Data-enabled predictive control for dynamic traffic routing

Kai Zhang[1], Kenan Zhang*[2], Linbin Huang[1], Giuseppe Belgioioso[1], John Lygeros[1], and Florian Dörfler[1]

[2]ENAC, EPFL, Switzerland
[1]Automatic Control Laboratory, ETH Zürich, Switzerland

Short summary

In the vision of wide adoption of autonomous vehicles (AVs), this paper proposes a dynamic traffic routing algorithm in the framework of data-driven predictive control (DeePC) for traffic congestion mitigation. Different from existing model-based approaches, DeePC does not require a parametric model to describe the traffic dynamics but directly predicts the network throughput given the route assignment based on historical observations. Compared to learning-based methods, DeePC is also more flexible in incorporating additional constraints. Yet, two challenges emerge when applying DeePC to dynamic traffic routing. First, DeePC is designed for linear time-invariant systems, whereas the behaviors of traffic networks are highly nonlinear. To address system nonlinearity, we adopt a relaxed formulation of DeePC and propose customized data collection strategies. Secondly, DeePC can hardly be centrally solved on large networks due to the great number of feasible routes and flow conservation constraints. To tackle this scalability issue, we develop a distributed optimization algorithm that decomposes the original DeePC problem into local problems and solves them in a coordinative way. Our numerical experiments on both small and medium-sized networks demonstrate the performance of DeePC in optimizing the network throughput and reducing traffic congestion. Further, DeePC archives reasonable fairness and shows a higher robustness towards the prediction error of traffic demand, which implies great potential in real practice.
Keywords: autonomous vehicles, data-driven control, dynamic traffic routing, model predictive control, traffic control

1 Introduction

Large cities around the world have long been suffering from serious traffic congestion and its consequent social costs (INRIX, 2020). In the meanwhile, autonomous-driving technologies have developed rapidly in recent years. Besides enhancing convenience, comfort, and safety in individual mobility, autonomous vehicles (AVs) are also believed to create unprecedented opportunities for urban traffic control and management (Li et al., 2023). In particular, the full controllability of AVs can be utilized at different levels. At the micro level, the car-following and lane-changing behaviors of AVs can be optimized to create platooning and achieve speed harmonization (Gong & Du, 2018; Malikopoulos et al., 2018). At the macro level, trajectories and routes of AVs can be centrally controlled to improve traffic throughput, reduce congestion, and even induce more desirable behaviors of human drivers (Lu et al., 2019; Zhang & Nie, 2018).

Motivated by the foreseeable adaptation of AVs, this study investigates network-level vehicle route control for traffic mitigation, whose potential has been demonstrated in several recent studies using static traffic assignment models (e.g., Zhang & Nie, 2018; Chen et al., 2020) and day-to-day dynamics (e.g., Guo et al., 2022). However, the route control in real time is rather complicated because, instead of the shortest path myopically, it has to determine the route assignment such that the total network throughput is maximized. To bypass such difficulty, some studies have turned to learning-based approaches (e.g., Lazar et al., 2021; Toghi et al., 2022). While learning-based approaches show promising performance, they are often computationally expensive and hard to generalize among use cases. Besides, it still remains challenging for existing learning algorithms to incorporate non-trivial constraints (Malikopoulos et al., 2018).

*Corresponding author: `kenan.zhang@epfl.ch`

In contrast to previous works, this study tackles the dynamic routing problem via model predictive control (MPC). The essence of MPC is to control a dynamic process with a predictive model with a finite rolling horizon Rawlings (2000); Kwon & Han (2005). At each time step, MPC solves an optimal control problem over the next horizon, where the future system states and outputs are estimated by the predicted model using measurements of the current state. Then, the controller employs one or several steps in the solution, moves to the next step, and repeats the same process. In this study, we adopt a data-driven MPC framework named *data-enabled predictive control* (DeePC) Coulson et al. (2019). In short, DeePC constructs a non-parametric predictive model using historical observations from the dynamic system. Different from the learning-based approaches, DeePC allows one to incorporate input and output constraints and thus can easily capture the physics in traffic networks (e.g., flow conservation). Although DeePC is developed for deterministic linear time-invariant (LTI) systems, it shows satisfactory performance and robustness in multiple nonlinear and stochastic systems (Elokda et al., 2021; Huang et al., 2019, 2023). Its application in transportation, however, is still rare and mostly limited at the vehicle level (Wang et al., 2023), whereas a recent study has applied DeePC for regional ride-hailing vehicle rebalancing (Zhu et al., 2023). To the best of our knowledge, this is the first study that applies DeePC in route control at the network level.

