DYNAMIC NETWORK CAPACITY ALLOCATION VIA SPARSE IDENTIFICATION OF NONLINEAR DYNAMICS WITH MODEL PREDICTIVE CONTROL

Qing-Long Lu, Corresponding Author
Chair of Transportation Systems Engineering, Technical University of Munich, Germany
Email: qinglong.lu@tum.de

Raphael Stern, Ph.D.
Department of Civil, Environmental, and Geo- Engineering, University of Minnesota, USA
Email: rstern@umn.edu

Mohammad Sadrani
Chair of Transportation Systems Engineering, Technical University of Munich, Germany
Email: m.sadrani@tum.de

Constantinos Antoniou, Ph.D.
Chair of Transportation Systems Engineering, Technical University of Munich, Germany
Email: c.antoniou@tum.de

Word Count: 7130 words + 1 table(s) × 250 = 7380 words

Submission Date: August 1, 2022

ABSTRACT

Demand variations throughout the day and differences in area popularity across the city result in spatiotemporal changes in traffic flow. One of the well-known phenomena arising from these changes is tidal traffic, characterized by an imbalance between inbound and outbound traffic on a given road. This issue reflects the fluctuation in the alignment between transportation system supply and demand. Lane reversal control has been a common supply-side measure for alleviating this urban traffic "sickness" by adapting road capacity allocation to the demand imbalance between two directions of a road. This study investigates the dynamic network capacity allocation control problem in the era of connected and autonomous vehicles (CAVs), which integrates dynamic traffic signal splits and lane reversal controls. Considering the high dimensionality and non-linearity of urban transportation systems, we apply the sparse identification of nonlinear dynamics (SINDy) technique to construct a sparse yet sufficiently accurate surrogate model. This model estimates the forthcoming network traffic state based on the current state and implemented control decisions. The surrogate model is then integrated into a model predictive control (MPC) framework, forming a SINDy-MPC framework, to assist in optimal network capacity allocation decision-making in real time. The experiments show that the system identified by SINDy exhibits stability in the presence of Gaussian noise disturbances. The proposed dynamic network allocation control scheme can effectively reduce traffic imbalance, improve traffic efficiency, and enhance traffic resilience against cyberattacks.

Keywords: Network capacity allocation, Connected and autonomous vehicle, Model predictive control, Sparse model identification, Traffic resilience, Cyberattack

INTRODUCTION

Daily commuting behaviors are a significant factor that contributes to the tidal traffic phenomenon in urban areas Fu et al. (*1*). The formation of tidal traffic exacerbates congestion to a great extent, especially during peak hours. To mitigate this impact, many cities have implemented reversible lanes on busy roads Wollenstein-Betech et al. (*2*). For instance, Park Avenue in Montreal, Canada, a five-lane road, allocates three lanes toward the city center and two lanes outward during the morning peak hours, and reverses in the evening. Reversible lanes serve as a capacity reallocation method between opposite directions on two-way roads Wollenstein-Betech et al. (*2*). Similarly, traffic signal control reallocates the capacity of intersection legs, with the effective capacity (also known as saturation flow) of these segments being determined by both design capacity and the signal plan. These strategies are commonly referred to as lane allocation (or lane reversal control) and traffic signal control, respectively, and they have been extensively studied in various contexts, such as during peak hours and under mixed traffic flow.

Intuitively, combining these measures and extending control to the entire network can achieve full control and management of network capacity. While both static and dynamic traffic signal control have been widely investigated, lane reversal control is mostly modeled as a static, temporary measure for handling severe traffic imbalances Ampountolas et al., Chen et al. (*3, 4*). This is because, nowadays, the implementation of lane reversal control for human drivers is costly Levin and Boyles (*5*), attributed to the requirements for implementation time, personnel and materials as well as the existing infrastructure design. More importantly, frequent lane direction changes inevitably cause confusion among human drivers Chen et al. (*4*).

This study is focused on the integration and coordination of lane reversal control and traffic signal control. We name this kind of measure as network capacity allocation, and dynamic network capacity allocation refers to the dynamic implementation of the measures involved. Clearly, this approach offers numerous benefits. Firstly, within-day demand variations often lead to significant flow imbalances between directions on two-way roads. An optimal dynamic capacity allocation strategy thus can be sought so that the network capacity aligns with varying demand. Secondly, transportation disruptions due to special events, such as road maintenance and marathons, can result in substantial societal and economic losses. They are also often related to changes in the supply and/or demand of the transportation system. Dynamic network capacity allocation can effectively mitigate these non-equilibrium states, thereby facilitating traffic efficiency. Moreover, dynamically adjusting the allocation strategy according to traffic conditions may be more cost-efficient than widening roads or extending the network in addressing traffic congestion issue Ampountolas et al. (*3*).

However, implementing dynamic network capacity allocation is challenging nowadays due to the involvement of human drivers. Here, we focus on a scenario where all vehicles are connected and autonomous (CAVs) following previous studies on dynamic lane allocation, such as (*5*), (*6*), and (*4*). Vehicles keep communicating with roadside units (RSUs) and the traffic control center (TCC) in regard to location information and network configuration. In this scenario, vehicles can react immediately to changes in the network capacity allocation strategy Chu et al. (*6*). Traffic signals, although possibly invisible, would exist as packages of messages informing right-of-way determinations. It is important to note that only roads showing unbalanced traffic in two directions will be included in the dynamic control system. This method can address the problem of unbalanced traffic and reduce the control system's dimension, allowing real-time implementation with limited computational resources.

Given the high dimensionality and non-linearity characteristics of transportation systems Qurashi et al. (*7*), implementing dynamic network capacity allocation control is particularly challenging, especially regarding optimal control decisions for a specific system state. However, many systems actually evolve on a low-dimensional attractor characterized by large-scale coherent structures Brunton et al. (*8*). Consequently, we can apply dimension reduction techniques to identify active features from traffic measurements and control decisions to construct a simpler dynamic system. Such a system facilitates fast solutions, enabling real-time network capacity allocation control.

This study aims to formulate the dynamic network capacity allocation problem, develop a low-dimensional dynamical system to describe the relationship between traffic state and capacity allocation control decisions, and evaluate the resilience of the developed control system. Specifically, we employ the sparse identification of nonlinear dynamics (SINDy) technique Brunton et al. (*8*) to construct a low-dimensional yet sufficiently accurate system from traffic measurements and capacity allocation control decisions. This system is then integrated into a model predictive control (MPC) framework to facilitate real-time capacity allocation control. Additionally, considering that CAVs are vulnerable to cyberattacks due to their reliance on electronics and software, we also analyze the resilience of the proposed dynamic capacity allocation control to such attacks.

The remainder of this paper is structured as follows. Section 3 formulates the dynamic network capacity allocation problem. Section 4 introduces the MPC framework, SINDy algorithm, and the development of resilient dynamic network capacity allocation problem. Then, we analyze the experiments of a case study in Section 5. Finally, conclusions are summarized in Section 6.

PROBLEM STATEMENT

In this study, we assume that the configuration of a transportation network can be dynamically altered to adapt to within-day traffic flow variations, particularly in the era of CAVs facilitated by vehicle-to-everything (V2X) technology. By "network configuration," we refer to the allocation of network capacity, which is jointly regulated through lane allocation and traffic signal planning. For example, denote the design capacity of a lane i as q_i^{cap}, then its effective capacity can be estimated by

$$q_i^{\text{eff}} = \frac{q_i^{\text{cap}} t_i^g}{T} \tag{1}$$

where t_i^g is the total green time assigned to the phases when lane i has the right of way, and T is the signal cycle. The capacity of a link l is then determined by its number of lanes and traffic signal plan:

$$q_l = \sum_{i=1}^{n_l} q_i^{\text{eff}} \tag{2}$$

where n_l indicates the number of lanes in link l. Clearly, the effective capacity of a link is controlled by its number of lanes and the related traffic splits. This is the basic idea behind the capacity allocation at the link level. Extending this concept to encompass the entire transportation network, we arrive at the network capacity allocation problem investigated in this study.

For dynamic or real-time network capacity allocation, this study envisions a future scenario where all vehicles are connected and autonomous and communicate with infrastructure and the traffic control center (TCC) in real-time via V2X technologies. Specifically, the real-time communication includes four key aspects, as shown in Figure 1. (1) Vehicles receive updated network configuration strategies for the upcoming time interval and adjust their driving behaviors

accordingly. (2) Vehicles within a neighborhood communicate with each other regarding their adjustments to prevent collisions. (3) Vehicles proactively send their location information to nearby sensors or are passively detected by sensors. (4) The sensor network updates the overall traffic state of the network and sends traffic measurements to the TCC for decision-making on network configuration. These communication means are also depicted in Figure 1 for reference.

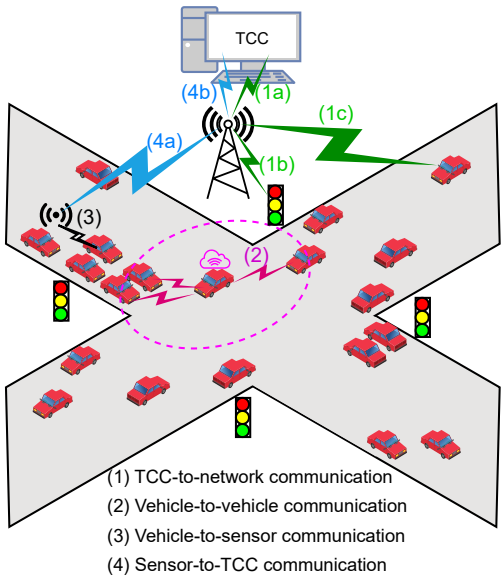


FIGURE 1: Architecture of an example vehicular sensor network.

Consider a transportation network defined as $G(V,A)$ with V and A denoting sets of nodes and links, respectively. Let $\mathbf{u}_t$ denote the network capacity allocation strategy in time interval t, a vector of decisions including all controllable links and traffic signals. Let D_t and $\mathbf{y}_t$ denote travel demand and traffic measurements within t. Then, we can characterize the dynamic capacity allocation problem as below.

