Transportation Research Part B 122 (2019) 309–326

Contents lists available at ScienceDirect

Transportation Research Part B

journal homepage: www.elsevier.com/locate/trb

Two-dimensional macroscopic model for large scale traffic networks

Stéphane Mollier[a], Maria Laura Delle Monache[b,*], Carlos Canudas-de-Wit[a], Benjamin Seibold[c]

[a] Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, GIPSA-Lab, 38000 Grenoble, France
[b] Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, GIPSA-Lab, 38000 Grenoble, France
[c] Department of Mathematics, Temple University, Philadelphia, PA 19122, US

ARTICLE INFO

Article history:
Received 19 June 2018
Revised 22 February 2019
Accepted 25 February 2019

Keywords:
Macroscopic traffic flow model
Two-dimensional PDE
Kernel density estimation
Validation & simulation

ABSTRACT

In this article we introduce a new traffic flow model for a dense urban area. We consider a two-dimensional conservation law in which the velocity magnitude is given by the fundamental diagram and the velocity direction is constructed following the network geometry and assuming we do not have precise information of drivers trajectories. We validate the model using synthetic data from Aimsun and propose a reconstruction technique to recover the 2D density from the data of individual vehicles. A comparison between the model and the data is shown.

1. Introduction

Different families of traffic models have been developed over the years (see van Wageningen-Kessels et al., 2014 for a review of model evolution). Microscopic models aim at representing each vehicle as a particle with dynamics based on Ordinary Differential Equations (ODEs). The first microscopic model called car-following was proposed in (Pipes, 1953; Kometani and Sasaki, 1961) with the idea that vehicles adapt their speed according to the position of the vehicle in front. Then, the global traffic state can be seen as a system of n coupled ODEs, where n is the number of vehicles in the network. These models allow to describe the behavior of every single driver but are not always used in practice due to their high computational cost and their difficulty to be calibrated as they require many parameters. Alternatively, macroscopic models do not model each individual vehicle but represent the traffic state as average quantities like vehicle density. In the thirties, Greenshields et al. (1934) found a link between vehicles density and flux. Following that work, a model inspired from fluid dynamics was introduced: the Lighthill, Whitham and Richards model (LWR model) (Lighthill and Whitham, 1955; Richards, 1956). This model is based on a partial differential equation and is able to describe dynamically the evolution of traffic density along a road. A discrete and easy to implement version, the Cell Transmission Model (CTM) which is equivalent to the Godunov method (Godunov, 1959) has been introduced in (Daganzo, 1994). This model has been extended, subsequently, to networks to model urban settings.

Subsequently, the extension to a network was developed in (Coclite et al., 2005), here the authors couple the LWR model with a junction model. The dynamics of the junction is modeled with an LP-optimization problem. Several others models

* Corresponding author.
E-mail address: ml.dellemonache@inria.fr (M.L. Delle Monache).

https://doi.org/10.1016/j.trb.2019.02.016
0191-2615/

have been introduced since then, see Garavello et al. (2016) and reference therein. These models assume knowledge of several parameters, for instance, the split ratio of vehicles at each intersection, which might be difficult to obtain, and also focus on a level of detail that my be not relevant for the size of the area considered that is why researchers on large scale traffic modeling started to develop models that describe traffic at a more aggregated level. In this framework, an important field of research concentrates on the notion of Macroscopic (or Network) Fundamental Diagram. Starting with some empirical observation of traffic in a city, Daganzo and Geroliminis (2008) and Geroliminis and Daganzo (2008) show that it is possible to exhibit a relation between the average density and the average flow over a whole network. This result enables the introduction of accumulation models — also called reservoir models — which consist of representing the traffic state of a network with a single scalar field variable representing the total number of vehicles in the network. These models are practical because they are understandable, with few parameters to tune and a low computational cost. However, they contain little information about the traffic states. For example, they are not able to describe precisely where vehicles are located over the reservoir. This problem was later on addressed in some papers in which the authors separated different areas of the city with different reservoirs, see for example Leclercq et al. (2015) and Hajiahmadi et al. (2013).

At the beginning of traffic modeling in urban areas, some continuous two dimensional model were investigated (Beckmann, 1952) but only in the case of static models. These authors represent the traffic state in the 2D-plane using a density of vehicles per area. An overview of static models in two dimensions is available in (Ho and Wong, 2006).

Finally, traffic in urban areas can be modeled with two-dimensional continuous and dynamic models. A review of some of these model have been done by Aghamohammadi and Laval (2018). These models represent the traffic density ρ as a variable over a 2D-plane $(x,\ y) \in \Omega$. Such models are based on a two-dimensional conservation law and take the following general structure:

$$\begin{cases} \dfrac{\partial \rho(t,x,y)}{\partial t} + \nabla \cdot \vec{\Phi}(\rho(t,x,y)) = 0 \\ \rho(0,x,y) = \rho_0(x,y) \end{cases}, \qquad \forall t \in \mathbb{R}^+, \forall (x,y) \in \Omega \tag{1}$$

where ρ is the aggregated density and $\vec{\Phi}$ the flow vector defined as the product of the density ρ and velocity field vector $\vec{v}$. This type of two-dimensional equations is commonly used in pedestrian modelling (Helbing, 1992; Hughes, 2002; Jiang et al., 2010). However, one should remark that crowds evolve in general in open space and are not constrained on roads as vehicles do. Thus, a first assumption for the application of this kind of equations to traffic is to consider that the urban network is dense enough to be approximated as a continuum. Generally, 2D models are not expected to describe very precisely the density evolution at space coordinates, but focus more on capturing the main traffic features and the global evolution of the density. The literature concerning this type of models is scarce, but there are several studies which start considering the problem. First, in Jiang et al. (2011) and Du et al. (2013), the authors take inspiration from pedestrian modeling in order to model vehicular traffic. They define the flux $\vec{\Phi}$ by solving an Eikonal equation such that the flow follows the path of the lowest cost — usually in terms of travel time, but other criteria could be used as well. Another extension of this model is done in Jiang et al. (2015). Their extension considers a second order equation such that it improves the description of the vehicle acceleration in the aim of pollutant estimation. In Romero Perez and Benitez (2008), an advection diffusion equation is introduced with a function of flux $\vec{\Phi}$ that depends on the space coordinates instead of depending on the density. Thus, the velocity is predetermined and the equation becomes linear. Another study (Della Rossa et al. (2010)) consider a model including a diffusion term and a drift term dependent on the density. The direction of the drift vector is fixed in some area and is determined by the shape of the network. In Saumtally (2012) and Sossoe and Lebacque (2016), the authors investigate the representation of intersection and how it could be interpreted in a 2D model. Following this idea, they take inspiration of junction models in one dimension like Lebacque and Khoshyaran (2004) and build extensions to 2D models. Lastly in (Herty et al., 2018; Chetverushkin et al., 2010; Sukhinova et al., 2009), the authors consider two-dimensional models for the case of multilane roads instead of a road network. Thus, the methodology and the model have several similarities but the objectives and the results are different. In Mollier et al. (2018), we consider a flux function consisting of a direction that depends on the space coordinates — as the one considered in this paper — and the simple fundamental diagram suggested first by Greenshields et al. (1934), however without any specific tuning of the parameters.

As 2D models are recent, there is little validation or calibration of these models. A first challenge in testing 2D models is to obtain a two-dimensional density function from real traffic data. In particular, the reconstruction of a density in the 2D-plane from vehicle data on the road network needs to be defined properly. The problem of reconstructing a probability density from observation is a well-known problem in statistics. One common method is Kernel Density Estimation (KDE) introduced first by Parzen (1962). Using this method, each data observation contributes to the density via a Gaussian centered at the position of the data observation. In Fan (2013) and Fan et al. (2014), the authors suggest to use this method for real traffic density but in the one-dimensional case when density represents a single road.

This article has two contributions. The first one is to present a 2D model with a geometry-dependent flux where the magnitude depends on the density and the direction depends on space. The second contribution is a methodology for the validation of 2D models using microsimulation. To this aim, a method to reconstruct a two-dimensional density from simulated data is described. A numerical method for the simulation is also shown and the results of comparison between the 2D model and the microsimulator Aimsun are presented. The organization of the paper follows these lines. In Section 2, we present the considered model with the construction of the flux function and the numerical method used for the simulation. Then in Section 3, we explain the methodology for the reconstruction of two-dimensional density. In Section 4, we deal

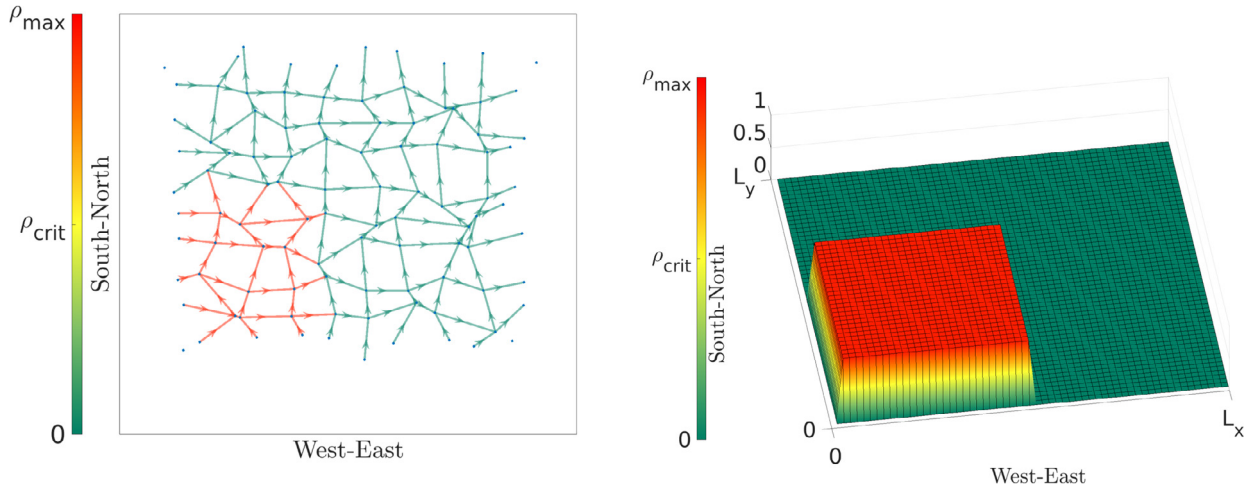


Fig. 1. Representation of an urban area as a road network (left) and as a continuum (right).

with the estimation from data of the parameters to construct a Fundamental Diagram consistent with the network. Finally in Section 5, we display the results of the comparison between the simulation of the 2D model and the equivalent scenarios run with the microsimulator Aimsun.

2. Model design

2.1. A two-dimensional conservation law for traffic modeling

In this paper we introduce a 2D model for traffic flow based on a two-dimensional conservation law. We consider a model in the two-dimensional plane where the density represents the number of vehicles per square area. An example can be seen in Fig. 1, giving a comparison between the same traffic situation described with a 1D density (left) and a 2D density (right).

Based on the different variables that could influence the flux function, several models could be designed. In this paper, we make the following assumptions :

1. The velocity magnitude is decreasing with respect to density.
2. The drivers do not adapt their paths with respect to density, i.e., there is no re-routing.
3. The maximal speed and the capacity is constant with respect to space.
4. The direction of the flow is given by the geometry of the network.