2 Methodology

Consider a pure AV environment where all vehicles' routes to their destination are centrally determined upon their entries into the road network. Let $\mathcal{W}$ denote the set of origin-destination (OD) pairs with $m = |\mathcal{W}|$. For each OD pair $w \in \mathcal{W}$, we denote $\mathcal{P}_w$ as its path set and $\mathcal{P} := \cup_{w \in \mathcal{W}} P_w$ as the network path set with $p = |\mathcal{P}|$. To describe the relationship between paths and OD pairs, we define a mapping matrix $M \in \{0,1\}^{p \times m}$ such that $M_{ij} = 1$ if path j connects OD pair i and zero otherwise. To describe the network traffic as a discrete-time dynamic system, we discretize the study horizon into time steps with an equal interval. Accordingly, the demand flow entering the network at each time step t is denoted by $d_t \in \mathbb{R}^m_{\geq 0}$ and the *control input* is given by the route assignment vector $u_t \in \mathbb{R}^p_{\geq 0}$ that satisfies the flow conservation constraint

$$Mu_t = d_t, \quad \forall t. \tag{1}$$

Let $y_t^w \in \mathbb{R}_{\geq 0}$ denote the number of vehicles that travel between OD pair w arriving at their destination at each time t. Then, the network throughput is given by $y_t = \sum_{w \in \mathcal{W}} y_t^w$, which is also considered as the *control output* in this study.

Construction of Hankel matrix

The key component of DeePC is the Hankel matrix, which stores all the information about the system dynamics. Let $\mathcal{D} = \{u_d, y_d\}_{d=1}^D$ represent a long trajectory of control inputs and outputs collected from the system and H be the predictive horizon in the optimal problem at each time step. The Handel matrix defined on data $\mathcal{D}$ and trajectory length $H < L < D$, denoted as $\mathcal{H}_L(\mathcal{D})$, is constructed as follows:

$$\mathcal{H}_L(\mathcal{D}) := \left[\begin{array}{cccc} u_1 & u_2 & \cdots & u_{D-L+1} \\ u_2 & u_3 & \cdots & u_{D-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ u_{L-H} & u_{L-H+1} & \cdots & u_{D-H} \\ \hline y_1 & y_2 & \cdots & y_{D-L+1} \\ y_2 & y_3 & \cdots & y_{D-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ y_{L-H} & y_{L-H+1} & \cdots & y_{D-H} \\ \hline u_{L-H+1} & u_{L-H+2} & \cdots & u_{D-H+1} \\ \vdots & \vdots & \ddots & \vdots \\ u_L & u_{L+1} & \cdots & u_D \\ \hline y_{L-H+1} & y_{L-H+2} & \cdots & y_{D-H+1} \\ \vdots & \vdots & \ddots & \vdots \\ y_L & y_{L+1} & \cdots & y_D \end{array}\right] = \left[\begin{array}{c} U_P \\ \hline Y_P \\ \hline U_F \\ \hline Y_F \end{array}\right], \tag{2}$$

where $U_P \in \mathbb{R}^{pH_{ini} \times H_c}, Y_P \in \mathbb{R}^{mH_{ini} \times H_c}, U_F \in \mathbb{R}^{pH \times H_c}, Y_F \in \mathbb{R}^{mH \times H_c}$ represent four blocks in the Hankel matrix with $H_{ini} = L - H$ as the observation horizon and $H_c = D - L + 1$ as the column number.

As per the Fundamental Lemma in Willems et al. (2005), if $\mathcal{D}$ is collected from a controllable LTI system and the control input $\{u_d\}_{d=1}^{D}$ is *persistently exciting* of order L, then any trajectory of the dynamic system $(u_{init}, y_{init}, u, y)$ satisfies

$$\begin{bmatrix} U_P \\ Y_P \\ U_F \\ Y_F \end{bmatrix} g = \begin{bmatrix} u_{ini} \\ y_{ini} \\ u \\ y \end{bmatrix} \tag{3}$$

for some $g \in \mathbb{R}^{H_c}$. The condition of persistent excitement is later shown to hold when the rank of the Hankel matrix is sufficiently large, i.e., $\text{rank}(\mathcal{H}) \geq mL + n$, where n is the order of the dynamic system (Markovsky & Dörfler, 2022).