Problem Statement (Dynamic network capacity allocation)**.** *Given a transportation network $G(V,A)$, the problem is to determine a set of lane allocation and traffic signal plans $\mathbf{u}_1, \ldots, \mathbf{u}_t$ that can optimize a pre-defined objective function $\boldsymbol{g}$ subject to dynamic travel demand $D_1, \ldots, D_t$. Mathematically, the problem can be expressed as*

$$\mathbf{u}_1^*, \ldots, \mathbf{u}_t^* = \underset{\mathbf{u}_1,\ldots,\mathbf{u}_t}{\operatorname{argmin}} \sum_{k=1}^{t} \boldsymbol{g}(\mathbf{y}_k | G, D_k, \mathbf{u}_k, \mathbf{y}_{k-1}) \tag{3}$$

$\boldsymbol{g}(\mathbf{y}_k | G, D_k, \mathbf{u}_k, \mathbf{y}_{k-1})$ indicates that the traffic state of the current time step $\mathbf{y}_k$ is dependent on the travel demand D_k, allocation decision $\mathbf{u}_k$, and the traffic state of the previous time step $\mathbf{y}_{k-1}$. The inputs of the problem include a given transportation network, dynamic travel demand, and traffic measurements for previous time steps, while the outputs will be the capacity allocation strategies.

METHODOLOGY

The dynamic network capacity allocation problem can be structured as a dynamic control problem. Given that transportation systems are complex, nonlinear, and high-dimensional, we propose a hybrid approach incorporating both data-driven and model-based techniques to address this challenge. Specifically, we integrate the Sparse Identification of Nonlinear Dynamics (SINDy) technique into a Model Predictive Control (MPC) framework. This integration allows us to identify a low-dimensional representation of the system and achieve dynamic control of network capacity.

Figure 2 illustrates the integrated SINDy-MPC framework for traffic control problems. Initially, SINDy is applied to model the relationship between the current traffic state, control signals, and the traffic state in the subsequent timestep. Subsequently, this identified system is embedded into the MPC framework to optimize control decisions for the next timestep. It is important to note that although the framework separates these two components, the model and the control law can be learned simultaneously. One can develop a streaming algorithm to continually refine the model with new observations, iterating between model identification and control optimization to achieve convergence and reliable results Kaiser et al. (*9*). Detailed descriptions of these two components are provided in Section 4.1 and Section 4.2, respectively.

This approach leverages the strengths of both data-driven and model-based methods, ensuring a robust and adaptive solution for the dynamic network capacity allocation problem.

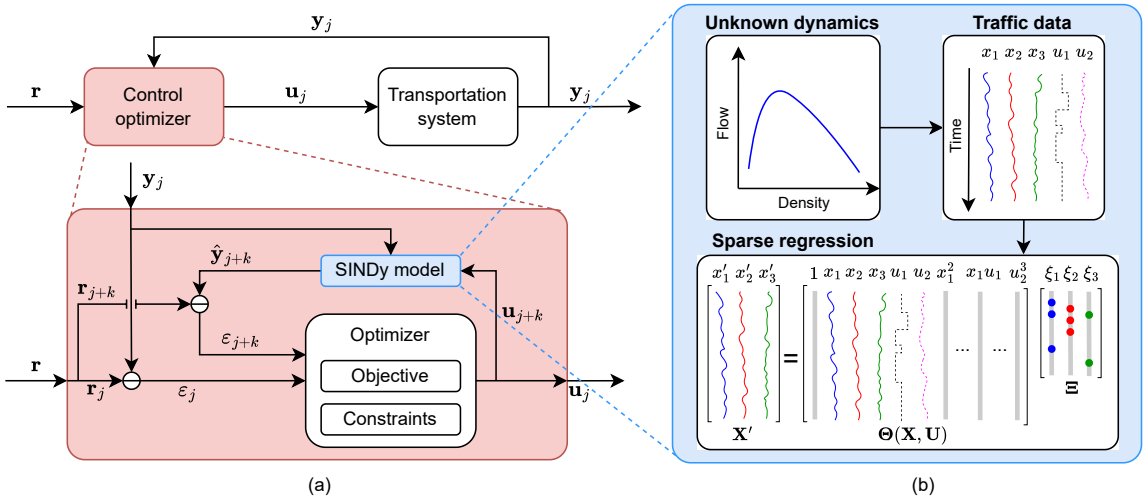


FIGURE 2: The SINDy-MPC framework for traffic control problems. (a) MPC framework; (b) SINDy.

Model predictive control

MPC is a flexible framework for controlling nonlinear systems with constraints, time delays, instability, and uncertainty. For high-dimensional complex systems, one can integrate various types of surrogate models into the framework to realize dynamic or even real-time control. This fits with the characteristics of our problem.

In MPC, the open-loop actuation input $\mathbf{u}$ is optimized over a receding horizon $t_c = m_c \Delta t$ to minimize a given cost function J over a prediction horizon $t_p = m_p \Delta t$, where Δt denotes the size

of a timestep. In general, the prediction horizon cannot be shorter than the receding horizon, i.e., $m_p \geq m_c$, and the input $\mathbf{u}$ is assumed to be constant from t_c to t_p. Given the system state $\mathbf{x}_j$, MPC optimizes a control sequence $\mathbf{u}(\cdot|\mathbf{x}_j) := \{\mathbf{u}_{j+1}, \cdots, \mathbf{u}_{j+k}, \cdots, \mathbf{u}_{j+m_c}\}$ to minimize the following objective:

$$J_{\mathbf{x}_j} = \sum_{k=0}^{m_p-1} \|\hat{\mathbf{x}}_{j+k} - \mathbf{r}_{j+k}\|^2_{\mathbf{Q}} + \sum_{k=1}^{m_c-1} (\|\mathbf{u}_{j+k}\|^2_{\mathbf{R}_u} + \|\Delta\mathbf{u}_{j+k}\|^2_{\mathbf{R}_{\Delta\mathbf{u}}}) \tag{4}$$

where $\mathbf{r}$ represents the reference trajectory to be tracked, $\hat{\mathbf{x}}$ indicates the predicted system state, $\|\mathbf{x}\|^2_{\mathbf{Q}} = \mathbf{x}^\top \mathbf{Q}\mathbf{x}$, and $\Delta\mathbf{u}_j = \mathbf{u}_j - \mathbf{u}_{j-1}$. $\mathbf{Q} \succeq 0$ (positive semi-definite), $\mathbf{R}_u \succ 0$ (positive definite) and $\mathbf{R}_{\Delta u} \succ 0$ (positive definite) are the weight matrices for the penalty terms of system state, actuation, and actuation difference, respectively. Note that we do not include a terminal cost in this formulation.

The following constraints are incorporated:

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{f}}(\hat{\mathbf{x}}_k, \mathbf{u}_k) \tag{5}$$

$$\mathbf{u}_{\min} \leq \mathbf{u}_k \leq \mathbf{u}_{\max} \tag{6}$$

$$\Delta\mathbf{u}_{\min} \leq \Delta\mathbf{u}_k \leq \Delta\mathbf{u}_{\max} \tag{7}$$

where $\hat{\mathbf{f}}(\cdot)$ represents a surrogate model describing system dynamics. Equations (6) and (7) represent the bounds on the actuation and its difference between two successive timesteps, respectively.

The optimal control is then applied for one time-step, i.e., only $\mathbf{u}_{j+1}$ is applied. The procedure is reinitialized and repeated at each subsequent timestep. This results in an implicit feedback control raw

$$\mathbf{K}(\mathbf{x}_j) = \mathbf{u}(j+1|\mathbf{x}_j) = \mathbf{u}_{j+1} \tag{8}$$

This equation reflects that the control action will be the first term in the optimized actuation sequence starting at the initial condition $\mathbf{x}_j$.

Sparse identification of nonlinear dynamics with control

Sparse identification of nonlinear dynamics (SINDy) Brunton et al. (*8*) was developed to build up parsimonious models for identifying nonlinear systems with limited measurement data. This technique focuses on discovering the underlying dynamics of a system by finding the simplest model that can accurately describe the observed data. Later on, SINDy with control Kaiser et al. (*9*) was proposed, extending the original SINDy framework to include control inputs. This advancement provides a way to evaluate the effects of control actuation inputs on the state of the identified system. Consider a nonlinear dynamical discretized system of the form:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \tag{9}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ represents the system state or measurements, $\mathbf{u}_k \in \mathbb{R}^q$ represents control inputs, and $\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) : \mathbb{R}^n \times \mathbb{R}^q \mapsto \mathbb{R}^n$ describes system dynamics.

The SINDy with control algorithm leverages sparse regression techniques to identify significant terms from a library of candidate linear and nonlinear model terms in the state $\mathbf{x}$ and control input $\mathbf{u}$. Function $\mathbf{f}$ is then developed to approximate the complex system of interest. The implementation of the SINDy with control algorithm consists of four steps, including data collection, construction of the library of candidate terms, sparse regression implementation, and system identification. In particular, SINDy with control implements the following steps:

Data collection. As a data-driven model identification approach, the successful implementation of the SINDy algorithm requires sufficient snapshots of the system state $\mathbf{x}$ and control inputs $\mathbf{u}$. In our case, we can construct a dataset consisting of various capacity allocation strategies and

their resulting traffic measurements. Previous applications illustrated that SINDy also performs well with limited data. If m snapshots were measured, then we have the following two matrices:

$\mathbf{X} = [\mathbf{x}_1 \quad \mathbf{x}_2 \cdots \mathbf{x}_m]^\top$ and $\mathbf{U} = [\mathbf{u}_1 \quad \mathbf{u}_2 \cdots \mathbf{u}_m]^\top$

Furthermore, we need to construct a matrix $\mathbf{X}'$ with the columns of $\mathbf{X}$ advanced one timestep, which represents the system output, subject to the current state and the control about to be effective.

$\mathbf{X}' = [\mathbf{x}_2 \quad \mathbf{x}_3 \cdots \mathbf{x}_{m+1}]^\top$

Library of candidate terms. Construct a set of candidate terms (either linear or nonlinear functions) that may be able to describe the complex system dynamics of interest according to the domain knowledge or empirical analysis. We provide the following set of terms as an example.

$$\mathbf{\Theta}(\mathbf{X},\mathbf{U}) = [\mathbf{1} \quad \mathbf{X} \quad \mathbf{U} \quad \mathbf{X}\otimes\mathbf{X} \quad \mathbf{X}\otimes\mathbf{U} \cdots \sin(\mathbf{X}) \quad \sin(\mathbf{U}) \quad \sin(\mathbf{X})\otimes\sin(\mathbf{U}) \cdots] \tag{10}$$

where $\mathbf{X}\otimes\mathbf{U}$ represents the matrix consisting of all product combinations of the components in $\mathbf{X}$ and $\mathbf{U}$. That is,

$$\mathbf{X}\otimes\mathbf{U} = [\mathbf{X}_1\circ\mathbf{U}_1 \quad \mathbf{X}_1\circ\mathbf{U}_2 \quad \ldots \mathbf{X}_n\circ\mathbf{U}_q] \tag{11}$$

where $\mathbf{X}_k (k = 1,\ldots,n)$ and $\mathbf{U}_k (k = 1,\ldots,q)$ are the k-th columns of $\mathbf{X}$ and $\mathbf{U}$, respectively, and $\circ$ represents the Hadamard product. The dimension of $\mathbf{X}\otimes\mathbf{U}$ would be $m \times nq$.

