Sensor Placement Considering the Observability of Traffic Dynamics: on the Algebraic and Graphical Perspectives

Xinyue Hu[a] and Yueyue Fan[a,*]

[a]*Department of Civil and Environmental Engineering, University of California, Davis, One Shields Ave, Davis 95616, USA*

Keywords:
dynamic traffic networks
sensor location problem
exact observability
structural observability

In this paper, we present a new sensor location model that aims to maximize the observability of link densities in a dynamic traffic network described using a piecewise linear ODE system. We develop an algebraic approach based on the eigenstructure to determine the sensor location for achieving full observability with a minimal number of sensors. Additionally, a graphical approach based on the concept of structural observability is developed. By exploiting the special property of flow conservation in traffic networks, we derive a simple analytical result that can be used to identify observable components in a partially observable system, which is the main contribution of this paper. The graphical and algebraic properties of observability are then integrated into a sensor location optimization model considering a wide range of traffic conditions. Through numerical experiments, we demonstrate the good performance of our sensor deployment strategies in terms of the average observability and estimation errors.

1 Introduction

Accurate and reliable traffic data is crucial for traffic operation and management. Traffic sensors serve as one of the most important sources of such data. Due to budget constraints, one often faces a challenge where a limited number of sensors need to be deployed in a way that the obtained information from the sensors is the richest.

In the context of maximizing the richness of information in a traffic network, the concept of observability has received much attention (Gentili and Mirchandani, 2012; Castillo et al., 2015), where the term observability refers to the ability to determine whether a given set of measurements is sufficient to uniquely infer the states of a traffic network, such as flows, speeds, and densities.

Most traffic sensor placement literature on observability focused on aggregated flow counts and are thus referred to as the static observability problem in Castillo et al. (2015). Existing approaches already

* Corresponding author. E-mail address: yyfan@ucdavis.edu.

provide simple and efficient exact solutions for full and partial observability in the static case (Hu et al., 2009; Ng, 2012; He, 2013; Viti et al., 2014).

For dynamical systems, observability is a measure of how well the internal states of a system can be inferred from knowledge of external outputs (from sensors). Observability and its mathematical dual controllability have been well-studied in the control theory since introduced by Kalman (1960). More recently, graph-theoretic approaches have been developed to explore an extended concept of observability, the structural observability, which focuses on the structure of the algebraic conditions (Lin, 1974; Liu et al., 2011, 2013). Structural observability bypasses the need for accurate/exact knowledge of system dynamics and in many cases, it is equivalent to observability. This concept and the graphical approaches are also borrowed in recent traffic network observability studies (Bekiaris-Liberis et al., 2017; Rostami-Shahrbabaki et al., 2020; Mousavi and Kouvelas, 2020). In recent years, more sensor placement studies incorporated the observability concept in a dynamic traffic system (Contreras et al., 2015; Agarwal et al., 2015; Bekiaris-Liberis et al., 2017). For example, based on nonlinear dynamic traffic network modeling, sensor placement strategies have been studied in Contreras et al. (2015) and Nugroho et al. (2021). The non-linearity has to be addressed by either linearizing the problem around an equilibrium point or assuming a known sample path of the dynamic process. However, traffic conditions are subject to variations and a fixed equilibrium point may not exist. Therefore, a systematic and effective sensor placement method that could maximize information gain while being capable of handling a large spectrum of dynamic traffic conditions is still needed.

This paper aims to bridge the gap by proposing sensor location optimization methods considering the observability of traffic density under various dynamic traffic conditions. The dynamics of a highway network is captured using a piecewise-linear macroscopic traffic dynamical model and we explore the algebraic and graphical properties of such systems. The main contributions of our research are the following:

- We establish new understandings of graphical and algebraic properties of observability beyond what has been known for general linear dynamic systems by exploiting unique characteristics of traffic networks. Our results advance the current understanding of observability concept in the transportation science literature.
- We incorporate the structural observability concept into the design of an optimal sensor location problem to maximize information gain from sensors under various traffic conditions. Numerical experiments demonstrate the effectiveness of our optimal sensor location strategies in terms of observability and traffic estimation quality.
- The analytical and computational advantages of our results are made possible by exploiting special features of the dynamic traffic networks, which showcases how domain expertise from the transportation field could be integrated with and extend the current state of knowledge of complex systems.

The rest of the paper is organized as follows. In Section 2, we establish the mathematical properties and analyses associated with structural observability, and introduce a sensor location optimization both from graphical and algebraic perspectives. In Section 3, we use various numerical examples to explain the definition, properties, and calculation of observability concept, and to test out the quality of sensor location solutions obtained from our model. The last section concludes the paper with discussion and possible future extensions.

2 Methodology

2.1 Dynamic traffic network premises

The dynamic traffic system in this study is described using a deterministic queueing model, the Link Queue Model (LQM). We choose a macroscopic traffic model since we are only interested in the aggregate behavior of vehicles. Plus, the LQM model keeps a good balance between physical realism and mathematical tractability, which is important for achieving computational scalability and theoretical understanding. We acknowledge that no mathematical model could perfectly represent the real-world traffic dynamics. Later on in the numerical experiments, we will showcase that even though our sensor placement model is based on analysis of a simplified dynamic traffic model, it results in rich information that leads to better traffic state estimation under various simulated conditions that deviate from the analytical model. The remainder of this subsection provides a quick summary of various dynamic traffic models, for which readers who are familiar with traffic dynamic systems literature may skip.

The famous LWR model (Lighthill and Whitham, 1955; Richards, 1956) is a macroscopic traffic flow model that captures shock and rarefaction waves of fluid-like traffic. The flow conservation equation can be written as:

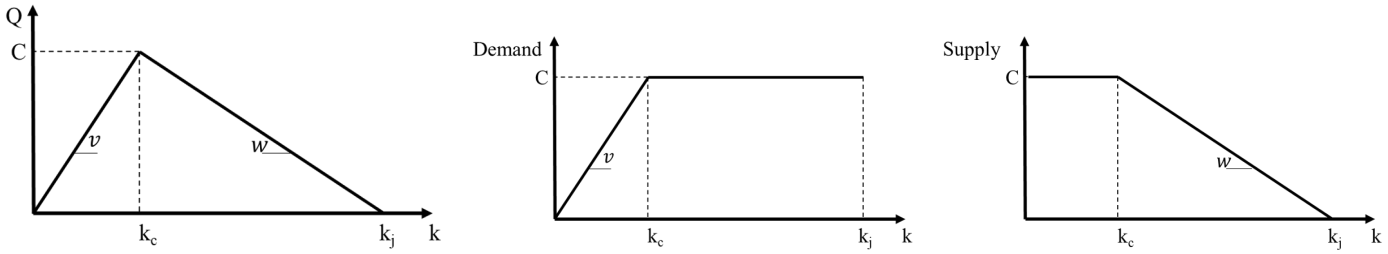
$$\frac{\partial k(x,t)}{\partial t} + \frac{\partial q(x,t)}{\partial x} = 0 \tag{1}$$

where k and q denote the traffic density and flow, respectively. In addition, an equilibrium speed-flow relationship is assumed which refers to the fundamental diagram. This model is a partial differential equation model.

