Urban traffic congestion control: a DeePC change

Alessio Rimoldi*, Carlo Cenedese*, Alberto Padoan*, Florian Dörfler*, John Lygeros*

***Abstract*— Urban traffic congestion remains a pressing challenge in our rapidly expanding cities, despite the abundance of available data and the efforts of policymakers. By leveraging behavioral system theory and data-driven control, this paper exploits the DeePC algorithm in the context of urban traffic control performed via dynamic traffic lights. To validate our approach, we consider a high-fidelity case study using the state-of-the-art simulation software package Simulation of Urban MObility (SUMO). Preliminary results indicate that DeePC outperforms existing approaches across various key metrics, including travel time and CO_2 emissions, demonstrating its potential for effective traffic management.**

I. Introduction

In the ever-growing urban landscapes of our times, the need for efficient and effective traffic congestion management systems has never been more pressing. Recent studies indicate that, on average, commuters in major cities spend approximately 54 hours each year stuck in traffic jams [1]. The economic and environmental toll of traffic congestion is staggering [2]. As the world continues to urbanize [3], these figures are projected to increase significantly, unless novel solutions for traffic congestion are developed.

In the era of digital transformation, data serves as a critical asset in this respect, revealing complex patterns of urban traffic and supporting data-informed decision-making [4]. For instance, machine learning traffic pattern analysis algorithms have demonstrated the potential to enhance traffic flow and alleviate congestion [5], [6]. In this context, the emergence of data-driven traffic control algorithms promises to revolutionize how we tackle the problem of urban traffic congestion [7]. With these premises, this paper addresses the urban traffic control problem using the Data-enabled Predictive Control (DeePC) algorithm [8].

Related work: In the past decades, the problem of urban traffic control has attracted the interest of many researchers. Early solutions focus on controlling traffic only in the proximity of traffic lights and intersections [9] and small traffic networks [10]. Over the years results have been developed for larger networks and have often considered as the main objective of the reduction of the total system travel time or the emissions. In [11] the authors model the city as a directed graph and propose a signal control problem based on efficiently solving a Quadratic Programs (QP). The authors in [12] propose two different macroscopic models of urban traffic to design an Model Predictive Control (MPC) able to compute a structured network-wide traffic controller. Control strategies that focus particularly on reducing the emission can be found in [13]–[16] and references therein.

Due to the unprecedented data collection, storage, and computation capabilities, there has been a resurgence of interest in the control community in "direct" control approaches (see, e.g., the recent survey [17]). Leveraging behavioral system theory [18], these approaches infer optimal decisions directly from observed data. A particularly successful direct data-driven algorithm is DeePC [8], which has been applied in a wide array of practical case studies [8], [19], [20]. DeePC has been applied in the context of traffic control, e.g., to solve control problems related to the coordination of Connected Autonomous Vehicles (CAVs) [21], to vehicle rebalancing in mobility-on-demand systems [22], and to the dissipation of stop-and-go waves [23]. However, to the best of our knowledge, DeePC has not been applied in the context of urban traffic congestion control due to the large-scale nature of the problem. Motivated and inspired by [24]–[26], we propose a tractable formulation by leveraging an aggregate description of the urban traffic dynamics of a city which mitigates the effect of unpredictable events.

Contributions: The main contributions of the paper are threefold. (i) We show that the DeePC algorithm, although fundamentally linear, excels in handling mildly nonlinear systems. Adopting the DeePC algorithm avoids the use of complex nonlinear models, but rather it generates optimal decisions directly from data. In particular, being able to directly provide control for traffic lights that do not directly regulate the flow among regions is extremely difficult via a traditional MPC based approach. (ii) We show that the building blocks necessary to deploy DeePC are simple to create after the partition of the city in homogenous regions, which should be performed only once as in [26]. This allows practitioners to bypass the identification of model parameters and, in case the infrastructure changes, new data can be collected and automatically updated in the algorithm. (iii) We show that the data-driven approach can outperform model-based control in terms of traffic congestion and emission reduction the current state-of-the-art model-based controller via SUMO mesoscopic simulations.

Notation For every $p \in \mathbb{N}$, the set of positive integers $\{1, 2, ..., p\}$ is denoted by $\mathbf{p}$. $(Y)^X$ denotes the collection of all maps from X to Y. for $x \in X \cap X'$. The restriction of $f : X \to Y$ to a subset $X' \subset X$ is denoted by $f|_{X'}$ and is defined by $f|_{X'}(x)$ If $\mathcal{F} \subset (Y)^X$, then $\mathcal{F}|_{X'}$ denotes $\{f|_{X'} : f \in \mathcal{F}\}$. The shift operator $\sigma : (\mathbb{R}^q)^{\mathbb{N}} \to (\mathbb{R}^q)^{\mathbb{N}}$ is defined as $w \mapsto \sigma w$, with $\sigma w(t) = w(t+1)$ for all $t \in \mathbb{N}$.

*Authors are with Automatic Control Laboratory, Department of Electrical Engineering and Information Technology, ETH Zürich, Physikstrasse 3 8092 Zürich, Switzerland {ccenedese,apadoan,fdorfler,jlygeros}@ethz.ch.

Research is supported by NCCR Automation and funded by the Swiss National Science Foundation (grant number 180545).

II. Preliminaries

A. Sequences and Hankel matrices

We consider finite and infinite sequences in $(\mathbb{R}^q)^{\mathbf{T}}$ and $(\mathbb{R}^q)^{\mathbb{N}}$, respectively. By a convenient abuse of notation, we identify each finite sequence $w \in (\mathbb{R}^q)^{\mathbf{T}}$ with the corresponding vector $\mathrm{col}(w(1), \dots, w(T)) \in \mathbb{R}^{qT}$. We use the terms sequence and trajectory interchangeably.

The *Hankel matrix* of depth $L \in \mathbf{T}$ associated with the finite sequence $w \in (\mathbb{R}^q)^{\mathbf{T}}$ is defined as

$$H_L(w) = \begin{bmatrix} w(1) & w(2) & \cdots & w(T-L+1) \\ w(2) & w(3) & \cdots & w(T-L+2) \\ \vdots & \vdots & \ddots & \vdots \\ w(L) & w(L+1) & \cdots & w(T) \end{bmatrix}.$$

B. Dynamical systems

A *dynamical system* (or, briefly, *system*) is a triple $\Sigma = (\mathbb{T}, \mathbb{W}, \mathcal{B})$, where $\mathbb{T}$ is the *time set*, $\mathbb{W}$ is the *signal set*, and $\mathcal{B} \subseteq (\mathbb{W})^{\mathbb{T}}$ is the *behavior* of the system. We exclusively focus on *discrete-time* systems, with $\mathbb{T} = \mathbb{N}$ and $\mathbb{W} = \mathbb{R}^q$.