Centralized DeePC

With the Hankel matrix, we are now ready to present the optimal control problem in DeePC at each time step t as follows:

$$\begin{aligned}
\min_{g,u,\sigma} \quad & -\sum_{k=0}^{H-1} y_{k|t} + \lambda_g ||g||_2^2 + \lambda_y ||\sigma||_2^2 && \text{(4a)} \\
s.t. \quad & \begin{bmatrix} U_P \\ Y_P \\ U_F \\ Y_F \end{bmatrix} g = \begin{bmatrix} u_{ini} \\ y_{ini} \\ u \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ \sigma \\ 0 \\ 0 \end{bmatrix}, && \text{(4b)} \\
& M u_{k|t} = d_{k|t}, \quad \forall k, && \text{(4c)} \\
& u_{k|t} \geq 0, \quad \forall k, && \text{(4d)} \\
& y_{k|t} \geq 0, \quad \forall k. && \text{(4e)}
\end{aligned}$$

In Problem (4), the subscript $k|t$ denotes the k-th time step after t, σ is introduced to capture the nonlinearity in system dynamics and the noise in observations, $\lambda_g, \lambda_\sigma$ are the parameters of the l_s-regularization on g and σ.

Distributed DeePC

The centralized DeePC can be efficiently solved for small networks. However, as the traffic network expands, both the number of OD pairs m and the size of the path set p grow rapidly. Furthermore, to satisfy the condition for persistency of excitation, the length of historical input and output trajectories must also increase accordingly. These two factors together dramatically increase the dimension of the Hankel matrix. As a result, the problem can hardly be solved centrally for large networks with a limited computational budget. To tackle this scalability issue, we propose a distributed algorithm by exploiting the structure of the original DeePC problem.

Note that the OD-path matrix M is arranged by OD pairs and can be decomposed based on origins as illustrated in Fig. 1. Similarly, we can decompose the Hankel matrix with respect to each origin, as well as other variables. Let $\mathcal{R}$ denote the set of origins, then, for each origin $r \in \mathcal{R}$, we can define a local optimal control problem as follows:

$$\begin{aligned}
\min_{g_r,u_r,\sigma_r} \quad & -\sum_{k=0}^{H-1} y_{k|t,r} + \lambda_g ||g_r||_2^2 + \lambda_y ||\sigma_r||_2^2 && \text{(5a)} \\
s.t. \quad & \begin{bmatrix} U_{P,r} \\ Y_{P,r} \\ U_{F,r} \\ Y_{F,r} \end{bmatrix} g_r = \begin{bmatrix} u_{init,r} \\ y_{init,r} \\ u_r \\ y_r \end{bmatrix} + \begin{bmatrix} 0 \\ \sigma_r \\ 0 \\ 0 \end{bmatrix}, && \text{(5b)} \\
& M_r u_{k|t,r} = d_{k|t,r}, \quad \forall k, && \text{(5c)} \\
& u_{k|t,r} \geq 0, \quad \forall k, && \text{(5d)} \\
& y_{k|t,r} \geq 0, \quad \forall k. && \text{(5e)}
\end{aligned}$$

Note that Problem (5) remains the same as Problem (4) except for the extra subscript r, though the problem complexity is largely reduced thanks to the decreases in the number of decision variables and constraints. For instance, the number of (5b) drops from $(m+p)L$ to $(m_r+p_r)L$, where m_r and p_r are the number of OD pairs with origin r and that of path starting from r, respectively.

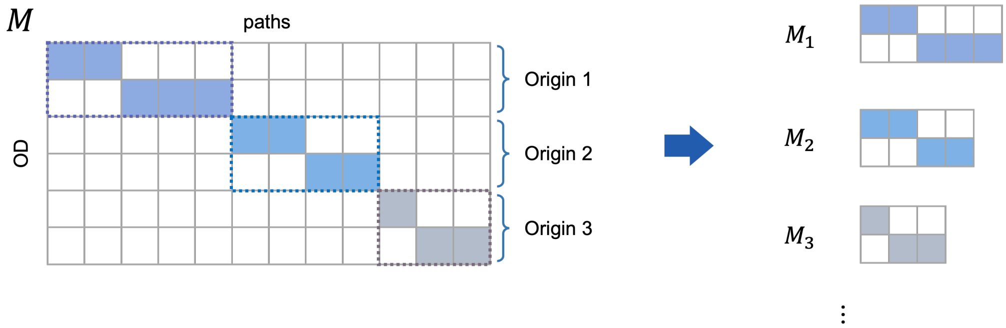


Figure 1: Decomposition of OD-path matrix.

By substituting u_r, y_r, and σ_r in Eq. (5a) with the system dynamics (5b), we can rewrite Problem (5) in a more compact form as follows:

$$\min_{g_r} \quad f_r(g_r) = -(Y_{F,r}g_r)^T\mathbf{1} + \lambda_g\|g_r\|_2^2 + \lambda_y\|Y_{P,r}g_r - y_{ini,r}\|_2^2 \tag{6a}$$

$$s.t. \quad g_r \in \mathcal{C}_r = \begin{cases} U_{P,r}g_r = u_{ini,r}, \\ U_{F,r}g_r \geq 0, \\ Y_{F,r}g_r \geq 0, \\ (I_H \otimes M_r)\, U_{F,r}g_r = d_r, \end{cases} \tag{6b}$$

where $\mathbf{1}$ denotes the unit vector, I_H denotes the identity matrix of dimension H, $\otimes$ represents the Kronecker product, and d_r denotes the demand vector starting from origin r over the prediction horizon.