The cell transmission model (CTM) is developed to approximate the LWR model (Daganzo, 1994, 1995). The CTM model divides a link into small cells with the same length and defines each cell's demand (maximum sending flow) and supply (maximum receiving flow) to describe interactions between adjacent freeway cells as well as shockwaves. The actual flow transmitted is determined by the minimum of the upstream demand and downstream supply. There are different ways to define demand and supply functions based on the assumption of the fundamental diagram. To avoid the overwhelmingly large number of cells for better efficiency, Yperman (2007) proposed the Link Transmission Model which requires only one sending flow and one receiving flow for an entire link. However such demand and supply functions are delayed functions in in- and out-flows, making it less suitable for observability-based sensor location problems.

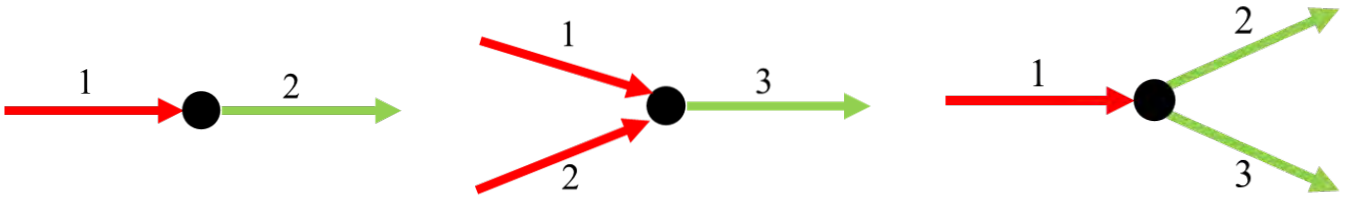
To avoid the nonlinearity in CTM, the switching-mode model (SMM) was proposed by Muñoz et al. (2003). The SMM is a piecewise affine system that switches among different sets of linear difference equations (representing different traffic states of the freeway), depending on boundary flow and the congestion status of the cells in a freeway segment and assuming a triangular fundamental diagram. Taking traffic density as the state variable, the SMM permits uneven cell lengths, which greatly reduces the dimension of the system. To further incorporate the merging and diverging behavior in highway networks, Jin (2021) developed a deterministic queueing model, the LQM whose junction model is realistic and link model is mathematically tractable. The LQM is a system of ordinary differential equations with link density being the state variable. Let k_a and l_a, respectively, denote the traffic density and length of link a, and the link density updates based on the following equation:

$$\dot{k}_a(t) = \frac{f_a(t) - g_a(t)}{l_a} \tag{2}$$



(a) Triangular fundamental diagram (b) Link demand (maximum sending flow) (c) Link supply (maximum receiving flow)

Figure 1: Fundamental diagram, demand and supply functions



(a) A linear junction (b) A merge (c) A diverge

Figure 2: Three Types of Junctions

where $f_a(t)$ and $g_a(t)$ are the in- and out-flux, which are computed from the demand and supply relationship based on the fundamental diagram.

Though LQM allows for various types of fundamental diagrams, we use triangular fundamental diagrams in the following discussion for simplicity. Note that other forms of fundamental diagrams such as trapezoid can work as well, as long as it is piecewise linear. A fundamental diagram depicts the flow-density relationship. A triangular fundamental diagram is shown in Figure 1a, where v is the free-flow speed and w is the congestion wave speed. The maximum possible flow rate C, also known as the capacity, is achieved at the critical density k_c. As shown in Figure 1b, 1c, the demand and supply functions of link of time-dependent density $k(t)$ is defined as:

$$d(t) = \begin{cases} vk(t) & \text{if } k(t) < k_c \\ C & \text{if } k(t) \geq k_c \end{cases} \tag{3a}$$

$$s(t) = \begin{cases} C & \text{if } k(t) < k_c \\ w(k_j - k(t)) & \text{if } k(t) \geq k_c \end{cases} \tag{3b}$$

where k_c, k_j are the critical density and jam density of the link, respectively; v and w are the free-flow speed and shock wave speed, respectively.

For a corridor (without merge/diverge/ramps) consisting of an upstream link 1 and a downstream link 2 as shown in Figure 2a, the flux between link 1 and 2 is governed by the demand of link 1 and the supply of link 2:

$$g_1(t) = f_2(t) = \min\{d_1(t), s_2(t)\} \tag{4}$$

The flux functions defined at merging and diverging junctions in the LQM is physically meaningful.

For a merge with two upstream links, 1 and 2, and a downstream link 3 (Figure 2b), the flux functions are defined as the following:

$$f_3(t) = \min\{d_1(t) + d_2(t), s_3(t)\} \tag{5a}$$
$$g_1(t) = \min\{d_1(t), \max\{s_3(t) - d_2(t), \alpha s_3(t)\}\} \tag{5b}$$
$$g_2(t) = f_3(t) - g_1(t) \tag{5c}$$

where α is the merging priority of link 1. If link 1 and link 2 have the same priority to merge, the situation is called a fair merge[1] and $\alpha = \frac{C_1}{C_1+C_2}$. Note that, any choice of α will be fine and fair merge is not the only choice.

For a diverge junction shown in Figure 2c, the flux functions are determined as follows:

$$g_1(t) = \min\left\{d_1(t), \frac{s_2(t)}{\xi_{1\to2}(t)}, \frac{s_3(t)}{\xi_{1\to3}(t)}\right\} \tag{6a}$$
$$f_2(t) = \xi_{1\to2}(t) g_1(t) \tag{6b}$$
$$f_3(t) = \xi_{1\to3}(t) g_1(t) \tag{6c}$$

where $\xi_{1\to2}(t)$ and $\xi_{1\to3}(t)$ are the proportions of vehicles on link 1 traveling to links 2 and 3, respectively.

Using Fig 2a as an example, when the two links are free-flow with length 1 and the demand of link 1 is less than supply of link 2, the network density dynamics is described as:

$$\begin{bmatrix}\dot{k_1}\\ \dot{k_2}\end{bmatrix} = \begin{bmatrix}-v_1 & 0\\ v_1 & 0\end{bmatrix}\begin{bmatrix}k_1\\ k_2\end{bmatrix} + \begin{bmatrix}f_1\\ -g_2\end{bmatrix} \tag{7}$$

The above system can be expressed as a linear time-invariant (LTI) system:

$$\dot{\boldsymbol{x}}(t) = A\boldsymbol{x}(t) + \boldsymbol{u}(t) \tag{8}$$

where $\boldsymbol{x}$ is the continuous state variables, $\boldsymbol{u}(t) = \begin{bmatrix}f_1 & -g_2\end{bmatrix}^T$ is the boundary flow vector. The dynamics of traffic network density can be effectively captured by an LTI dynamical system during short periods of consistent traffic conditions, such as free-flow conditions. However, for longer durations characterized by varying traffic conditions, the model involves switching among multiple LTI dynamical systems. This aspect will be elucidated in detail in subsection 2.6.