A system Σ is *linear* if the corresponding behavior $\mathcal{B}$ is a linear subspace, *time-invariant* if $\mathcal{B}$ is shift-invariant, *i.e.*, if $w \in \mathcal{B}$ implies $\sigma w \in \mathcal{B}$, and *complete* if $\mathcal{B}$ is closed in the topology of pointwise convergence [18, Proposition 4]. The model class of all complete Linear Time-Invariant (LTI) systems is denoted by $\mathcal{L}^q$. By a convenient abuse of notation, we write $\mathcal{B} \in \mathcal{L}^q$.

The structure of an LTI system is characterized by a set of integer invariants known as *structure indices* [18]. The most important ones are the *number of inputs* m, *number of outputs* p, the *lag* ℓ, and the *order* n, see, [18, Section 7] for definitions. Every finite-dimensional LTI system admits a minimal representation and can be described by the equations

$$\sigma x = Ax + Bu, \quad y = Cx + Du, \tag{1}$$

where $\begin{bmatrix} A & B \\ C & D \end{bmatrix} \in \mathbb{R}^{(n+p)\times(n+m)}$ and m, n, and p are the number of inputs, the order, and the number of outputs of $\mathcal{B}$, respectively.

C. Data-driven representations of dynamical systems

Given a trajectory $w_{\mathrm{d}} \in \mathbb{R}^{qT}$ of a system $\mathcal{B} \in \mathcal{L}^q$, it is possible to derive a non-parametric representation of its finite-horizon behavior using raw data. We summarize a version of this principle known as the *fundamental lemma* [27].

Lemma 1: [28, Corollary 19] Consider a system $\mathcal{B} \in \mathcal{L}^q$ with lag $\ell \in \mathbb{N}$ and a trajectory of the system $w_{\mathrm{d}} \in \mathcal{B}|_{[1,T]}$. Assume $L > \ell$. Then $\mathcal{B}|_{[1,L]} = \mathrm{im}\, H_L(w_{\mathrm{d}})$ if and only if

$$\mathrm{rank}\, H_L(w_{\mathrm{d}}) = mL + n, \tag{2}$$

where n and m are the order and the number of inputs of the system, respectively.

Lemma 1 is a key result in data-driven control [17]. It characterizes all trajectories of given length of an LTI system in terms of the image of a Hankel matrix. This foundational principle can be adapted in various ways, see the recent survey [17]. Remarkably, non-parametric representations have found practical use in data-driven control even when dealing with *nonlinear* systems [29].

The rank condition (2) is known as the *generalized persistency of excitation* condition [28]. Note that upper bounds on the structure indices of $\mathcal{B}$ are necessary to check this condition from data. Alternatively, the rank condition (2) can be guaranteed to hold for controllable systems if a certain rank condition on the inputs holds [27].

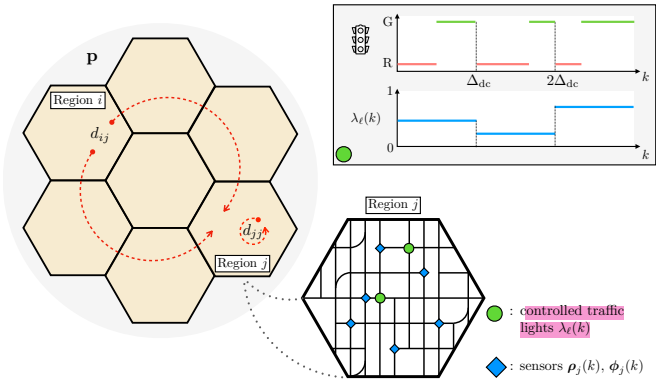


Fig. 1: The city is divided into $\mathbf{p}$ regions (the hexagons) based on their Macroscopic Fundamental Diagram (MFD) and the demand $\boldsymbol{d}$ among them. The qualitative detail of how sensors $\mathbf{s}_j$ (blue diamonds) and traffic lights $\mathbf{l}$ (green circles) might be positioned within a region $j \in \mathbf{p}$. In the top part, how the control input λ_ℓ influences the duty cycle of a traffic light $\ell \in \mathbf{l}$.

III. Methodology

A. Urban traffic: sensing and dynamics

The transportation network of a city is composed of a collection of roads, highways, intersections, and points of interest such as bus stops. As in [24], [26], the city can be partitioned in $p \in \mathbb{N}$ regions leveraging the concept of MFD. These regions are identified such that the aggregate drivers' behavior traveling through it is as homogeneous as possible. The set of all regions is denoted by $\mathbf{p} := \{1, \cdots, p\}$. Modern cities can choose among different types of sensors to assess traffic conditions. Among these, the most common (and reliable) are *Eulerian* sensors, such as single-loop detectors [30]. They can measure the number of vehicles that cross them during a fixed period and the occupancy, that is, the fraction of time during the same period that the sensor detected a vehicle above it [31]. Similar data can also be retrieved from other types of sensors, such as *Lagrangian* (or mobile) sensors. This includes vehicles equipped with transponders and GPS traveling through the city. This allows smaller cities that do not have (or cannot afford) fixed sensors to implement the control scheme designed in the following sections [31]. For simplicity, however, we consider only Eulerian sensors throughout the paper.