Note that vectors g_r in all local problems not only share the same dimension H_c but also must reach consensus to ensure the constraint Eq. (4b). Problem (4) is thus reformulated as a distributed optimization problem:

$$\begin{aligned} \min_{\{g_r\}_{r\in\mathcal{R}}} \quad & \sum_{r\in\mathcal{R}} f_r(g_r) \\ s.t. \quad & g_r = g_{r'}, \quad \forall r, r' \\ & g_r \in \mathcal{C}_r, \ \forall r. \end{aligned} \tag{7}$$

Problem (7) can be solved by a semi-decentralized algorithm based on the Douglas-Rauchford operator splitting method (Bauschke & Combettes, 2011), which is detailed in Algorithm 1.

Algorithm 1 Semi-decentralized algorithm for DeePC

Inputs: $\alpha > 0$; $\{\gamma^i\}_{i\in\mathbb{Z}_{\geq 0}}$ s.t. $\gamma^i \in [0,2]\ \forall i$ and $\sum_{i=0}^{\infty}\gamma^i(2-\gamma^i) = \infty$.
Outputs: g^*, u^*.
Initialization: Set $i = 0$ and $z_r^i = 0, \forall r$.
Iterate until convergence:

1. Central coordination:
$$\bar{z}^i = \frac{1}{|\mathcal{R}|}\sum_{r\in\mathcal{R}} z_r^i$$

2. Local update: for $r \in \mathcal{R}$,
$$\begin{aligned} g_r^{i+1} &= \arg\min_{g_r\in\mathcal{C}_r} \quad f_r(g_r) + \frac{1}{2\alpha}\|g - (2\bar{z}^i - z_r^i)\|_2^2 \\ z_r^{i+1} &= z_r^i + \gamma^i\left(g_r^{i+1} - \bar{z}^i\right) \end{aligned}$$

Set $g^* = \bar{z}^i$, $u^* = U_F g^*$.

---

3 Results and discussion

Simulation setup and evaluation metrics

The simulation experiments reported in this section are conducted in SUMO, where the controller is coded in Python 3.9 using OSQP library[1] and interacts with the simulation environment through the SUMO Traffic Control Interface (TraCI)[2].
The widely studied Braess and Sioux Falls (SF) networks are used to evaluate the performance of DeePC. Specifically, the Braess network is applied for a sanity check because the optimal routing policy is known a priori. On the other hand, the SF network is used to demonstrate centralized and distributed DeePC for more complex traffic dynamics. Since our local workstation cannot solve centralized DeePC on the full SF network, we also generate a cropped SF network with fewer OD pairs (29/528) that contribute to the majority of travel demand, while the full SF network is used to demonstrate the performance of distributed DeePC.
In all tested networks, we first specify the maximum demand for each OD pair, denoted by $d_{max} = \{d_{max,w}\}_{w \in \mathcal{W}}$, and then design a time-vary ratio $\{\eta_t\}$ to capture the traffic pattern during the day. To replicate the fluctuations in demand, we further introduce Gaussian noise $\varepsilon_t \sim \mathcal{N}(0, \sigma^2)$ to the temporal, which yields the actual demand as

$$d_t = (\eta_t + \varepsilon_t) d_{max}. \tag{8}$$

An example of the demand profile is given in Figure 2.

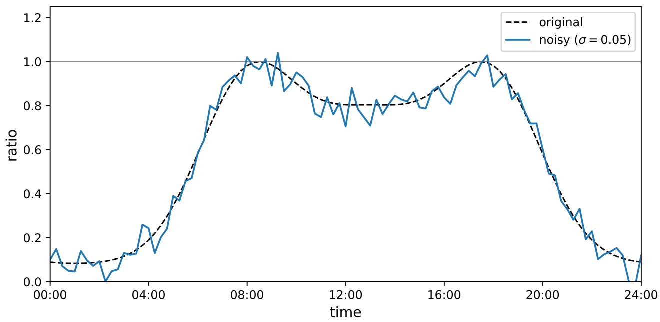


Figure 2: Example of daily demand profile (dash line represents the noise-free demand).