2.2 Premises on observability of a linear time-invariant system

Consider the following LTI dynamical system with $\boldsymbol{x}$ being the state variable (link density) and $\boldsymbol{u}$ being the known input,

$$\dot{\boldsymbol{x}}(t) = A\boldsymbol{x}(t) + \boldsymbol{u}(t),\ \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \text{ is unknown} \tag{9}$$

With sensors, we obtain the corresponding measurements

$$\boldsymbol{y}(t) = C\boldsymbol{x}(t) \tag{10}$$

[1]Note that fair merge is only presented as an example. Our model and analyses presented in this paper can accommodate various node merge models, not only fair merge. As long as the merge node model after the min, max, median function results in piecewise linear functions, the results are applicable.

where $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{y} \in \mathbb{R}^p$, $A \in \mathbb{R}^{n\times n}$ and $C \in \mathbb{R}^{p\times n}$. Here n is the number of states (links) and p is the number of sensors. C is the mapping from the states to measurements. Each row of the observation matrix C is a standard unit vector since the sensor directly measures the link density.

The LTI system is called "observable" if the knowledge of $\boldsymbol{u}(t)$ and $\boldsymbol{y}(t)$ over some finite time interval is sufficient to uniquely determine $\boldsymbol{x}_0$. With a uniquely determined initial value and knowledge of the system dynamics (A), we are able to "observe" the state variables at any time.

The observability matrix $\mathscr{O}$ is defined as $\mathscr{O} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$. The system is observable when $rank(\mathscr{O}) = n$ (the observability matrix has full column rank) and $\boldsymbol{x}_0$ can be uniquely determined. This condition is called the Kalman's rank test. This is also referred to as the exact observability, since the exact parameterization of the system dynamics is known. One might recall that in a static problem, observability is also measured by rank of a matrix. The difference is the matrix - in a static network problem, one focuses on the matrix from the system of linear equations based on flow conservation, which only reflects the connection of various information pieces (such as link flows) over the spatial dimension. While in a dynamic problem, the focus is on the observability matrix $\mathscr{O}$, which reflects the connection of various information pieces (i.e. state variables such as link densities) in both spatial and time dimensions described by the system of ODEs.

2.3 An Algebraic Approach of sensor placement for full observability

In the current subsection, we propose a new Algebraic Approach to identify sensor locations that guarantee full observability of an LTI traffic system. Furthermore, we demonstrate the optimality of this approach by proving its ability to achieve the minimal number of sensors. Note that the proposed Algebraic Approach works for any possible realization of the parameters in A. Our theoretical contribution is made possible by exploring the unique algebraic structure of the observability test within the transportation network, where each sensor directly measures its corresponding link density.[2]

Let us first start from the well-known Popov-Belevitch-Hautus Test (PBH rank test) in the control theory. PBH rank test is equivalent to the Kalman's rank test and it reveals the connection between observability and eigenvalues. The PBH rank test states the following:

Theorem 1. *A linear time-invariant system pair (A,C) is observable, if and only if for any eigenvalue λ of A, we have rank* $\left(\begin{bmatrix} A-\lambda I \\ C \end{bmatrix}\right) = n$.

The sensor location problem now reduces to finding an observation matrix C to satisfy the above condition. Yuan et al. (2013) developed an algebraic procedure to minimize the number of driver nodes in complex networks for exact controllability. Due to the mathematical duality of observability and controllability, the result can be applied to find the minimum number of sensors.

However, in complex networks, a driver node has the ability to control multiple components' states simultaneously. When applying this approach to observer design, it means sensors can measure a linear combination of the component states, resulting in rows in C that are not unit vectors. Traffic sensors, on the

[2]This is different from the context studied in the general complex systems literature where a sensor may measure a linear combination of the system states.

other hand, can only observe the state of one link, not a combination of states. Thus, each row of the matrix C is a unit vector and this extra constraint makes this problem harder to solve.

The rows of C are unit vectors with ith element to be 1 if the ith link is equipped with a sensor and its density is directly observed. Due to the simple yet special structure of C, we developed an algebraic algorithm to find sensor location that guarantees full observability using the minimal number of sensors. This approach can be summarized in the following 4 steps:

Algebraic Approach

Step 1: Compute the eigenvalues of matrix A.

Step 2: For each eigenvalue λ, choose C s.t. $\begin{bmatrix} A-\lambda I \\ C \end{bmatrix}$ is of full rank. This is done by finding the non-pivot columns in the reduced row echelon form (RREF) of $A-\lambda I$.

Step 3: Combine the necessary sensor locations found for each eigenvalue to finalize the sensor location for full system observability

Step 4: Examine if any sensor is redundant. This is to compute the column rank of $\mathcal{O}$ (Kalman's rank test) with a modified observation matrix. This modification involves removing the ith row in C for all i, one at a time. If the column rank remains n after excluding a sensor, it will be dropped from the necessary sensor list.

The Algebraic Approach ensures full observability. **Step 2** and **Step 3** returns a sensor set that is sufficient for full observability, since Theorem 1 condition will be satisfied. However, it might induce redundant sensors, as shown in the example in Appendix A. **Step 4** will guarantee the sensor locations that are necessary for full observability, without including redundant sensors. In contrast to optimization model-based methods, which often encounter numerical challenges due to their combinatorial nature, this Algebraic Approach is numerically efficient, making it particularly suitable for tackling large-scale problems.

The Algebraic Approach also gives an optimal sensor placement strategy with **minimum number of sensors** for full observability.

Proof. We prove our claim by contradiction. $C \in R^{p\times n}$ is the observation matrix obtained by our Algebraic Approach. Assume there is an optimal observation matrix $C^* \in R^{q\times n}$ $(q < p)$ that uses fewer sensors and yet satisfies the PBH rank test.

Based on **Step 2**, for all λ of A, we have rank $\left(\begin{bmatrix} A-\lambda I \\ C \end{bmatrix}\right) = n$. Also, **Step 4** ensures that removing any row in C will leads to unobservability, i.e., $\exists\lambda_i, \text{rank}\left(\begin{bmatrix} A-\lambda_i I \\ C \end{bmatrix}\right) = n$ and each row in C is linearly independent of the rows in $A-\lambda_i I$. Since C contains orthogonal unit vectors, $A-\lambda_i I$ has rank $n-p$.

For the optimal observation matrix C^*, we also have rank $\left(\begin{bmatrix} A-\lambda_i I \\ C^* \end{bmatrix}\right) = n$. However, the maximum rank of matrix $\begin{bmatrix} A-\lambda_i I \\ C^* \end{bmatrix} = \text{rank}\,(A-\lambda_i I) + \text{rank}\,(C^*) = n-p+q < n$, which leads to a contradiction.

Therefore, we conclude that there cannot exist any C^* that has a fewer dimension than C constructed through steps 1-4, meaning C constructed in our Algebraic Approach is optimal in the sense of a minimal number of sensors used.

□

The Algebraic Approach works for any realization of the parameters in A without imposing restrictions on the structure of the system matrix. Thus, this approach is applicable for any dynamical system, not limited to transportation network density dynamics. The only requirement is that the rows of observation matrix C consist of unit vectors.

Next, we apply our approach to the following example and demonstrate that caution is needed while translating conclusions from other fields to a domain specific context where problem structure and assumptions may differ.

Consider the system matrix $A = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 1 & 2 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$.

Step 1

Compute the eigenvalues of A. The eigenvalues are -1, -2 and 0, all with algebraic multiplicity 2.