We denote the number of sensors in each region $i \in \mathbf{p}$ as $s_i \in \mathbb{N}$ and the set of all sensors in the region is $\mathbf{s}_i := \{1, \cdots, s_i\}$. From each sensor $j \in \mathbf{s}_i$, we can attain, from

the occupancy and number of vehicles, the traffic density $\rho_j(t)$ [veh/km] and flow $\phi_j(t)$ [veh/h] during the time interval indexed by $t \in \mathbb{N}$ of length $\Delta \in \mathbb{R}_+$. We denote the density and flows measured in region $i \in \mathbf{p}$ during t by

$$\boldsymbol{\rho}_i(t) \coloneqq \mathrm{col}((\rho_1(t)), \cdots, \rho_{s_i}(t))) \in \mathbb{R}^{s_i} \tag{3a}$$
$$\boldsymbol{\phi}_i(t) \coloneqq \mathrm{col}((\phi_1(t), \cdots, \phi_{s_i}(t))) \in \mathbb{R}^{s_i}. \tag{3b}$$

Similarly, the collective vectors of all densities and flows in the city during t are, respectively,

$$\boldsymbol{\rho}(t) \coloneqq \mathrm{col}((\boldsymbol{\rho}_i(t))_{i\in\mathbf{p}}) \in \mathbb{R}^s \tag{4a}$$
$$\boldsymbol{\phi}(t) \coloneqq \mathrm{col}((\boldsymbol{\phi}_i(t))_{i\in\mathbf{p}}) \in \mathbb{R}^s, \tag{4b}$$

where $s = \sum_{i\in\mathbf{p}} s_i$.

The evolution of traffic within the city depends on the number of commuters entering the system during every time interval within each region. We call this flow of vehicles *demand*. The flow of vehicles starting their trip in region $i \in \mathbf{p}$ at time t and aiming to end the trip in region $j \in \mathbf{p}$ is denoted by $d_{ij}(t)$ [veh/h], while $d_{ii}(t)$ [veh/h] is the *internal trip completion flow* [24], which describes the flow of vehicles whose final destination is region i. The collective vector of demands associated with region i is denoted by $\boldsymbol{d}_i(t) \coloneqq \mathrm{col}((d_{ij}(t))_{j\in\mathbf{p}}) \in \mathbb{R}^p$ while the vector of all the demands among all regions is $\boldsymbol{d}(t) \coloneqq \mathrm{col}((\boldsymbol{d}_i(t))_{i\in\mathbf{p}}) \in \mathbb{R}^{p^2}$. Hereafter, we consider the *nominal* value of $\boldsymbol{d}(t)$ as an exogenous and known quantity. How to estimate the demand among regions is a well-known problem that has been extensively studied [32], [33].

Policymakers throughout the years have designed a plethora of interventions to reduce urban traffic congestion [34]. Among dynamic policies, there is the installation of controllable traffic lights [35]. By strategically controlling the flow of vehicles traveling through an intersection, it is possible to indirectly influence $\boldsymbol{\phi}_i(t)$ and $\boldsymbol{\rho}_i(t)$ in each region. Given two regions $i, j \in \mathbf{p}$, if one (or multiple) intersection used by drivers to move from i to j allows for a smaller flow to pass through, then the vehicles will accumulate increasing the density in i and decreasing the flow from i to j. The flow of vehicles that can travel through a intersection can be controlled by varying the ratio green $\lambda(t)$ of the duty cycle $\Delta_{\mathrm{dc}} \in \mathbb{N}$, see Fig. 1. Therefore, the control input $\lambda_\ell(t) \in [\underline{\lambda}_\ell, \overline{\lambda}_\ell] \subseteq [0, 1]$ where $\lambda_\ell(t) = 1$ means that the traffic light is always green and $\lambda_\ell(t) = 0$ red for the whole duty cycle. The vector of all input associated with the l controlled traffic lights is denoted by $\boldsymbol{\lambda}(t) = \mathrm{col}((\lambda_\ell(t))_{\ell\in\mathbf{l}})$, where $\mathbf{l} = \{1, \cdots, l\}$.

We model the nonlinear dynamics associated with the density evolution measured by the sensors as

$$\boldsymbol{\rho}(k+1) = f(\boldsymbol{\rho}(t), \boldsymbol{\lambda}(t), \boldsymbol{d}(t)), \tag{5}$$

where $f : \mathbb{R}^{s+l+p^2} \to \mathbb{R}^s$ is not known. This choice mitigates the course of dimensionality associated with the proposed data-driven approach. Moreover, as demonstrated by our numerical investigations, the density usually has smoother trajectories that are easier to predict via a data-driven approach.

As discussed in [36], aggregating the data from the sensor within a region helps reduce the variability and capture the macroscopic variation in density. An aggregating function has the role of mapping the sensors' measured densities to the associated region density. So, the *average* density of region i is attained as $\overline{\rho}_i(t) = h_i(\boldsymbol{\rho}_i(t))$, where $h_i : \mathbb{R}^{s_i} \to \mathbb{R}$ is the aggregating function of region $i \in \mathbf{p}$. Consequently, we define the complete aggregation function as $h : \mathbb{R}^s \to \mathbb{R}^p$ and

$$\overline{\boldsymbol{\rho}}(t) = h(\boldsymbol{\rho}(t)) \coloneqq \mathrm{col}\left((h_i(\boldsymbol{\rho}_i(t)))_{i\in\mathbf{p}}\right). \tag{6}$$

Using the density dynamics of the city regions in (5)–(6) and the definition in Section II, we now introduce the behavior $\mathcal{B}_{\mathrm{c}}$ that describes the associated dynamical system

$$\mathcal{B}_{\mathrm{c}} = \{(\boldsymbol{\lambda}, \boldsymbol{d}, \overline{\boldsymbol{\rho}}) \in \mathbb{R}^{m+p} : \exists \boldsymbol{\rho} \in \mathbb{R}^s \text{ s.t (5)–(6) hold}\}, \tag{7}$$

where $m = l + p^2$ and $u = (\boldsymbol{\lambda}, \boldsymbol{d}) \in \mathbb{R}^m$ is the complete input, divided in exogenous and controllable, and $\overline{\boldsymbol{\rho}}$ the output.