As most relevant roads are commonly bi-directional, this last assumption requires some justification. Large-scale urban traffic modeling is of particular relevance during (morning and afternoon) rush hour times of peak congestion. During those times, most urban areas exhibit a dominant direction of traffic flow (e.g., from the suburbs to downtown in the morning), and it is generally known (from historic data and/or travel demand modeling) which roads in the network carry this peak flow. In that spirit, we consider the following model.

$$\begin{cases} \dfrac{\partial \rho}{\partial t}(t,x,y) + \nabla \cdot \vec{\Phi}(x,y,\rho(t,x,y)) = 0, & \forall t \in \mathbb{R}^+, \forall (x,y) \in \Omega \subset \mathbb{R}^2 \\ \rho(0,x,y) = \rho_0(x,y). & \forall (x,y) \in \Omega \end{cases} \tag{2}$$

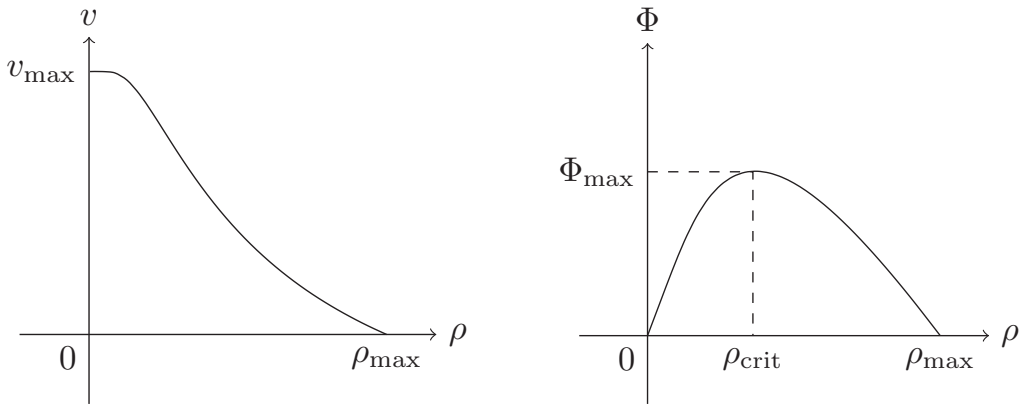
The density ρ is a two-dimensional quantity (number of vehicles per square meter) and it is a function of time t and space (x, y) defined in an area $\Omega \subset \mathbb{R}^2$. The flux function $\vec{\Phi}(x,y,\rho) : [0, \rho_{\max}] \times \Omega \to [0, \Phi_{\max}]$ is given by

$$\vec{\Phi}(x,y,\rho) = \rho \vec{v}(\rho) \tag{3}$$

where the velocity field $\vec{v}(x,y,\rho) : [0, \rho_{\max}] \times \Omega \to [0, v_{\max}]$ is given by

$$\vec{v}(x,y,\rho) = \underbrace{v(\rho)}_{\text{magnitude}} \cdot \underbrace{\vec{d}_\theta(x,y)}_{\textit{direction}} \tag{4}$$

We denote by θ the angle between $\vec{d_\theta}$ and the x-axis. The magnitude of the velocity, $v(\rho)$, is determined by the Fundamental Diagram (FD). In this paper, the FD chosen is the one introduced by Newell (1961) and Francklin (1961), whose velocity



(a) Relation velocity density (b) Relation flux density: FD

Fig. 2. Speed and flow vs. density.

function is:

$$v(\rho) = v_{\max}\left(1 - \exp\left(\frac{c}{v_{\max}}\left(1 - \frac{\rho_{\max}}{\rho}\right)\right)\right) \tag{5}$$

This specific FD has been considered as the best one to fit the data among the concave FD. This function possesses three parameters: The maximum velocity $v_{\max}$, the maximum density $\rho_{\max}$, and a velocity c that determines how rapidly the velocity decreases with increasing density, and thus affects the value of the critical density. The resulting flux function is strictly concave down. An example of this FD is displayed in Fig. 2. The existence and uniqueness of solutions of equation (2) are guaranteed under conditions of smoothness of the function flux Φ. We refer the reader to the papers of Kruzhkov (1970) (see p.223 for the conditions of Uniqueness and p. 230 for the Existence), Rossi (2017) for the scalar case in two space dimension as considered in this article.

2.2. Construction of the velocity direction field

In the model description (2), we have presented a velocity field that is the product of the a magnitude that depends on the density only, and a direction that depends on position only. In this section, we suggest one possible way to construct the direction function $\vec{d_\theta}$ from the geometry of the road network similarly to what is done in Mollier et al. (2018). The motivation behind this is due to the fact that, contrary to what happens with crowds Hughes (2002), vehicles are constrained to the physical road network.

Remark 1. This method is valid only for a traffic network defined by an oriented graph, i.e. without two directional roads. We assume as well, that we do not have any detail on driver trajectories. The split ratio, the origin-destination matrices or any information on driver behavior are unknown. The method presented here aims to analyze which traffic features a simple two dimensional model can capture in the situation where we just know that a preferred flow direction exists.

Before describing in detail how the function $\vec{d_\theta}$ is constructed we need to introduce some notation. We describe a road as a path from one intersection to another. We denote by $q \in \{1, ., Q\}$ the different roads of the network. The spatial path of each road is described by a parametric curve $\Psi^q : s \in [0, s_{\max}] \to (\Psi_1^q(s), \Psi_2^q(s)) \in \mathbb{R}^2$. The variable $s \in [0, s_{\max}]$ allows to progress along the road curvature from an intersection to the next one. Let $\vec{\tau^q}(\Psi^q(s))$ be the tangent vector of the road q at position $(\Psi_1^q(s), \Psi_2^q(s))$. For example in a network with only straight roads, this tangent vector is constant along each road. An example is given in Fig. 3. The estimation of the unit vector $\vec{d_\theta}$ at the discrete cell level is done by a spatial interpolation method called Inverse Distance Weighting:

$$\vec{d_\theta}(x,y) = \frac{\sum_{q=1}^{Q} \int_{s\in[0,1]} w(\|(x,y) - (\Psi_1^q(s), \Psi_2^q(s))\|) \quad \vec{\tau^q}(\Psi^q(s)) \quad ds}{\left\| \sum_{q=1}^{Q} \int_{s\in[0,1]} w(\|(x,y) - (\Psi_1^q(s), \Psi_2^q(s))\|)\, \vec{\tau^q}(\Psi^q(s)) \quad ds \right\|} \tag{6}$$

Eq. (6) constructs a direction at any point in the domain as a (normalized) weighted average of the road direction $\vec{\tau^q}$ of all points in the network, where the weight of a contributing point on a road depends on its distance to the evaluation point. The weight function $w : \mathbb{R}^+ \to \mathbb{R}^+$ should be a decreasing function of the distance. Here we use an exponential function:

$$w : X \to e^{-\beta X} \qquad \text{with} \qquad \beta > 0.$$

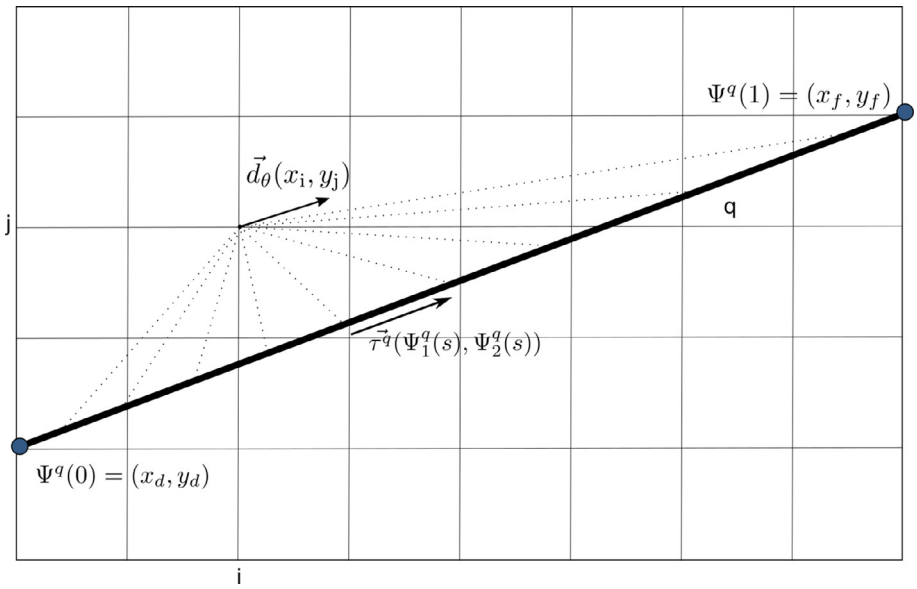


Fig. 3. Variables considered for the estimation of the velocity direction field. ($s_{\max} = 1$).

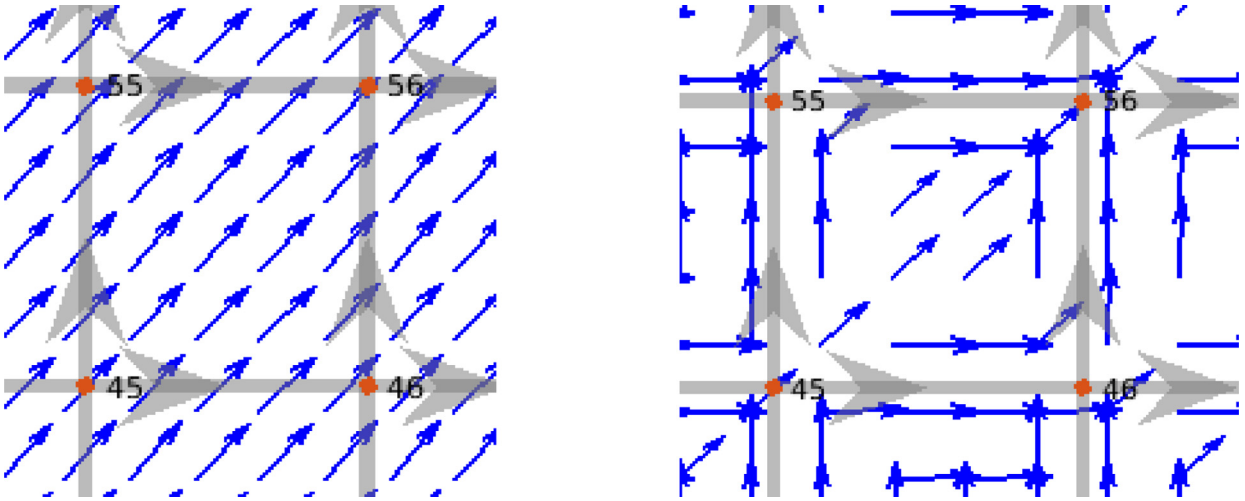


Fig. 4. Comparison between the estimation of the velocity direction for a small β (left) and large β (right).

However, other weight functions, particularly compactly supported ones, are also possible. The parameter β represents the localization of the weighted average: for small β, the velocity direction field provides only the global trend of the direction, while with a large β, the velocity field follows the detailed features of the roads. Fig. 4 demonstrates these two extreme cases in an example. As we are interested in a model that captures the large-scale two-dimensional vehicle transport, but without over-resolving the details of the network, we need to choose a value of β that lies between these two extremes.

Fig. 5 shows the velocity direction field on the network considered for simulation: as the β chosen is not large, the direction field is smooth and quite close to the global direction of the network which is towards the North-East direction. It is important to stress that for a general network, the weighted average (6) could potentially generate an undefined direction (due to the numerator and denominator vanishing). However, for dominant direction flow networks considered here, this scenario cannot happen.