Step 2

1. For $\lambda = -1$,

$A - \lambda I = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$, our sensor placement (choice of C) should ensure $\begin{bmatrix} A - \lambda I \\ C \end{bmatrix}$ to be full column ranked. Since each row in C is a unit vector, we can identify the columns that need a leading 1 by using the row reduction. To make $\begin{bmatrix} A - \lambda I \\ C \end{bmatrix}$ full ranked, we can construct the row reduced echelon form of $\text{RREF}\left(\begin{bmatrix} A - \lambda I \\ C \end{bmatrix}\right)$ since elementary row operations do not change the rank.

$\text{RREF}\,(A - \lambda I) = \begin{bmatrix} \mathbf{1} & 2 & 0 & 0 & 1 & 0 \\ 0 & 0 & \mathbf{1} & 0 & 1 & 0 \\ 0 & 0 & 0 & \mathbf{1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{1} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$, for columns without a leading 1 (column 2 and column 5), we need to include a 1 in matrix C. Thus, C has to include rows $\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$, meaning placing sensors on link 2 and 5.

2. For $\lambda = -2$,

$$A-\lambda I=\begin{bmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\1&2&0&0&0&0\\0&0&0&0&0&0\\0&0&1&2&2&0\\0&0&0&1&0&2\end{bmatrix} \text{ and RREF } (A-\lambda I)=\begin{bmatrix}\mathbf{1}&0&0&0&0&0\\0&\mathbf{1}&0&0&0&0\\0&0&\mathbf{1}&0&2&-4\\0&0&0&\mathbf{1}&0&2\\0&0&0&0&0&0\\0&0&0&0&0&0\end{bmatrix}$$

C has to include rows $\begin{bmatrix}0&0&0&0&1&0\\0&0&0&0&0&1\end{bmatrix}$.

3. For $\lambda=0$,

$$A-\lambda I=\begin{bmatrix}-1&0&0&0&0&0\\0&-1&0&0&0&0\\1&2&-2&0&0&0\\0&0&0&-2&0&0\\0&0&1&2&0&0\\0&0&0&1&0&0\end{bmatrix} \text{ and RREF } (A-\lambda I)=\begin{bmatrix}\mathbf{1}&0&0&0&0&0\\0&\mathbf{1}&0&0&0&0\\0&0&\mathbf{1}&0&0&0\\0&0&0&\mathbf{1}&0&0\\0&0&0&0&0&0\\0&0&0&0&0&0\end{bmatrix}$$

C again has to include rows $\begin{bmatrix}0&0&0&0&1&0\\0&0&0&0&0&1\end{bmatrix}$.

Step 3

Combining all the necessary rows in C, we obtain the sensor locations that guarantee full observability. In this example, final choice of C is $\begin{bmatrix}0&1&0&0&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\end{bmatrix}$.

Step 4

Not observing any one of the link 2, 5 and 6 will lead to unobservability (remove the each row on at a time and check the rank of observability matrix). Thus, there is no redundant sensor.

It concludes that for full system observability, sensors are needed on Link 2, 5, and 6 (after the ordering of node indices). Note that the conclusion in (Yuan et al., 2013) of using the maximum geometric multiplicity (which is $\mu(\lambda^M)=2$ in this example) to determine the minimum sensor requirement for full system observability is not applicable here. This is due to the more strict constraint for our observer design, i.e. one sensor can only observe the density of one link.

2.4 Structural observability and inference diagram

Structural observability is an extended concept of observability stemmed from structural controllability, first introduced by Lin (1974). Compared to the exact observability, whose analysis requires knowing the value of matrix A, the structural observability only considers the structure of matrices A and C, meaning whether the matrices' elements are either fixed zeros or independent free parameters.

Definition 2.1 (Structural observablity)**.** The system (A,C) is observable for as least some realizations of parameters, without changing the structure (i.e. the location of zero and nonzero elements) of the system matrix and observation model (A,C).

This concept is particularly relevant to the context of this study for the following reasons. Firstly, for a dynamic traffic system, the exact values of parameters in the system matrix A may change depending on the

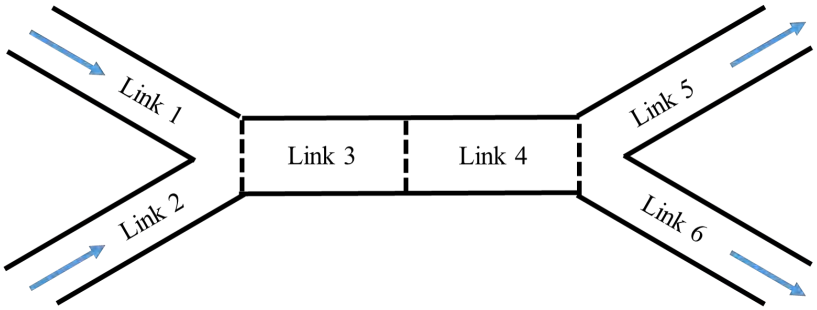


Figure 3: 6-link network example

traffic condition, and one may not be able to always measure the parameters. Secondly, sensor deployment strategies are typically designed for a long-term planning period, during which the exact traffic situation (which determines the exact values of system parameters) may evolve but the connectivity of the physical road infrastructure that would affect the structure of the system is likely to be stable. Thirdly, even though a structurally observable system may not be observable in theory, for such occasion to happen in a dynamic traffic network is rare (ex. 2 merging links have exactly the same free-flow travel speed).

Structural observability/controllability is found to be closely related to the graph theory (Lin, 1974; Liu et al., 2011, 2013). We adopt the same method in Liu et al. (2013) to construct an **inference diagram** from the system matrix A.

Definition 2.2 (Inference diagram)**.** Each state variable is considered as a node in the inference diagram and we draw a directed edge from node i pointing to node j if x_j appears in x_i's state update equation ($a_{ij} \neq 0$ in A).

The edge indicates that the information of state j is captured by state i.

Let us use a a 6-link toy network (shown in Figure 3) as an example to show how we construct the inference diagram. This toy network, even though is small, has a linear junction, a merge, and a diverge, which are building blocks of a highway network.

Consider a mode where link 5 is congested and all other links are uncongested. In this case, the downstream links of uncongested links capture information about the upstream links. Link 5 is congested and the flow from link 4 and 5 is determined by supply of link 5, which is a function of link 5 density. The inflow for link 6 is thus also a function of link 5 density. The inference diagram corresponding to this mode, and the matrix A, with the (i, j)th element denoted as a_{ij}, are plotted in Figure 4. Apparently, different traffic modes can result in different matrix A and inference diagram, as demonstrated by the examples given in Appendix B.

Definition 2.3 (Self-edge)**.** A self-edge is an edge starting from a node and pointing to the same node.

In the above example, node 1,2,3,5 have a self-edge. In some literature, self-edge is called a self-loop.

Definition 2.4 (Reachable node)**.** A reachable node in the Inference Diagram is a node such that there exists a sequence of adjacent nodes (i.e. walk) from a given observed node (such as a sensor location) to it. A k-step reachable node is a node that is reached from an observed node by a walk with k edges.