B. The DeePC algorithm

1) Setup and assumptions: Consider a (possibly unknown) LTI system $\mathcal{B} \in \mathcal{L}^{m+p}$, with m inputs and p outputs. Assume that data recorded offline from system $\mathcal{B}$ are available. Specifically, assume that an input sequence $u_{\mathrm{d}} = \mathrm{col}(u_{\mathrm{d}}(1), \cdots, u_{\mathrm{d}}(T)) \in \mathbb{R}^{mT}$ of given length $T \in \mathbb{N}$ is applied to the system $\mathcal{B}$ and that the corresponding output sequence is $y_{\mathrm{d}} = \mathrm{col}(y_{\mathrm{d}}(1), \cdots, y_{\mathrm{d}}(T)) \in \mathbb{R}^{pT}$. The subscript "d" is used to indicate that these are sequences of data samples collected offline. Finally, let $T_{\mathrm{ini}} \in \mathbb{N}$ and $T_{\mathrm{f}} \in \mathbb{N}$, with $T_{\mathrm{ini}} + T_{\mathrm{f}} \leq T$, and assume that the sequence $w_{\mathrm{d}} \in \mathbb{R}^{(m+p)T}$, defined as $w_{\mathrm{d}}(t) = \mathrm{col}(u_{\mathrm{d}}(t), y_{\mathrm{d}}(t))$ for $t \in \mathbf{T}$, satisfies the generalized persistency of excitation condition (2), where $L = T_{\mathrm{ini}} + T_{\mathrm{f}}$ and n is the order (or an upper bound on the order) of the system.

2) Data organization: Next, partition the input/output data into two parts which we call *past data* and *future data*. Formally, given the time horizons $T_{\mathrm{ini}} \in \mathbb{N}$ and $T_{\mathrm{f}} \in \mathbb{N}$ associated with the past data and the future data, define

$$\begin{pmatrix} U_{\mathrm{p}} \\ U_{\mathrm{f}} \end{pmatrix} = H_{T_{\mathrm{ini}}+T_{\mathrm{f}}}(u_{\mathrm{d}}), \quad \begin{pmatrix} Y_{\mathrm{p}} \\ Y_{\mathrm{f}} \end{pmatrix} = H_{T_{\mathrm{ini}}+T_{\mathrm{f}}}(y_{\mathrm{d}}), \tag{8}$$

where $U_{\mathrm{p}} \in \mathbb{R}^{(mT_{\mathrm{ini}})\times(T-T_{\mathrm{ini}}+1)}$ consists of the first T_{ini} block-rows of the matrix $H_{T_{\mathrm{ini}}+T_{\mathrm{f}}}(u_{\mathrm{d}})$ and $U_{\mathrm{f}} \in \mathbb{R}^{(mT_{\mathrm{f}})\times(T-T_{\mathrm{f}}+1)}$ consists of the last T_{f} block-rows of the matrix $H_{T_{\mathrm{ini}}+T_{\mathrm{f}}}(u_{\mathrm{d}})$ (similarly for Y_{p} and Y_{f}), respectively. In the sequel, past data denoted by the subscript "p" is used to estimate the initial condition of the underlying state, whereas the future data denoted by the subscript "f" is used to predict the future trajectories.

3) State estimation and trajectory prediction: By the fundamental lemma, any trajectory of the finite-horizon behavior $\mathcal{B}|_{[1,T_{\mathrm{ini}}+T_{\mathrm{f}}]}$ of length $T_{\mathrm{ini}} + T_{\mathrm{f}}$ can be constructed using the data collected offline. Under the assumptions of Lemma 1, a trajectory $\mathrm{col}(u_{\mathrm{ini}}, y_{\mathrm{ini}}, u, y)$ belongs to $\mathcal{B}|_{[1,T_{\mathrm{ini}}+T_{\mathrm{f}}]}$ if and

only if there exists $g \in \mathbb{R}^{T-T_{\text{ini}}-T_{\text{f}}+1}$ such that

$$\begin{pmatrix} U_{\text{p}} \\ Y_{\text{p}} \\ U_{\text{f}} \\ Y_{\text{f}} \end{pmatrix} g = \begin{pmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u \\ y \end{pmatrix}. \tag{9}$$

For $T_{\text{ini}} \geq \ell$, the lag of the system, the output y is uniquely determined [17]. Intuitively, the trajectory $\text{col}(u_{\text{ini}}, y_{\text{ini}})$ specifies the underlying initial state from which the trajectory $\text{col}(u, y)$ evolves. This allows one to predict future trajectories based on a given initial trajectory $\text{col}(u_{\text{ini}}, y_{\text{ini}}) \in \mathcal{B}|_{[1,T_{\text{ini}}]}$, and the precollected data in U_{p}, U_{f}, Y_{p}, and Y_{f}. Indeed, given an initial trajectory $\text{col}(u_{\text{ini}}, y_{\text{ini}}) \in \mathcal{B}|_{[1,T_{\text{ini}}]}$ of length $T_{\text{ini}} \geq \ell$ and a sequence of future inputs $u \in \mathbb{R}^{mT_{\text{f}}}$, the first three block equations of (9) can be solved for g. The sequence of future outputs is then given by $y = Y_{\text{f}} g$. Conversely, given a desired reference output y an associated control input u can be calculated.

4) DeePC algorithm: Given the future time horizon $T_{\text{f}} \in \mathbb{N}$, a reference trajectory for the output $\hat{y} = (\hat{y}_0, \hat{y}_1, \cdots) \in (\mathbb{R}^p)^{\mathbb{N}}$ and input $\hat{u} = (\hat{u}_0, \hat{u}_1, \cdots) \in (\mathbb{R}^m)^{\mathbb{N}}$, past input/output data $w_{\text{ini}} = \text{col}(u_{\text{ini}}, y_{\text{ini}}) \in \mathcal{B}|_{[1,T_{\text{ini}}]}$, input constraint set $\mathcal{U} \subseteq \mathbb{R}^{mT_{\text{f}}}$, output constraint set $\mathcal{Y} \subseteq \mathbb{R}^{pT_{\text{f}}}$, output cost matrix $Q \in \mathbb{R}^{p \times p}$, and control cost matrix $R \in \mathbb{R}^{m \times m}$, and regularization function $\psi : \mathbb{R}^{T-L+1} \to \mathbb{R}$, the DeePC algorithm relies on the solution of the following optimization problem:

$$\begin{aligned} \min_{u,y,g} \; & \sum_{k=1}^{T_{\text{f}}} \|y(k) - \hat{y}(t+k)\|_Q^2 + \|u(k) - \hat{u}(t+k)\|_R^2 + \psi(g) \\ \text{s.t.} \; & \begin{pmatrix} U_{\text{p}} \\ Y_{\text{p}} \\ U_{\text{f}} \\ Y_{\text{f}} \end{pmatrix} g = \begin{pmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u \\ y \end{pmatrix}, \\ & u \in \mathcal{U}, \; y \in \mathcal{Y}. \end{aligned} \tag{10}$$