In all experiments, we set each time step as 15 min, the observation horizon as $H_{init} = 10$ (2.5h) and the prediction horizon as $H = 4$ (1h). Three metrics are used to assess the performance of DeePC and compare it with other benchmark policies: (i) the number of vehicles in the network; (ii) the daily average travel time; and (iii) the unfairness defined as the average difference between the realized and minimum travel times. The first two are the primary indicators of system efficiency, or the severity of traffic congestion, while the last one is considered an equity measure. Mathematically, the unfairness is computed as follows:

$$w_t = \frac{u_t^T (c_t - \mu_t)}{u_t^T \mathbf{1}}, \tag{9}$$

where $c_t \in \mathbb{R}^p$ is the path travel times, $\mu_t \in \mathbb{R}^p$ is the minimum path travel times between the OD pair corresponding to each path. Hence, the larger w_t, the more serious unfairness induced by the routing policy.

Data sampling and benchmark policies

For simple LTI systems, a random control policy is sufficient to generate input-output trajectories that well represent the system dynamics Huang et al. (2021). However, our preliminary experiments show that the random policy performs badly for data collection. When demand is high, random route control often leads to serious congestion and finally yields a simulation breakdown. As alternatives, we propose to use more reasonable routing policies to collect data. For the Braess

[1] `https://osqp.org/docs/index.html`
[2] `https://sumo.dlr.de/docs/TraCI.html`

network, we consider the route assignment under static user equilibrium (UE) (Sheffi, 1985), while for the Sioux Falls network, we apply both static UE and myopic routing policies. The latter distributes vehicles based on the path travel time upon their arrival. At time t, the proportion of vehicles on path $j \in \mathcal{P}_w$ between OD pair w is given by

$$\rho_{t,j} = \frac{\exp\left(-\theta c_{t-1,j}\right)}{\sum_{j' \in \mathcal{P}_w} \exp\left(-\theta c_{t-1,j'}\right)}, \tag{10}$$

where θ is the dispersion parameter. These two data collection policies are essentially motivated by real practice, where we are more likely to observe route choices between UE (e.g., commute trips) and myopic (e.g., non-regular trips).

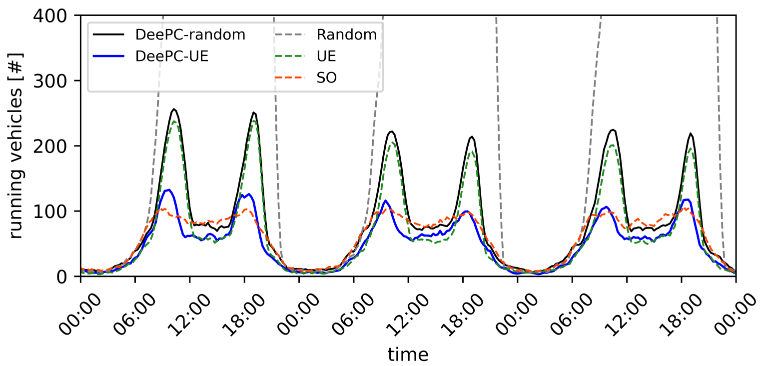
As benchmarks, we implement static user equilibrium (UE) and system optimum (SO) policies on both networks, while for Sioux Falls, we further numerically solve the dynamic UE and SO policies, referred to as "pseudo-DUE" and "pseudo-DSO" hereafter. For pseudo-DUE, we update the route assignment fraction $\rho_{t,j}$ on a daily basis according to the realized path travel time until it converges. As for pseudo-DSO, the updates are guided by the marginal travel time numerically computed between two consecutive time steps.

Main findings

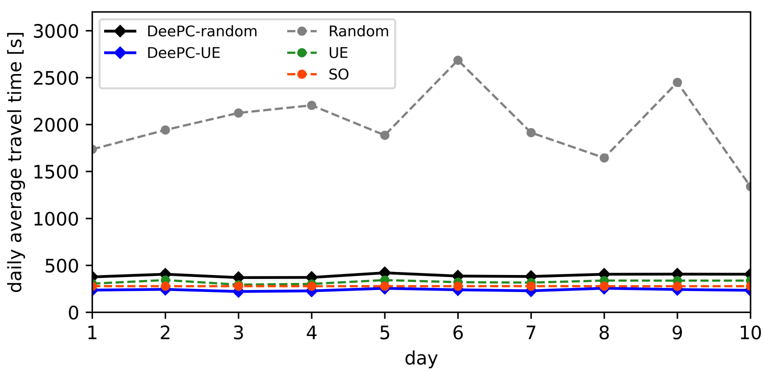
Due to the space limit, we only report the main findings from the experiments as follows:

- Recovery of optimal control

  Our experiments on the Braess network demonstrate the potential of DeePC in recovering the optimal route control policy. As illustrated in Figure 3, DeePC with UE sampling approaches the lower bound between static UE and SO in both congestion measures. Interestingly, DeePC with random sampling performs closely to UE. This result showcases the importance of the data sampling strategy of DeePC in a highly non-linear dynamic system.