Using Figure 4 as an example, node 1 is reachable from node 4. It is 2-step reachable from node 4 through edge (4,3) $\rightarrow$ edge (3,1). Also, it is 3-step reachable from node 4 through edge (4,3) $\rightarrow$edge (3,3) $\rightarrow$ edge (3,3), or edge (4,3) $\rightarrow$ edge (3,1) $\rightarrow$ edge (1,1).

$$A = \begin{bmatrix} a_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 & 0 & 0 \\ 0 & 0 & a_{43} & 0 & a_{45} & 0 \\ 0 & 0 & 0 & 0 & a_{55} & 0 \\ 0 & 0 & 0 & 0 & a_{65} & 0 \end{bmatrix}$$

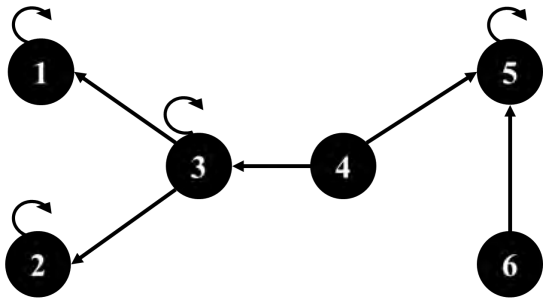


Figure 4: Inference diagram for the 6-link network

2.5 A Graphical Approach to identify observable components

The existence of self-edges in an inference diagram (see Proposition 1 and its proof) is a special feature in the dynamic traffic network context, which may not be true for general complex networks. The existence of self-edges greatly increases observability, as we shall see later.

Proposition 1. *In an inference diagram of a traffic system, any node with an incoming edge also has a self-edge.*

Proof. A node with an incoming edge implies the flow between this node and its neighbor(s) is determined by this node's own density. Due to the flow conservation, this flow will also act on the node itself, causing a self-edge. □

As an example, in Figure 4, the nonzero entry a_{43} of system matrix A indicates the change of density of link 4 is a function of density of link 3. This implies the flow between link 4 and 3 is determined by the density of link 3. This flow also affects link 3 itself, leading to a nonzero entry a_{33} in A. Consequently, we observe a self-edge in the inference diagram, reflecting the influence of link 3's density on its own dynamics.

We are now ready to show the following important theorem to identify observable states (network link densities):

Theorem 2. *Based on the inference diagram, any reachable nodes from observed nodes are also (structurally) observable. Moreover, if each node has at most one outgoing edge (excluding self-edge), the observability is exact.*

Proof. Each row of observation matrix C is a unit vector. Assume the ith row of C is unit vector $\boldsymbol{e}_k^T$ where all the elements are zero except for the kth element. The ith sensor is used to observe the kth state. Consider the observability matrix $\mathcal{O}_i = \begin{bmatrix} C_{i,:} \\ C_{i,:}A \\ C_{i,:}A^2 \\ \vdots \\ C_{i,:}A^{n-1} \end{bmatrix}$ where $C_{i,:}$ is the ith row of C (i.e., $\boldsymbol{e}_k^T$). Since multiplying $C_{i,:}$ to a matrix is simply selecting the kth row of the matrix, the rows of matrix $\mathcal{O}_i$ contain kth row of $A, A^2, \ldots, A^{n-1}$, leading to a matrix whose columns are $\mathbf{0}$ for unreachable nodes starting from node k. In the case where each node has at most one outgoing edge, the leading nonzero element positions of the nonzero columns are different. Therefore, the nonzero columns are linearly independent.

In the general case, due to the existence of the self-edges, entries under the first nonzero entry will not vanish for each column. The structural assumption on our system leads to linearly independent nonzero columns (more discussion on this is provided in the remark below). Thus, the nonzero linearly independent columns of $\mathscr{O}_k$ form a full column rank observability matrix, representing an observable subsystem consisting of the observed node and its reachable nodes. This is true for any k. The observed nodes and the consequent reachable nodes are the observable components. For unreachable nodes, we can consider them as if they are not part of the system and they will not affect the observability of the reachable nodes. □

Remark on structural assumption: Readers familiar with the complex systems literature may recall that the structural assumption leading to the conclusion that the nonzero columns are linearly independent was based on the assumption that all entries in matrix A are independent variables. This is certainly not the case in the traffic dynamic network case. For example, consider the problem shown in Figure 4, $a_{11} = -\frac{v_1}{l_1}$ and $a_{31} = \frac{v_1}{l_3}$ where v_1 is the free-flow travel speed on link 1 and l_1, l_3 are the link lengths of link 1 and link 3, respectively. Regardless of the choice of the free parameter v_1, entries a_{11} and a_{31} are dependent: $a_{11} = -\frac{l_3}{l_1} a_{31}$. With these dependent elements in A, can we still assume linear independence of the nonzero columns in the observability matrix? We answer this question next.

Proposition 2. *We denote the inference diagram as graph $G(V,E)$. $\forall (i_1, j_1), (i_2, j_2) \in E$, the corresponding entries in A, $a_{i_1 j_1}$ and $a_{i_2 j_2}$ are independent parameters given $j_1 \neq j_2$.*

Proof. For nonzero parameters $a_{i_1 j_1}$ and $a_{i_2 j_2}$, the flows between (i_1, j_1) and (i_2, j_2) are functions of the link densities of j_1 and j_2 respectively. Consequently, the parameters $a_{i_1 j_1}$ and $a_{i_2 j_2}$ are solely dependent on the fundamental diagrams of different links j_1 and j_2, each possessing independent characteristics such as free-flow travel speed. Thus, the two parameters $a_{i_1 j_1}$ and $a_{i_1 j_2}$ are independent. □

To prove Theorem 2, we claimed that the structural assumption will make the column vectors linearly independent in the observability matrix $\mathscr{O}$. It is evident that if all the nonzero leading entries of the columns are positioned in distinct rows, these columns will be linearly independent. The problem remains to verify for the case where the nonzero leading entries of two columns reside in the same row, and the columns can still be linearly independent. The (i, j)th entry of A^k is the sum of the products of all weights of all walks from node i to node j of length exactly k. For each of the two columns (denoted by j_1 and j_2) in $\mathscr{O}$ to be verified, assuming they are k-step reachable, the leading nonzero entries is the product of all weights of the k-length walk from the observed node (denoted by i) to the column node. Thus, **the leading nonzero entries for column j_1 and j_2 are functions of a_{ij_1} and a_{ij_2}** respectively, which are independent parameters based on Proposition 2. The entries under the leading nonzero entries are the products of the weights in $(k+1)$-length, which are functions of $a_{ij_1} \cdot a_{j_1 j_1}$ and $a_{ij_2} \cdot a_{j_2 j_2}$. Since Proposition 1 implies $a_{jj} \neq 0$ if $a_{ij} \neq 0$, moreover, a_{ij} and a_{jj} are dependent parameters, scaled by a factor, **the two entries under the leading nonzero entires of column j_1 and j_2 are functions of $a_{ij_1}^2$ and $a_{ij_2}^2$**. Therefore, there exist choices of parameters a_{ij_1} and a_{ij_2}, such that columns j_1 and j_2 are linearly independent.