Algorithm 1 DeePC

Input: Data trajectories $w_{\text{d}} = \text{col}(u_{\text{d}}, y_{\text{d}}) \in \mathbb{R}^{(m+p)T}$, most recent input/output measurements $w_{\text{ini}} = \text{col}(u_{\text{ini}}, y_{\text{ini}}) \in \mathbb{R}^{(m+p)T_{\text{ini}}}$, a reference trajectories $\hat{y} = (\hat{y}_0, \hat{y}_1, \cdots) \in (\mathbb{R}^p)^{\mathbb{N}}$, $\hat{u} = (\hat{u}_0, \hat{u}_1, \cdots) \in (\mathbb{R}^m)^{\mathbb{N}}$, input constraint set $\mathcal{U} \subseteq \mathbb{R}^{mT_{\text{f}}}$, output constraint set $\mathcal{Y} \subseteq \mathbb{R}^{pT_{\text{f}}}$, output cost matrix $Q \in \mathbb{R}^{p \times p}$, and control cost matrix $R \in \mathbb{R}^{m \times m}$.

1: Solve (10) for $g^\star$.
2: Compute optimal input sequence $u^\star = U_{\text{f}} g^\star$.
3: Apply optimal input sequence $(u_t, \cdots, u_{t+j-1}) = (u_1^\star, \cdots, u_j^\star)$ for some $j \leq T_{\text{f}}$.
4: Set t to $t+j$ and update u_{ini} and y_{ini} to the T_{ini} most recent input/output measurements.
5: Return to 1.

Remark 1: As demonstrated by the numerical investigation in Section IV, our formulation leads to satisfactory performance. However, extra slack variables can be used to account for the presence noise explicitly [8]. A thorough exploration of this issue is deferred to a future publication.

C. DeePC for urban traffic control

We are now ready to introduce DeePC for the dynamical system $\mathcal{B}_{\text{c}}$ describing the evolution of urban traffic.

1) Behavioral representation and constraints: The data matrix on the left-hand side in (9) can be constructed using historical data. For $t \in \mathbf{T}$ the collection of input and output data used are respectively defined as

$$\begin{aligned} u_{\text{d}}(t) &:= \text{col}(\boldsymbol{\lambda}(t), \boldsymbol{d}(t)) \\ y_{\text{d}}(t) &:= \text{col}(\overline{\boldsymbol{\rho}}(t)). \end{aligned}$$

Using these sequences, we can construct the matrices in (8) as discussed in Section III-B.2. The data are collected offline and thus can be chosen among many different traffic scenarios that excited the dynamical system $\mathcal{B}_{\text{c}}$ in different ways capturing a wide variety of the behaviors of the system. To construct u_{ini} and y_{ini} for the current time interval $t \in \mathbb{N}$, we simply collect the previous T_{ini} values of the input and outputs,

$$\begin{aligned} u_{\text{ini}} &:= \text{col}(\boldsymbol{\lambda}(t - T_{\text{ini}}), \boldsymbol{d}(t - T_{\text{ini}}), \cdots, \boldsymbol{\lambda}(t), \boldsymbol{d}(t)), \\ y_{\text{ini}} &:= \text{col}(\overline{\boldsymbol{\rho}}(t - T_{\text{ini}}), \cdots, \overline{\boldsymbol{\rho}}(t)). \end{aligned}$$

The control input has to satisfy the box constraints $\boldsymbol{\lambda}(k) \in [\underline{\boldsymbol{\lambda}}, \overline{\boldsymbol{\lambda}}] \subseteq \mathbb{R}^l$, where $\underline{\boldsymbol{\lambda}} = \text{col}(\underline{\lambda}_1, \cdots, \underline{\lambda}_l)$ and $\overline{\boldsymbol{\lambda}}$ is defined similarly. Moreover, the green ratio cannot be changed during a duty cycle but only at the end of it, hence every Δ_{dc} time intervals of length Δ. This translates into the following linear constraint on the variable $\boldsymbol{u}$

$$Mu = \mathbf{0},$$

where $M \in \mathbb{R}^{mT_{\text{f}} \times mT_{\text{f}}}$ is used to impose for every $\ell \in \mathbf{l}$ and $k \leq T_{\text{f}}$ multiple of Δ_{dc} the constraint

$$\lambda_\ell(k+1) = \lambda_\ell(k+2) = \cdots = \lambda_\ell(k + \Delta_{\text{dc}}).$$

On the contrary, the demand $\boldsymbol{d}$ is assumed to be a known exogenous input. We impose it to be fixed and equal to the known demand $\overline{\boldsymbol{d}} = \text{col}((\boldsymbol{d}(t), \cdots, \boldsymbol{d}(t + T_{\text{f}}))) \in \mathbb{R}^{p^2 T_{\text{f}}}$ over the prediction horizon, i.e., $Du = \overline{\boldsymbol{d}}$ with $D \in \mathbb{R}^{mT_{\text{f}} \times p^2 T_{\text{f}}}$ being a suitable matrix. Therefore, the set of constraints for the inputs over the whole prediction horizon reads as

$$\mathcal{U} = \left\{ u \in \left([\underline{\boldsymbol{\lambda}}, \overline{\boldsymbol{\lambda}}] \times \mathbb{R}_+^{p^2} \right)^{T_{\text{f}}} \mid Mu = 0, \; Du = \overline{\boldsymbol{d}} \right\}. \tag{11}$$

The box constraints on the output ensure that the density in the region remains below the *grid-lock density* $\overline{\boldsymbol{\rho}}_{\max} \in \mathbb{R}_+^p$, hence $y(t) \in [0, \overline{\boldsymbol{\rho}}_{\max}]$ and $\mathcal{Y} := [0, \overline{\boldsymbol{\rho}}_{\max}]^{T_{\text{f}}}$.

Finally, in the third step of the DeePC algorithm, we apply to the system the first part of the computed optimal input $\boldsymbol{\lambda}^\star$. In this application of urban traffic control, we use $j = \Delta_{\text{dc}}$ since the control input $\boldsymbol{\lambda}$ cannot be changed during that interval.

