies on a model, and is therefore labeled model-based. The survey [2] gives a good overview of many different modern techniques. The use of probe vehicles in this context is however quite recent and the reader can refer to [3], [4], [5]. Nevertheless, for more complex models or robustness issues, it is today almost impossible to use macroscopic models. This is in part due to the fact that it is very difficult, mathematically speaking, to derive convergence properties for infinite-dimensional nonlinear systems.

Another approach that is widely used today is called data-driven [6]. Such a methodology uses measurement data to derive system properties or predict the near future. This has been used in traffic state reconstruction in [7] for instance. The approach is quite powerful since it does not require many assumptions and the generality is quite high. However, there are not many practical applications since it requires many in-domain measurements. Using vehicles as sensors (probe vehicles) requires a large penetration rate, meaning that most of the vehicles are capable of probing.

To get the advantages of both methods while avoiding the aforementioned difficulties, it is of interest to develop a

data-model driven methodology. This has recently been done in [8], [9] using the notion of physics-informed learning. This data-based technique enforces a physical model on the measurements such that the generalization error is kept small even when the measurements are few and sparse.

This approach has had a very large impact on the scientific community. The problem of traffic state reconstruction has recently been investigated using this methodology in [10] and further in [11], [12]. The use of probe vehicles in particular is studied in [13]. The application to traffic flow reconstruction with external simulators is considered in [14].

The objective of this paper requires some definitions before we are able to state it precisely. However, informally speaking, the aim is to derive an algorithm capable of estimating the density of cars from measurements taken by a subset of the cars on the road. The main contribution is the development of a machine learning framework for joint identification and state estimation of traffic flow from sparse measurements taken by probe vehicles. In comparison with our previous papers [13], [14], which focused on state estimation, here we consider in addition the problem of identifying the velocity function. Such a difference implies an explosion of the number of variables and physical costs. The second contribution lies in the enhanced training procedure so that it can deal with many constraints simultaneously.

The paper is organized as follows. In Section 2, we introduce two traffic models. Section 3 is dedicated to the mathematical formulation of the objective in terms of an optimization problem and introduces a relaxed version of the problem. Section 4 proposes a learning solution focusing on the training procedure. Section 5 discusses the results obtained in different settings. Finally, Section 6 concludes with some perspectives on future research directions.

**Notation:** We define  $L^1_{\text{loc}}(\mathcal{S}_1, \mathcal{S}_2)$ ,  $L^\infty(\mathcal{S}_1, \mathcal{S}_2)$  and  $C^k(\mathcal{S}_1, \mathcal{S}_2)$  as the spaces of locally integrable functions, bounded functions and continuous functions of class  $k$  from  $\mathcal{S}_1$  to  $\mathcal{S}_2$ , respectively. For a differentiable function  $f$  of a single variable,  $f'$  refers to its derivative, while partial derivatives of multivariable functions are indicated by subscripts.

## II. COUPLED MACRO-MICRO MODEL OF TRAFFIC FLOW

There are two main kinds of traffic models in the literature, namely macroscopic and microscopic models [1]. These two model types are briefly discussed in this first section to explain the benefit of using a coupled micro-macro model of traffic flow.

<sup>1</sup> Division of Decision and Control Systems and Digital Futures, KTH Royal Institute of Technology, Stockholm, Sweden (e-mail: {barreau, aguiar, johnliu, kallej}@kth.se).

This research is partially funded by the KAUST Office of Sponsored Research under Award No. OSR-2019-CRG8-4033, the Swedish Foundation for Strategic Research and Knut and Alice Wallenberg Foundation.

### A. Microscopic model

Assume here that there are  $N > 1$  vehicles located at position  $y_i \in \mathbb{R}$  for  $i \in \{1, \dots, N\}$ . First order microscopic models assume that the following dynamic equation holds:

$$\begin{cases} y'_i(t) = V(y_{i+1}(t) - y_i(t)) & \text{if } i \in \{1, \dots, N-1\}, \\ y'_N(t) = V_{\text{lead}}(t), \\ y_1(0) < \dots < y_N(0). \end{cases}$$

This is a follow-the-leader dynamical system when  $V_{\text{lead}} \geq 0$ . Usually, the velocity function  $V$  depends on the intra-vehicular space and is decreasing, bounded and positive.

Such models perform badly in practice since they are too simple. Second-order models have been introduced to improve the correlation with observations [1]. A drawback is that the system dimension becomes very large when dealing with many vehicles, so that an infinite-dimensional model (referred to as a macroscopic model) is introduced.

### B. Macroscopic model

1) *Notion of density:* The normalized density of vehicles  $\rho : \mathbb{R}_{\geq 0} \times \mathbb{R} \rightarrow [0, 1]$  is defined such that

$$N_{\text{veh}}(x_1, x_2, t) \propto \int_{x_1}^{x_2} \rho(t, x) dx,$$

where  $N_{\text{veh}}(x_1, x_2, t)$  is the number of vehicles at a given time  $t$  on the road segment  $[x_1, x_2]$ . As shown in [13], [15], the normalized density is a solution of the equation

$$\rho_t(t, x) + f(\rho(t, x))_x = 0 \quad (1)$$

with an initial condition  $\rho(0, \cdot) = \rho_0$ , and  $f : [0, 1] \rightarrow \mathbb{R}_{\geq 0}$  is a smooth concave function which is related to  $V$ .