Sparse regression. We expect to include a large number of functions in the library given by Equation (10) such that the sparse regression described in the next step can identify the system dynamics accurately. Assuming that $\mathbf{\Theta}(\mathbf{X},\mathbf{U})$ contains all functions necessary for describing the system, then the system can be written as

$$\mathbf{X}' = \mathbf{\Theta}(\mathbf{X},\mathbf{U})\mathbf{\Xi} + \mathbf{E} \tag{12}$$

where $\mathbf{\Xi}$ indicates the coefficient matrix relating the terms in $\mathbf{\Theta}(\mathbf{X},\mathbf{U})$ to the elements in $\mathbf{X}'$.

However, many dynamical systems are often governed by relatively few terms Fasel et al. (*10*). In addition, it is beneficial to reduce the complexity of the identified system for the sake of accelerating the solution of the optimization problem to be addressed in the control framework. This can also enable the real-time control of complex systems. Therefore, the least squares regularized with an l^1 norm can be implemented to find a sparse representation of the system.

$$\hat{\boldsymbol{\xi}}_k = \underset{\boldsymbol{\xi}_k}{\operatorname{argmin}} \frac{1}{2m}\|\mathbf{X}'_k - \mathbf{\Theta}(\mathbf{X},\mathbf{U})\boldsymbol{\xi}_k\|_2^2 + \alpha\|\boldsymbol{\xi}_k\|_1 \tag{13}$$

where $\mathbf{X}'_k$ and $\boldsymbol{\xi}_k$ are the k-th columns of $\mathbf{X}'$ and $\mathbf{\Xi}$, respectively, α is a hyperparameter determining the regularization strength. $\|\cdot\|_1$ and $\|\cdot\|_2$ indicate the l^1 norm and l^2 norm, respectively.

Identified system. $\hat{\boldsymbol{\xi}}_k$ is a vector of coefficients determining the active terms in the k-th row in Equation (10). In other words, it indicates the terms that are significant in describing the dynamics of the k-th state measurement. By solving the problem expressed by Equation (13), we obtain the identified system

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{f}}(\mathbf{x}_k,\mathbf{u}_k) = \mathbf{\Theta}(\mathbf{x}_k,\mathbf{u}_k)\hat{\mathbf{\Xi}} \tag{14}$$

where $\mathbf{\Theta}(\mathbf{x_k},\mathbf{u}_k)$ represents applying the same transformation to $\mathbf{x}_k$ and $\mathbf{u}_k$ as $\mathbf{\Theta}(\mathbf{X},\mathbf{U})$, and $\hat{\mathbf{\Xi}}$ denotes the estimated coefficient matrix.

The identified sparse model can then be integrated into the MPC framework and works as a surrogate of the complex system for evaluating different control strategies.

Resilient dynamic network capacity allocation problem

In Section 3, we have formulated the dynamic network capacity allocation problem as Equation (3). Considering the increasing frequency of transportation disruptions due to natural disasters (e.g., flooding) and their severe aftermath, we aim to develop a dynamic network capacity allocation

problem targeting improving urban traffic resilience by leveraging the SINDy-MPC framework in this section. To this end, we need to integrate the evaluation of traffic resilience into the objective function of the problem, i.e., $\boldsymbol{g}(\cdot)$ in Equation (3) and $J_{\mathbf{x}_j}$ in Equation (4), and adapt the MPC framework to the network capacity allocation problem.

Traffic resilience objective development

In this study, we want to examine the efficacy of dynamic network capacity allocation in improving urban traffic resilience in the era of fully autonomous vehicles. Traffic resilience has been defined as the "ability of an urban road transportation system to prepare for different kinds of disruptions, effectively serve vehicles, and recover rapidly to its optimal serving rate" Lu et al. (*11*). Also, traffic resilience loss due to congestion can be evaluated by the deviation of the trip completion rate from its optimal value, which is given by

$$\delta_k^d = D_c - D_k \tag{15}$$

where D_k is the trip completion rate at timestep k, and D_c is its critical (optimal) value. The problem is to minimize the traffic resilience loss over the time period under investigation. Therefore, in this case, the objective function can be expressed as

$$J = \sum_{k=1}^{m_p} \|D_k - D_c\|^2 \tag{16}$$

Given the linear relationship between the trip completion rate and weighted space-mean flow uncovered by the MFD curve, i.e., $D(k) = \gamma \bar{q}_k$, the objective function can be transformed to

$$J = \sum_{k=1}^{m_p} \|\bar{q}_k - q_c\|^2 \tag{17}$$

Since $\bar{q}_k = \sum_{i\in\mathbb{L}} b_i q_{i,k} / \sum_{i\in\mathbb{L}} b_i$ where $\mathbb{L}$ is the set of links being detected, b_i is the length of link i, we can further transform the objective to

$$J = \sum_{k=1}^{m_p} \left\| \mathbf{b}^\top \mathbf{q}_k - \mathbf{b}^\top \mathbf{q}_c \right\|^2 \tag{18}$$

where, for simplicity, we construct a vector $\mathbf{q}_c$ which has the same dimension as $\mathbf{q}_k$ and all elements are q_c. Moreover, for generality, we would like to insert a weight matrix into the objective function to incorporate the consideration of heterogeneous reliability of the state measurements. Mathematically, this can be achieved by

$$J = \sum_{k=1}^{m_p} \left\| \mathbf{b}^\top \mathbf{W} \mathbf{q}_k - \mathbf{b}^\top \mathbf{W} \mathbf{q}_c \right\|^2 \tag{19}$$

$$= \sum_{k=1}^{m_p} (\mathbf{q}_k - \mathbf{q}_c)^\top \mathbf{W}^\top \mathbf{b}\mathbf{b}^\top \mathbf{W} (\mathbf{q}_k - \mathbf{q}_c) \tag{20}$$

$$= \sum_{k=1}^{m_p} \|\mathbf{q}_k - \mathbf{q}_c\|^2_{\mathbf{W}^\top \mathbf{b}\mathbf{b}^\top \mathbf{W}} \tag{21}$$

Ultimately, the resilience evaluation objective also has the same format as the first summation term in Equation (4). Considering that implementing and changing capacity allocation control strategies leads to no additional expense in the context under consideration, the second summation term in Equation (4) would not be evaluated in the dynamic capacity allocation control framework.

Problem formulation

One of the critical factors affecting the effectiveness of the SINDy model as a surrogate lies in the construction of the library of candidate terms to be included in the sparse regression, i.e., the second step of SINDy. Equation (21) implies that the system state is represented by the traffic flows. According to the urban traffic network modeling approach proposed in (*12*), the traffic flow dynamics of a link z at a junction j_1 can be estimated by

$$x_z(t+1) = x_z(t) + \Delta t \left[\sum_{w \in I_{j_1}} \tau_{w,z} \sum_{i \in V_w} u_{j_1,i}(t) \frac{S_w}{T} - \sum_{i \in V_z} u_{j_2,i}(t) \frac{S_z}{T} + e_z(t) \right] \tag{22}$$

where I_j denotes the set of incoming links of junction j, $\tau_{w,z}$ is the turning rate giving the rate of vehicles that reach junction j_1 from an incoming link $z \in I_{j_1}$ and turn into the link of interest z, $u_{j_1,i}$ denotes the green time of phase i, V_w denotes the set of phases that link w has the right of way, S_w represents the saturation flow of link w, and $e_z(t)$ represents the disturbance in demand and exit rates. Simply speaking, the first term within the square brackets calculates the traffic demand from other links to the link of interest, while the second term calculates the exit flow from the link of interest to its connecting links. In both terms, multiplications of control signals u and saturation flows S are present. Inspired by this and given that saturation flow often cannot be met, we mainly incorporate the term $\mathbf{X} \otimes \mathbf{U}$ in the library, that is

$$\boldsymbol{\Theta}(\mathbf{X}, \mathbf{U}) = [\mathbf{1} \quad \mathbf{X} \quad \mathbf{U} \quad \mathbf{X} \otimes \mathbf{U}] \tag{23}$$

To make the notations consistent, we replace the $\mathbf{q}$ in Equation (21) with $\mathbf{x}$, and define $\mathbf{Q} = \mathbf{W}^\top \mathbf{b}\mathbf{b}^\top \mathbf{W}$. Then, we reach the following formulation of the dynamic network capacity allocation problem with SINDy-MPC:

$$\min_{\mathbf{u}(\cdot|\mathbf{x}_j)} J_{\mathbf{x}_j} = \sum_{k=1}^{m_p} \|\hat{\mathbf{x}}_{j+k} - \mathbf{x}_c\|_{\mathbf{Q}}^2 \tag{24}$$

$$\text{s.t.} \quad \hat{\mathbf{x}}_{k+1}^\top = [1 \quad \hat{\mathbf{x}}_k^\top \quad \mathbf{u}_k^\top \quad (\hat{\mathbf{x}}_k \otimes \mathbf{u}_k)^\top] \begin{bmatrix} — & \hat{\boldsymbol{\xi}}_1^\top & — \\ — & \hat{\boldsymbol{\xi}}_{\mathrm{x}}^\top & — \\ — & \hat{\boldsymbol{\xi}}_{\mathrm{u}}^\top & — \\ — & \hat{\boldsymbol{\xi}}_{\mathrm{xu}}^\top & — \end{bmatrix} \tag{25}$$

$$\mathbf{u}_{\min} \leq \mathbf{u}_k \leq \mathbf{u}_{\max} \tag{26}$$

where $\mathbf{x}_c$ indicates the vector of the given reference value for the critical weighted space-mean flow, $\hat{\boldsymbol{\xi}}_1$, $\hat{\boldsymbol{\xi}}_{\mathrm{x}}$, $\hat{\boldsymbol{\xi}}_{\mathrm{u}}$ and $\hat{\boldsymbol{\xi}}_{\mathrm{xu}}$ denote the estimated coefficients for the corresponding terms indicated by their subscripts.