(a) Number of running vehicles

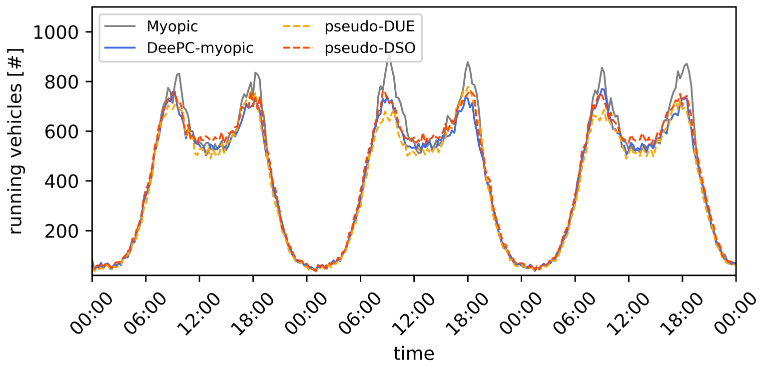


(b) Daily average travel time

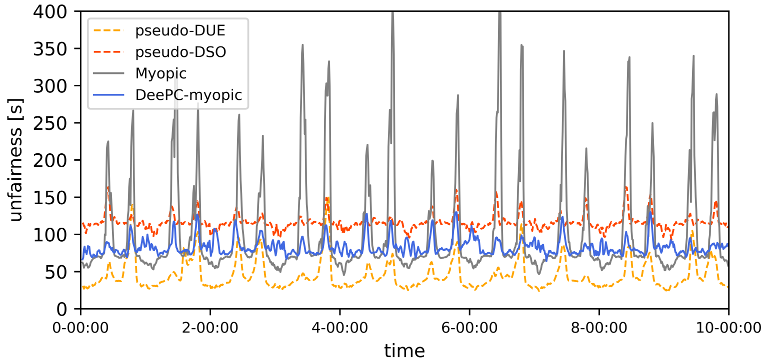
Figure 3: Performance of centralized DeePC on Braess network.

- Balance between efficiency and equity

Our experiments of centralized DeePC on the cropped SF network further demonstrate its efficiency. Meanwhile, the results show that DeePC also achieves satisfactory fairness. As shown in Figure 5a, DeePC performs similarly to pseudo-DUE and pseudo-DSO and outperforms myopic routing. The gap between all benchmark policies is however not substantial possibly due to (i) the network topology and demand profile, and (ii) pseudo-DUE and pseudo-DSO are not solved to optimum. On the other hand, Figure 5b confirms that pseudo-DSO induces more serious unfairness compared to pseudo-DUE, where some vehicles suffer from much longer detours. DeePC, however, manages to retain the unfairness at a medium level without the sudden spikes occurring in myopic routing.



(a) Number of running vehicles



(b) Unfairness

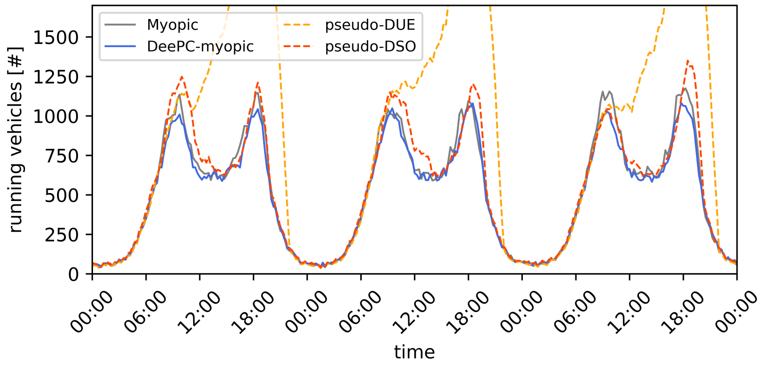
Figure 4: Performance of centralized DeePC on cropped SF network.