2.3. Model discretization

2.3.1. Internal scheme

Numerical methods for conservation laws have been broadly studied in one dimension, multidimension and variable coefficient Toro (2013). In Lie (1999), the authors consider space-dependence for quasilinear equations and show that it

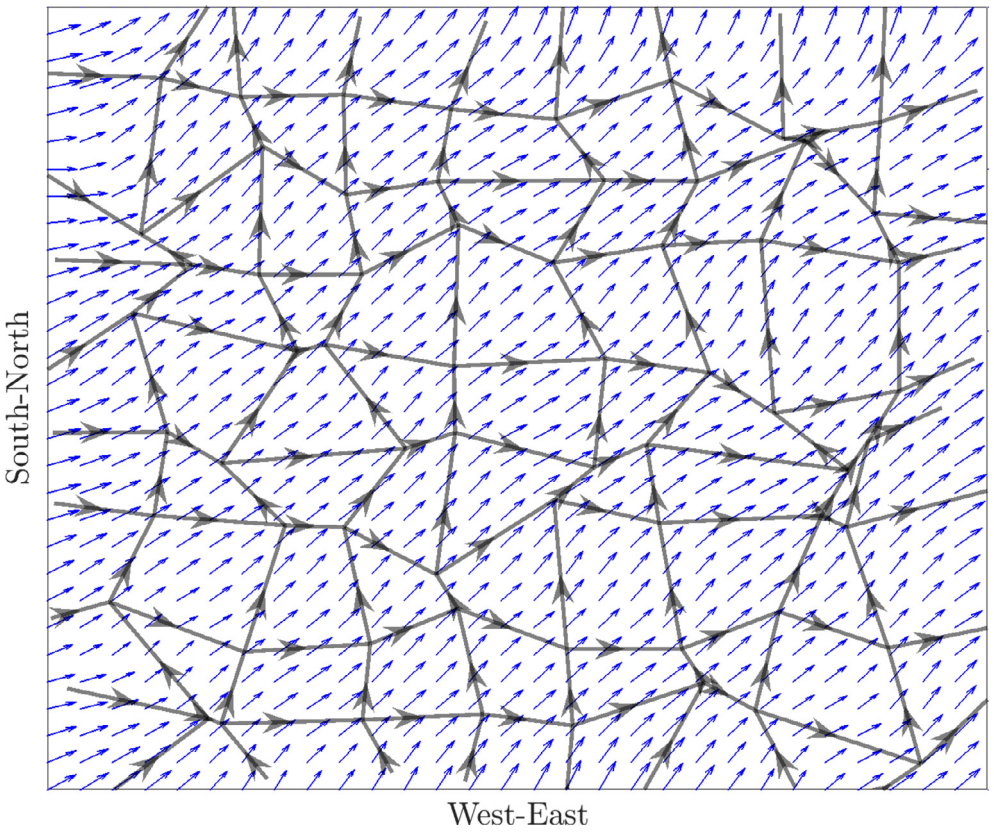


Fig. 5. Example of a velocity direction field in a modified Manhattan grid network.

is possible to use dimensional splitting if the flux function is bounded and Lipschitz continuous. As the space and density dependencies of the flux can be split, one can rewrite equation (2) to have an equation with a quasilinear term and a source term: $\forall t \in \mathbb{R}^+, \forall (x, y) \in \Omega$,

$$\frac{\partial \rho}{\partial t} + \underbrace{\cos\left(\theta\right)\frac{\partial\ \rho v(\rho)}{\partial x} + \sin\left(\theta\right)\frac{\partial\ \rho v(\rho)}{\partial y}}_{Quasilinear} = \underbrace{-\rho v(\rho)\left(\frac{\partial\ \cos(\theta)}{\partial x} + \frac{\partial\ \sin(\theta)}{\partial y}\right)}_{Source}. \tag{7}$$

The splitting method, or method of fractional steps, was considered first by Godunov (1959) and then properly introduced by Strang (1968). The principle of dimensional splitting is to compute separately the different term of the equation for each discrete interval of time. Thus, the equation of our model can be split in three parts:

$$\frac{\partial \rho}{\partial t} + \underbrace{\cos(\theta(x,y))\frac{\partial\ \rho v(\rho)}{\partial x}}_{1} + \underbrace{\sin(\theta(x,y))\frac{\partial\ \rho v(\rho)}{\partial y}}_{2} = \underbrace{-\rho v(\rho)\left(\frac{\partial\ \cos(\theta)}{\partial x} + \frac{\partial\ \sin(\theta)}{\partial y}\right)}_{3}.$$

Then, the dimensional and operator splitting consist of dividing for each time step, the computation of the solution by 3 steps. In the first step, the propagation of the density along the x-coordinates is computed. Then in the second step, the propagation of density along the y-coordinates is updated. Finally, using the operator splitting method (Toro, 2013; Gosse, 2014) the source term is taken into account.

For each dimension, the flux is computed using the Godunov scheme (Godunov, 1959). Note that the splitting approach presented here is only one, simple, way to discretize the model equations. Other, potentially more efficient computationally, discretizations are possible. In the numerical results, the numerical approximation is conducted with a fine spatio-temporal resolution, so that the errors due to the discretization are negligible compared to the model and upscaling errors.

Let $(C_{i,j})_{(i,j)\in[1..I]\times[1..J]}$ be the cell space discretization and $(x_{C_{i,j}}, y_{C_{i,j}})$ be the coordinates of each cell center. Applying the method given in Section 2.2, we can define a flux direction for each cell $\vec{d}_\theta(x_i, y_j)$. Let us define the density in the cells with $(\rho_{i,j})_{(i,j)\in[1..I]\times[1..J]}$, then the numerical flux is defined at cell interfaces with the notation $F_{i+\frac{1}{2},j} = F(\rho_{i,j}, \rho_{i+1,j})$ and the

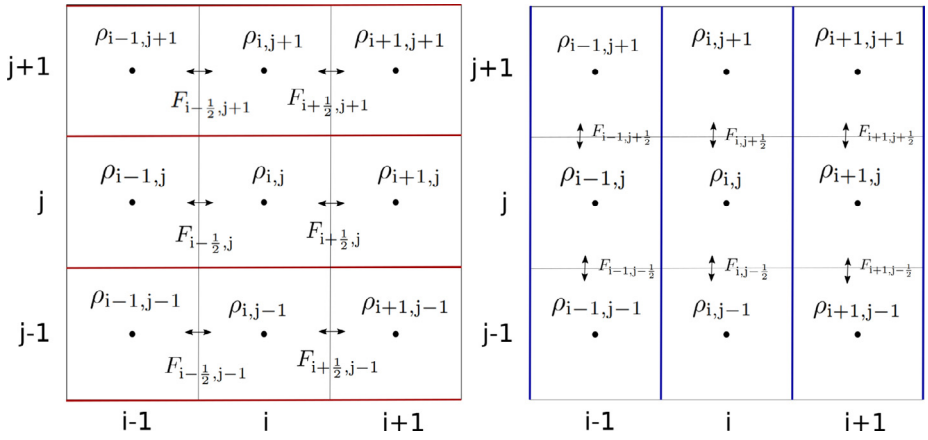


Fig. 6. Application of the dimensional splitting with the representation of the cell interfaces.

function F is defined as follows:

$$F(\rho_{i,j},\rho_{i+1,j}) = \begin{cases} \min(\Phi(\rho_{i,j}),\Phi(\rho_{i+1,j})), & \text{if } \rho_{i,j} < \rho_{i+1,j} \\ \Phi(\rho_{i,j}), & \text{if } \rho_{i+1,j} \le \rho_{i,j} \le \rho_{\text{crit}} \\ \Phi(\rho_{i+1,j}), & \text{if } \rho_{\text{crit}} \le \rho_{i+1,j} \le \rho_{i,j} \\ \Phi_{\max}, & \text{if } \rho_{i+1,j} \le \rho_{\text{crit}} \le \rho_{i,j}. \end{cases} \tag{8}$$

The vertical flux $F_{i,j+\frac{1}{2}}$ is defined analogously.

Let Δt be the time step, and Δx and Δy the space discretization with respect to the x-axis and the y-axis, respectively. The time step Δt is chosen in practice to respect the Courant–Friedrichs–Lewy (CFL) condition in order to garantee the stability of the numerical scheme. In two dimensions, this condition can be described as follows:

$$\Delta t \le \frac{\Delta x \Delta y}{(\Delta x + \Delta y) v_{\max}}. \tag{9}$$

Then the global scheme for the computation of the model can be defined as follows:

$$\rho^{*}_{\mathrm{i,j}} = \rho^{\mathrm{n}}_{\mathrm{i,j}} - \cos(\theta_{\mathrm{i,j}})\frac{\Delta t}{\Delta x}(F^{n}_{\mathrm{i}+\frac{1}{2},\mathrm{j}} - F^{n}_{\mathrm{i}-\frac{1}{2},\mathrm{j}}), \tag{10}$$

$$\rho^{**}_{\mathrm{i,j}} = \rho^{*}_{\mathrm{i,j}} - \sin(\theta_{\mathrm{i,j}})\frac{\Delta t}{\Delta y}(F^{*}_{\mathrm{i,j}+\frac{1}{2}} - F^{*}_{\mathrm{i,j}-\frac{1}{2}}), \tag{11}$$

$$\rho^{\mathrm{n}+1}_{\mathrm{i,j}} = \rho^{**}_{\mathrm{i,j}} - m(\rho^{**})\left(\frac{\Delta t}{\Delta x}(\cos(\theta_{\mathrm{i}+\frac{1}{2},\mathrm{j}}) - \cos(\theta_{\mathrm{i}-\frac{1}{2},\mathrm{j}})) + \frac{\Delta t}{\Delta y}(\sin(\theta_{\mathrm{i,j}+\frac{1}{2}}) - \sin(\theta_{\mathrm{i,j}-\frac{1}{2}}))\right). \tag{12}$$

Here $\theta_{\mathrm{i,j}}$ is the angle of the unit vector direction of the flux $\vec{d}_{\theta}(x_{\mathrm{i}}, y_{\mathrm{j}})$ defined in Section 2.2 at cell $C_{i,j}$, and $cos(\theta_{\mathrm{i}+\frac{1}{2},\mathrm{j}})$ is equal to $\frac{cos(\theta_{\mathrm{i}+1,\mathrm{j}})+cos(\theta_{\mathrm{i,j}})}{2}$ the average between cosinus of the angles in cell $C_{i,j}$ and $C_{i+1,j}$. Fig. 6 shows a graphical representation of the dimensional splitting, on the left the propagation of the density and the flow interface in the x-axis, and on the right the y-axis propagation. In summary, the splitting separates the computation of every time iteration into three steps: propagation along the horizontal axis, propagation along the vertical axes, and source term.

2.3.2. Boundary conditions

The boundary conditions are defined by using Ghost Cells. The concept of this method is to compute the flux crossing the boundary $F_{\mathrm{l_x}+\frac{1}{2},\mathrm{j}}$ by considering fictitious cells $\rho_{\mathrm{l_x}+\frac{1}{2},\mathrm{j}}$ outside of the domain. The method have been introduced first by Fedkiw et al. (1999). The representation of this situation are displayed in Fig. 7.

The flux across the boundary$F_{\mathrm{l_x}+\frac{1}{2},\mathrm{j}} = F(\rho_{l_x,j}, \rho_{l_x+1,j})$ is defined in the same way as the internal numerical flux using the internal density at the boundary and the fictitious ghost cell outside the domain.

