Unjamming urban traffic: data-driven control and actuator selection strategies

A. Rimoldi[a,*], C. Cenedese[a], A. Padoan[a], J. Lygeros[a], F. Dörfler[a]

[a] ETH Zürich, Zürich, Switzerland
arimoldi@student.ethz.ch, ccenedese@ethz.ch, apadoan@ethz.ch jlygeros@ethz.ch
doerfler@control.ee.ethz.ch
[*] Corresponding author

Extended abstract submitted for presentation at the Conference in Emerging Technologies in Transportation Systems (TRC-30)
September 02-03, 2024, Crete, Greece

29 April 2024

Keywords: Data-driven, Traffic, Control, Actuator, Selection

1 INTRODUCTION

Due to the recent expansion of modern cities, the need for efficient traffic management systems has reached critical heights. Recent studies starkly highlight the problematic reality of urban traffic congestion, with commuters spending an average of 54 hours annually trapped in gridlock Schrank *et al.* (2019). In this landscape, data emerges as an indispensable asset Atzori *et al.* (2010) and data-driven traffic control algorithms are poised to redefine our approach to urban congestion Li *et al.* (2020). Recently, Data-enabled Predictive Control (DeePC) has been successfully applied to dynamic traffic lights to reduce urban traffic congestion, see Rimoldi *et al.* (2023). Despite the inherent complexity of the problem, the DeePC algorithm has proven effective at capturing the nonlinear dynamics of urban traffic networks, outperforming traditional model-based approaches. In this work, we introduce a Variable Speed Limits (VSL) control scheme to reduce congestion. Moreover, we delve into the critical aspect of optimally selecting actuators in a data-driven fashion, namely, the most important roads in the dense network of a city. Motivated and inspired by Sirmatel & Geroliminis (2018), this method capitalizes on an aggregate description of urban traffic dynamics based on Macroscopic Fundamental Diagram s (MFDs). We validate the effectiveness of our approach through the use of the *Unjam* Application Programming Interface (API) Unjam (2024) that allows for a simple interconnection of closed-loop controllers with complex Simulation of Urban MObility (SUMO) SUMO (2001) microsimulations.

Contributions: (i) We present a novel application of the DeePC algorithm with the goal of VSL control. (ii) We summarize the theoretical formulation of DeePC to perform data-driven actuator selection and heuristics used to lower the computational burden. (iii) We show the effectiveness of the control policy and actuator selection strategies via the SUMO microsimulation software automated through *Unjam* attaining an improvement in total travel time and emission metrics such as CO2 emitted and fuel consumption.

2 DeePC for VSL control and actuator selection

2.1 Data-driven control of urban VSL

We borrow most of the notation from Rimoldi *et al.* (2023). The s sensors within the city measure the densities $\boldsymbol{\rho} \in \mathbb{R}^s$. The density dynamics are described by $f : \mathbb{R}^{s+l+p^2} \to \mathbb{R}^s$ and read as $\boldsymbol{\rho}(t+1) = f(\boldsymbol{\rho}(t), \boldsymbol{\lambda}(t), \boldsymbol{d}(t))$ where $\boldsymbol{\lambda}(t) \in \mathbb{R}^l$ is the speed limit during the interval t applied to the l streets controlled, and $\boldsymbol{d}$ the demand (assumed to be known) among the p regions in which the city is divided using MFDs, see (Rimoldi *et al.*, 2023, Sec. III.A). To make the problem tractable and reduce fluctuations due to local phenomena, we use $\boldsymbol{\lambda}$ to steer the average density of each region, i.e., $\overline{\boldsymbol{\rho}}(t) = h(\boldsymbol{\rho}(t))$ with $h : \mathbb{R}^s \to \mathbb{R}^p$ being a suitable averaging function.