- Robustness towards demand prediction errors

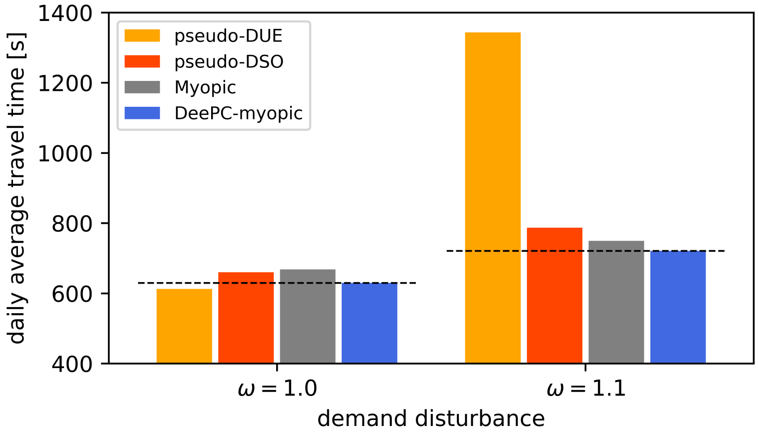
  Another advantage of DeePC verified by the simulation results is its robustness towards inaccurate predictions of demand. Figure 5 reports the performance of DeePC and other benchmarks under a slightly twisted demand profile ωd. As expected, myopic routing can adapt to the demand increase because it produces route assignments based on real-time traffic conditions. Pseudo-DSO shows reasonable robustness whereas pseudo-DUE leads to serious congestion. Surprisingly, DeePC performs the best among all benchmarks.

- Demonstration on scalability

  We finally tested the distributed DeePC on the full SF network. The original problem has 44,868 variables and 45,258 constraints. After decomposition, the size of each local problem reduces to 30,330 variables and 870 constraints. With a mild consensus threshold, Algorithm 1 manages to terminate in 20 seconds. The results shown in Figure 6 are similar to Figure 3, where DeePC outperforms the static UE and SO policies in terms of network efficiency.

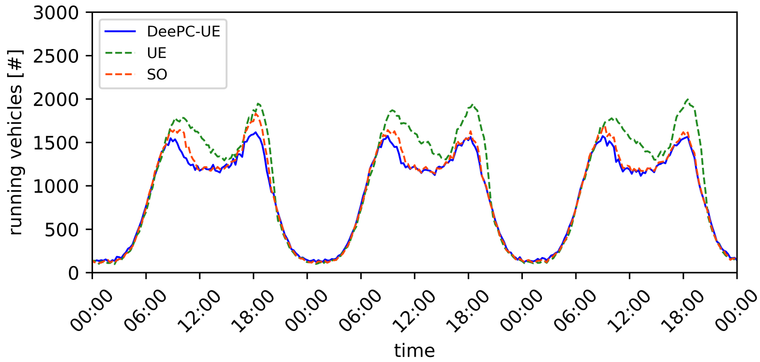


(a) Number of running vehicles ($\omega = 1.1$)

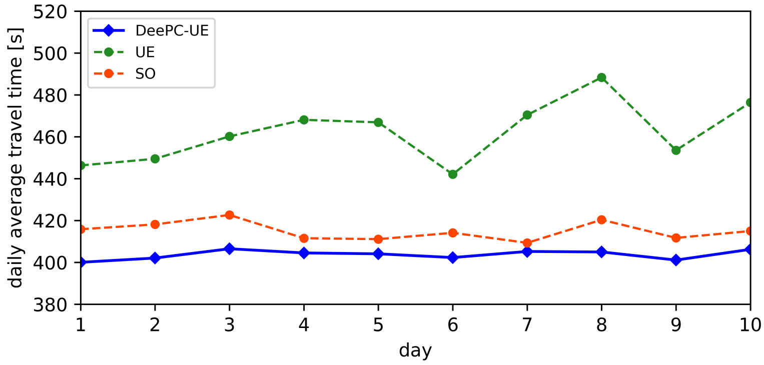


(b) Average travel time

Figure 5: Performance of centralized DeePC on cropped SF network with twisted demand.



(a) Number of running vehicles



(b) Daily average travel time

Figure 6: Performance of distributed DeePC on full SF network.

4 Conclusions

This study investigated the potential of DeePC, a data-driven model predictive control approach, for dynamic traffic routing with a vision that all vehicles are fully controllable autonomous vehicles (AVs). Our simulation experiments on small- and medium-sized networks demonstrate promising performances of DeePC, especially with a better-designed data sampling strategy. To address the scalability, we reformulated the step-wise control problem in DeePC as a distribution optimization problem and developed a distributed solution algorithm. The distributed DeePC also shows satisfactory control performance as well as computational efficiency. Besides improving network throughput and reducing traffic congestion, DeePC is also capable of maintaining the unfairness at a reasonable level and shows high robustness towards inaccurate demand predictions. These properties further strengthen the significance of DeePC in real practice.
This study opens several directions for future study. First, the original DeePC was developed for linear systems, which is clearly not the case for traffic networks. Hence, it is worthwhile to explore the recent extensions of DeePC for non-linear systems. Additionally, the distributed DeePC may still not be sufficient to solve the route control problem in larger networks. A possible direction is to develop a hierarchical control scheme with regional coordination. Last but not least, considering the coexistence of human-driven vehicles is essential, given that the mixed traffic is likely to last for a while before the era of full autonomy (Di & Shi, 2021).