Remark on the importance of self-edges: In the example of 6-link network with the inference diagram of Figure 4, if a sensor is placed on link 4, the observability matrix is then in the following form, where we

use $\times$ to indicate a nonzero entry:

$$\begin{bmatrix} 0 & 0 & 0 & \times & 0 & 0 \\ 0 & 0 & \times & 0 & \times & 0 \\ \times & \times & \times & 0 & \times & 0 \\ \times & \times & \times & 0 & \times & 0 \\ \times & \times & \times & 0 & \times & 0 \\ \times & \times & \times & 0 & \times & 0 \end{bmatrix} \begin{array}{l} \longleftarrow C_{i,:} \text{ (note: link 4 is directly observed by the sensor)} \\ \longleftarrow C_{i,:}A \text{ (links 3 and 5 are 1-step reachable from link 4)} \\ \longleftarrow C_{i,:}A^2 \text{ (links 1, 2, 3 and 5 are 2-step reachable from link 4)} \\ \longleftarrow C_{i,:}A^3 \\ \longleftarrow C_{i,:}A^4 \\ \longleftarrow C_{i,:}A^5 \end{array}$$

The structural observability assumption leads to the conclusion that the nonzero columns are linearly independent. Thus, the observability matrix of the subsystem consisting of link 1,2,3,4, and 5 has full column rank. This subsystem will be observable. Now imagine if the inference diagram did not have the self-edges. All the nodes in the graph that are previously k-step reachable are no longer $(k+1)$-step reachable, leading to the following observability matrix:

$$\begin{bmatrix} 0 & 0 & 0 & \times & 0 & 0 \\ 0 & 0 & \times & 0 & \times & 0 \\ \times & \times & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{array}{l} \longleftarrow C_{i,:} \\ \longleftarrow C_{i,:}A \\ \longleftarrow C_{i,:}A^2 \\ \longleftarrow C_{i,:}A^3 \\ \longleftarrow C_{i,:}A^4 \end{array}$$

where the first 5 columns can never be linearly independent no matter what values the non-zero elements take. This example shows the importance of the self-edges in increasing the observability.

In summary, Theorem 2 provides an efficient way to identify the observable subspace of the states - the number of reachable nodes in the inference diagram measures the observability, and the reachable nodes correspond to the observable states. This is important since it provides an explicit quantification of the quality of observability (unlike other metrics such as observability Gramian). It enables us to evaluate a sensor placement strategy for a partially observable system. A natural follow-up result is that to fully observe all link densities, traffic sensors should be located at links corresponding to the source nodes (nodes that have only outgoing edges) in the inference diagram. The result is consistent with the graph-theoretical approaches developed in Liu et al. (2011) and Liu et al. (2013): observing the source nodes is indeed the solution applying the maximum matching algorithm. Note that the graph-theoretical approaches developed in Liu et al. (2011) and Liu et al. (2013) find sensor location for full observability only. Our finding on the partial observability property is blessed by the special graphical properties of the traffic density dynamics. To our knowledge, this is the first study to reveal the partial observability property of the dynamic traffic system in the literature.

2.6 A mathematical programming model maximizing number of observable components under various traffic conditions

In the above sections, we have established algebraic and graphical approaches for observability analysis of an LTI traffic system. However, the dynamics of traffic network density can only be modeled by an LTI dynamical system for short periods of time with consistent traffic conditions. A single set of linear equations is inadequate for capturing the evolving nature of traffic conditions within networks. In this

section, we integrate observability and mode switching concepts in a mathematical programming modeling framework to maximize observable components under various traffic conditions.

Let us first explain the mode switching concept. Due to the min function to determine the actual flow in Eq (4), (5) and (6), the system is nonlinear. However, with a triangular fundamental diagram, the whole system can be considered as a piecewise linear system, switching within different linear models, which we call **modes**. Using Fig 2a as an example, when the two links are free-flow with length 1 and the demand of link 1 is less than supply of link 2, the system dynamics is described as:

$$\begin{bmatrix} \dot{k_1} \\ \dot{k_2} \end{bmatrix} = \begin{bmatrix} -v_1 & 0 \\ v_1 & 0 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} + \begin{bmatrix} f_1 \\ -g_2 \end{bmatrix} \tag{11}$$

When link 2 becomes congested, the supply of link 2 is less than the demand of link 1, the system dynamics then evolves to:

$$\begin{bmatrix} \dot{k_1} \\ \dot{k_2} \end{bmatrix} = \begin{bmatrix} 0 & w_2 \\ 0 & -w_2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} + \begin{bmatrix} f_1 - w_2 k_{j,2} \\ -g_2 + w_2 k_{j,2} \end{bmatrix} \tag{12}$$

In a more concise form, the above system with two modes can be expressed as a piecewise affine system:

$$\dot{\boldsymbol{x}}(t) = A(\lambda_t)\boldsymbol{x}(t) + \boldsymbol{b}(\lambda_t) + \boldsymbol{u}(t) \tag{13}$$

where $\boldsymbol{x}$ is the continuous state variables, $\boldsymbol{u}(t) = \begin{bmatrix} f_1 & -g_2 \end{bmatrix}^T$ is the boundary flow vector. A and $\boldsymbol{b}$ are mode specific, which switch among K different modes $\lambda_t \in \{1, 2, \cdots, K\}$. In Appendix B, we show a bigger example of mode switching in the 6-link network.

Now we are ready to present our sensor optimization model that considers a wide spectrum of traffic modes. Let the binary decision variable z_i control the sensor location and the auxiliary binary variable $x_{k,i}$ indicate the inferred observability.

$$\text{let } z_i = \begin{cases} 1, \text{if a sensor is put on link } i, \\ 0, \text{otherwise} \end{cases}, x_{k,i} = \begin{cases} 1, \text{if link } i \text{ is observable in mode } k, \\ 0, \text{if link } i \text{ is not observable in mode } k \end{cases}$$

where $i = 1, 2, ..., n$ and $k = 1, 2, ..., K$, K is the total number of different modes of the system under consideration. Note that in a typical decision environment, K does not need to include all possible modes; one may only include representative modes that are more likely to appear. The importance of each mode k may be expressed by a non-negative weight factor w_k. In practice, this can refer to the frequency of occurrence of a mode. A practical way of obtaining major modes and their occurrence rates is introduced in the case study in Section 3.3.

We then have the following mixed-integer program formulation:

$$\max_{z,x} \sum_{k=1}^{K} w_k \sum_{i=1}^{n} x_{k,i} \tag{14a}$$

$$s.t. \sum_i z_i = p \tag{14b}$$

$$x_{k,i} \leq \sum_{(i^-,i) \in \delta_k^-(i)} x_{k,i^-} + z_i, \forall k, i \tag{14c}$$

$$z_i \in \{0, 1\}, \forall i \tag{14d}$$

$$x_{k,i} \leq 1, \forall k, i \tag{14e}$$

where $\delta_k^-(i)$ is the set of incoming edges of node i (here, the self-edges are **not** considered as incoming edges) in the inference diagram of the k-th mode. The objective function (14a) maximizes the average number of observable states across all possible modes, in a weighted sense. Constraint (14b) specifies the total number of sensors to be placed where p is pre-specified. Constraints (14c) impose that a node is not observable if it is not directly observed or it has no observable predecessor nodes in the inference diagram. One might notice that x is not enforced to be binary in the formulation. This is not a mistake as the objective function and constraints will naturally push x to be binary. With a non-negative weight $w_{k,i}$, $x_{k,i}$ will be binary given constraints (14c) and (14e). This could provide computational advantages because handling integer constraints typically adds burden to large-scale linear programming problems.