Equation (25) can be transformed to

$$\hat{\mathbf{x}}_{k+1}^\top = \hat{\boldsymbol{\xi}}_1^\top + \hat{\mathbf{x}}_k^\top \hat{\boldsymbol{\xi}}_{\mathrm{x}}^\top + \mathbf{u}_k^\top \hat{\boldsymbol{\xi}}_{\mathrm{u}}^\top + (\hat{\mathbf{x}}_k \otimes \mathbf{u}_k)^\top \hat{\boldsymbol{\xi}}_{\mathrm{xu}}^\top \tag{27}$$

which can be further simplified as

$$\hat{\mathbf{x}}_{k+1} = \hat{\boldsymbol{\xi}}_1 + \hat{\boldsymbol{\xi}}_{\mathrm{x}} \hat{\mathbf{x}}_k + \hat{\boldsymbol{\xi}}_{\mathrm{u}} \mathbf{u}_k + \hat{\boldsymbol{\xi}}_{\mathrm{xu}} (\hat{\mathbf{x}}_k \otimes \mathbf{u}_k) \tag{28}$$

The control inputs $\mathbf{u}$ in the capacity allocation problem, as explained in Section 3, include lane allocation controls and traffic signal splits, which are discrete and continuous variables, respectively. Therefore, we can decompose $\mathbf{u}$ into two parts based on the nature of controls, resulting in a clearer presentation of the problem. Similarly, $\hat{\boldsymbol{\xi}}_{\mathrm{u}}$ is decomposed into $\hat{\boldsymbol{\xi}}_{\mathrm{c}}$ and $\hat{\boldsymbol{\xi}}_{\mathrm{d}}$ correspond-

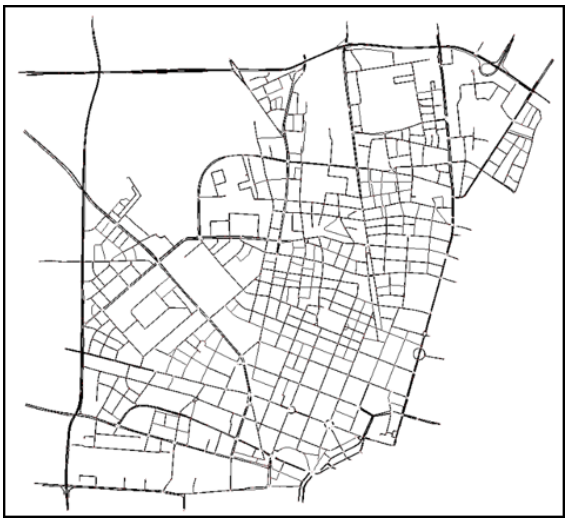


FIGURE 3: Network of the study area.

ingly, and $\hat{\boldsymbol{\xi}}_{\text{xu}}$ is also decomposed into $\hat{\boldsymbol{\xi}}_{\text{xc}}$ and $\hat{\boldsymbol{\xi}}_{\text{xd}}$. The problem can then be expressed as

$$\min_{\mathbf{u}(\cdot|\mathbf{x}_j)} J_{\mathbf{x}_j} = \sum_{k=1}^{m_p} \|\hat{\mathbf{x}}_{j+k} - \mathbf{x}_c\|^2_{\mathbf{Q}} \tag{29}$$

$$\text{s.t.} \quad \hat{\mathbf{x}}_{k+1} = \hat{\boldsymbol{\xi}}_1 + \hat{\boldsymbol{\xi}}_{\text{x}}\hat{\mathbf{x}}_k + \hat{\boldsymbol{\xi}}_{\text{c}}\mathbf{c}_k + \hat{\boldsymbol{\xi}}_{\text{xc}}(\hat{\mathbf{x}}_k \otimes \mathbf{c}_k) + \hat{\boldsymbol{\xi}}_{\text{d}}\mathbf{d}_k + \hat{\boldsymbol{\xi}}_{\text{xd}}(\hat{\mathbf{x}}_k \otimes \mathbf{d}_k) \tag{30}$$

$$\mathbf{c}_{\min} \leq \mathbf{c}_k \leq \mathbf{c}_{\max} \tag{31}$$

$$\mathbf{d}_{\min} \leq \mathbf{d}_k \leq \mathbf{d}_{\max} \tag{32}$$

We can see that this is a nonlinear mixed-integer problem due to the presence of $\hat{\mathbf{x}}_k \otimes \mathbf{c}_k$ and $\hat{\mathbf{x}}_k \otimes \mathbf{d}_k$. While nonlinear models can improve the accuracy and reliability of MPC outputs, they also pose a challenge to real-time control due to the increase in computational complexity. Fortunately, the development of computing power and advanced algorithms are increasingly removing the stumbling stones that have prevented nonlinear MPC from being used in real-time applications Kaiser et al. (*9*).

CASE STUDY AND RESULTS

Experiment setup

We conducted experiments using the busiest residential area (Maxvorstadt and Schwabing, see Figure 3) in the city center of Munich, Germany. This network covers about 25 km^2 and includes 2,535 links. The travel demand for this network was carefully calibrated with real traffic measurement observations using the simultaneous perturbation stochastic approximation (SPSA) algorithm Spall (*13*) and its variant combining SPSA and principle component analysis (PCA) Qurashi et al. (*7*). The traffic light coordination and adaptation were also optimized accordingly. Readers are referred to (*14*) for more detailed information on the calibration procedure. To examine the effectiveness of the proposed dynamic capacity allocation control approach in improving traffic resilience, we utilized the Simulation of Urban MObility (SUMO) simulator Lopez et al. (*15*). All experiments were conducted at a microscopic resolution, employing a dynamic stochastic user assignment method to approximate dynamic user equilibrium (DUE).

As discussed in Section 3, all vehicles are assumed connected and autonomous. To simulate

the driving behavior of these vehicles, the Intelligent Driver Model (IDM), a well-known car-following model. The IDM was calibrated using driving trajectory data from Toyota Camry cars equipped with an adaptive cruise control (ACC) system. We refer readers to (*16*) and (*17*) for more details about this dataset and the calibration procedure, respectively. IDM determines car-following actions by controlling vehicle acceleration, which is given by

$$a_{\text{IDM}}(s, v, \Delta v) = a\left[1 - \left(\frac{v}{v_0}\right)^{\delta} - \left(\frac{s^*(v, \Delta v)}{s}\right)^2\right] \tag{33}$$

where

$$s^*(v, \Delta v) = s_0 + vT_0 + \frac{v\Delta v}{2\sqrt{ab}} \tag{34}$$

$$\Delta v = v - v^{(\text{p})} \tag{35}$$

where a_{IDM} represents the acceleration to be executed, v and $v^{(\text{p})}$ denote the vehicle velocity at the current and previous time step, respectively, and v_0 is the free-flow velocity. The explanation of the rest notations and the corresponding calibrated parameter values are provided in Table 1.

TABLE 1: Calibrated IDM parameters for autonomous (ACC) vehicles.

Parameter	Calibrated value	Description
T_0	2.2 s	Desired time headway
s_0	6.3 m	Jam distance
a	0.6 m/s^2	Maximum acceleration
b	5.2 m/s^2	Desired deceleration
δ	15.5	Free acceleration exponent

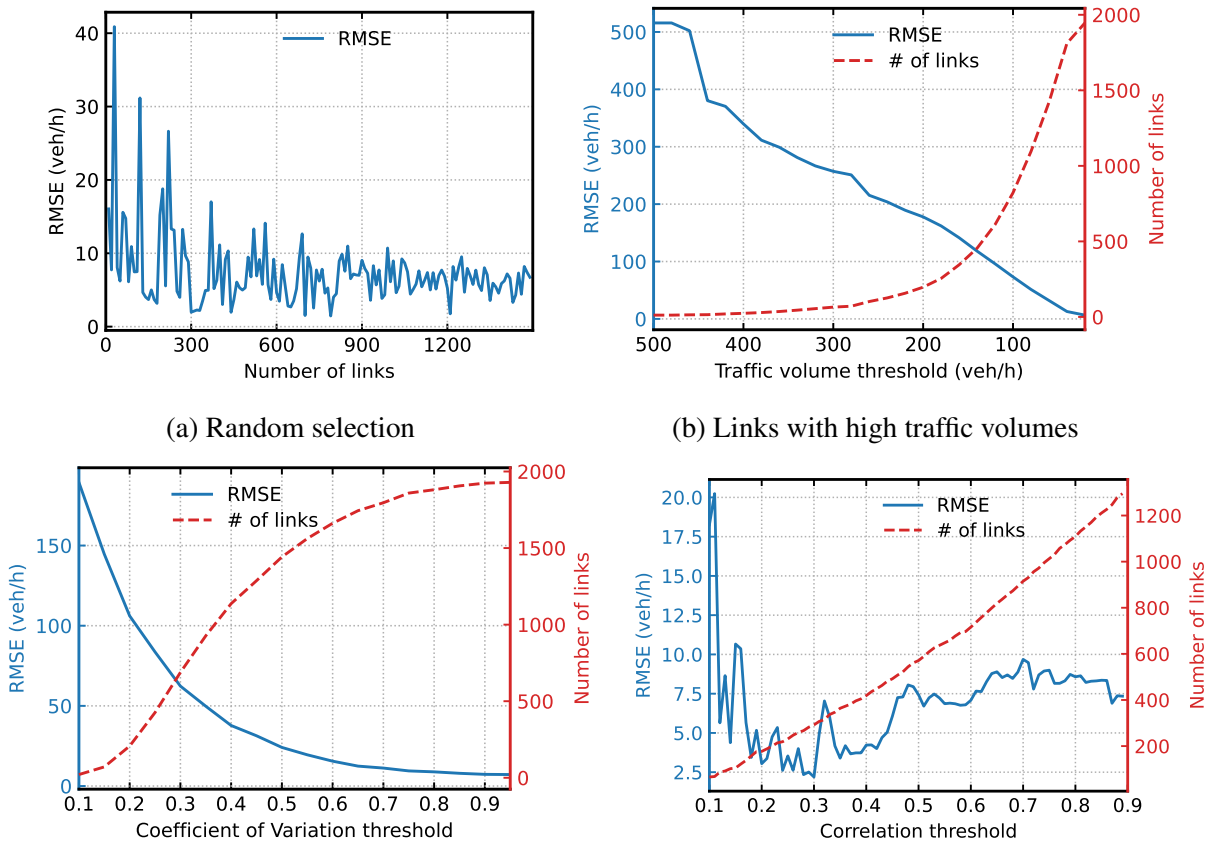
In terms of the constraints on network capacity allocation, we disallow the full blocking of any links to maintain the completeness of the route set. This measure prevents vehicles from running out of available routes and ensures connectivity to destinations throughout their trips. In this case, $\mathbf{d}_{\min} = \mathbf{1}$ and $\mathbf{d}_{\max} = \mathbf{a} - \mathbf{1}$, where $\mathbf{a}$ is the vector representing the number of lanes on all links. Similarly, for signal timing constraints, we set $\mathbf{c}_{\min} = 0.1\boldsymbol{T}$ and $\mathbf{c}_{\max} = 0.9\boldsymbol{T}$, where $\boldsymbol{T}$ denotes the cycle times of the intersections under consideration. These constraints ensure that the duration of any signal phase must be longer than 10% of the cycle time but cannot exceed 90% of the entire cycle. This treatment guarantees a balanced and effective allocation of green times at intersections, promoting smoother traffic flow.