We can use the above model to frame the VSL control problem in the classical DeePC form. The behavior $\mathcal{B} \in \mathcal{L}^{m+p}$ represents the dynamical system defined by the traffic network, with m inputs $u := (u^{\boldsymbol{\lambda}}, u^{\boldsymbol{d}}) = (\boldsymbol{\lambda}, \boldsymbol{d}) \in \mathbb{R}^m$ and p outputs $y = \overline{\boldsymbol{\rho}}$. Next, we define the input/output data $w_{\mathrm{d}} = \mathrm{col}(u_{\mathrm{d}}, y_{\mathrm{d}}) \in \mathcal{B}|_{[1,T]}$ of given length $T \in \mathbb{N}$ recorded offline, a future time horizon $T_{\mathrm{f}} \in \mathbb{N}$, the past input/output data $w_{\mathrm{ini}} = \mathrm{col}(u_{\mathrm{ini}}, y_{\mathrm{ini}}) \in \mathcal{B}|_{[1,T_{\mathrm{ini}}]}$ of given length $T_{\mathrm{ini}} \in \mathbb{N}$, an output constraint set $\mathcal{Y} \subseteq \mathbb{R}^{pT_{\mathrm{f}}}$, and an input constraint set $\mathcal{U} \subseteq \mathbb{R}^{mT_{\mathrm{f}}}$, where we impose that $u^d_t = d(t)$ for all t. The reference trajectory for the output $\hat{y} = (\hat{y}_0, \hat{y}_1, \cdots) \in (\mathbb{R}^p)^{\mathbb{N}}$ is assumed to be constant and equal to the critical densities of the p regions extracted directly from the associated MFDs. As reference input $\hat{u} = (\hat{u}_0, \hat{u}_1, \cdots) \in (\mathbb{R}^m)^{\mathbb{N}}$, we use the standard speed limit imposed on the controlled streets $\hat{\boldsymbol{\lambda}}$, hence for all t we define $\hat{u}_t = (\hat{\boldsymbol{\lambda}}, \boldsymbol{d}(t))$.

The DeePC algorithm relies on solving in receding horizon the following optimization problem

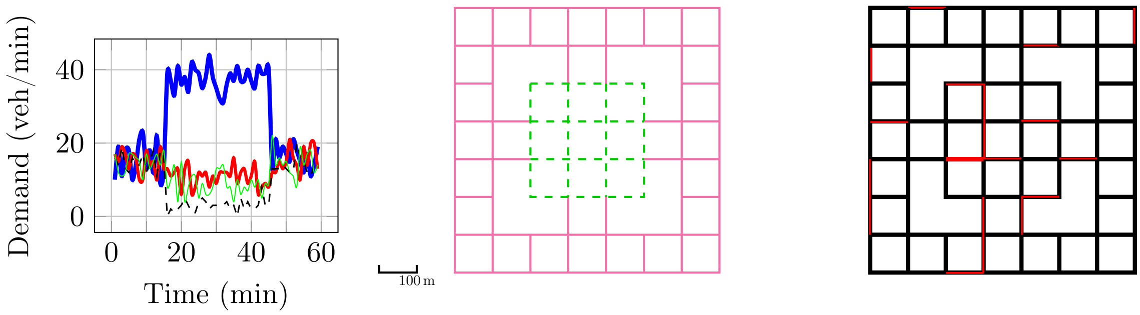
$$
\begin{aligned}
\min_{u,y,g} \ & \sum_{k=1}^{T_{\mathrm{f}}} \|y(k) - \hat{y}(t+k)\|_Q^2 + \|u(k) - \hat{u}(t+k)\|_R^2 + \psi(g) + \lambda_y ||\sigma||_1 \\
\text{s.t.} \ & \begin{pmatrix} U_{\mathrm{p}} \\ Y_{\mathrm{p}} \\ U_{\mathrm{f}} \\ Y_{\mathrm{f}} \end{pmatrix} g = \begin{pmatrix} u_{\mathrm{ini}} \\ y_{\mathrm{ini}} \\ u \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ \sigma \\ 0 \\ 0 \end{pmatrix}, \\
& u \in \mathcal{U}, \ y \in \mathcal{Y},
\end{aligned}
\tag{1}
$$

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. Moreover, $Q \succ 0$ and $R \succ 0$ are weights in the cost and we used the regularization function $\psi : \mathbb{R}^{T-L+1} \to \mathbb{R}$ and parameter $\lambda_y \in \mathbb{R}$. For a more in-depth explanation, see Coulson *et al.* (2019).