Equation (1) corresponds to the law of conservation of mass of some substance with density  $\rho$  and  $f(\rho)$  is the rate at which the substance is passing through a point with density  $\rho$ . Then we can also define the substance velocity  $v : [0, 1] \rightarrow \mathbb{R}_{\geq 0}$  through the relation

$$\rho v(\rho) = f(\rho). \quad (2)$$

In traffic applications, the ‘substance’ in question refers to the vehicles on a road, as from a macroscopic point of view one assumes that the flow of vehicles can be approximated by the flow of a continuous substance. Then  $f(\rho)$  gives the amount of vehicles flowing through some point of space per unit time, and  $v(\rho)$  is the mean vehicle velocity.

2) *Existence and uniqueness:* It is well known that there may be no smooth solution of (1), even if  $\rho_0$  is smooth. Hence we consider solutions in a weak sense, and one then has the following existence and stability result [13]:

*Theorem 1:* If  $\rho_0 \in L^\infty(\mathbb{R}, [0, 1])$  and  $f \in C^2([0, 1], \mathbb{R}_{\geq 0})$ , then there exist weak solutions  $\rho$  of (1) with regularity  $\rho \in C^0(\mathbb{R}_{\geq 0}, L^1_{\text{loc}}(\mathbb{R}, [0, 1]))$ .

Uniqueness is ensured by restricting the set of weak solutions to those satisfying the Lax-E condition [16, Chap. 14].

Alternatively, one may consider the unique smooth solution of the related equation

$$\bar{\rho}_t(t, x) + f(\bar{\rho}(t, x))_x = \gamma^2 \bar{\rho}_{xx}(t, x) \quad (3)$$

for small  $\gamma > 0$ . If  $\bar{\rho}_\gamma$  is the solution of (3), then the limit  $\rho := \lim_{\gamma \downarrow 0} \bar{\rho}_\gamma$  exists in  $L^1_{\text{loc}}$  [16, p. 157] and is the unique entropic solution of (1).

3) *Examples of flux functions:* The two most well-known flux functions used in traffic applications are perhaps the Greenshields [17] and the Newell-Daganzo [18], [19] flux functions. The Greenshields flux function  $f_G$  is given by

$$f_G(\rho) = \rho v_G(\rho), \quad v_G(\rho) = V_f(1 - \rho). \quad (4)$$

At zero traffic density the vehicles move at the free flow velocity  $V_f > 0$ , and the velocity  $v_G$  decreases linearly with density, going to zero as the road becomes more congested.

The Newell-Daganzo flux function is given by

$$f_{\text{ND}}(\rho) = \min \{V_f \rho, W(1 - \rho)\}. \quad (5)$$

The velocity function in this case is then

$$v_{\text{ND}}(\rho) = \begin{cases} V_f, & \rho < \sigma, \\ W \frac{1-\rho}{\sigma}, & \rho \geq \sigma, \end{cases}$$

where  $\sigma \in [0, 1]$  is such that  $V_f \sigma = W(1 - \sigma)$ . Hence vehicles move at a constant average velocity up to a critical density  $\sigma$  where the road becomes congested, and the velocity goes to zero as  $\rho$  approaches one.

*Remark 1:* The flux function in this case is not differentiable everywhere, but one can consider a smooth approximation of  $f_{\text{ND}}$  such that Theorem 1 applies.  $\square$

### C. Relation between the models

The two models introduced so far are mathematically related. The Greenshields model is a limit case of the first order follow-the-leader dynamics [20] and the integral of the normalized density is proportional to the number of vehicles. Consequently,  $\rho$  is directly related to the intra-vehicular space and one can use the velocity function  $v$  of the macroscopic scheme as the vehicle velocity  $V$  [14]. This leads to the cascaded system

$$\begin{cases} \rho_t(t, x) + f(\rho(t, x))_x = \gamma^2 \rho_{xx}(t, x), \\ y'_i(t) = v(\rho(t, y_i(t))), \quad i \in \{1, \dots, N\}, \end{cases} \quad (6)$$

for  $t \geq 0$  and  $x \in \mathbb{R}$ , with appropriate initial conditions. In [13] the authors give conditions for (6) to be mathematically well-posed.

*Definition 1:* For the two models to be self-consistent, one must ensure the following:

- 1)  $f$  must be positive, concave and  $C^2$ ;
- 2)  $v$  must be positive and decreasing;
- 3)  $v$  must be always larger than or equal to  $f'$ .

*Remark 2:* The last item has the following physical motivation. If we consider that the flow is a wave,  $f'$  is related to the group velocity (the speed of a characteristic) and  $v$  to

the phase velocity. Requiring  $f' \leq v$  means that a particle must move faster than the envelop of the wave.  $\square$

The following proposition gives a simple sufficient condition for consistency of the models.

*Proposition 1:* If  $f$  is concave and  $v \in C^2([0, 1], \mathbb{R}_{\geq 0})$ , then equation (6) is a consistent model for describing micro- and macroscopic behaviors of traffic flow.