Selection of traffic measurements

For large-scale networks, it is crucial to select representative links for measurement. As the network scale increases, the number of candidate terms in the library also grows dramatically. Limiting the number of traffic measurements used to represent the system state can not only reduce the dimensionality of system terms but also mitigate noise from uncertain and unreliable traffic measurements. The selected links should be able to provide reliable estimates to the MFD dynamics estimated from all links. Thus, we use the root mean squared error (RMSE) between the average traffic volumes estimated by all links and those by the selected links to determine the effective set of links to measure. Four selection methods are compared: (1) selecting links randomly; (2) selecting links with higher traffic volumes; (3) selecting links with more stable traffic over the period

of interest; (4) selecting links with less correlation.

Figure 4 shows the results obtained from these selection methods by using data from 1000 simulation replications. From Figure 4a, it is evident that randomly selecting links leads to fluctuating RMSE values. According to the law of large numbers, the fluctuation gradually diminishes with the increment of the number of selected links. In terms of the second method, intuitively, averaging the traffic states of links with high traffic volumes will lead to an overestimation of the traffic state of the entire network. This is confirmed by the high RMSEs shown in Figure 4b. For instance, if we only consider links with traffic volumes higher than 200, the resulting RMSE and the number of links will be 189.33 and 157, respectively. In essence, the third method assumes that traffic stability is a good indicator of link representativeness regarding traffic states. Here, we use the coefficient of variation across all link-wide observations as an indicator. However, as shown in Figure 4c, the RMSEs associated with small numbers of links are still very large. It only shows a minor improvement compared to the second method. Finally, the last method validates that selecting links with less correlation provides accurate estimations of network-wide traffic states. With fewer than 200 links selected, the RMSE is even less than 20. In contrast, the first method is unstable with fewer than 100 links. Therefore, in the following experiments, we employ the last method to determine the links to measure. More specifically, we set the correlation threshold to 0.2 so that 177 links are selected, resulting in a RMSE of 3.04.



(a) Random selection

(b) Links with high traffic volumes

(c) Links with stable traffic

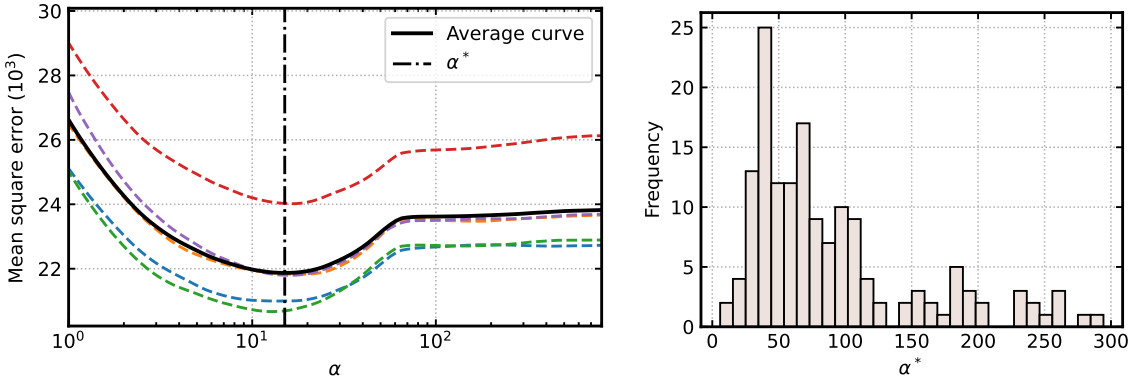
(d) Links with less correlation

FIGURE 4: Determination of representative links to measure.

SINDY model estimation performance

The problem expressed by Equation (13) is a critical step in SINDy for estimating a sparse model based on observed system dynamics. The least squares method, regularized with an l^1 norm, leads to the lasso regression technique. The problem can be efficiently solved by a bunch of optimization algorithms, such as coordinate descent and the sequential thresholded least squares (STLS) algorithm Brunton et al. (*8*). In this study, we apply the coordinate descent algorithm implemented in the *Lasso* module of the *scikit-learn* Python library.

Moreover, we incorporate cross-validation to identify the optimal regularization hyperparameter α^* to determine the active terms and their corresponding coefficients. Figure 5a shows cross-validation results for the final link within the set of links selected in Section 5.2. This figure depicts the performance of different α values under five-fold cross-validation, together with the average performance across all folds. The optimal α is located by the minimum point of the average performance curve. Figure 5b displays the distribution of the optimal α values for different links. While the optimal α varies among the links, most values fall in the range of 10 to 130.



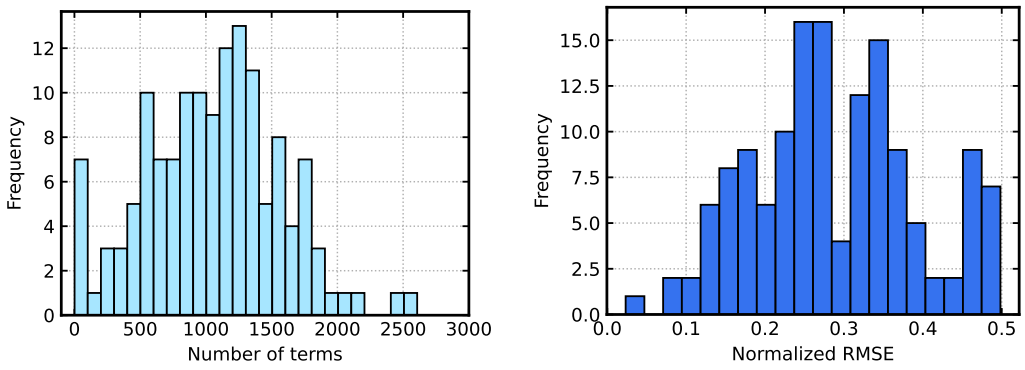
(a) (b)

FIGURE 5: Determination of the regularization penalty.

Figure 6 shows the distributions of the number of active terms identified by the SINDy technique and the RMSE between observed traffic measurements and those estimated by the identified system. The results indicate that the traffic states of most links are governed by 100 to 2,000 active terms out of about 42,000 potential terms. These models achieve normalized RMSEs, calculated relative to the corresponding mean values, ranging from 0 to 0.5 for the majority of links, demonstrating a satisfactory balance between model complexity and estimation accuracy.

Control evaluation of SINDy-MPC

We evaluate the proposed SINDy-MPC framework through three aspects. First, we analyze its stability by replacing the "Transportation system" component in the framework with the identified system via SINDy, i.e., the transportation system is modeled as an analytical function. Let $\tilde{\mathbf{x}}$ be the "observed" traffic measurements, then $\tilde{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_{k+1}$ in this case. Second, we introduce some noise into the analytical "transportation system" to evaluate the stability of the control system. Namely, $\tilde{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_{k+1} + \boldsymbol{\delta}_0$, where $\boldsymbol{\delta}_0 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_0)$. In the following associated experiment, we set $\boldsymbol{\sigma}_0 = 0.1\hat{\mathbf{x}}$. Third, we incorporate the complex traffic simulator as the "transportation system" to simulate the traffic states under different control actions. We have $\tilde{\mathbf{x}}_{k+1} = \mathscr{S}(\hat{\mathbf{x}}_k, \hat{\mathbf{u}}_k)$ where $\mathscr{S}$



(a) (b)

FIGURE 6: Summary of the models identified.

represents the traffic simulator. This scenario is used to assess the performance of the proposed dynamic network capacity control strategy in real applications. In these experiments, both m_p and m_c are set to 3, i.e., the prediction horizon is as wide as the receding horizon.

Figure 7 illustrates the MPC results from 20 control steps. The bottom subplot shows that the system converges after around 6 iterations, with the value of the objective reducing by 3.76%. Three examples are provided for the traffic measurements, traffic signals, and lane allocations under control in the first three subplots, respectively. Traffic volumes of the selected three example links also observe a convergence within 20 iterations. We also found that most traffic measurements obtain an improvement in traffic volumes. As the output of the control system, the convergence of $\mathbf{x}$ indicates that this system is a stable system. Traffic signal configurations, on the other hand, may differ significantly in consecutive intervals like signal 1, or gradually stabilize to fixed configurations like signal 2 and signal 3. Likewise, the selected links for lane allocation control present a similar phenomenon.

Then, to validate the stability of the system with stochasticity involved, we define a stochastic model by introducing a disturbance term to the analytical system. The disturbances are imposed on the traffic measurements, proportional to their deterministic values. Figure 8 compares the final stabilized $\mathbf{x}$'s obtained from the deterministic and stochastic system. Their closeness evidences that the presented control system preserves its stability nature under a stochastic context and is robust to noisy measurements. Furthermore, it is worth mentioning that only one continuous control decision (i.e., signal split) and three discrete control decisions change in the stochastic system compared to the results obtained from the deterministic one.