3. Reconstruction of a 2D density from individual vehicle trajectories of microsimulation

Two-dimensional model validation is a topic that is not commonly addressed in the literature: while many references can be found concerning the validation and reconstruction of 1D density, only few references exists for two-dimensional

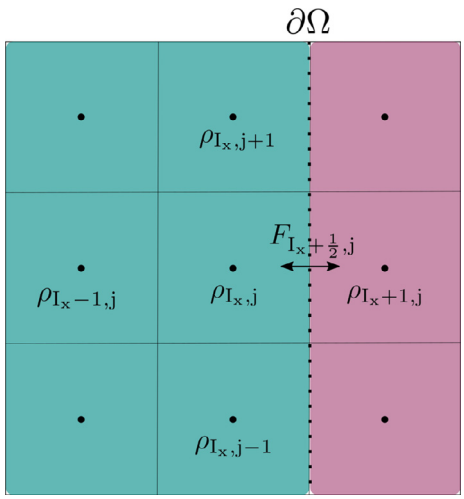


Fig. 7. Representation of the boundary conditions using Ghost Cell: the domain Ω is represented in green and the area outside the domain is represented in purple. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

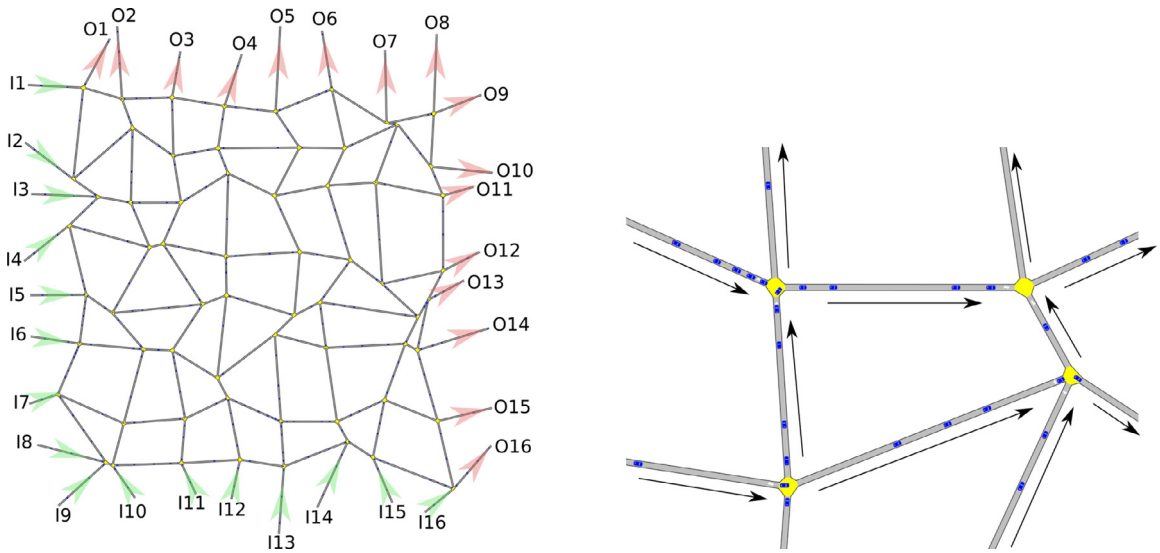


Fig. 8. Manhattan grid oriented towards the North-East direction in Aimsun (left) and a zoom of the road network (right).

models. In Della Rossa et al. (2010), the authors compare their results against microsimulation but without providing a systematic methodology. In Herty et al. (2018), the authors compare their model with real data from a highway, however, for the case of a multilane road instead of a road network. In this paper we propose a comparison between the model that we introduce and a reconstructed density obtained from microsimulations done in Aimsun (https://www.aimsun.com/). We start the simulations with same initial conditions and we want to compare the evolution of the main traffic density with respect to time. A first question that arises is the exact definition of density over the plane and how to estimate it from data. In this Section, we show a method to reconstruct density from GPS logs and then use this reconstruction to compare microscopic and macroscopic model results.

3.1. Data collection from Aimsun

We consider several scenarios in the microsimulator Aimsun. We use a test network, shown in Fig. 8, of a deformed 10×10 Manhattan grid, laid out on a $1\,km \times 1\,km$ domain. To cut the regularity and therefore have a more generic network, we start from a regular Manhattan grid and add a normally distributed perturbation (of standard deviation 30 m) to the position of each node. Horizontal edges go East-bound, and vertical edges go North-bound. In line with this, all roads in the West and South are entrance roads ("I" number 1 to 16), whereas the roads in the East and North are exit roads ("O" number 1 to 16). Each network edge is a single-lane road. Vehicles are assumed to have a length of 4 m and the minimum

gap between two vehicles is 2 m. This results in a road capacity of 166.67 vehicles per kilometer. It is important to note that this 1D road capacity does not equal the maximum density of the 2D model. The time step of the Aimsun simulation $\Delta t_{\text{Aim}} = 0.8$s. During the simulations, the vehicles' positions in the network are collected at each time step, yielding the data points x_k^n, y_k^n for vehicle indices $k \in [1, \dots, K(n)]$ and at times $t^n = n\Delta t_{Aim},\ n \in [0, \dots, N]$.

3.2. Density estimation with Gaussian kernel

In this section, the Kernel Density Estimation (KDE) method is presented. The idea of this method is that each observation has a spatial contribution to the estimated density, defined by the function chosen as kernel. The final estimated density then corresponds to the superposition of all those contributions. Note that, in contrast to the estimation of probability densities that are normalized to integrate to 1, the resulting vehicle density is scaled to integrate to the total number of vehicles. We start with the one-dimensional case to establish the concepts and notations.

Reconstruction of density with kernel estimation in 1D:

Let x_k^n be the position of vehicle k at time t^n. Then, the density at that time can be estimated as follows:

$$\tilde{\rho}^n(x) = \sum_{k=1}^{K(n)} G_{1d}(x - x_k^n) \tag{13}$$

where G_{1d} is the kernel function used to describe the contribution of each vehicle. We choose the kernel to be a Gaussian function:

$$G_{1d}(x) = \frac{e^{-\frac{x^2}{2d_0^2}}}{\sqrt{2\pi}d_0} \quad \text{which satisfies} \quad \int_{\mathbb{R}} G_{1d}(x)\, dx = 1$$

Here the parameter d_0 is a length scale determining the width of the Gaussian. The integral of the estimated density over space is equal to the number of vehicles:

$$\int_{\mathbb{R}} \tilde{\rho}^n(x)\, dx = \text{Number of vehicles on the road at time } t^n. \tag{14}$$

Reconstruction of density with kernel estimation in 2D:

Let $(x_k^n, y_k^n)_{k\in[1,\dots,K(n)],n\in[0,\dots,N]}$ be the position of the vehicles at time t^n. Then the density over the 2D-plane can be estimated as follows:

$$\tilde{\rho}^n(x, y) = \sum_{k=1}^{K(n)} G_{2d}\left(\begin{pmatrix} x \\ y \end{pmatrix} - \begin{pmatrix} x_k^n \\ y_k^n \end{pmatrix}\right) \tag{15}$$

with

$$G_{2d}(x, y) = \frac{e^{-\frac{x^2+y^2}{2d_0^2}}}{2\pi d_0^2} \quad \text{which satisfies} \quad \iint_{\mathbb{R}^2} G_{2d}(x, y)\, dx\, dy = 1.$$

The density $\tilde{\rho}_{i,j}^n$ in the cell $C_{i,j}$ is estimated with the value at the center of the cell. An example of the reconstruction of the density is given in Fig. 9.

3.3. Parameters fitting for the kernel reconstruction

In practice, the parameter d_0, which determines the range impact of the Gaussian kernel, has to be properly chosen. There are several works regarding the optimal choice of this parameter including some applications on traffic modeling, see for example (Fan et al., 2014; Fan, 2013). In these articles, the authors study the choice of this parameter for an application of the kernel density approximation for a 1D model of traffic. They based their choice of the parameter d_0 on the idea that if the headway between vehicles is constant then the reconstructed density should be constant as well. However, there must be an exception to this principle when the density of vehicles is very low. An example is given in Fig. 10.

We can observe that a choice of a small d_0 (smaller than the vehicle spacing) leads to the creation of peaks in the reconstructed density. Conversely, a d_0 chosen too high invalidates the reconstructed density due to boundary effects and leads to a bell-shaped reconstruction. For an intermediate value of d_0, the reconstructed density is almost constant, with only small layers of reduced density near the boundaries.

Next we try to extend this idea for density reconstructions in two dimensions. First, let us consider equidistant vehicles in the 2D-plane without taking into account the network (e.g., vehicles on a parking lot). The density reconstructed in this case can be seen in Fig. 11.

The same qualitative effects as in one dimension can be observed in the 2D case. When the d_0 is chosen too large or too small, the reconstructed density tends to a bell or to have multiple peaks, respectively. If d_0 takes an intermediate value, then the reconstructed density is almost constant in space and has the shape of a plateau.

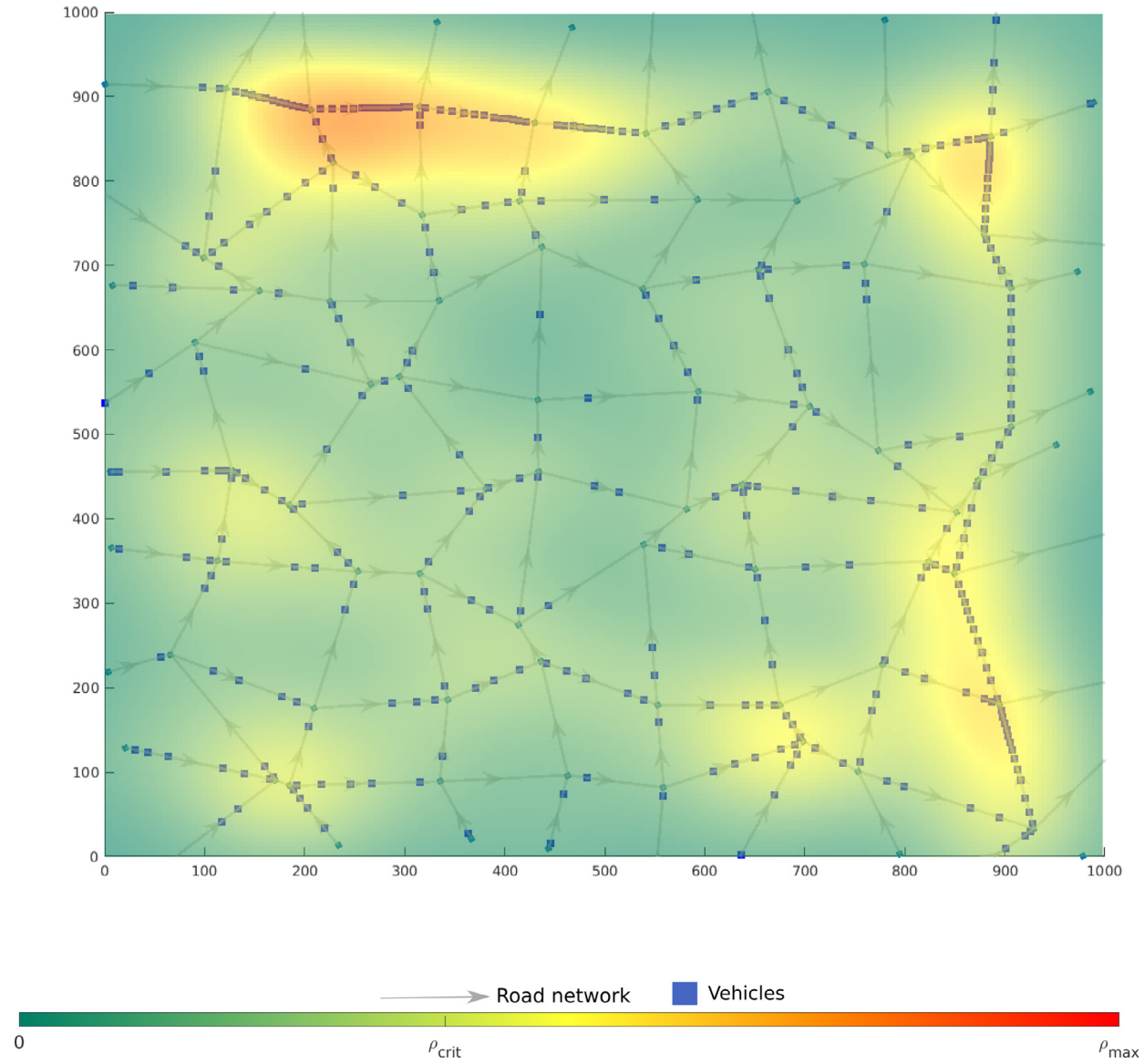


Fig. 9. Example of 2d density reconstruction from data by kernel density estimation method: the density is represented by the colormap, the blue squares represent the positions of vehicles and the considered network can be seen in the background. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Next we consider the test network and place vehicles equidistantly along the roads. This means that on the whole network, we consider that vehicles are distributed with a constant spacing. Fig. 12 shows the result of the reconstructed density for this scenario. In the cases that d_0 is too high or too low, we face the same problem than encountered previously.