*Proof:* From the assumptions and using (2), we get that  $f(0) = 0$  and  $f$  is positive. By concavity of  $f$ , the Chordal Slope Lemma implies that  $v$  is decreasing. Since  $f'(\rho) = v(\rho) + \rho v'(\rho)$ , we get that  $f' - v$  has the same sign as  $v'$  which is negative. Consequently, Definition 1 indeed holds and the model is consistent.  $\blacksquare$

For the sequel, we need to express  $f'$  and  $f''$  as functions of  $v$ . Using (2), we get:

$$f'(\rho) = v(\rho) + \rho v'(\rho), \quad f''(\rho) = 2v'(\rho) + \rho v''(\rho). \quad (7)$$

Combining (6), (7), Proposition 1 and Definition 1 leads to the following for  $i \in \{1, \dots, N\}$ :

$$\begin{cases} \mathcal{N}_y[y_i] := y'_i - v(\rho(\cdot, y_i)) = 0, \\ \mathcal{N}_\rho[\rho, v] := \rho_t + (v(\rho) + \rho v'(\rho)) \rho_x - \gamma^2 \rho_{xx} = 0, \\ \mathcal{N}_v[v] := 2v' + \rho v'' \leq 0, \\ v \geq 0 \text{ and } C^2. \end{cases} \quad (8)$$

*Remark 3:* Note that with the Greenshields or the regularized Newell-Daganzo speed function, the two last statements of (8) are indeed verified, resulting in a consistent model.  $\square$

#### D. Measurements

In this paper  $N$  probe vehicles located at  $y_1(t), \dots, y_N(t)$  are used as mobile sensors within the flow of vehicles, with dynamics given by the first equation in (6). These vehicles can measure the following values in real-time:

- 1) Their positions  $y_i(t)$ ;
- 2) The local density at their locations:  $\rho_i(t) = \rho(t, y_i(t))$ ;
- 3) And their instantaneous speed  $v_i(t) = y'_i(t)$ ;

Requiring position and speed measurements is reasonable since these can be obtained using GPS and/or inertial sensors. The density measurements are more delicate. In a single-lane environment, one can measure the inter-vehicular distances and deduce an approximation of the local density. In a multi-lane context, a camera or radar sensors might be needed [21].

*Remark 4:* Compared to the authors' previous work [14], we have added the instantaneous speed measurements. These are however not essential but preferable since we do not make any assumptions on the velocity-density relation.  $\square$

Based on equation (8), one can investigate the state-reconstruction problem.

### III. PROBLEM STATEMENT

The general definition of partial-state reconstruction is proposed in [13]. Here, we adapt this definition to the case of density reconstruction.

*Definition 2:* Let  $\Omega = [0, T]$ ,  $T > 0$ . A **density reconstruction**  $\hat{\rho}$  in  $\mathcal{H}_c \subseteq H^1(\Omega, H^2(\mathbb{R}, [0, 1]))$  is defined by

$$\hat{\rho} \in \underset{\bar{\rho} \in \mathcal{H}_c}{\text{Argmin}} \sum_{i=1}^N \int_0^T |\rho_i(t) - \bar{\rho}(t, y_i(t))|^2 dt. \quad (9)$$

The overall quality of the reconstruction is measured by the generalization error (GE):

$$\text{GE}_T(\hat{\rho}) = \int_0^T \int_{y_1(t)}^{y_N(t)} |\rho(t, x) - \hat{\rho}(t, x)|^2 dx dt. \quad (10)$$

**Problem statement:** The objective of this paper is to propose an efficient algorithm to compute a density reconstruction with low GE using noisy measurements from PVs.

To obtain a small GE, there are two options:

- 1) There are many probe vehicles so that  $\text{GE}_T(\hat{\rho}) \simeq \frac{1}{N} \sum_{i=1}^N \int_0^T |\rho_i(t) - \hat{\rho}(t, y_i(t))|^2 dt$  and a density reconstruction will lead to a small error;
- 2) The set  $\mathcal{H}_c$  is defined such that the minimizers have low GE.

Of course, we assume that the penetration rate is low, meaning that there are few probe vehicles compared to the number of vehicles. Hence the generalization error in the first case will be high if  $\mathcal{H}_c = H^1(\Omega, H^2(\mathbb{R}, \mathbb{R}))$ .

In this paper we are interested in the second case and the set  $\mathcal{H}_c$  is constructed using the fact that (8) must hold. In other words, we define

$$\mathcal{H}_c = \left\{ \bar{\rho} \in H^1(\Omega, H^2(\mathbb{R}, \mathbb{R})) \mid \int_0^T \int_{y_1}^{y_N} \mathcal{N}_\rho[\bar{\rho}, v]^2 = 0 \text{ for } \right. \\ \left. v \in \underset{\bar{v} \in C^2([0, 1], \mathbb{R}_{\geq 0})}{\text{Argmin}} \sum_{i=1}^N \int_0^T |v_i(t) - \bar{v}(\rho_i(t))|^2 dt \right. \\ \left. \text{s. t. } \int_0^1 \max \left( 0, \mathcal{N}_v[\bar{v}] \Big|_{\rho=u} \right)^2 du = 0 \right\}.$$

*Remark 5:* It has been proven in [3] (with  $\gamma = 0$ ) that the conditions of Definition (1) imply perfect reconstruction for finite  $T > 0$ . For any density reconstruction  $\hat{\rho} \in \mathcal{H}_c$ , we get  $\text{GE}_t(\hat{\rho}) = \text{GE}_T(\hat{\rho})$  for all  $t \geq T$ .  $\square$

In the following section we derive a relaxed version of (9) and discuss how to locally solve it.