Further, we applied the SUMO simulator to simulate the response of the transportation system to the control decisions. Compared to analytical disturbance, the traffic simulator incorporates more complex disturbance patterns by integrating various uncertainties ranging from uncertain driving behaviors to uncertain traffic assignments. In addition, the demand changes that cannot be captured by the system also render significant uncertainty in the demand side of the simulator, thereby imposing unpredictability on the system state. As a result, it is hard to evaluate the stability of the integrated system. Instead, we compare the simulated traffic measurements with the values estimated by the system identified by SINDy, as shown in Figure 9. It is clear that the traffic measurements estimated by the identified SINDy model fit well with the simulated values, albeit with

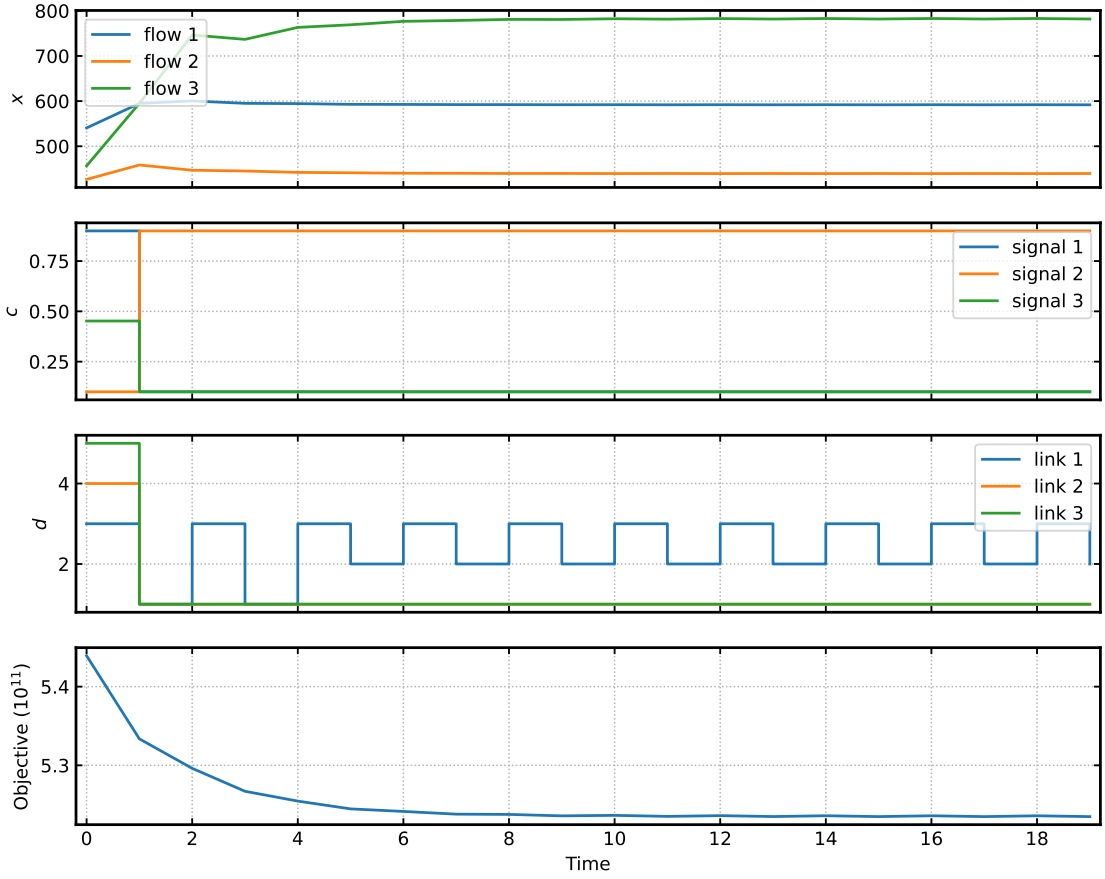


FIGURE 7: Performance of the SINDY-MPC approach under the deterministic model (some example flows, signals, and links).

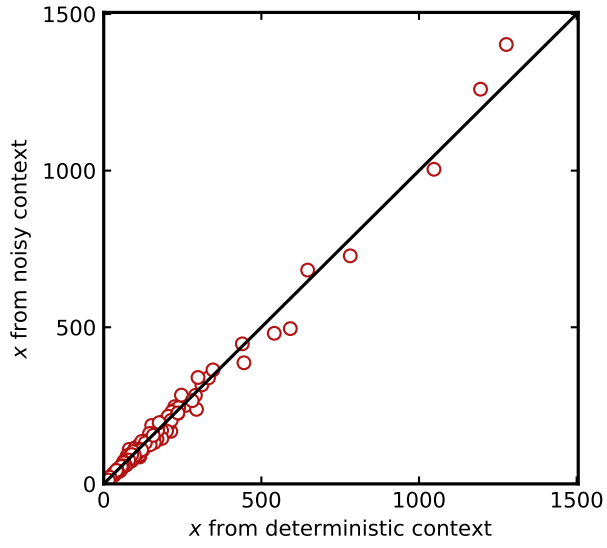


FIGURE 8: Comparing the stabilized **x** under deterministic and stochastic analytical models.

a slight tendency towards overestimation. The suggested control decisions by the MPC-SINDy framework are thus reliable subject to acceptable prediction errors.

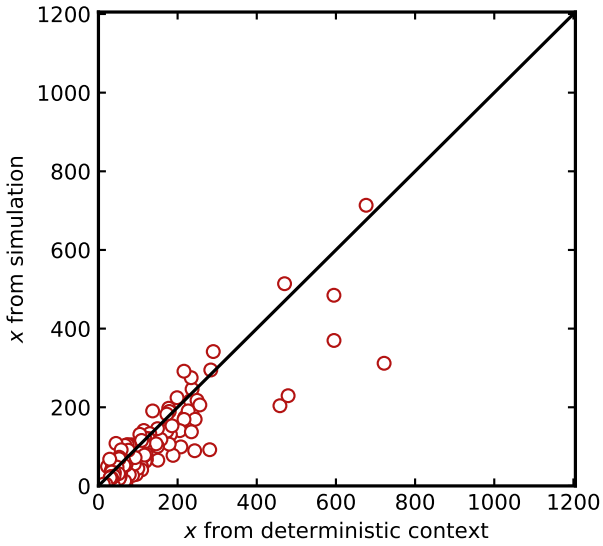


FIGURE 9: Comparing the simulated **x** and the estimated **x**.

Resilience to cyberattacks

Cyberattack disruption scenario design

Aside from testing the performance of the proposed approach under normal operational conditions, it is also important to validate its robustness in the context of cyberattacks. CAVs, which are infused with numerous electronics and software to support various autonomous features, resemble computers more than traditional vehicles. Therefore, cybersecurity emerges as a critical issue in the era of CAVs Katrakazas et al. (*18*). As a typical example, we devised a scenario involving a type of sensor attack, specifically targeting Light Detection and Ranging (LiDAR) sensors. LiDAR is commonly used in CAVs for collision avoidance, ACC, and object recognition. Given that the car-following model adopted in this study is calibrated with trajectories of ACC vehicles, this scenario design is particularly relevant. In our experiments, we simulate attacks on the acceleration reaction, introducing disturbances to the acceleration decision-making process of the vehicles. Denote p as the proportion of vehicles being attacked. The acceleration after attacks will be given by

$$a_{\text{ATK}}(s,v,\Delta v) = a_{\text{IDM}}(s,v,\Delta v) + \delta_1 v \tag{36}$$

where the second term indicates the disturbance intensity on acceleration as a proportion δ_1 of the vehicle velocity. Given a specific δ_1, faster vehicles observe greater uncertainty in acceleration determination. Here, we simulate an attack scenario where $\delta_1 \sim \mathcal{N}(\mu_1, \sigma_1)$. Traffic states under multiple μ_1's from -0.09 to -0.01 with intervals of 0.01 and a fixed $\sigma_1 = 0.005$ are evaluated. ACC is dependent on the relative velocity and headway between the follower and the leader derived from LiDAR measurements Boddupalli et al. (*19*). Hence, this type of intra-vehicle communication attack can be regarded as attacking on the LiDAR measurements.

Impacts of cyberattacks

It is beneficial to understand how cyberattacks will affect traffic externalities. Thus, in Figure 11, we provide the contour plots of the metrics evaluating the pedestrian risk exposure, noise emission, fuel consumption, and exhaust emissions, respectively. In particular, pedestrian risk is measured by an enhanced Routledge indicator, which describes the probability of a vehicle hitting a pedestrian under certain congestion conditions if the pedestrian crosses the road randomly. We refer readers to (*14*) for the derivation of this indicator, as well as the detailed calculation methods for the other three metrics. Note that we have to make δ_1 in Equation (36) a constant in each attacking scenario in order to draw these contours accurately. Figure 11a offers the contour plot of pedestrian risk, which reduces as the attack intensity (δ_1) and proportion (p) increase. Additionally, the contour lines are more vertical in the upper-right part, while they are more horizontal in the lower-left part. This implies that when $|\delta_1|$ is small and p is large, altering δ_1 can lead to more obvious changes in pedestrian risk exposure. In contrast, the risk exposure is more sensitive to p when $|\delta_1|$ is large and p is small. The contours of traffic noise exposure present a similar shape as those of pedestrian risk. The only difference lies in the density of contour lines. In Figure 11b, the distance between every two neighboring lines remains nearly the same across the investigated range of parameters. This suggests that any changes at any point will result in a similar influence on the noise exposure. However, the contours of fuel consumption are almost straight lines, indicating that the sensitivities to δ_1 and p are very close. Yet, the increase of contour density from the small value area to the large value area implies that the impact of the alterations in these two parameters is more significant when their original values are large. On the other hand, since exhaust emissions are estimated by using fuel consumption data, the contours of different kinds of emissions will also present a similar shape to those of fuel consumption. Figure 11d shows the contours of CO_2 emission as an example, while the other emissions, such as CO, PM_x, and NO_x, are similar.

Figure 10 compares the MFD dynamics (aggregated every five minutes) under different configurations of the cyberattack under consideration. Each subplot in the first row compares the MFDs of the scenarios with different percentages of attacked vehicles under the same attack intensity, i.e., p's are different but μ_1's are the same. In contrast, each subplot in the second row compares the MFDs of the cases where the percentages of attacked vehicles are the same but are attacked differently, i.e., p's are the same but μ_1's are different. Clearly, when p is small, it is hard to recognize the difference in MFDs compared to the one free of cyberattacks. Surprisingly, when the absolute value of μ_1 is also small, such as Figure 10(a), we can even observe a slight improvement in MFD. This implies that adding appropriate driving heterogeneity can improve traffic efficiency. However, the improvement is very limited, and it will disappear as the attacking intensity increases. Moreover, the MFDs of scenarios with high attacking penetration demonstrate clear reductions, regardless of the attacking intensity. From the second row of subplots, it is hard to differentiate the MFDs under different μ_1 when $p = 0.1$. Their differences become clearer as p increases.