In practice, it seems difficult to obtain a totally flat estimation of the density in this case even when d_0 is chosen with an intermediate value. Thus, the parameter d_0 is chosen by an optimization process. Consider vehicles placed at a minimal distance — one vehicle every 6m — over the network and let $\tilde{\rho}$ be the corresponding estimated density. The parameter d_0 is chosen such that it reduces as much as possible the distance between maximum of the density evaluated over Ω and the rest of the estimated density.

$$d_0 = \underset{d_0 \in \mathbb{R}_+^*}{\arg\min} \| \tilde{\rho}(.,.) - \max_{(x,y)\in\Omega} \tilde{\rho}(x,y) \|_2 \tag{16}$$

The density $\tilde{\rho}$ depend on the parameter d_0 in a non linear way given by the Eq. (15). For the 10×10 Manhattan square grid of 1km length considered, the resulting value of d_0 is 85.1 m. An important remark is that the method suggested for the estimation of d_0 is required in order to remain consistent with the assumption that the FD is defined as constant over the 2D-plane.

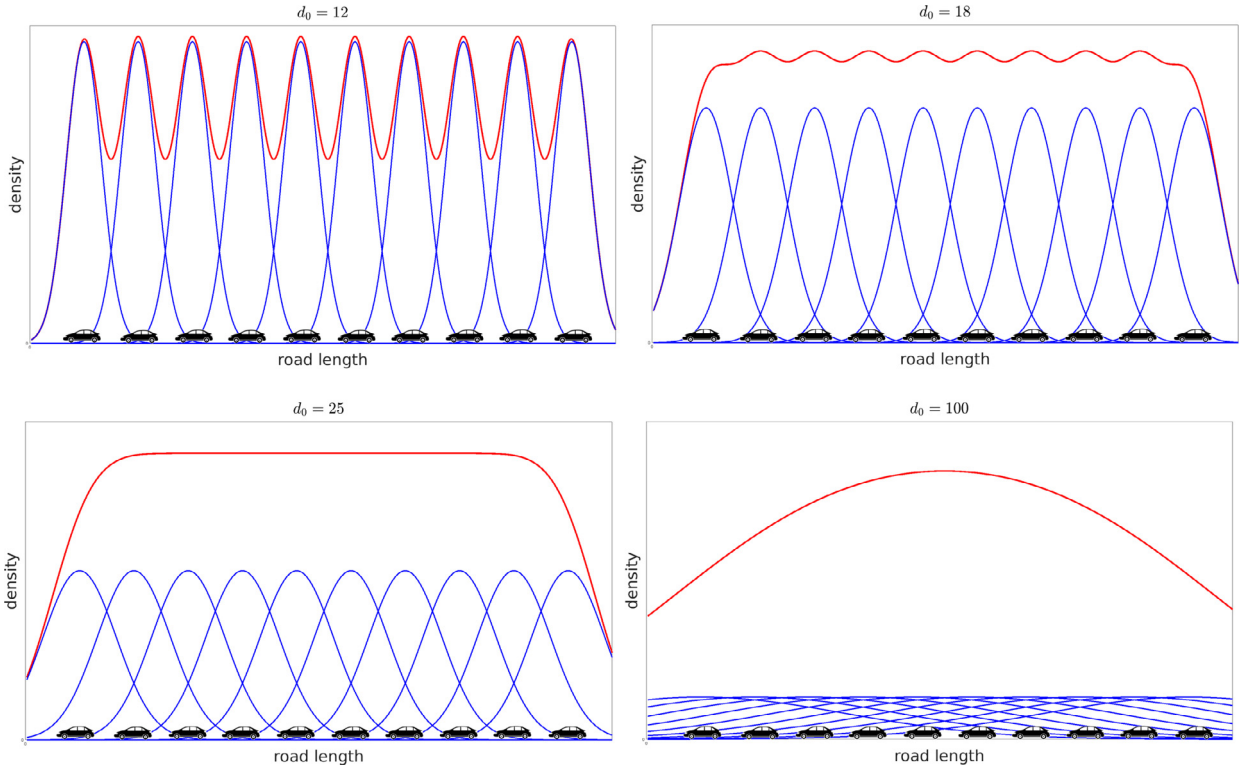


Fig. 10. Reconstruction of density in 1D for vehicles with a constant headway of 37 m and for a d_0 of respectively 12 m, 18 m, 25 m, and 100 m.

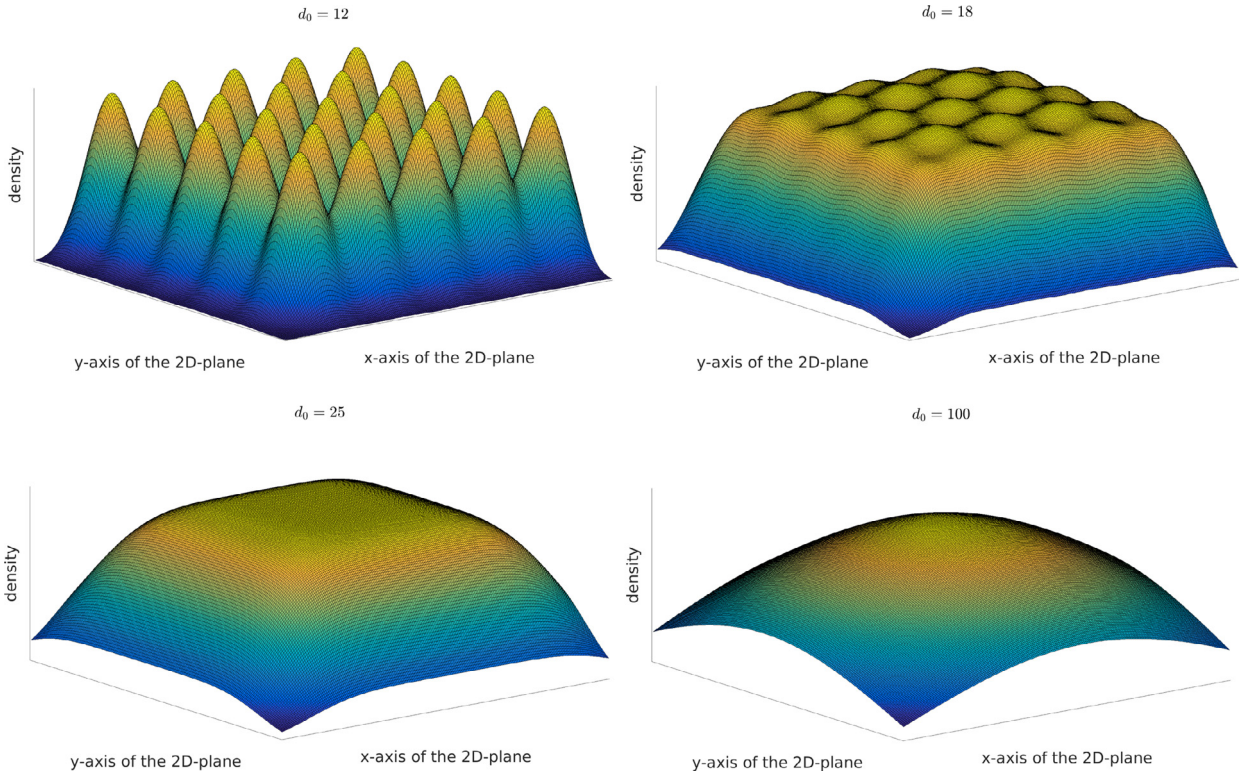


Fig. 11. Reconstruction of 2D density for vehicles uniformly distributed every 37 m in the 2D-plane and for d_0 of respectively 12 m, 18 m, 25 m, and 100 m.

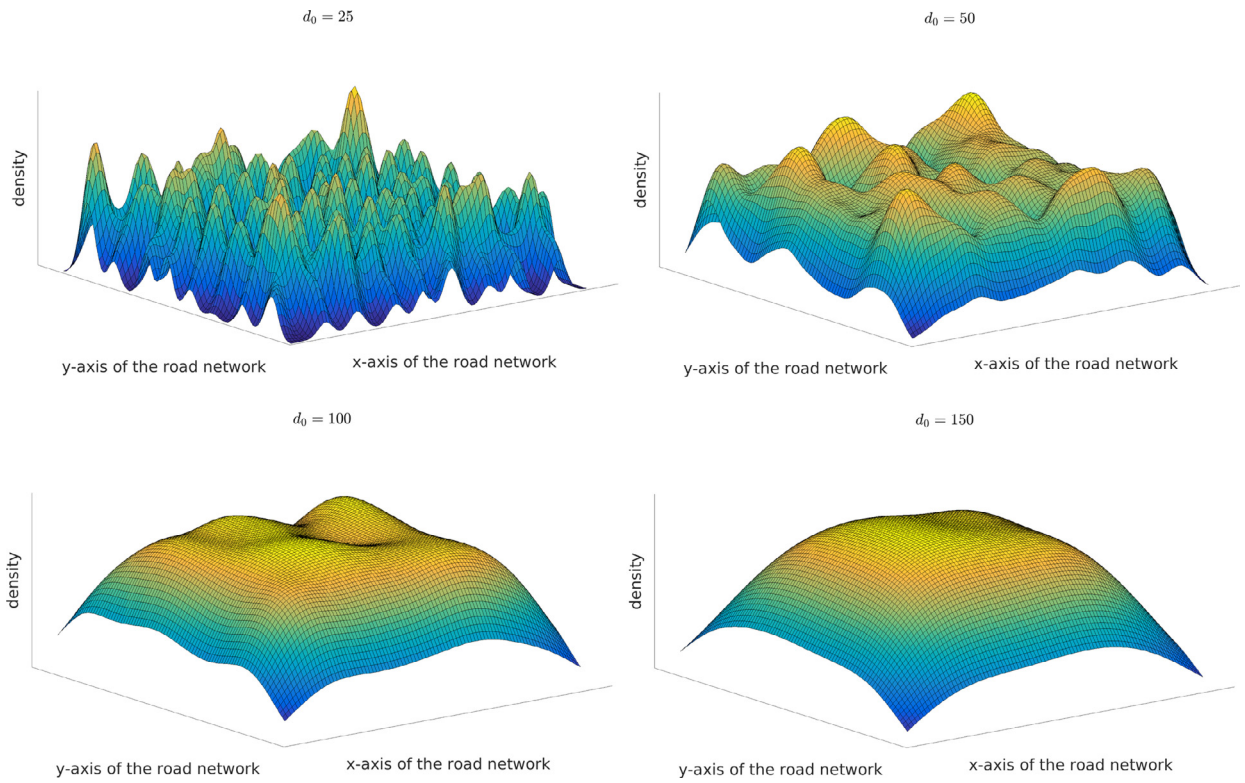


Fig. 12. Reconstruction of 2D density for vehicles with a constant spacing of 37 m on the network and for d_0 of respectively 25 m, 50 m, 100 m, and 150 m.