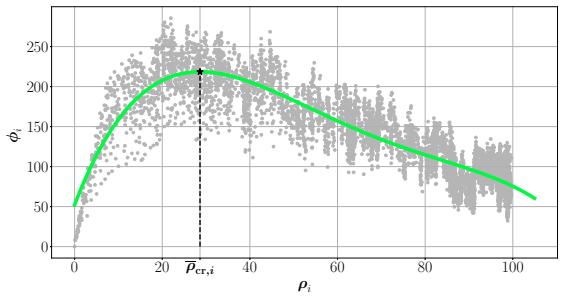


Fig. 2: The MFD for the region $i \in \mathbf{p}$, the historic measured data $\boldsymbol{\rho_i}, \boldsymbol{\phi_i} \in \mathbb{R}^{s_i}$ by the sensors $\mathbf{s_i}$ (light gray dots). The associated MFD for region $i \in \mathbf{p}$ (solid green line) displaying the relation between $\overline{\boldsymbol{\rho}}_i$ and $\overline{\boldsymbol{\phi}}_i$. The critical density $\overline{\boldsymbol{\rho}}_{\mathrm{cr},i}$ is the density maximizing the flow in the region $\overline{\boldsymbol{\phi}}_i$, while $\overline{\boldsymbol{\rho}}_{\max}$ is the density at which the flow is zero.

2) Reference trajectory and cost function: Motivated and inspired by [25], we take advantage of the concept of MFD to estimate what is the optimal density for each region, $\hat{y}$. The concept of MFD formalizes the relation between the flow $\boldsymbol{\phi}_i$ and density $\boldsymbol{\rho}_i$ within a region $i \in \mathbf{p}$. As shown in Fig. 2, we can use the MFD to identify each region's $\overline{\rho}_{\mathrm{cr},i}$ that corresponds to the density ensuring the maximum flow within the region, thus $\overline{\boldsymbol{\rho}}_{\mathrm{cr}} = \mathrm{col}((\overline{\rho}_{\mathrm{cr},i})_{i\in\mathbf{p}})$ and $\hat{y} = (\overline{\boldsymbol{\rho}}_{\mathrm{cr}}, \overline{\boldsymbol{\rho}}_{\mathrm{cr}}, \cdots)$. In (10), we minimize the distance between the regions' densities and the critical ones. In a congested scenario, this leads to also maximizing the flow within the region. The value of $\overline{\boldsymbol{\rho}}_{\mathrm{cr}}$ can be initially obtained via historical data, yet while running the algorithm the closed-loop behavior of the system may vary and the newly collected data can be used to recompute the MFD and update $\hat{y}$ if necessary. Traffic evolution throughout a single day has two major peaks during the morning and evening commute. To achieve better performance, practitioners can select a time-varying $\overline{\boldsymbol{\rho}}_{\mathrm{cr}}$. These changes would not affect the theoretical formulation proposed and are relatively straightforward to implement.

The reference value of the inputs $\hat{u}$ is composed of the one associated with $\boldsymbol{\lambda}$, and the one for the exogenous output. The latter is exogenous and set equal to $\boldsymbol{d}$ according to (11). The former represents the normal operation mode of the traffic lights and can be chosen constant if the amount of time traffic lights are green during a duty cycle does not vary throughout the day.

In the cost function in (10), the matrices R and Q are chosen positive definite. Notice that the components of $\|u - \hat{u}\|_R$ associated with $\boldsymbol{d}$ are always equal to zero due to the chosen reference for it and the constraints on u. The regularization term ψ has been chosen as in [8].

IV. Simulations

We illustrate the performance of our urban traffic congestion control framework with two case studies: a lattice network and a preliminary study of the traffic network of Zürich. We have tested the DeePC algorithm against a model-based predictive control (MPC) formulation devised in [37]. We studied the behavior of DeePC and MPC in two scenarios: a congested and an uncongested scenario. The main difference between the two is the demand experienced by the network. Both traffic networks are modeled using the open-source state-of-the-art simulation software package SUMO [38] which offers the possibility of simulating multi-model large-scale traffic networks both in a microscopic and mesoscopic fashion and provides estimates on the emissions produced. All simulations are conducted on a 3.6 GHz AMD Rizen 7 4000 Series processor.

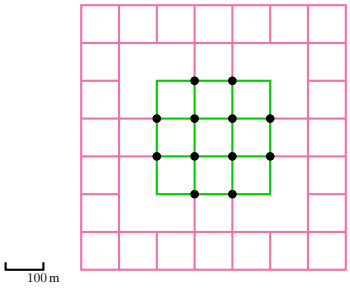


Fig. 3: Lattice traffic network. The network is partitioned into an outer region (magenta) and in inner region (green). Black dots represent the controlled traffic lights.

A. Case study 1: lattice network

Consider a lattice network whose topology is shown in Fig. 3. The network is composed of 208 roads (each one equipped with a sensor) connected by 64 intersections, of which 12 are controlled and act as actuators, represented by black dots in Fig. 3. All vehicles are assumed to be cars. The simulation covers a period of one hour. Perimeter control requires the network to be partitioned into regions, composed of roads that are similar in terms of average density. This allows for the production of a low scatter MFD. Fig. 3 shows the partitioning used, where we define two regions the outer one (magenta) and the inner one (green).

The demand in the network has a clear direction, all trips generated move from the outer region to the inner region. This simulates the flow of vehicles traveling during the morning peak hours from the outside of a city to the city center. The demand has a triangular shape, peaking at minute 30 with a base of length 800 [s]. The maximum flow is 180 [veh/h] in the uncongested setting and 1080 [veh/h] in the congested scenario. The average travel time without control in the uncongested scenario is 9.3 minutes and 6.7 minutes of waiting time due to traffic congestion. In the congested case, the average travel time is 20.6 minutes with 18.4 minutes of waiting time.