Since the urban traffic resilience indicator adopted in the study is derived from MFD dynamics, we further compare the traffic resilience losses across different scenarios in Figure 12. Three demand scenarios are considered, with small, medium, and large demand scenarios representing the original demand, 1.2, and 1.5 multiples of the original demand, respectively. The results are in line with the findings from Figure 10. The impact of $|\mu_1|$ is minor when the proportion of attacked vehicles is small, with resilience losses remaining similar across the first two cyberattack scenarios regardless of the demand level. However, as the proportion of attacked vehicles increases

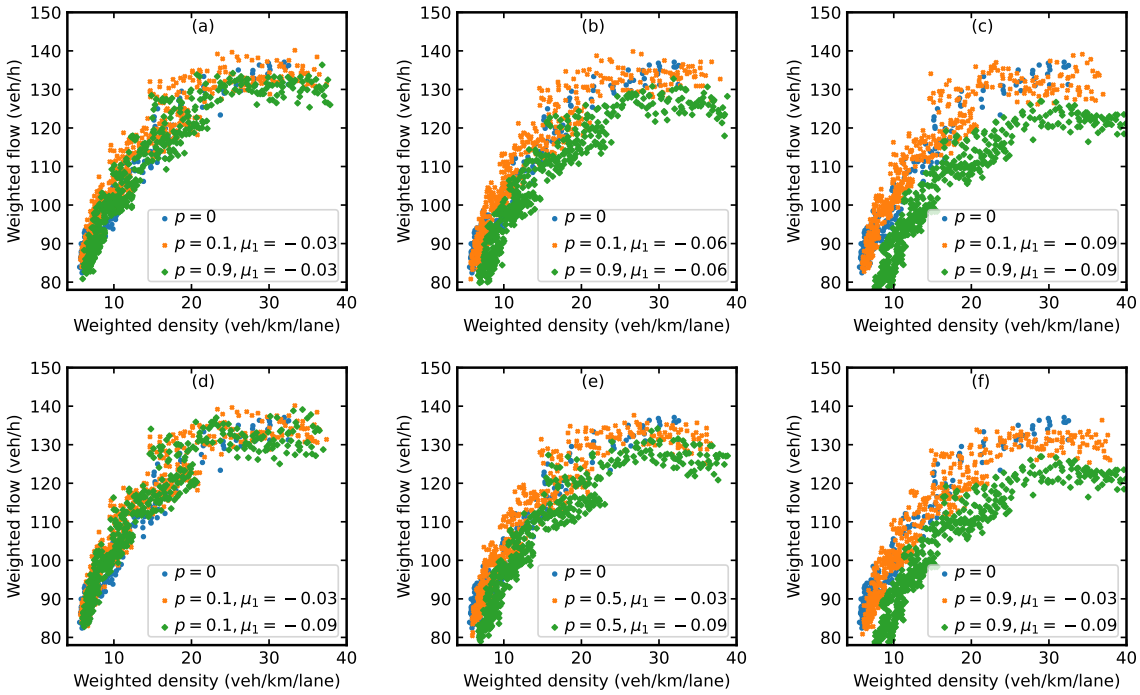


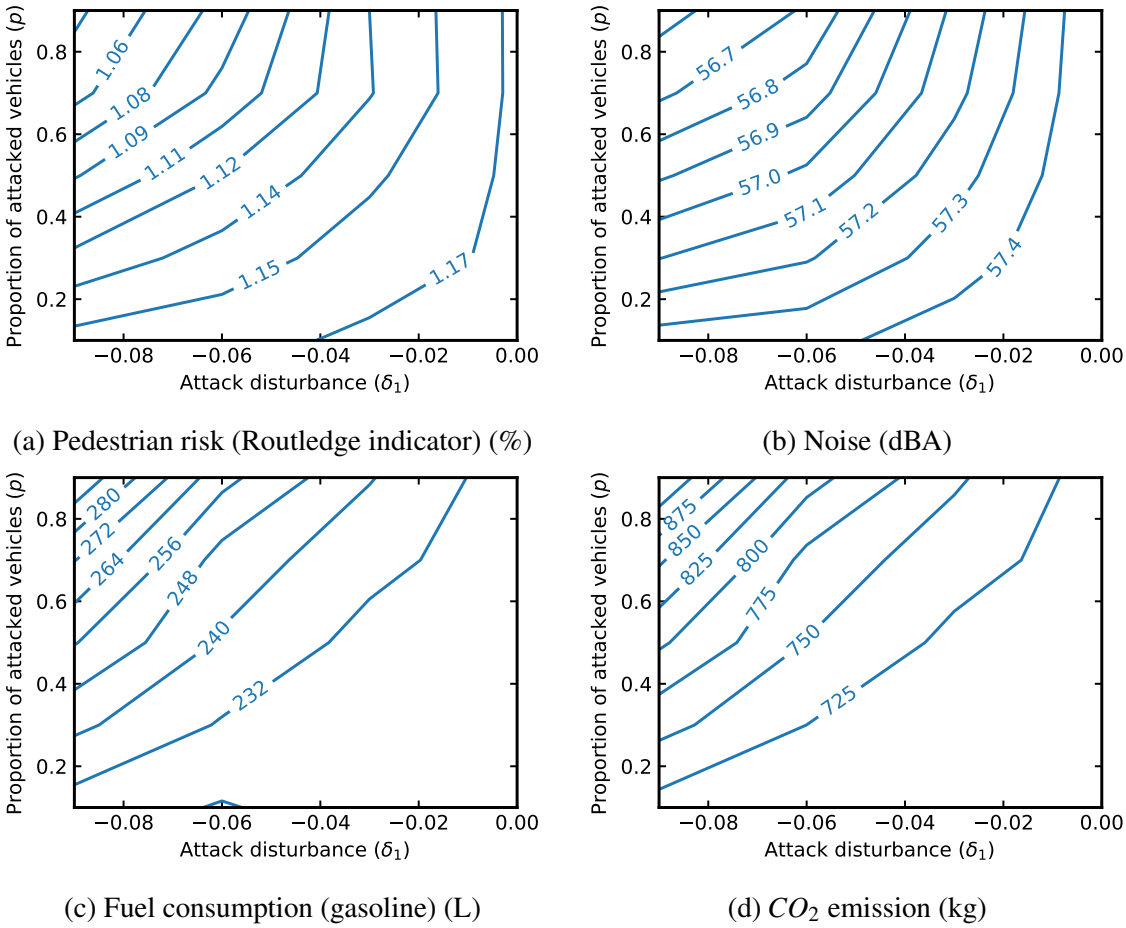
FIGURE 10: MFD comparison under different cyberattack scenarios.

to 90%, traffic resilience losses escalate by about three times as $|\mu_1|$ increases from 3% to 9%.

To summarize, cyberattacks hold the potential of deteriorating fuel consumption and exhaust emissions. Moreover, they can also significantly reduce MFD dynamics by causing transportation disruptions and impeding traffic efficiency, thereby deteriorating urban traffic resilience. Although they can improve the pedestrian risk and noise exposure, the improvement is marginal compared to the negative effects. Therefore, it is imperative to validate the resilience of the proposed dynamic capacity allocation against cyberattacks.

Control performance under disruption scenarios

As introduced in Section 4.3.1, the traffic resilience indicator is calculated based on the deviations in trip completion rates. To assess the resilience of the proposed dynamic network capacity allocation control to cyberattacks, we compare the trip completion rates achieved under this control with those obtained from the baseline control configuration, i.e., the one with the current lane allocation plan and optimized static traffic signal splits. Three hours of simulations are run, with the first and the last half an hour as the simulation warm-up and dissipation periods, respectively. Traffic dynamics were aggregated every five minutes, resulting in 24 time intervals. To account for simulation randomness in traffic assignment, we averaged the results across 10 replications and calculated the corresponding standard deviations. Similar to Figure 12, four different cyberattack scenarios are considered. While all simulations were conducted at the small demand level, similar patterns were observed in medium and high demand scenarios. We can see from each subplot that trip completion rates under the SINDy-MPC solution are lower than those under the baseline control before the 8th interval. However, after this point, the SINDy-MPC solution consistently outperforms the baseline control. Overall, the SINDy-MPC framework improves trip completion



(a) Pedestrian risk (Routledge indicator) (%)

(b) Noise (dBA)

(c) Fuel consumption (gasoline) (L)

(d) CO_2 emission (kg)

FIGURE 11: Impact of cyberattacks on traffic externalities.

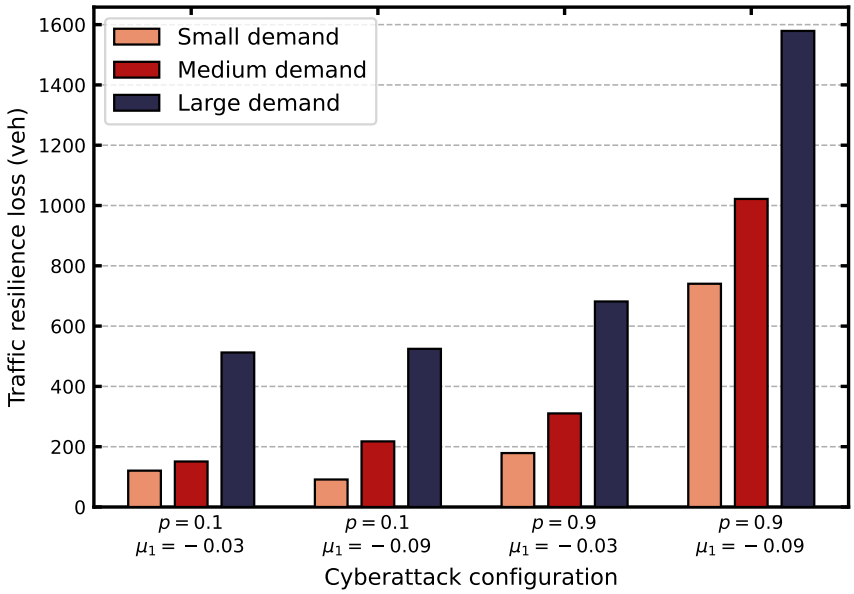
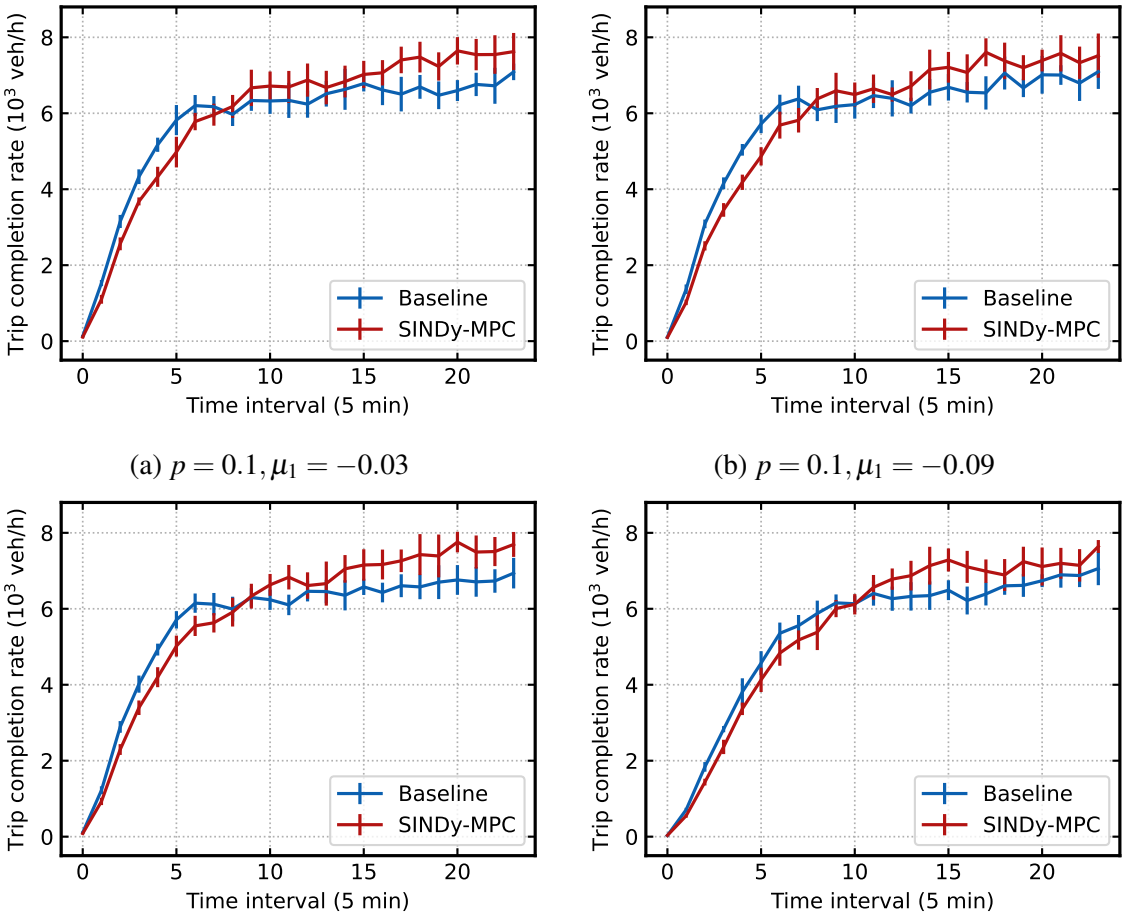