It is beneficial to recognize that the mathematical model (14) is indeed a stochastic program when the weight factor w_k is set to be probability of mode k. Recall a general two-stage stochastic nonlinear program (Ruszczyński and Shapiro, 2003)

$$\begin{aligned} &\min_x && L(x) + E_\xi[Q(x,\xi)] \\ &\text{s.t.} && g_i^1(x) \le 0 \\ &\text{with} && \\ &Q(x,\xi) = \min_y && f(y;x,\xi) \\ &\text{s.t.} && g_i^2(y;x,\xi) \le 0 \end{aligned} \tag{15}$$

Typically, the goal can be interpreted as to make a planning decision that minimizes the total costs including the current and the expected future costs. The first stage planning decision x with cost $L(x)$ has to be made before any possible outcome of the random parameter ξ becomes certain. In the second stage, the actual realization of ξ becomes known and a recourse decision y can be taken. $Q(x,\xi)$ is the objective function of the second stage problem given a particular choice of x and a realization of ξ.

In the current context, each mode k corresponds to an uncertain scenario with a probability of w_k. The sensor placement decision z corresponds to the planning decision. Note we do not impose a sensor deployment cost, so there is no $L(x)$. Instead, we limit the budget of sensor deployment by imposing a hard constraint on the total number of sensors. The observability of links x corresponds to the operational decisions in a classic stochastic programming (SP) framework. The second-stage subproblem $Q(x,\xi) = \min_y f(y;x,\xi)$ in the generic formulation (15) becomes $Q(z,k) = \max_x \sum_{i=1}^n x_{k,i}$ subject to the feasibility constraints of x (14c and 14e) in the current context.

We make an association of our model with classic SP framework so that one can exploit the very rich SP literature such as the various decomposition approaches (Rockafellar and Wets, 1991; Carøe and Schultz, 1999; Collado et al., 2012) developed for solving large-scale SP models if needed, even though numerical treatment is not an emphasis of this study.

2.7 Notes on computation of the Algebraic Approach exploiting the network structure

In the Algebraic Approach, the first step is to calculate the eigenvalues of the system matrix A based on the special structure of A for traffic network dynamics. The following assumption is made for the inference diagram we construct for the system dynamics:

$$A = \begin{bmatrix} a_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{44} & 0 & 0 \\ 0 & 0 & a_{53} & a_{54} & 0 & 0 \\ 0 & 0 & 0 & a_{64} & 0 & 0 \end{bmatrix}$$

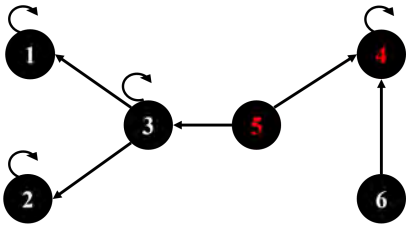


Figure 5: 6-link network example: after the topological ordering

Assumption: The inference diagram of the traffic density dynamics is an acyclic graph.

This assumption holds true if the information does not flow in one direction that forms a loop. If the physical network does not contain any cycle, this assumption will be true. Even if it contains cycles, the information is unlikely to flow in one direction (all links are free-flow or congested), which is also a case that we are not very interested in.

With this assumption, our Algebraic Approach has some advantageous properties.

Proposition 3. *Since the inference diagram is acyclic, there exists a topological ordering such that the matrix A is lower-triangular. Moreover, the eigenvalues are simply the diagonal elements of A.*

A (reverse) topological ordering is a vertex ordering for a directed acyclic graph such that for every directed edge (i, j) from vertex i to vertex j, i comes after j in the ordering. After the topological sort, $a_{ij} = 0$ if $i < j$ and A is lower-triangular. This is for analytical convenience only, as the eigenvalues are the diagonal elements for an upper or lower triangular matrix.

We implement the (reverse) topological ordering and obtain matrix A as an lower triangular matrix. For the 6-link network example in Figure 3, the reordered graph and its system matrix A is shown in Figure 5. The node 4 and 5 are switched to satisfy the ordering requirement. Matrix A is now an lower triangular matrix.

3 Numerical Examples

This section includes numerical experiments on three networks, with a purpose of showcasing the applicability of the proposed observability measures and sensor placement approach under different network settings.

In all examples, observability is quantified as the average number of observable components across all possible modes, computed in Eq.(14a). In subsection 3.1, a small toy network is used to illustrate the idea of mode switching as the system gets congested. This example also showcases that better observability leads to better estimation quality even when the actual traffic dynamics deviates from the LQM model on which the observability analysis is based. In subsection 3.2, we study the observability in a larger network to further investigate the benefit of higher observability considering various possible modes for state estimation. In subsection 3.3, we propose a practical solution for obtaining the mode occurrence information using readily available data sources (Google Maps data in this case) even before we install any sensor in the network.