### IV. LEARNING-BASED DENSITY RECONSTRUCTION

There are several ways of solving constrained optimization problems. One is to define the optimized variables in such a way that the constraints are already satisfied as done in [22] where the boundary conditions are enforced in the solution. However, this technique does not work well in the presence of noise or for complex systems since the boundary condition cannot be enforced. We will explore here another technique which consists of penalizing unfeasible solutions using an extended Lagrangian cost function.

*Remark 6:* We assume here that the measurements over the interval  $[0, T]$  are given as a finite dataset consisting of  $N_{\text{mea}}$  values at the sampling instants  $\{t_k\}_{k=1}^{N_{\text{mea}}}$ .  $\square$

The Lagrange multiplier method relaxes the optimization problem (9) to:

$$\hat{\rho} \in \underset{\bar{\rho} \in H^1(\Omega, H^2(\mathbb{R}, [0,1]))}{\text{Argmin}} \left\{ \min_{v \in C^2([0,1], \mathbb{R}_{\geq 0}), \gamma > 0} \sum_k \lambda_k \mathfrak{L}_k \right\} \quad (11)$$

for  $\lambda_k \geq 0$  and where the costs  $\mathfrak{L}_k$  are given in the sequel.

#### A. Interpretation of the different costs

1) *Data-based costs:* The first kind of cost function is typical in regression problems in machine learning [23]. It consists of the mean square error (MSE) between the measurements and the estimated function. Given the three types of measurements described above, we arrive at the following cost functions:

- 1) We measure the position of each PV at given instants  $t_k$  so that the dataset is  $\{y_i(t_k)\}_{i,k}$  and the MSE is

$$\mathfrak{L}_1 = \frac{1}{N} \sum_{i=1}^N \frac{1}{N_{\text{mea}}} \sum_{k=1}^{N_{\text{mea}}} [y_i(t_k) - \hat{y}_i(t_k)]^2.$$

Here,  $\hat{y}_i : [0, T] \rightarrow \mathbb{R}$  is the estimated position of the  $i$ -th PV.

- 2) The second data-based cost function is related to the density measurements  $\{\rho_i(t_k)\}_{i,k}$ :

$$\mathfrak{L}_2 = \frac{1}{N} \sum_{i=1}^N \frac{1}{N_{\text{mea}}} \sum_{k=1}^{N_{\text{mea}}} [(\rho_i(t_k) - n_{\rho_i}) - \hat{\rho}(t_k, y_i(t_k))]^2,$$

where  $\hat{\rho} : [0, T] \times \mathbb{R} \rightarrow [0, 1]$  is the estimated density and  $n_{\rho_i}$  is a variable introduced to suppress the bias in the noise (as discussed in [13]).

- 3) Finally, including the velocity measurements enables the identification of the model (8) by estimating the velocity function:

$$\mathfrak{L}_3 = \frac{1}{N} \sum_{i=1}^N \frac{1}{N_{\text{mea}}} \sum_{k=1}^{N_{\text{mea}}} [v_i(t_k) - \hat{v}(\rho_i(t_k))]^2,$$

where  $\hat{v} : [0, 1] \rightarrow \mathbb{R}_{\geq 0}$  is the estimated velocity.

For  $n_{\rho_i} = 0$ , these cost functions are based on the measurement data only, so they provide quantitative information which can be used to reject *unbiased* noise.

*Remark 7:* There is no need to remove the biases on the trajectories since they are defined up to a constant. There is also no correction on the velocity measurements since they are assumed to be noiseless.  $\square$

However, this type of cost function has some drawbacks. In the presence of biased noise, the previous cost functions do not perform well. Secondly, we must have as many density measurements as position and speed measurements. One solution to these problems is to introduce two other cost functions:

- 1) The first corrects for errors in the trajectory measurements:

$$\mathfrak{L}_4 = \frac{1}{N} \sum_{i=1}^N \frac{1}{N_{\text{mea}}} \sum_{k=1}^{N_{\text{mea}}} [(\rho_i(t_k) - n_{\rho_i}) - \hat{\rho}(t_k, \hat{y}_i(t_k))]^2.$$

- 2) The second cost function is related to velocity measurements:

$$\mathfrak{L}_5 = \frac{1}{N} \sum_{i=1}^N \frac{1}{N_{\text{mea}}} \sum_{k=1}^{N_{\text{mea}}} [v_i(t_k) - \hat{v}(\hat{\rho}_i(t_k, \hat{y}_i(t_k)))]^2.$$

Finally, the most important deficiency when using only data-based costs is the necessity of a large dataset. More specifically, the measurements should be well spread across the set  $[0, T] \times [y_1, y_N]$ . This is not the case in our problem since the measurements are taken along the trajectories of the PVs. To ensure a small generalization error we need to consider additional cost functions.

2) *Physics costs:* These cost functions do not depend on the measurements, but only on the reconstructed functions. Consequently, they can be seen as regularizers.