4. Parameter tuning

4.1. Estimation of the fundamental diagram

Since we consider densities as number of vehicles per meter square, we need to introduce a new way to estimate from synthetic data the fundamental diagram for the two-dimensional models. The definition of a fundamental diagram is closely linked to the definition of density. In Section 3.2, a way to reconstruct density from traffic data was presented. Thus, it is consistent that the fundamental diagram is also based on the kernel reconstruction method. For clarity purposes, one may emphasize that this FD aims to be applied for network modeling, but it is not related to the notion of Macroscopic Fundamental Diagram (MFD). Indeed, this FD links the two dimensional density and the velocity of vehicles and it is applied locally.

The data for the construction of the FD is collected from a simulation in Aimsun in which we recorded the position and the instantaneous velocity of vehicles every 0.8s. We are then able at each time step, to reconstruct an estimated density with the kernel method defined Section 3.2. In order to construct a FD we need to reconstruct a velocity or a flux over the 2D plane using the data collected from Aimsun as well. In particular, we construct an interpolation of the velocity of all vehicles. Let $(v_k^n)_{(k,n)\in[1,\dots,K(n)]\times[1,\dots,N]}$ be the speeds of the vehicles of subscript k over the $K(n)$ vehicles present in the network at time t^n. The corresponding positions of these vehicles are $(x_k^n, y_k^n)_{(k,n)\in[1,\dots,K(n)]\times[1,\dots,N]}$. Using these data, we can — consistent with (15) — estimate the density $(\tilde{\rho}_{i,j}^n)_{(i,j)\times n\in[1,\dots,I]\times[1,\dots,J]\times[1,\dots,N]}$ by the Kernel Density Estimation method:

$$\tilde{\rho}_{i,j}^n = \sum_{k=1}^{K(n)} G_{2d}\left(\begin{pmatrix} x_{C_{i,j}} \\ y_{C_{i,j}} \end{pmatrix} - \begin{pmatrix} x_k^n \\ y_k^n \end{pmatrix}\right) \tag{17}$$

We can also estimate the velocity field $(\tilde{v}_{i,j}^n)_{(i,j)\times n\in[1,\dots,I]\times[1,\dots,J]\times[1,\dots,N]}$, by interpolation of each individual vehicle speed and then deduce the flow rate field $(\tilde{\Phi}_{i,j}^n)_{(i,j)\times n\in[1,\dots,I]\times[1,\dots,J]\times[1,\dots,N]}$ respectively, as follows:

$$\tilde{v}_{i,j}^n = \frac{\sum_{k=1}^{K(n)} G_{2d}\left(\begin{pmatrix} x_{C_{i,j}} \\ y_{C_{i,j}} \end{pmatrix} - \begin{pmatrix} x_k^n \\ y_k^n \end{pmatrix}\right) v_k^n}{\left|\sum_{k=1}^{K(n)} G_{2d}\left(\begin{pmatrix} x_{C_{i,j}} \\ y_{C_{i,j}} \end{pmatrix} - \begin{pmatrix} x_k^n \\ y_k^n \end{pmatrix}\right)\right|} \quad \text{and} \quad \tilde{\Phi}_{i,j}^n = \tilde{v}_{i,j}^n \, \tilde{\rho}_{i,j}^n \tag{18}$$

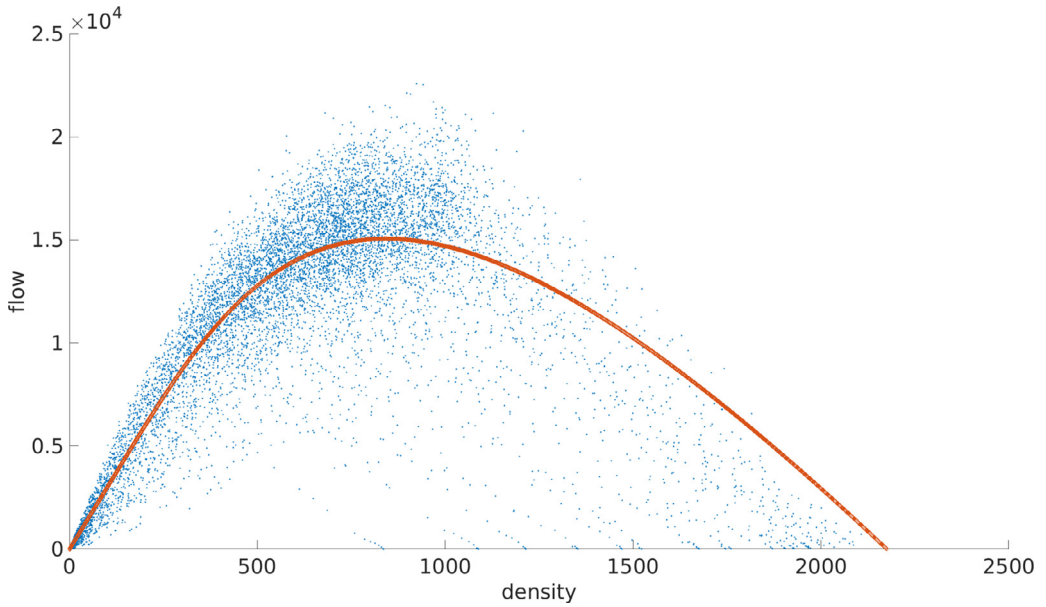


Fig. 13. Estimation of the fundamental diagram with the function of Newell and Franklin.

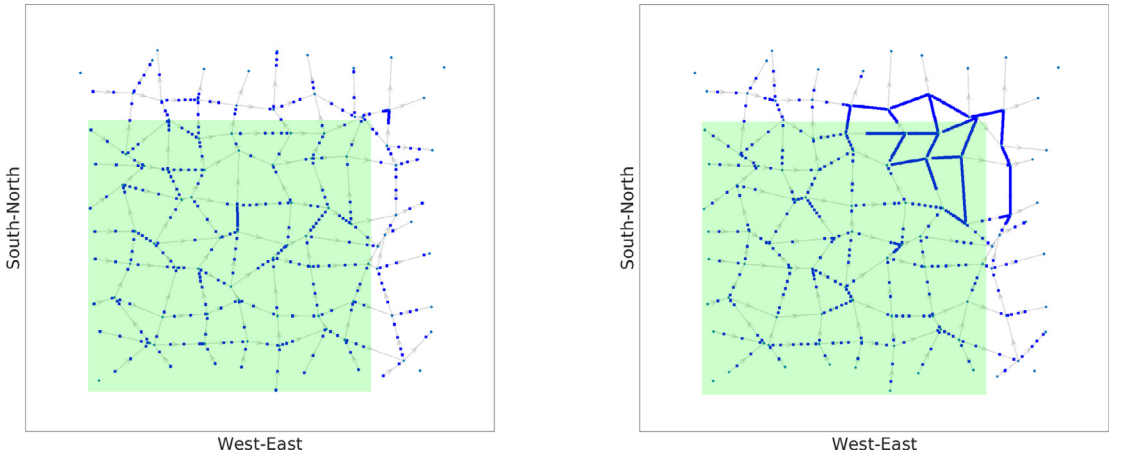


Fig. 14. Distribution of vehicles during the Aimsun simulation for an individual run of the scenario 1 at the initial and final time of the simulation.

The number of data observation we obtain with the method presented just above can be really huge: for each discrete point and each time step we have a measure. This leads to redundant data, difficulty to read data and inconsistency of the estimation. For these reason, we apply a sampling over time and an aggregation over space. Let s_t the ratio between the time step of the sampling and the time step of the simulation. Let H and L be the number of cells aggregated over space. We denote by $(\bar{\rho}^n_{i,j})_{(i,j)\times n\in[1,\dots,\bar{I}]\times[1,\dots,\bar{J}]\times[1,\dots,\bar{N}]}$ and $(\bar{\Phi}^n_{i,j})_{(i,j)\times n\in[1,\dots,\bar{I}]\times[1,\dots,\bar{J}]\times[1,\dots,\bar{N}]}$ respectively the density and the flow after the sampling and aggregation and constructed as follows:

$$\bar{\rho}^n_{i,j} = \frac{1}{H.L}\sum_{h=H.(i-1)+1}^{H.i}\ \sum_{l=L.(j-1)+1}^{L.j} \tilde{\rho}^{s_t.n}_{h,l} \quad \text{and} \quad \bar{\Phi}^n_{i,j} = \frac{1}{H.L}\sum_{h=H.(i-1)+1}^{H.i}\ \sum_{l=L.(j-1)+1}^{L.j} \tilde{\Phi}^{s_t.n}_{h,l} \tag{19}$$

where $s_t = 20$, $H = 10$ and $L = 10$.

Applying these reconstructions for all data, a fundamental diagram density–flow relation can be obtained. The result is shown as the points in Fig. 13. We then fit a function $\tilde{\Phi}$ to these data points that satisfies the following constraints:

1. The flux is zero for a vanishing density: $\tilde{\Phi}(0) = 0$.
2. The flux returns to zero when the density reaches its maximum: $\tilde{\Phi}(\rho_{\max}) = 0$
3. The function $\tilde{\Phi}$ must be a concave down function.
4. The function $\tilde{\Phi}$ must be smooth.

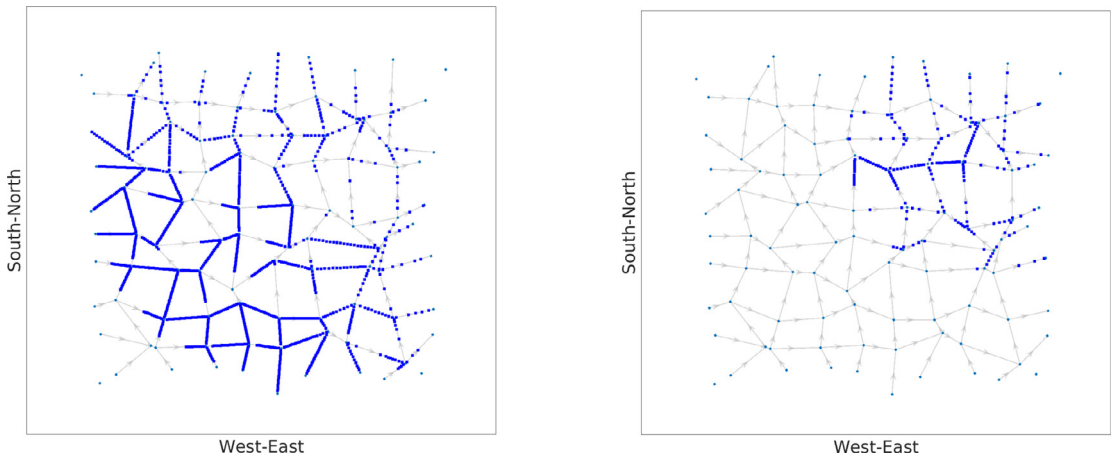


Fig. 15. Distribution of vehicles during the Aimsun simulation for an individual run the scenario 2 at the initial and an interdiate time of the simulation.