FIGURE 12: Impact of cyberattacks on urban traffic resilience.

rates in all scenarios, demonstrating greater resilience to cyberattacks. Nonetheless, the degree of improvement varies across scenarios. Specifically, the improvements in the two scenarios with $\mu_1 = -0.03$ are greater than the two with $\mu_1 = -0.09$. This suggests that the resilience of the proposed control scheme is subject to the intensity of the attack. Additionally, in the scenario where $p = 0.9$ and $\mu_1 = -0.09$, the trip completion rate curves are significantly lower than in other scenarios, indicating more substantial traffic resilience losses. This observation is consistent with the findings presented in Figure 12.



(a) $p = 0.1, \mu_1 = -0.03$ (b) $p = 0.1, \mu_1 = -0.09$

(c) $p = 0.9, \mu_1 = -0.03$ (d) $p = 0.9, \mu_1 = -0.09$

FIGURE 13: Comparison of trip completion rates between the baseline control and the SINDy-MPC solution under different cyberattack scenarios.

CONCLUSIONS

This study investigated the dynamic network capacity allocation control problem in the era of CAVs, in which traffic signal splits and lane allocation plans were optimized to improve traffic efficiency in real time. The development of this control scheme was motivated by the common tidal traffic phenomenon observed in urban transportation networks, resulting from the spatiotemporal variations in travel demand. Its realization was dependent on a SINDy-MPC framework, which integrated a sparse yet accurate model identified through the SINDy technique to estimate system

dynamics, coupled with MPC for decision-making. The proposed control scheme was evaluated on a large-scale real network in the city center of Munich, Germany. The microscopic SUMO traffic simulator was utilized to validate the effectiveness of the control decisions.

Experiments demonstrated that selecting representative links to estimate the overall traffic state based on correlation was superior to methods based on random selection, traffic volume, and traffic stability. This approach resulted in a smaller RMSE and fewer selected links, reducing the system dynamics' dimensionality. The identified system by SINDy was found to be sparse in the dynamic expression of each link compared to the number of terms involved in the candidate library, while maintaining accuracy in terms of RMSEs. We evaluated the performance of the proposed control scheme in the contexts of deterministic system and stochastic system models, as well as a complex simulation model. Results from the deterministic and stochastic system models illustrated system stability in the presence of no stochasticity or Gaussian noise disturbances. Moreover, as evident by simulations, the proposed network capacity control effectively reduced traffic imbalances on the associated roads and improved overall traffic efficiency.

Further, considering the vulnerability of CAVs to cyberattacks, we systematically assessed the impacts of such attacks from the perspectives of externalities, MFD dynamics, and traffic resilience. The impacts on traffic safety, environmental conditions, fuel consumption, and exhaust emissions were not negligible. We found that although cyberattacks were imposed on the demand side, the shape of MFD also observed a degradation. The impact of the proportion of attacked vehicles was more significant than that of the intensity of the attack. This finding was also validated by the comparison of traffic resilience losses under different cyberattack configurations. Specifically, the impact of the attack intensity is minor when the proportion of attacked vehicles is small, with resilience losses remaining similar across different intensity levels regardless of the demand level. However, as the proportion of attacked vehicles increases, traffic resilience losses escalate significantly as the attack intensity increases. We then applied the proposed control scheme to these cyberattack scenarios, demonstrating its robustness and resilience. The result suggested that the resilience of the proposed control scheme is subject to the intensity of the attack.

Future research can focus on integrating dynamic capacity allocation control decisions into reinforcement learning methods, given their ability to model complex relationships. How to design the reinforcement learning procedure to manage the high dimensionality issue of the problem will be the key challenge of the work. Additionally, since the case study presented here integrated an unchanged SINDy model into the MPC framework, further improvement could involve developing an alternating optimization procedure to determine optimal control decisions and update the SINDy surrogate model. In this case, special treatments will be needed to enhance computational efficiency.

ACKNOWLEDGEMENTS

This research was supported by the CONCERT-Japan DARUMA project, funded by the German Federal Ministry of Education and Research (BMBF) under Grant Number 01DR21010.

REFERENCES

1. Fu, Q., Y. Tian, and J. Sun, Modeling and Simulation of Dynamic Lane Reversal Using a Cell Transmission Model. *Journal of Intelligent Transportation Systems*, Vol. 26, No. 6, 2022, pp. 717–729.
2. Wollenstein-Betech, S., I. C. Paschalidis, and C. G. Cassandras, Optimizing Lane Reversals in Transportation Networks to Reduce Traffic Congestion: A Global Optimization Approach. *Transportation Research Part C: Emerging Technologies*, Vol. 143, 2022, p. 103840.
3. Ampountolas, K., J. A. Dos Santos, and R. C. Carlson, Motorway Tidal Flow Lane Control. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 4, 2020, pp. 1687–1696.
4. Chen, S., H. Wang, and Q. Meng, An Optimal Dynamic Lane Reversal and Traffic Control Strategy for Autonomous Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 4, 2022, pp. 3804–3815.
5. Levin, M. W. and S. D. Boyles, A Cell Transmission Model for Dynamic Lane Reversal with Autonomous Vehicles. *Transportation Research Part C: Emerging Technologies*, Vol. 68, 2016, pp. 126–143.
6. Chu, K.-F., A. Y. S. Lam, and V. O. K. Li, Dynamic Lane Reversal Routing and Scheduling for Connected and Autonomous Vehicles: Formulation and Distributed Algorithm. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 6, 2020, pp. 2557–2570.
7. Qurashi, M., Q.-L. Lu, G. Cantelmo, and C. Antoniou, Dynamic Demand Estimation on Large Scale Networks Using Principal Component Analysis: The Case of Non-Existent or Irrelevant Historical Estimates. *Transportation Research Part C: Emerging Technologies*, Vol. 136, 2022, p. 103504.
8. Brunton, S. L., J. L. Proctor, and J. N. Kutz, Discovering Governing Equations from Data by Sparse Identification of Nonlinear Dynamical Systems. *Proceedings of the National Academy of Sciences*, Vol. 113, No. 15, 2016, pp. 3932–3937.
9. Kaiser, E., J. N. Kutz, and S. L. Brunton, Sparse Identification of Nonlinear Dynamics for Model Predictive Control in the Low-Data Limit. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, Vol. 474, No. 2219, 2018, p. 20180335.
10. Fasel, U., E. Kaiser, J. N. Kutz, B. W. Brunton, and S. L. Brunton, Sindy with control: A tutorial. In *2021 60th IEEE Conference on Decision and Control (CDC)*, IEEE, 2021, pp. 16–21.
11. Lu, Q.-L., W. Sun, J. Dai, J.-D. Schmöcker, and C. Antoniou, Traffic resilience quantification based on macroscopic fundamental diagrams and analysis using topological attributes. *Reliability Engineering & System Safety*, 2024, p. 110095.
12. De Oliveira, L. B. and E. Camponogara, Multi-Agent Model Predictive Control of Signaling Split in Urban Traffic Networks. *Transportation Research Part C: Emerging Technologies*, Vol. 18, No. 1, 2010, pp. 120–139.
13. Spall, J. C., Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 34, No. 3, 1998, pp. 817–823.
14. Lu, Q.-L., M. Qurashi, and C. Antoniou, Simulation-Based Policy Analysis: The Case of Urban Speed Limits. *Transportation Research Part A: Policy and Practice*, Vol. 175, 2023, p. 103754.

15. Lopez, P. A., M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, Microscopic traffic simulation using sumo. In *2018 21st international conference on intelligent transportation systems (ITSC)*, IEEE, 2018, pp. 2575–2582.
16. Gunter, G., D. Gloudemans, R. E. Stern, S. McQuade, R. Bhadani, M. Bunting, M. L. Delle Monache, R. Lysecky, B. Seibold, J. Sprinkle, et al., Are commercially implemented adaptive cruise control systems string stable? *IEEE Transactions on Intelligent Transportation Systems*, Vol. 22, No. 11, 2020, pp. 6992–7003.
17. De Souza, F. and R. Stern, Calibrating microscopic car-following models for adaptive cruise control vehicles: Multiobjective approach. *Journal of Transportation Engineering, Part A: Systems*, Vol. 147, No. 1, 2021, p. 04020150.
18. Katrakazas, C., A. Theofilatos, G. Papastefanatos, J. Härri, and C. Antoniou, Cyber Security and Its Impact on CAV Safety: Overview, Policy Needs and Challenges. In *Advances in Transport Policy and Planning*, Elsevier, Vol. 5, 2020, pp. 73–94.
19. Boddupalli, S., A. S. Rao, and S. Ray, Resilient Cooperative Adaptive Cruise Control for Autonomous Vehicles Using Machine Learning. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 9, 2022, pp. 15655–15672.