- 1) Dynamics of the traffic density: given  $N_{\text{phy}}^{\rho}$  sample points  $\{(t_k^{\rho}, x_k^{\rho})\}_{k=1}^{N_{\text{phy}}^{\rho}}$ ,

$$\mathfrak{L}_6 = \frac{1}{N_{\text{phy}}^{\rho}} \sum_{k=1}^{N_{\text{phy}}^{\rho}} \left( \mathcal{N}_{\rho}[\hat{\rho}, \hat{v}] \Big|_{(t,x)=(t_k^{\rho}, x_k^{\rho})} \right)^2.$$

In addition, the dissipation coefficient  $\gamma$  that is introduced to smooth  $\hat{\rho}$  should be as small as possible, leading to the introduction of another cost:

$$\mathfrak{L}_7 = \gamma^2.$$

- 2) Dynamics of the PV trajectories: given  $N_{\text{phy}}^y$  sample points  $\{t_k^y\}_{k=1}^{N_{\text{phy}}^y}$ ,

$$\mathfrak{L}_8 = \frac{1}{N} \sum_{i=1}^N \frac{1}{N_{\text{phy}}^y} \sum_{k=1}^{N_{\text{phy}}^y} \left( \mathcal{N}_y[\hat{y}_i] \Big|_{t=t_k^y} \right)^2.$$

- 3) Constraint on the concavity of  $f$ : given  $N_{\text{phy}}^v$  sample points  $\{\rho_k^v\}_{k=1}^{N_{\text{phy}}^v}$ ,

$$\mathfrak{L}_9 = \frac{1}{N_{\text{phy}}^v} \sum_{i=1}^{N_{\text{phy}}^v} \max \left( 0, \mathcal{N}_v[\hat{v}] \Big|_{\rho=\rho_k^v} \right)^2.$$

These three costs are based on the assumption that the real dynamics follow equation (8). The advantages are the same as in any model-based framework: they can effectively reduce the noise.

3) *Total cost:* The total cost is a weighted sum of the ten previously introduced loss functions weighted by the  $\lambda_k$ . If  $\lambda_2, \lambda_3, \lambda_6$  and  $\lambda_8$  are all strictly positive, the penalty method [24] states that solving a sequence of problems with these weight variables increasing to infinity leads to a solution of the discretization of problem (9).

The remaining cost functions are used to enhance the quality of the reconstruction by restricting the set of possible solutions and rejecting the noise. Indeed, the trajectory reconstructions suppress unbiased noise on the locations of the local density measurements. The costs related to the velocity are used for identification. It appears that using a subset of the velocity measurements is enough since the identification is also supported by  $\mathfrak{L}_5, \mathfrak{L}_6, \mathfrak{L}_8$  and  $\mathfrak{L}_9$ .

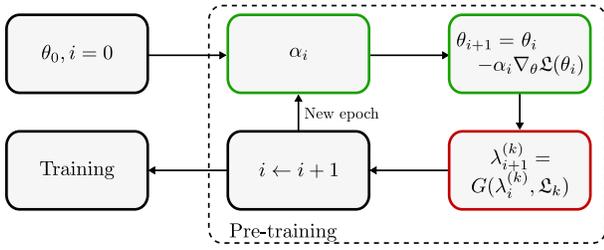


Fig. 1. Flow diagram of the update process. The boxes in green refer to traditional 1st order optimization scheme. The red box refers to the hyper-parameters  $\lambda_k$  update. The function  $G$  is defined in [26].

*Remark 8:* The physics-based cost functions we considered above are based only on the dynamics (8) and do not depend on the measurement data at all. We could additionally consider ‘mixed’ data- and model-based cost functions. For instance, relating the density measurements to the vehicle dynamics one can consider the cost function

$$\mathcal{L}_{\text{mixed}} = \frac{1}{N} \sum_{i=1}^N \frac{1}{N_{\text{mea}}} \sum_{k=1}^{N_{\text{mea}}} [\hat{y}'_i(t_k) - \hat{v}(\rho_i(t_k) - n_{\rho_i})]^2,$$

allowing for simultaneous noise rejection and identification. This type of cost is referred to as a *first-order* physics cost, while those considered above are called *second-order* physics costs. Note that unlike the second-order physics costs, this type of cost cannot correct for the problem of sparse measurements, since it depends on the data.

We observed that this type of cost is not necessary to obtain good performance in the numerical examples we consider here, hence in this paper the total cost function  $\mathcal{L}$  contains only second-order physics costs.  $\square$

### B. Neural network approximation of a density reconstruction

Let  $\Theta_\theta$  be a general neural network, where  $\theta$  is a tensor containing the parameters of the network. We propose to approximate a function  $\hat{\rho}$  satisfying (11) by  $\Theta_\theta$ . It has been proved in [9] and [13, extended version] that  $\hat{\rho}$  can be approximated arbitrarily well by  $\Theta_\theta$  provided that the number of neurons is large enough. The main issue becomes solving (11) for  $\hat{\rho} = \Theta_\theta$ .