The two first constraints ensure that the density remains between 0 and $\rho_{\max}$. For a review of existing fundamental diagram and their properties we refer the reader to the paper of Carey and Bowers (2012). For our case, we consider the FD suggested by Newell (1961) and Francklin (1961) that satisfies the above properties and that seems the most appropriate to the shape of the data we have:

$$\Phi(\rho) = v_{\max}\rho\left(1 - \exp\left(\frac{c}{v_{\max}}(1 - \frac{\rho_{\max}}{\rho})\right)\right) \tag{20}$$

For the simulations, we consider that the maximum density $\rho_{\max}$ is fixed and corresponds to the maximum density reconstructed (before doing sampling and aggregation) of the data collected from the simulation.

$$\rho_{\max} = \max_{(i,j)\times n} \tilde{\rho}^n_{i,j}, \qquad (i,j)\times n \in [1,\dots,I]\times[1,\dots,J]\times[1,\dots,N] \tag{21}$$

The estimation of the parameters $v_{\max}$ and c is done using the aggregated and sampled data. The optimization problem is solve with a toolbox of Matlab considering the following minimization problem:

$$\arg\min_{v_{\max},c}\left(\sum_{i=1}^{\bar{I}}\sum_{j=1}^{\bar{J}}\sum_{n=1}^{\bar{N}}\|\Phi(\tilde{\rho}^n_{i,j}) - \bar{\Phi}^n_{i,j}\|_2^2\right) \tag{22}$$

The scenario of simulation considered for the collection of data is the following. We consider the network described in Fig. 8 initially congested. Then we open the network exits and stop the inflows. We collect data during the dissipation of the congestion which lasts 15 min. Every 0.8 s, measures are collected from vehicle position and speed over the whole space. The aggregated and sampled data can be seen in Fig. 13. We can emphasize that these data are only used for the estimation of $v_{\max}$ and c whereas $\rho_{\max}$ is given as the maximum of the untreated data. The curve is obtained for the value $\rho_{\max} = 2175$ vehicles/km^2, $v_{\max} = 29.9110$ km/h and $c = 17.2089$ km/h.

5. Simulation results

5.1. Description of the scenario

In Section 3, we established a way to reconstruct a two-dimensional density from data of the microsimulator Aimsun. We are now able to compare a simulation of the 2D model with the microsimulator. With this aim, we build two scenarios. Recall that we assume that we have only a flow oriented towards the North-East direction for all simulation. The network is the same deformed 10×10 Manhattan grid as before. The minimum spacing between vehicles along the road is 6m. For the two scenarios, the parameters of the 2D model are the same as the ones estimated in the previous sections: $d_0 = 85.1$ m, $\rho_{\max} = 2175$ vehicles/km^2, $v_{\max} = 29.911$ km/h, and $c = 17.2089$ km/h. The time step of Aimsun microsimulator is equal 0.8 s and it could not be changed because the reaction time of drivers rely on this value. For this reason, the time step Δt of the 2D model has to be a multiple of this value. The discretization Δx is chosen in order to respect the CLF condition.

In the Aimsun simulator, there is an important role of randomness (for instance at each intersection, the direction taken by a vehicle is a random variable with a probability dependent of the turning ratios). This implies that different runs of the same scenario could yields noticeably different results on a micro-scale. Thus, some congestion at a local level may appear at different place and time in two different run of microsimulation. Thus, to improve the robustness of our model comparison, we compute 100 different runs of the Aimsun simulation for each scenario. For each of these runs, we reconstruct a 2D density and then we use the average of these reconstructed densities as a means of comparison for our 2D model.

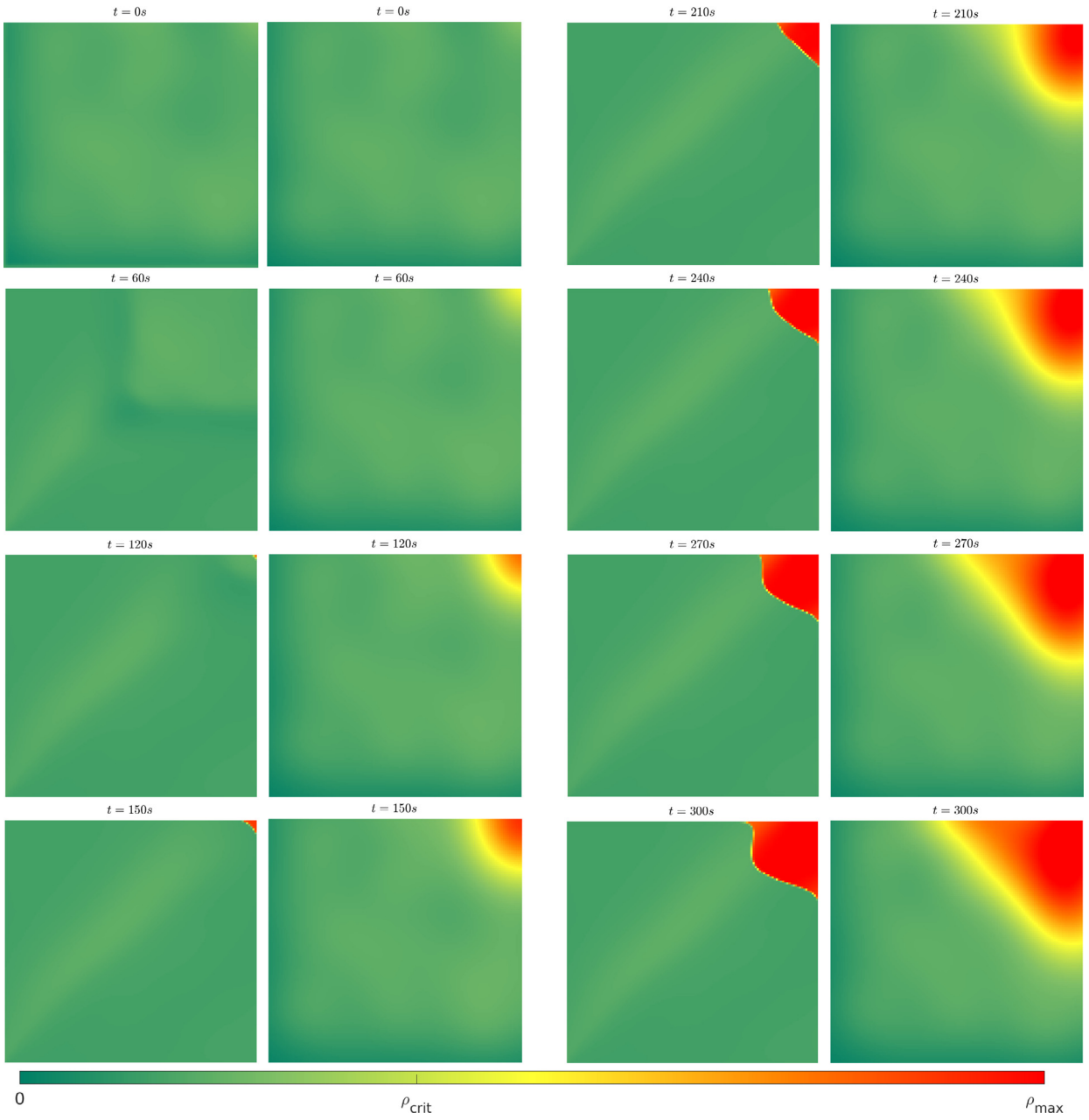


Fig. 16. Comparison of two-dimensional (left) and the average microsimulator behavior (right) during the creation of a congestion. Video of the full simulation available at https://youtu.be/Y9RGLFTIGSs.

Scenario 1: Creation of a congestion. We set up Aimsun that there is an accident in the North-East corner of the domain, which causes a congestion to propagate over the network that is initially in free flow. It is not possible to define directly boundary condition in a comparable way for the two-dimensional model and the Aimsun simulator. For this reason, the comparison between the 2D model and Aimsum microsimulation is done on a subdomain, a square of 800m which starts in the left bottom corner, of the real network defined in the microsimulator. Outside this subdomain, a two dimensional density is reconstructed from the microsimulation by the method described in Section 3.2. Finally, this density is used to feed the Ghost Cell of the 2D model as defined in Section 2.3.2. The split ratio at every intersection is set equal to 50%. This choice is consistent with the way of estimating the direction field in Section 2.2. The inflows in both the 2D model and the Aimsum simulator are identical: 800 vehicles per hour at each of the 16 entrances (see Fig. 8 for an example of an Aimsun simulation which correspond to 6400 vehicles per hour and per km over the 2 km of entrance boundary of the 2D model. The simulation starts 3 min after the beginning of the congestion and lasts 5 min 00 s. In Fig. 14 the initial and final states

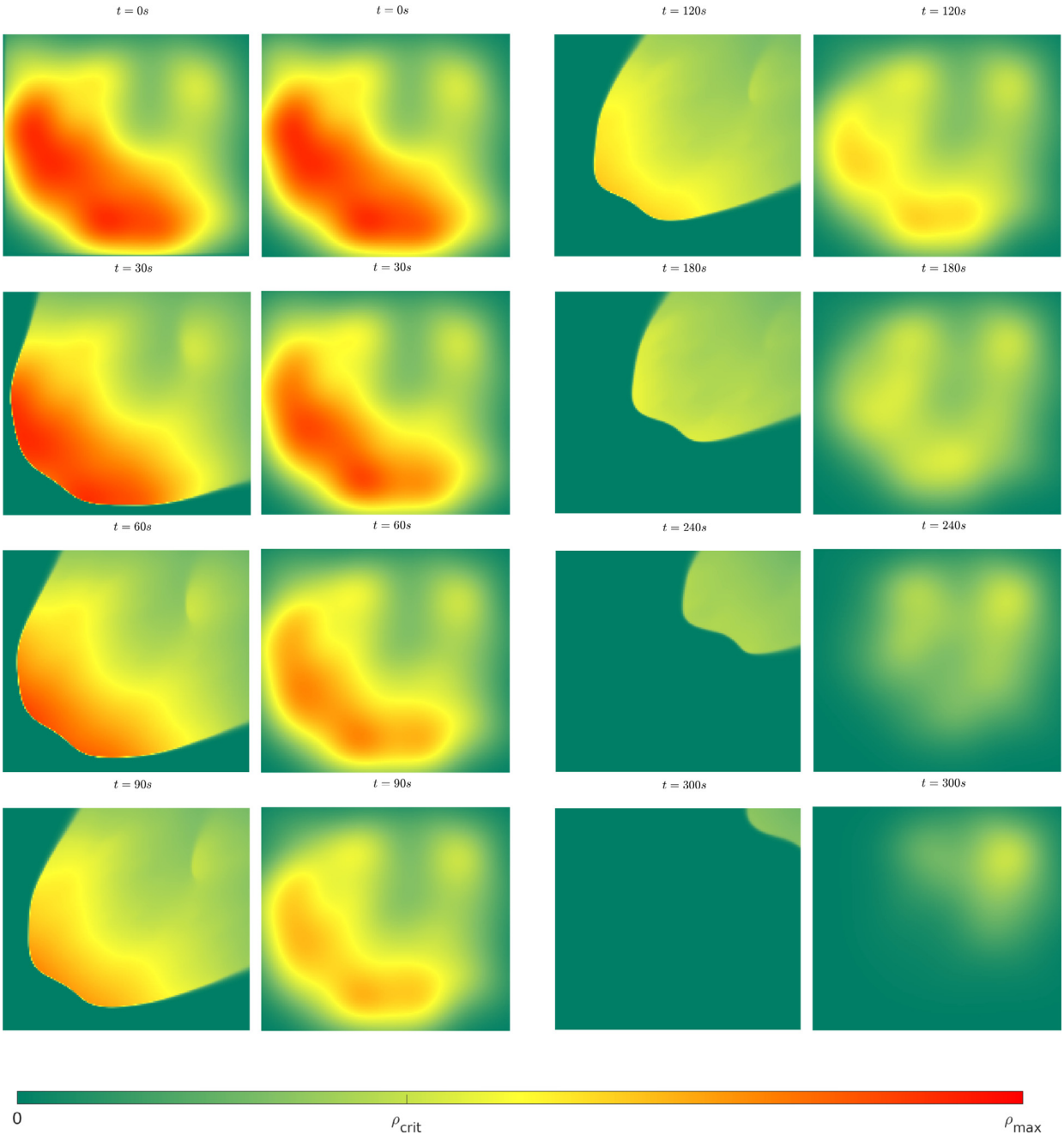


Fig. 17. Comparison of two-dimensional (left) and the average microsimulator behavior (right) during the dissipation of a congestion. Video of the full simulation available at https://youtu.be/OLGGyWU2jz4.

of the microsimulation for scenario 1 are displayed. A video that shows the reconstruction of density and the distribution of vehicle for one instance of this scenario is available at https://youtu.be/Nb-m2-fDxHY. The 2D-model for this scenario are simulated and compared only in the green subdomain. The parts of the network that are not considered in the 2D model are used to provide the boundary condition such that the condition of the 2D-model and the microsimulation are similar.