Tables I and II show that DeePC outperforms not having a control and the MPC baseline both in terms of average travel time reduction and CO_2 emissions. DeePC also decreases the

TABLE I: Uncongested scenario

	DeePC	MPC	No Control
Travel Time (min)	**5.84**	6.56	9.33
Waiting Time (min)	**2.89**	3.51	6.74
CO Emissions(kg)	**0.039**	0.046	0.076
CO_2 Emissions (kg)	**0.958**	1.075	1.501
HC Emissions (10^{-4}kg)	**2.03**	2.38	3.84
PMx Emissions (10^{-4}kg)	**0.20**	0.23	0.34
NOx Emissions (10^{-4}kg)	**4.16**	4.69	6.68

TABLE II: Congested scenario

	DeePC	MPC	No Control
Travel Time (min)	**18.41**	20.55	20.63
Waiting Time (min)	**15.56**	18.35	18.43
CO Emissions (kg)	**0.168**	0.192	0.192
CO_2 Emissions (kg)	**2.892**	3.241	3.254
HC Emissions (10^{-4}kg)	**8.35**	9.50	9.54
PMx Emissions (10^{-4}kg)	**0.70**	0.79	0.79
NOx Emissions (10^{-4}kg)	**13.13**	14.74	14.80

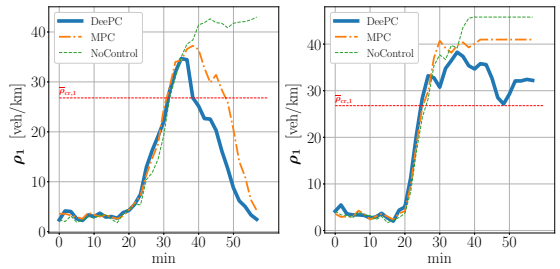
average travel time in the uncongested case by 37.4% over the no control baseline, compared with the 29.6% obtained by MPC. Fig.4 (a) highlights that DeePC maintains the density close to the critical one $\overline{\rho}_{\text{cr}}$, which in turn maximizes the flow Fig.4 (a) leading to the lower travel time. By taking the average time saved and CO_2 emissions and multiplying them for the number of vehicles, we get the total travel time and CO_2 that have been saved, respectively 151 hours and 1411 kg for DeePC. In the congested case DeePC the gain in travel time is 10,7%, while MPC does not perform significantly better than the no control baseline. This is because DeePC prevents the gridlock state, as shown in Fig. 4 (b), and achieves substantially higher flows. The total travel time and CO_2 emissions saved in this scenario are 318 hours and 3113 kg.

B. Case study 2: preliminary analysis of the Zürich network

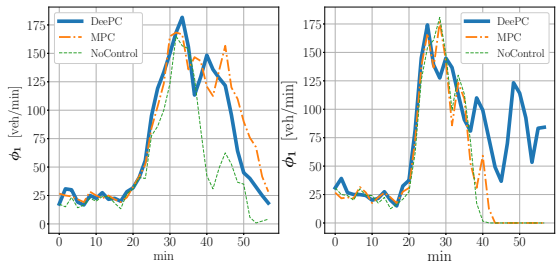
We present a preliminary study on DeePC over a larger simulation that is the entire traffic network of the city of Zürich, Switzerland. Leveraging a detailed map of the city, counting 14566 roads and 6917 intersections, the simulation offers a good approximation of the urban traffic dynamics by capturing many of the non-linear behaviours observed in reality. Fig. 5 illustrates the effectiveness of DeePC in controlling the system, resulting in increased flows and a reduced average density in the city center compared to the no-control baseline. This research direction will be explored in greater depth in an upcoming publication.

V. Conclusion

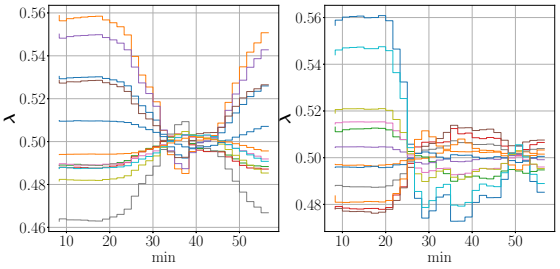
As urban areas continue to expand, exacerbating traffic congestion and emissions, data-driven traffic management strategies offer effective solutions to reduce congestion and emissions. This paper has leveraged the data-driven control algorithm DeePC to address the urban traffic control problem. Our preliminary results, based on a case study using the high-fidelity SUMO simulation software, suggests the great promise of the data-driven approach exemplified by DeePC in outperforming existing methods, particularly in key metrics such as travel time and CO_2 emissions. Future research should explore the engineering relevance of this approach with a real-world case study.



(a) Density region 1



(b) Flow in region 1



(c) Control input of the traffic lights

Fig. 4: Density (a), flow (b), optimal inputs of DeePC (c) related to the inner region are here shown divided in uncongested (left column) and congested (right column).

References

[1] D. Schrank, B. Eisele, and T. Lomax. (2019) 2019 urban mobility report. Texas A&M Transportation Institute. [Online]. Available: https://mobility.tamu.edu/umr/

[2] Council and the European Parliament. (2019) Fit for 55 package. [Online]. Available: https://www.consilium.europa.eu/en/policies/green-deal/timeline-european-green-deal-and-fit-for-55/

[3] U. Nations. (2018) 68% of the world population projected to live in urban areas by 2050, says un. [Online]. Available: https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html

[4] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[5] G. Ou, Z. Gao, and Y. Chen, "Reinforcement learning for adaptive traffic signal control: A q-learning approach," *Transportation Research Part C: Emerging Technologies*, vol. 86, pp. 298–317, 2018.

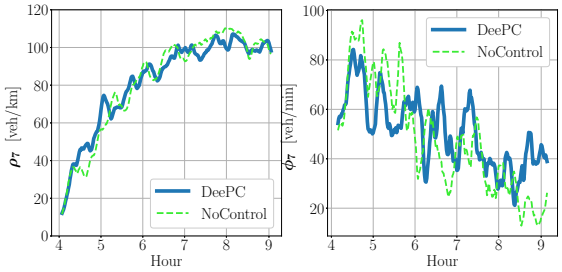


Fig. 5: Density (a) and the flow (b) of the center of Zürich under a no control policy and the DeePC control.

[6] W. Zhang, D. Wang, and N. Zheng, "A survey on urban traffic signal control: From classic to deep reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, pp. 7334–7352, 2021.

[7] Y. Li, J. Hong, and L. Zhang, "Urban traffic signal control: A comprehensive review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 1, pp. 380–394, 2020.

[8] J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control: In the shallows of the deepc," in *Proc. Eur. Control Conf.*, Naples, Italy, 2019, pp. 307–312.

[9] D. I. Robertson and R. D. Bretherton, "Optimizing networks of traffic signals in real time-the scoot method," *IEEE Transactions on vehicular technology*, vol. 40, no. 1, pp. 11–15, 1991.