1) *Training procedure:* The optimization problem (11) is particularly difficult to solve since it involves many ‘adversarial’ costs. In that scenario, standard optimization algorithms might not minimize the costs that are slowly varying with respect to  $\theta$ . To the best of the authors’ knowledge, the article [25] is the first to investigate this problem by adapting the weights  $\{\lambda_k\}$  during the training. However, the weight update algorithm is sensitive to noise and based on empirical considerations. Based on the same idea, in [26], the authors propose a set of algorithms to deal with this issue based on algorithms for constrained optimization.

The overall training procedure is summarized in Figure 1 and is divided into two parts:

- Pre-training step: we use a first order algorithm (such as stochastic gradient descent or ADAM) combined with a suitable algorithm for updating the  $\lambda_i$  for a fixed number of epochs.

- Training step: with the  $\lambda_i$  now fixed at appropriate values, a second-order optimizer (e.g., BFGS) is used to ensure convergence to a global optimum.

More details about the weight update algorithm can be found in Appendix A.

The main disadvantage of the proposed training procedure is that there is no guarantee on the convergence to a global minimum of (11), as well as no upper bound on the GE. Only convergence to a local minimum in parameter space is theoretically guaranteed (as is typically the case when using deep neural networks, since the resulting optimization problem is usually non-convex).

2) *Network architecture:* As discussed in [13], fully connected deep neural networks with hyperbolic tangent activation functions provide an effective and simple architecture for this problem.

The network used to approximate the density has 5 hidden layers with 10 neurons each. In order to ensure that the network output remains bounded, a tanh function is applied after the final layer. For the trajectory reconstruction, we use a small neural network with 3 hidden layers and 5 neurons per layer for each probe vehicle.

The speed function  $v$  is also approximated by a neural network. Since this function should not be too complex (otherwise it will be very difficult to solve the PDE), we use a very simple multi-layer neural network with 2 hidden layers and 5 neurons per layer. We enforce  $\hat{v}(1) = 0$  by multiplying the network output by  $1 - \rho$ .

## V. SIMULATION RESULTS AND DISCUSSION

In this section, we first compare simulation results and then investigate the performance of the training procedure using a statistical analysis<sup>1</sup>.

### A. SUMO simulation

Figure 2a shows the traffic density obtained from a SUMO simulation (see Appendix B for more details). One can clearly see stop and go waves which originate from a traffic light located at  $x = 2.5$  km.

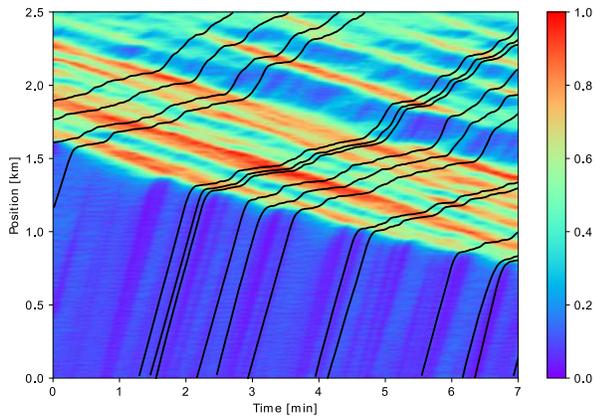
The neural network reconstruction gives the estimated density in Figure 2b. A low number of physics points is used ( $N_{\text{phy}}^\rho = 500$ ) as SUMO uses second order follow-the-leader dynamics and approximating it using a first order hyperbolic equation leads to discrepancies. With low values of  $N_{\text{phy}}^\rho$ , the model is enforced only when data is not available, thus ensuring a relatively good fit [26]. The characteristics are well-identified and the generalization error is small ( $\approx 0.3$ ), although the result is smoother than the original density.

### B. Training performance

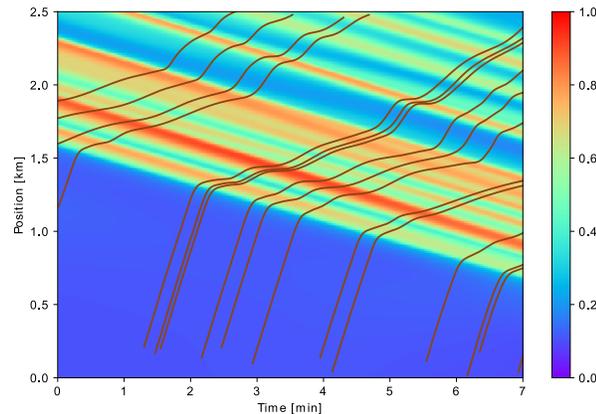
We investigate here the computation time and error for three simulation cases:

- 1) density measurements computed using a Godunov solver of (1) with the Greenshields flux function (4);

<sup>1</sup>The code and data are available at <https://github.com/mBarreau/TrafficReconstructionIdentification>



(a) Normalized density from a SUMO simulation. Black lines represent the PV trajectories.



(b) Reconstructed density using a neural network. Brown lines represent the estimated PV trajectories.

Fig. 2. Density reconstruction using data from a SUMO simulation.

- 2) density measurements computed using a Godunov solver of (1) with the Newell-Daganzo flux function (5);
- 3) density measurements from the SUMO simulation described above.

We compute the density reconstruction with and without the pre-training step described in V-B. The distribution of the computation time and the generalization error (10) for 50 simulation runs of each case is shown in Figure 3. The simulations were run on a laptop with an Intel(R) Core(TM) i5-8365U CPU @ 1.60 GHz with four processor cores.