Scenario 2: Congestion dissipation. The second scenario consists of the dissipation of a congested area. We create a congestion in the microsimulation that fill the network during 15 min and then we start to dissipate it by opening all the outputs and stopping the inflow at the entrances of the model. One minute after the dissipation start, we reconstruct the 2D density from the Aimsun simulator and feed this density as initial state to the 2D model. Then we compare the evolution of the micro and the macro models over the next 5 min 00 s. The flow of vehicles can cross the boundary without any

constraint. For this reason, it is not necessary to consider a specific subdomain as it is the case in scenario 1. The split ratio in the entire network are set equal to 50%. In Fig. 15 the initial and final states of the microsimulation for one sample of scenario 2 are displayed. A video that shows the reconstruction of density and the distribution of vehicle for one instance of this scenario is available at https://youtu.be/L8Q9MgYyBK4.

As the two models are very different, it might be difficult to have a good precision in the results. However, the aim of this study is more to be able to capture the large scale features of traffic than to focus on detail at a local level.

5.2. Comparison micro vs. macro simulation

Results of simulation scenario 1.

In this scenario, we are considering a creation of congestion. The results of the simulation can be seen in Fig. 16. The figure represents a comparison of simulation results between the 2D model and the average of the reconstructed density from 100 runs in Aimsun.

First, we can notice that the main traffic features, which is the propagation of a wave moving backward is captured by the 2D model. During the first 2 min, for the considered selected square of 800 m, the network is in free flow conditions because the congestion needs a sufficient time to appear. From $t = 120$ s, a delay can be seen in the creation of the congestion in the 2D model. This might be explained by the choice of boundary condition. Indeed, the boundary conditions in the 2D model are defined using Ghost Cells. The values of these cells are computed by reconstruction of a two dimensional density from the microsimulator. However, one feature of the two dimensional reconstruction is its smoothness due to the Gaussian kernel, whereas the 2D model propagates sharp discontinuities. This difference is causing a delay at the time at which appears the congestion. This delay propagates during the duration of the simulation.

Furthermore, we can observe some difference regarding the shape of the shock wave. Indeed, the shock in the 2D model is really discontinuous whereas for the reconstruction from the microsimulation it is smooth. But, as the kernel use for the reconstruction is a Gaussian with a quite large range we could have expected this phenomenon. The simulation time is short in comparison with usual study case but this scale directly with the size of the network smaller than usual.

Results of simulation scenario 2.

In the second scenario, we are considering a dissipation of a congestion. As in the previous simulation, the results represent a comparison of the density evolution between the 2D model and the average of 100 Aimsun simulation runs. The results can be seen in Fig. 17.

The simulation of the dissipation of congestion seems to fit well the reconstruction of the Aimsun simulation. The speed of the dissipation looks also similar. Nevertheless, there is some fundamental difference. First starting from time $t = 30$ s, one may notice that the 2D model includes a shock in the South-West of the congested area. This phenomena could not be captured by the reconstructed density which is smooth due to the Gaussian Kernel. On the other side of the congested area, a rarefaction wave that reduced can be observed. If the simulation time is short in comparison with usual study case, this could be directly link with the scaling of the size of the network.

6. Conclusion and future work

This paper investigates a method for the validation of a two-dimensional model. We construct a 2D model and explain methods for the tuning of parameters using the network for the direction of the flux and data for an estimation of a specific fundamental diagram. Then we present results of simulations which compare the results of a 2D model and the microsimulator Aimsun for a similar scenario. Future steps that need to be done is to consider a heterogeneous network with capacity and speed varying over the 2D-plane and also to extend the model with multiple layers of density to be able to describe several direction of traffic. Indeed, this would be necessary to get closer to more realistic scenarios. We could notice that the tools described in this article for the comparison between the microscopic and the 2D model can be used for the validation of more complex situation.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement 694209). B. Seibold wishes to acknowledge support by the National Science Foundation via grant CNS–1446690.

References

Aghamohammadi, R., Laval, J.A., 2018. Dynamic traffic assignment using the macroscopic fundamental diagram: a review of vehicular and pedestrian flow models. Transp. Res. Part B. In Press. Online at https://doi.org/10.1016/j.trb.2018.10.017.

Beckmann, M., 1952. A continuous model of transportation. Econometrica: J. Econometric Soc. 643–660.

Carey, M., Bowers, M., 2012. A review of properties of flow–density functions. Transp. Rev. 32 (1), 49–73.

Chetverushkin, B., Churbanova, N., Trapeznikova, M., 2010. Traffic flow simulation by 2d macro- and microscopic models. Latest Trends on Urban Planning Transportation.

Coclite, G.M., Garavello, M., Piccoli, B., 2005. Traffic flow on a road network. SIAM J. Math. Anal. 36 (6), 1862–1886.

Daganzo, C.F., 1994. The cell transmission model: a dynamic representation of highway traffic consistent with the hydrodynamic theory. Transp. Res. Part B 28 (4), 269–287.
Daganzo, C.F., Geroliminis, N., 2008. An analytical approximation for the macroscopic fundamental diagram of urban traffic. Transp. Res. Part B 42 (9), 771–781.
Della Rossa, F., D'Angelo, C., Quarteroni, A., 2010. A distributed model of traffic flows on extended regions.. NHM 5 (3), 525–544.
Du, J., Wong, S., Shu, C.-W., Xiong, T., Zhang, M., Choi, K., 2013. Revisiting Jiang's dynamic continuum model for urban cities. Transp. Res. Part B 56, 96–119.
Fan, S., 2013. Data-fitted generic second order macroscopic traffic flow models. Temple University.
Fan, S., Herty, M., Seibold, B., 2014. Comparative model accuracy of a data-fitted generalized aw-Rascle-Zhang model. Netw. Heterogeneous Media 9, 239–268.
Fedkiw, R.P., Aslam, T., Merriman, B., Osher, S., 1999. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). J. Comput. Phys. 152 (2), 457–492.
Francklin, R.E., 1961. The structure of a traffic shock wave. Civil Eng. Public Works Rev. 56, 1186–1188.
Garavello, M., Han, K., Piccoli, B., 2016. Models for Vehicular Traffic on Networks, 9. American Institute of Mathematical Sciences (AIMS), Springfield, MO.
Geroliminis, N., Daganzo, C.F., 2008. Existence of urban-scale macroscopic fundamental diagrams: some experimental findings. Transp. Res. Part B 42 (9), 759–770.
Godunov, S.K., 1959. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. Matematicheskii Sbornik 89 (3), 271–306.
Gosse, L., 2014. A two-dimensional version of the Godunov scheme for scalar balance laws. SIAM J. Numer. Anal. 52 (2), 626–652.
Greenshields, B.D., Thompson, J., Dickinson, H., Swinton, R., 1934. The photographic method of studying traffic behavior. Highway Res. Board Proc. 13, 382–399.
Hajiahmadi, M., Knoop, V.L., De Schutter, B., Hellendoorn, H., 2013. Optimal dynamic route guidance: a model predictive approach using the macroscopic fundamental diagram. In: Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on. IEEE, pp. 1022–1028.
Helbing, D., 1992. A fluid dynamic model for the movement of pedestrians. Complex Syst. 6, 391–415.
Herty, M., Fazekas, A., Visconti, G., 2018. A two-dimensional data-driven model for traffic flow on highways. Netw. Heterogeneous Media 13, 217–240.
Ho, H., Wong, S., 2006. Two-dimensional continuum modeling approach to transportation problems. J. Transp. Syst. Eng.Inf. Technol. 6 (6), 53–68.
Hughes, R.L., 2002. A continuum theory for the flow of pedestrians. Transp. Res. Part B 36 (6), 507–535.
Jiang, Y., Wong, S., Ho, H., Zhang, P., Liu, R., Sumalee, A., 2011. A dynamic traffic assignment model for a continuum transportation system. Transp. Res. Part B 45 (2), 343–363.
Jiang, Y.-Q., Ma, P.-J., Zhou, S.-G., 2015. Macroscopic modeling approach to estimate traffic-related emissions in urban areas. Transp. Res. Part D 60, 41–55.
Jiang, Y.-Q., Zhang, P., Wong, S., Liu, R.-X., 2010. A higher-order macroscopic model for pedestrian flows. Physica A 389 (21), 4623–4635.
Kometani, E., Sasaki, T., 1961. Dynamic Behaviour of Traffic with a Non-linear Spacing-speed Relationship, 105. Elsevier Publishing Co.
Kruzhkov, S.N., 1970. First order quasilinear equations in several independent variables. Matematicheskii Sbornik 123 (2), 228–255.
Lebacque, J., Khoshyaran, M., 2004. First order macroscopic traffic flow models for networks in the context of dynamic assignment. Transp. Planning 119–140.
Leclercq, L., Parzani, C., Knoop, V.L., Amourette, J., Hoogendoorn, S.P., 2015. Macroscopic traffic dynamics with heterogeneous route patterns. Transp. Res. Procedia 7, 631–650.
Lie, K.-A., 1999. A dimensional splitting method for quasilinear hyperbolic equations with variable coefficients. BIT Numer. Math. 39 (4), 683–700.
Lighthill, M.J., Whitham, G.B., 1955. On kinematic waves. ii. a theory of traffic flow on long crowded roads. In: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, 229. The Royal Society, pp. 317–345.
Mollier, S., Delle Monache, M.L., Canudas-de Wit, C., 2018. A simple example of a two-dimensional model for traffic: discussion about assumptions and numerical methods. Transp. Res. Rec. 2672 (20), 249–261.
Newell, G.F., 1961. A theory of traffic flow in tunnels. Theory Traffic Flow 193–206.
Parzen, E., 1962. On estimation of a probability density function and mode. AnnalsMath.Stat. 33 (3), 1065–1076.
Pipes, L.A., 1953. An operational analysis of traffic dynamics. J. Appl. Phys. 24 (3), 274–281.
Richards, P.I., 1956. Shock waves on the highway. Oper. Res. 4 (1), 42–51.
Romero Perez, L., Benitez, F.G., 2008. Outline of diffusion advection in traffic flow modeling. Transportation Research Board 87th Annual Meeting.
Rossi, E., 2017. Definitions of solutions to the IBVP for multi-dimensional scalar balance laws. J. Hyperbolic Differ. Eq. 15 (02), 349–374.
Saumtally, T., 2012. Modèles bidimensionnels de trafic. Université Paris-Est.
Sossoe, K.S., Lebacque, J.-P., 2016. Reactive dynamic assignment for a bi-dimensional traffic flow model. In: International Conference on Systems Science. Springer, pp. 179–188.
Strang, G., 1968. On the construction and comparison of difference schemes. SIAM J. Numer. Anal. 5 (3), 506–517.
Sukhinova, A.B., Trapeznikova, M.A., Chetverushkin, B.N., Churbanova, N.G., 2009. Two-dimensional macroscopic model of traffic flows. Math. Models Comput. Simul. 1 (6), 669.
Toro, E.F., 2013. Riemann Solvers and Numerical Methods for Fluid Dynamics: APractical Introduction. Springer Science & Business Media.
van Wageningen-Kessels, F., Van Lint, H., Vuik, K., Hoogendoorn, S., 2014. Genealogy of traffic flow models. EURO J. Transp. Logist. 4 (4), 445–473.