2.2 Actuator Selection

Expanding on our prior work Rimoldi *et al.* (2023), we present a version of the DeePC algorithm enhanced with an optimal actuator selection strategy tailored for the dynamical system $\mathcal{B}$. Our methodology capitalizes on the fundamental lemma (see J.C. (2007)). Specifically, we aim to transform the raw data matrix representation of behavior $\mathcal{B} \in \mathbb{L}^{m+p}$ into a refined behavior with fewer inputs, denoted as $\hat{\mathcal{B}} \in \mathbb{L}^{\hat{m}+p}$, while minimizing a predefined cost function. Formally, we aim to find a selection matrix F that, given a desired trajectory $w^\star = \mathrm{col}(u^\star, y^\star) \in \mathbb{R}^{(p+m)L}$, minimizes a specified cost function. The matrix F is square, with only one non-zero element for each row and column, thus enabling us to select pertinent components from $\mathcal{B}$. Under the assumptions of the fundamental lemma, any trajectory $w^\star = \mathrm{col}(u^\star, y^\star) \in \mathbb{R}^{(p+m)L}$ is represented in $\mathcal{B}|_{[1,T]}$ if and only if a corresponding vector $g \in \mathbb{R}^{T-L+1}$ exists, satisfying the equation

$$
\begin{pmatrix} U \\ Y \end{pmatrix} g = \begin{pmatrix} u^\star \\ y^\star \end{pmatrix},
$$



(a) *The demands are divided by origin and destination.*

(b) *Lattice traffic network partitioned into an outer region (solid magenta line) and the inner region (dashed green line).*

(c) *The actuators selected via the data-driven algorithm (in red). The pre-selected ones are the 16 streets in region 0 connecting it to 1.*

Figure 1 – *The network, the demand, and the actuators used in the simulations.*

where $U := \mathrm{col}(U_\mathrm{p}, U_\mathrm{f})$, $Y := \mathrm{col}(Y_\mathrm{p}, Y_\mathrm{f})$. Therefore, our objective is to determine the selection matrix F that minimizes the discrepancy between observed and desired trajectories, effectively optimizing actuator utilization within the urban traffic control framework.

3 Simulations

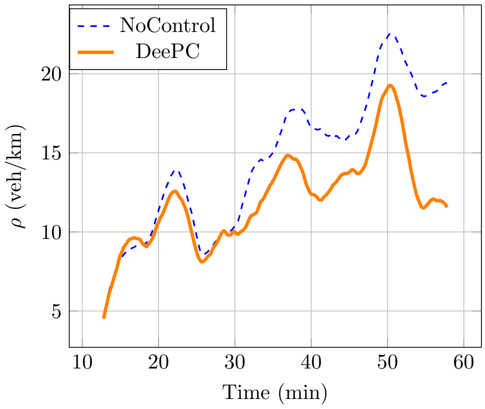
Numerical simulations have been performed to validate our approach using the state-of-the-art microscopic traffic simulation software SUMO and *Unjam* API. Following Rimoldi *et al.* (2023) consider a traffic network with a lattice structure, composed by an outer *region 0* and an inner *region 1*, as shown in Fig. 1b. The network is composed of $p = 208$ roads, each of which includes one traffic signal which can be used as an actuator to implement VSL. We consider a randomly generated noisy demand represented in Fig. 1a, which is kept fixed throughout all our experiments. The demand is designed to emulate the morning traffic peak observed in a real traffic system, with the majority of the trips being directed from the outer to the inner region during a limited window of time. The simulation spans the time of one hour, with 3600 vehicles being introduced during this time.