In the first two cases, we observe that using the pre-training leads to slightly higher computational time but the mean and variance of the generalization error decrease significantly. This implies that the pre-training step improves the solution quality and makes the training procedure more robust with respect to the parameter initialization.

In the third case, the performance is worsened by the pre-training step, due to the fact that this step attempts to enforce the dynamics (8) more strictly. As mentioned above, the first-order dynamics we considered here do not match those used by the SUMO simulator, so we do not observe the same performance improvement as when the data is generated by the model dynamics.

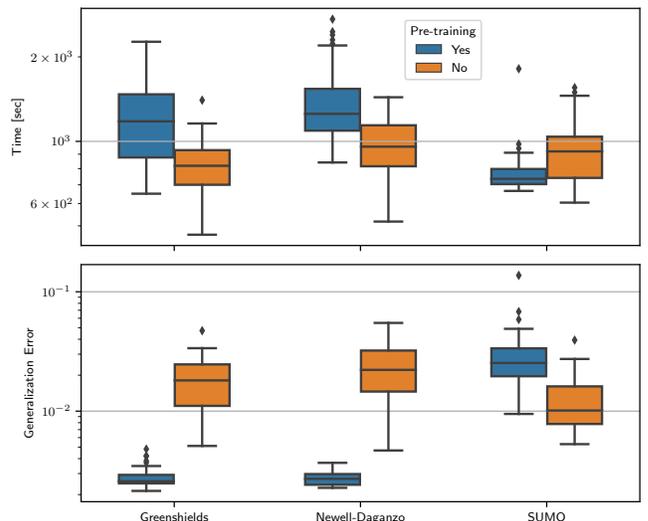


Fig. 3. Distribution of the computation time and GE for the three simulation cases described in V-B

## VI. CONCLUSION & PERSPECTIVES

In this article, we have investigated the use of physics-informed learning for identification and state reconstruction of traffic flow. The results show a good reconstruction quality with a moderate computational burden. The proposed training procedure helps in dealing with many physics costs and ensures a faster and more robust convergence. However, there are some discrepancies in the reconstruction using the data from SUMO, which are due to the model mismatch. A second order scheme with speed dynamics should be investigated. The scalability of the method to a large number of PVs is also to be studied, as the method requires one small neural network per PV. In addition, the extension to predictions in the near future should be considered, and comparisons with other methods [2], [3], [5] have to be conducted.

### APPENDIX

#### A. The training procedure

A complete study of training procedures useful in physics-informed learning is done in [26]. Here we give a brief summary of the techniques used in this article.

With a gradient-descent based optimization algorithm, the weights are updated according to

$$\theta_{i+1} = \theta_i - \alpha_i \sum_{k=1}^9 \lambda_i^{(k)} \nabla_{\theta} \mathcal{L}_k(\theta_i),$$

where  $\alpha_i$  is the learning rate.

The main idea (developed initially in [27], [28]) is to perform a dual update on the  $\lambda_i$  during training:

$$\lambda_{i+1}^{(k)} = \lambda_i^{(k)} - \alpha_{\lambda} \nabla_{\lambda^{(k)}} \mathcal{L}_k(\theta_i).$$

The update of a weight  $\lambda$  depends on the nature of the associated cost function. Data-based costs (as in part IV-A.1) have the same importance throughout the entire training and

can provide a good estimate in the initial training steps, so the corresponding weight should be positive and constant. Physics-based costs, meanwhile, represent constraints which can be of two types:

- 1) a soft constraint: the cost cannot be reduced to zero (finite-size neural network, noisy measurements...). Consequently, this term must be weighted such that its contribution relative to the data-based costs at the optimum is given by the user. This prior knowledge reflects the feasibility and reliability of the constraint.
- 2) a hard constraint: the cost must be reduced to (approximately) zero. These constraints are typically related to some physical law underlying the data that must hold. After training, the cost value must be as small as possible, regardless of the values of the other costs.

These two kinds of costs require different weight update procedures. For a soft constraint, the weight is upper bounded and should approach the desired final value given by the user. On the contrary, for a hard constraint, the weight is not upper bounded and the growth rate depends on the value of the corresponding cost.

### B. The SUMO simulation

The SUMO simulation is relatively simple and the full code is available on GitHub. The road is of length  $L = 3$  km with a traffic light at  $x = 2.5$  km. Since the traffic light is not represented in the model, we propose here to reconstruct the traffic for  $x \in [0, 2.5]$  km. The car dynamics are described in [29]. The probability of a car being a probe vehicle is set to 0.1.

The output of the simulation is the trajectories of all simulated vehicles. From this dataset, we reconstruct the number of cars in a given cell on a grid. The density is then obtained through convolution with a Gaussian filter [30]. The parameters of the Gaussian kernel depend on the grid and the speed of a vehicle in a given context. Here we assume the parameters to be the same, independently of the traffic state:  $\sigma_{\text{space}} = 0.01$  km and  $\sigma_{\text{time}} = 0.06$  min.