[10] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Y. Wang, "Review of road traffic control strategies," *Proceedings of the IEEE*, vol. 91, no. 12, pp. 2043–2067, 2003.

[11] K. Aboudolas, M. Papageorgiou, A. Kouvelas, and E. Kosmatopoulos, "A rolling-horizon quadratic-programming approach to the signal control problem in large-scale congested urban road networks," *Transportation Research Part C: Emerging Technologies*, vol. 18, no. 5, pp. 680–694, 2010.

[12] S. Lin, B. De Schutter, Y. Xi, and H. Hellendoorn, "Efficient network-wide model-based predictive control for urban traffic networks," *Transportation Research Part C: Emerging Technologies*, vol. 24, pp. 122–140, 2012.

[13] ——, "Integrated urban traffic control for the reduction of travel delays and emissions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1609–1619, 2013.

[14] C. Cenedese, P. Stokkink, N. Geroliminis, and J. Lygeros, "Incentive-based electric vehicle charging for managing bottleneck congestion," *European Journal of Control*, vol. 68, p. 100697, 2022.

[15] P. D. Grontas, C. Cenedese, M. Fochesato, G. Belgioioso, J. Lygeros, and F. Dörfler, "Designing optimal personalized incentive for traffic routing using big hype algorithm," *Proc. 62th Conf. Decision Control*, 2023.

[16] Q. Guo, L. Li, and X. J. Ban, "Urban traffic signal control with connected and automated vehicles: A survey," *Transportation research part C: emerging technologies*, vol. 101, pp. 313–334, 2019.

[17] I. Markovsky and F. Dörfler, "Behavioral systems theory in data-driven analysis, signal processing, and control," *Ann. Rev. Control*, vol. 52, pp. 42–64, 2021.

[18] J. C. Willems, "From time series to linear system—Part I. Finite dimensional linear time invariant systems," *Automatica*, vol. 22, no. 5, pp. 561–580, 1986.

[19] P. G. Carlet, A. Favato, S. Bolognani, and F. Dörfler, "Data-driven predictive current control for synchronous motor drives," in *IEEE Energy Conversion Congress and Exposition*, Detroit, MI, USA, 2020, pp. 5148–5154.

[20] L. Huang, J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control for grid-connected power converters," in *Proc. 58th Conf. Decision Control*, Naples, Italy, 2019, pp. 8130–8135.

[21] J. Wang, Y. Zheng, K. Li, and Q. Xu, "DeeP-LCC: data-enabled predictive leading cruise control in mixed traffic flow," *IEEE Transactions on Control Systems Technology*, 2023.

[22] P. Zhu, G. Ferrari-Trecate, and N. Geroliminis, "Data-enabled predictive control for empty vehicle rebalancing," in *European Control Conference*, 2023, pp. 1–6.

[23] J. Wang, Y. Zheng, J. Dong, C. Chen, M. Cai, K. Li, and Q. Xu, "Implementation and experimental validation of data-driven predictive control for dissipating stop-and-go waves in mixed traffic," *IEEE Internet of Things Journal*, 2023.

[24] I. I. Sirmatel and N. Geroliminis, "Economic model predictive control of large-scale urban road networks via perimeter control and regional route guidance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1112–1121, 2018.

[25] N. Geroliminis, J. Haddad, and M. Ramezani, "Optimal perimeter control for two urban regions with macroscopic fundamental diagrams: A model predictive approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 348–359, 2013.

[26] M. Saeedmanesh and N. Geroliminis, "Clustering of heterogeneous networks with directional flows based on "snake" similarities," *Transportation Research Part B: Methodological*, vol. 91, pp. 250–269, 2016.

[27] J. C. Willems, P. Rapisarda, I. Markovsky, and B. L. M. De Moor, "A note on persistency of excitation," *Syst. Control Lett.*, vol. 54, no. 4, pp. 325–329, 2005.

[28] I. Markovsky and F. Dörfler, "Identifiability in the behavioral setting," *IEEE Trans. Autom. Control*, vol. 68, no. 3, pp. 1667–1677, 2023.

[29] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer, "Linear tracking mpc for nonlinear systems—part ii: The data-driven case," *IEEE Transactions on Automatic Control*, vol. 67, no. 9, pp. 4406–4421, 2022.

[30] Z. Jia, C. Chen, B. Coifman, and P. Varaiya, "The PeMS algorithms for accurate, real-time estimates of g-factors and speeds from single-loop detectors," in *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, 2001, pp. 536–541.

[31] C. G. Claudel and A. M. Bayen, "Guaranteed bounds for traffic flow parameters estimation using mixed lagrangian-eulerian sensing," in *2008 46th Annual Allerton Conference on Communication, Control, and Computing*, 2008, pp. 636–645.

[32] L. R. Foulds, H. A. do Nascimento, I. C. Calixto, B. R. Hall, and H. Longo, "A fuzzy set-based approach to origin–destination matrix estimation in urban traffic networks with imprecise data," *European Journal of Operational Research*, vol. 231, no. 1, pp. 190–201, 2013.

[33] L. E. Olmos, S. Çolak, S. Shafiei, M. Saberi, and M. C. González, "Macroscopic dynamics and the collapse of urban traffic," *Proceedings of the National Academy of Sciences*, vol. 115, no. 50, pp. 12 654–12 661, 2018.

[34] E. Elokda, C. Cenedese, K. Zhang, J. Lygeros, and F. Dörfler, "Carma: Fair and efficient bottleneck congestion management with karma," *arXiv preprint arXiv:2208.07113*, 2022.

[35] I. Tomar, S. Indu, and N. Pandey, *Traffic Signal Control Methods: Current Status, Challenges, and Emerging Trends*, D. Gupta, Z. Polkowski, A. Khanna, S. Bhattacharyya, and O. Castillo, Eds. Singapore: Springer Nature Singapore, 2022.

[36] N. Geroliminis and J. Sun, "Properties of a well-defined macroscopic fundamental diagram for urban traffic," *Transportation Research Part B: Methodological*, vol. 45, no. 3, pp. 605–617, 2011.

[37] N. Geroliminis, M. Saeedmanesh, and K. A., "A linear formulation for model predictive perimeter traffic control in cities," 2017.

[38] "Simulation of urban mobility : Sumo," https://sumo.dlr.de/docs/index.html, accessed: 2023-09-11.