3.1 Results

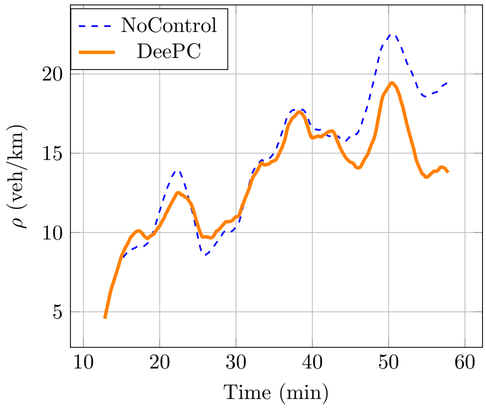
We now give a brief overview of the results. The baseline simulation we use to compare the DeePC control with is a No Control Baseline Simulation (NoControl) which is a simulation where no control policy is applied. Furthermore we compare the performance of DeePC with data-driven actuator selection (DeePC D^{2}AS) against a set of manually pre-selected edges (DeePC PA) informed only by the network partitioning between the outer *region 0* and the inner *region 1*. Fig. 2 shows the evolution of the traffic density and flow for the inner *region 1*. The density achieved under the DeePC control policy are consistently lower then under the no control policy. Comparing Fig. 2a and Fig. 2b we can infer that using the data-driven selected actuators, instead of the pre-selected ones, enables the control algorithm to achieve lower traffic densities. Table 1 shows travel time and emissions metrics under the different control conditions.

4 Discussion

As infrastructural changes in urban areas become less and less economically feasible, worsening traffic congestion and emissions call for an exploitation of data-driven procedures to reduce



(a) *Evolution of the density in the inner region under the DeePC control policy using the data-driven selected actuators*



(b) *Evolution of the density in the inner region under the DeePC control policy using the pre-selected actuators*

Figure 2 – *Time history of the density of the lattice network*

Table 1 – *Metrics*

	No Control	DeePC PA	DeePC D^2AS
Travel Time (h)	264,24	255,06	$\mathbf{244,83}$
Waiting Time (h)	119,8	112,29	$\mathbf{104,5}$
CO2 Emissions (kg)	2632,9	2559,2	$\mathbf{2471,5}$
Fuel consumption (kg)	839,81	816,32	$\mathbf{788,33}$
Trips completed	3579	**3584**	3583

congestion. In this paper we devised a data-driven selection and control pipeline for actuator selection and control, leveraging the control algorithm DeePC our study shows how data-driven procedures can be used to decrease congestion in urban settings. Given the encouraging results on the lattice network example a future expansion of the work will use a real-scale simulation of the city of Zürich.

References

Atzori, Luigi, Iera, Antonio, & Morabito, Giacomo. 2010. The Internet of Things: A survey. *Computer Networks*, **54**(15), 2787–2805.

Coulson, J., Lygeros, J., & Dörfler, F. 2019. Data-enabled predictive control: In the shallows of the DeePC. *Pages 307–312 of: Proc. Eur. Control Conf.*.

J.C., Willems. 2007. The Behavioural Approach to Open and Interconnected Systems.

Li, Y., Hong, J., & Zhang, L. 2020. Urban Traffic Signal Control: A Comprehensive Review. *IEEE Transactions on Intelligent Transportation Systems*, **21**(1), 380–394.

Rimoldi, Alessio, Cenedese, Carlo, Padoan, Alberto, Dörfler, Florian, & Lygeros, John. 2023. *Urban traffic congestion control: a DeePC change*.

Schrank, D., Eisele, B., & Lomax, T. 2019. *2019 Urban Mobility Report*.

Sirmatel, Isik Ilber, & Geroliminis, Nikolas. 2018. Economic Model Predictive Control of Large-Scale Urban Road Networks via Perimeter Control and Regional Route Guidance. *IEEE Transactions on Intelligent Transportation Systems*, **19**(4), 1112–1121.

SUMO. 2001. *Simulation of Urban Mobility : SUMO*. https://sumo.dlr.de/docs/index.html. Accessed: 2023-09-11.

Unjam. 2024. *Unjam : Intelligent traffic control for smarter mobility*. https://unjam.godaddysites.com/. Accessed: 2024-03-19.