References

Bauschke, H. H., & Combettes, P. L. (2011). Convex analysis and monotone operator theory in hilbert spaces. *CMS books in mathematics). DOI*, *10*, 978–1.

Chen, Z., Lin, X., Yin, Y., & Li, M. (2020). Path controlling of automated vehicles for system optimum on transportation networks with heterogeneous traffic stream. *Transportation Research Part C: Emerging Technologies*, *110*, 312–329.

Coulson, J., Lygeros, J., & Dörfler, F. (2019). Data-enabled predictive control: In the shallows of the deepc. In *2019 18th european control conference (ecc)* (pp. 307–312).

Di, X., & Shi, R. (2021). A survey on autonomous vehicle control in the era of mixed-autonomy: From physics-based to ai-guided driving policy learning. *Transportation research part C: emerging technologies*, *125*, 103008.

Elokda, E., Coulson, J., Beuchat, P. N., Lygeros, J., & Dörfler, F. (2021). Data-enabled predictive control for quadcopters. *International Journal of Robust and Nonlinear Control*, *31*(18), 8916–8936.

Gong, S., & Du, L. (2018). Cooperative platoon control for a mixed traffic flow including human drive vehicles and connected and autonomous vehicles. *Transportation research part B: methodological*, *116*, 25–61.

Guo, Z., Wang, D. Z., & Wang, D. (2022). Managing mixed traffic with autonomous vehicles–a day-to-day routing allocation scheme. *Transportation research part C: emerging technologies*, *140*, 103726.

Huang, L., Coulson, J., Lygeros, J., & Dörfler, F. (2019). Data-enabled predictive control for grid-connected power converters. In *2019 ieee 58th conference on decision and control (cdc)* (pp. 8130–8135).

Huang, L., Coulson, J., Lygeros, J., & Dörfler, F. (2021). Decentralized data-enabled predictive control for power system oscillation damping. *IEEE Transactions on Control Systems Technology*, *30*(3), 1065–1077.

Huang, L., Zhen, J., Lygeros, J., & Dörfler, F. (2023). Robust data-enabled predictive control: Tractable formulations and performance guarantees. *IEEE Transactions on Automatic Control*, *68*(5), 3163–3170.

INRIX. (2020). *2019 global traffic scorecard.*

Kwon, W. H., & Han, S. H. (2005). *Receding horizon control: model predictive control for state models*. Springer Science & Business Media.

Lazar, D. A., Bıyık, E., Sadigh, D., & Pedarsani, R. (2021). Learning how to dynamically route autonomous vehicles on shared roads. *Transportation research part C: emerging technologies*, *130*, 103258.

Li, J., Yu, C., Shen, Z., Su, Z., & Ma, W. (2023). A survey on urban traffic control under mixed traffic environment with connected automated vehicles. *Transportation Research Part C: Emerging Technologies*, *154*, 104258.

Lu, G., Nie, Y. M., Liu, X., & Li, D. (2019). Trajectory-based traffic management inside an autonomous vehicle zone. *Transportation Research Part B: Methodological*, *120*, 76–98.

Malikopoulos, A. A., Hong, S., Park, B. B., Lee, J., & Ryu, S. (2018). Optimal control for speed harmonization of automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, *20*(7), 2405–2417.

Markovsky, I., & Dörfler, F. (2022). Identifiability in the behavioral setting. *IEEE Transactions on Automatic Control*, *68*(3), 1667–1677.

Rawlings, J. B. (2000). Tutorial overview of model predictive control. *IEEE control systems magazine*, *20*(3), 38–52.

Sheffi, Y. (1985). *Urban transportation networks* (Vol. 6). Prentice-Hall, Englewood Cliffs, NJ.

Toghi, B., Valiente, R., Sadigh, D., Pedarsani, R., & Fallah, Y. P. (2022). Social coordination and altruism in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, *23*(12), 24791–24804.

Wang, J., Zheng, Y., Li, K., & Xu, Q. (2023). Deep-lcc: Data-enabled predictive leading cruise control in mixed traffic flow. *IEEE Transactions on Control Systems Technology*.

Willems, J. C., Rapisarda, P., Markovsky, I., & De Moor, B. L. (2005). A note on persistency of excitation. *Systems & Control Letters*, *54*(4), 325–329.

Zhang, K., & Nie, Y. M. (2018). Mitigating the impact of selfish routing: An optimal-ratio control scheme (orcs) inspired by autonomous driving. *Transportation Research Part C: Emerging Technologies*, *87*, 75–90.

Zhu, P., Ferrari-Trecate, G., & Geroliminis, N. (2023). Data-enabled predictive control for empty vehicle rebalancing. In *2023 european control conference (ecc)* (pp. 1–6).