## REFERENCES

- [1] A. Ferrara, S. Sacone, and S. Siri, *Freeway Traffic Modelling and Control*. Springer Nature, 2018.
- [2] T. Seo, A. M. Bayen, T. Kusakabe, and Y. Asakura, "Traffic state estimation on highway: A comprehensive survey," *Annual Reviews in Control*, vol. 43, pp. 128–151, 2017.
- [3] M. L. Delle Monache, T. Liard, B. Piccoli, R. Stern, and D. Work, "Traffic reconstruction using autonomous vehicles," *SIAM Journal on Applied Mathematics*, vol. 79, no. 5, pp. 1748–1767, 2019.
- [4] M. Barreau, A. Selivanov, and K. H. Johansson, "Dynamic traffic reconstruction using probe vehicles," in *Conference on Decision and Control (CDC)*, Jeju Island, Republic of Korea, 2020.
- [5] M. Čičić, M. Barreau, and K. H. Johansson, "Numerical investigation of traffic state reconstruction and control using connected automated vehicles," in *23rd International Conference on Intelligent Transportation Systems (ITSC)*, 2020.
- [6] J. Berberich, A. Koch, C. W. Scherer, and F. Allgöwer, "Robust data-driven state-feedback design," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 1532–1538.
- [7] J. C. Herrera and A. M. Bayen, "Incorporation of lagrangian measurements in freeway traffic state estimation," *Transportation Research Part B: Methodological*, vol. 44, no. 4, pp. 460–481, 2010.
- [8] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.
- [9] J. Sirignano and K. Spiliopoulos, "DGM: A deep learning algorithm for solving partial differential equations," *Journal of computational physics*, vol. 375, pp. 1339–1364, 2018.
- [10] J. Huang and S. Agarwal, "Physics informed deep learning for traffic state estimation," in *2020 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2020.
- [11] R. Shi, Z. Mo, K. Huang, X. Di, and Q. Du, "Physics-informed deep learning for traffic state estimation," *preprint arXiv:2101.06580*, 2021.
- [12] B. T. Thodi, Z. S. Khan, S. E. Jabari, and M. Menendez, "Incorporating kinematic wave theory into a deep learning method for high-resolution traffic speed estimation," *arXiv preprint arXiv:2102.02906*, 2021.
- [13] M. Barreau, J. Liu, and K. H. Johansson, "Learning-based state reconstruction for a scalar hyperbolic PDE under noisy lagrangian sensing," in *Learning for Dynamics and Control*. PMLR, 2021.
- [14] J. Liu, M. Barreau, M. Čičić, and K. H. Johansson, "Learning-based traffic state reconstruction using probe vehicles," in *16th IFAC Symposium on Control in Transportation Systems (CTS)*, 2021.
- [15] M. J. Lighthill and G. B. Whitham, "On kinematic waves II. a theory of traffic flow on long crowded roads," *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, vol. 229, no. 1178, pp. 317–345, 1955.
- [16] C. M. Dafermos, *Hyperbolic Conservation Laws in Continuum Physics; 3rd ed.*, ser. Grundlehren der mathematischen Wissenschaften. Dordrecht: Springer, 2010.
- [17] B. D. Greenshields, J. R. Bibbins, W. S. Channing, and H. H. Miller, "A study of traffic capacity," in *Highway research board proceedings*, vol. 14. National Research Council (USA), Highway Research Board, 1935.
- [18] G. F. Newell, "A simplified theory of kinematic waves in highway traffic, part i: General theory," *Transportation Research Part B: Methodological*, vol. 27, no. 4, pp. 281–287, 1993.
- [19] 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.
- [20] R. M. Colombo and E. Rossi, "On the micro-macro limit in traffic flow," *Rendiconti del Seminario Matematico della Università di Padova*, vol. 131, 2014.
- [21] R. Chellappa, G. Qian, and Q. Zheng, "Vehicle detection and tracking using acoustic and video sensors," in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3. IEEE, 2004, pp. iii–793.
- [22] I. E. Lagaris, A. Likas, and D. I. Fotiadis, "Artificial neural networks for solving ordinary and partial differential equations," *IEEE transactions on neural networks*, vol. 9, no. 5, pp. 987–1000, 1998.
- [23] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
- [24] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.
- [25] S. Wang, Y. Teng, and P. Perdikaris, "Understanding and mitigating gradient pathologies in physics-informed neural networks," *arXiv preprint arXiv:2001.04536*, 2020.
- [26] M. Barreau, M. Aguiar, and K. H. Johansson, "On the efficiency of PINNs training depending on their nature," *in preparation*, 2021.
- [27] F. Fioretto, P. Van Hentenryck, T. W. K. Mak, C. Tran, F. Baldo, and M. Lombardi, "Lagrangian duality for constrained deep learning," *arXiv preprint arXiv:2001.09394*, 2020.
- [28] Y. Nandwani, A. Pathak, and P. Singla, "A primal dual formulation for deep learning with constraints," in *Advances in Neural Information Processing Systems (NeurIPS)*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 12 157–12 168.
- [29] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO – Simulation of Urban MObility; An Overview," *German Aerospace Centre, Institute for Transportation Research*, 2011.
- [30] R. Scheepens, N. Willems, H. Van de Wetering, G. Andrienko, N. Andrienko, and J. J. Van Wijk, "Composite density maps for multivariate trajectories," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2518–2527, 2011.