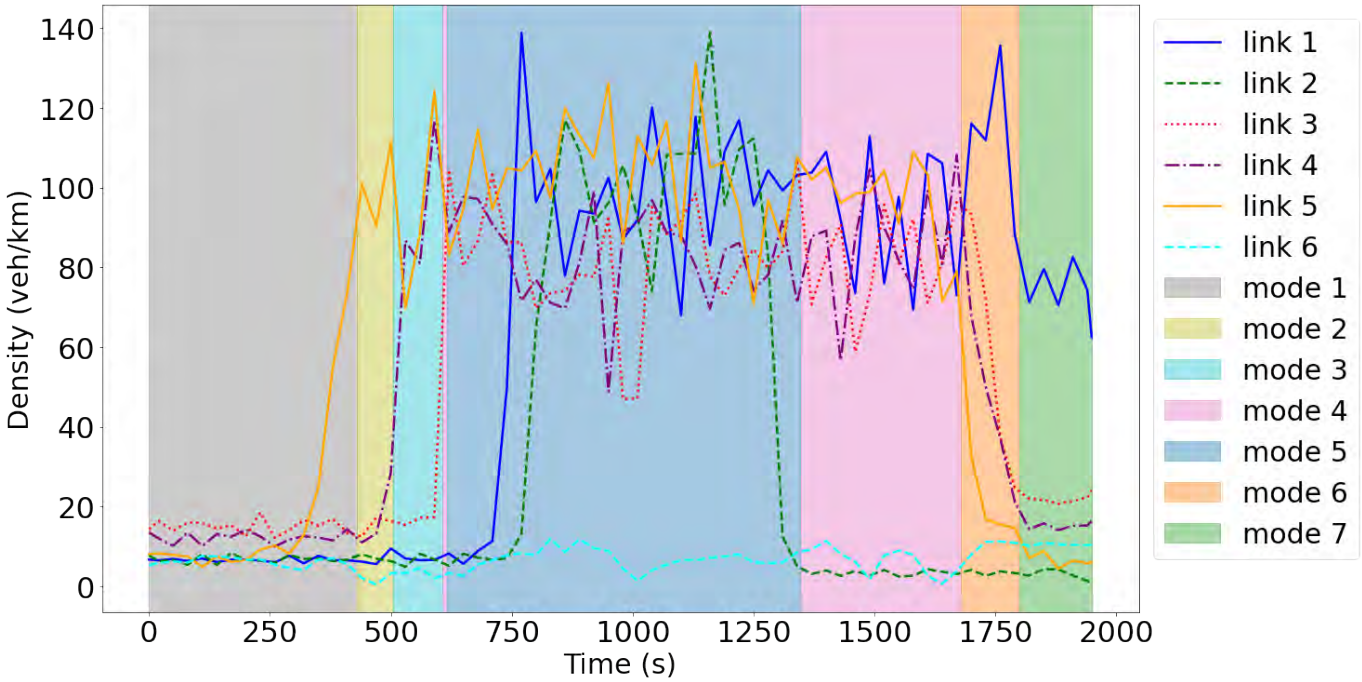


Figure 6: Link densities and mode switches in the 6-link network experiment

3.1 A 6-link toy network

Let us first start with the aforementioned 6-link network example. The traffic is simulated using the simulation software SUMO. The ground truth data is intentionally generated using simulation rather than the LQM to answer the following questions: What if the traffic network dynamics does not exactly follow the LQM, which after all is a simplification of the reality? Is the observability property calculated based on the mode-switching link queue model still meaningful?

In the simulation, each link is set to be 200 meters long. Link 1,3,4,5 are considered as the mainstream corridor and have higher free-flow travel speeds than link 2 and 6. The outflow at link 5 is intentionally controlled to generate congestion that propagates to link 1, 2, 3, and 4. Starting from free-flow condition, the queue spillback generates 7 distinct modes. The evolution of link densities and the corresponding mode switches in this experiment is illustrated in Figure 6. During the simulation, boundary flows are captured by additional loop detectors at the immediate upstream/downstream of the 6-link network. Real-time occupancy data collected at the midpoint of each link is then converted to density, serving as the ground truth. Note that the ground truth simulated by SUMO is independent of our choice of the macroscopic traffic model.

Let us first look at the results showing how different sensor location strategies could impact the observability outcome. We set the number of sensors to be deployed as 4. By solving the optimization problem in (14), we identify the optimal sensor locations as links 1, 2, 5 and 6, which leads to 5.81 observable links on average. This optimal sensor deployment strategy is compared with all other possible sensor location choices, which lead to an average observability ranging between 4.33 and 5.62. Detailed result is reported in Table 1.

Next, we address possible questions that reader might be wondering: the definition and analysis associated with observability seem to be centered around a piecewise linear dynamic traffic model, which

might deviate far from the reality. Besides serving our intellectual curiosity in a theoretical sense, does the established obervability understanding have any practical meaning? The following experiment shows that increased observability leads to better estimation quality, which indicates that a sensor placement strategy that maximizes observability should benefit downstream traffic state estimation applications.

Among the many traffic state estimation methodologies (Seo et al., 2017), we adopt the Interacting Multiple Model (IMM) filtering, which has demonstrated its effectiveness in traffic state estimation, both in urban and freeway traffic systems (Thai and Bayen, 2014; Panda et al., 2019; Zhang and Mao, 2015; Wang and Work, 2014). The IMM filtering is a powerful yet cost-effective filtering algorithm with the capability of estimating the state of a dynamic system that may switch across different traffic modes, which makes it a suitable choice for our needs. To implement the IMM filtering, one needs to describe the stochastic process to be estimated. In this example, the stochastic process is described following stochastic LQM to incorporate process noise and measurement noise:

$$\boldsymbol{x}_{t+1} = A_{\lambda_t}\boldsymbol{x}_t + \boldsymbol{b}_{\lambda_t} + \boldsymbol{u}_t + \boldsymbol{\omega}_t \tag{16}$$

$$\boldsymbol{y}_t = C\boldsymbol{x}_t + \boldsymbol{\nu}_t \tag{17}$$

where $\boldsymbol{x}$ is the discretized state variables, whose dynamics is switching among a number of K different modes $\lambda_t \in \{1,2,\cdots,K\}$. $\boldsymbol{\omega}_t$ and $\boldsymbol{\nu}_t$ are zero-mean Gaussian noises with known covariance. This model is similar to the one in Thai and Bayen (2014). The covariance of $\boldsymbol{\omega}_t$ and $\boldsymbol{\nu}_t$ are calibrated based on the SUMO data. In IMM filtering, at each time step, the model is no longer in a single true mode; instead, it is a probabilistic combination of all possible modes (K different modes). This filtering algorithm dynamically recalibrates and updates the likelihood of different modes for the current state based on the observed data. The filtering operates in parallel, employing Kalman filters for each mode, and the resulting individual estimates are combined to produce the final estimate. For a comprehensive understanding of the detailed steps involved in IMM filtering, interested readers are directed to Thai and Bayen (2014) and Panda et al. (2019). To maintain brevity, this paper refrains from duplicating these exhaustive details.

Table 1 compares the degree of observability with the estimation quality, measured by the mean absolute percentage error (MAPE), under different sensor location strategies. From these results, we hope to better understand the correlation between observability and state estimation. This is not trivial because good observablity does not necessarily guarantee good estimation quality since estimation also involves the identification of the mode switches – for estimation, the exact mode switching time and the true mode remain unknown and are inferred simultaneously with the state estimation; the only given information are the boundary flows, fundamental diagrams, and the covariance of the Gaussian noise term.

We observe that sensor placement with higher observability generally results in better estimation quality. Conversely, instances of poor observability can lead to substantial MAPE, primarily because loss of observability leads to uninformed guesses of the state and a lack of correction. For example, our IMM algorithm does not enforce the link density to be in the range from zero to jam density. When a state is unobservable under the current system mode, the measurement lacks information about the unobservable state. Consequently, the estimation quality for the unobservable states is very poor. Let us take a closer look at the case of sensor placement (1,2,3,4) as shown in Figure 7. During the period when both link 5 and 6 are unobservable, the estimation error consistently grows. When link 5 becomes observable at 430 seconds, the estimation of link 5 density now can be corrected by the measurement and the estimated density aligns

Table 1: Different sensor locations comparison

Sensor location	Observability	MAPE (%)
(1, 2, 5, 6)	5.81	17.21
(1, 2, 4, 6)	5.62	33.15
(1, 3, 5, 6)	5.56	25.37
(1, 4, 5, 6)	5.49	31.34
(2, 3, 5, 6)	5.29	385.24
(1, 3, 4, 6)	5.28	46.51
(2, 3, 4, 6)	5.00	396.32
(1, 2, 3, 6)	4.96	72.91
(1, 2, 3, 5)	4.96	384.03
(1, 2, 4, 5)	4.94	206.92
(2, 4, 5, 6)	4.94	24.99
(1, 2, 3, 4)	4.68	517.96
(3, 4, 5, 6)	4.66	301.70
(1, 3, 4, 5)	4.60	550.70
(2, 3, 4, 5)	4.33	1224.88

closely with the actual value. Subsequently, when link 5 becomes unobservble again, the estimation quality soon deteriorates. While for link 6, since it is not observable throughout the entire period, its estimated state is consistently poor. Thus, although observability and state estimation are two different concepts, we demonstrate that it is advantageous to place sensors in a way that maximizes the average number of observable components. This holds true even when the mode switching time and identification are not provided.

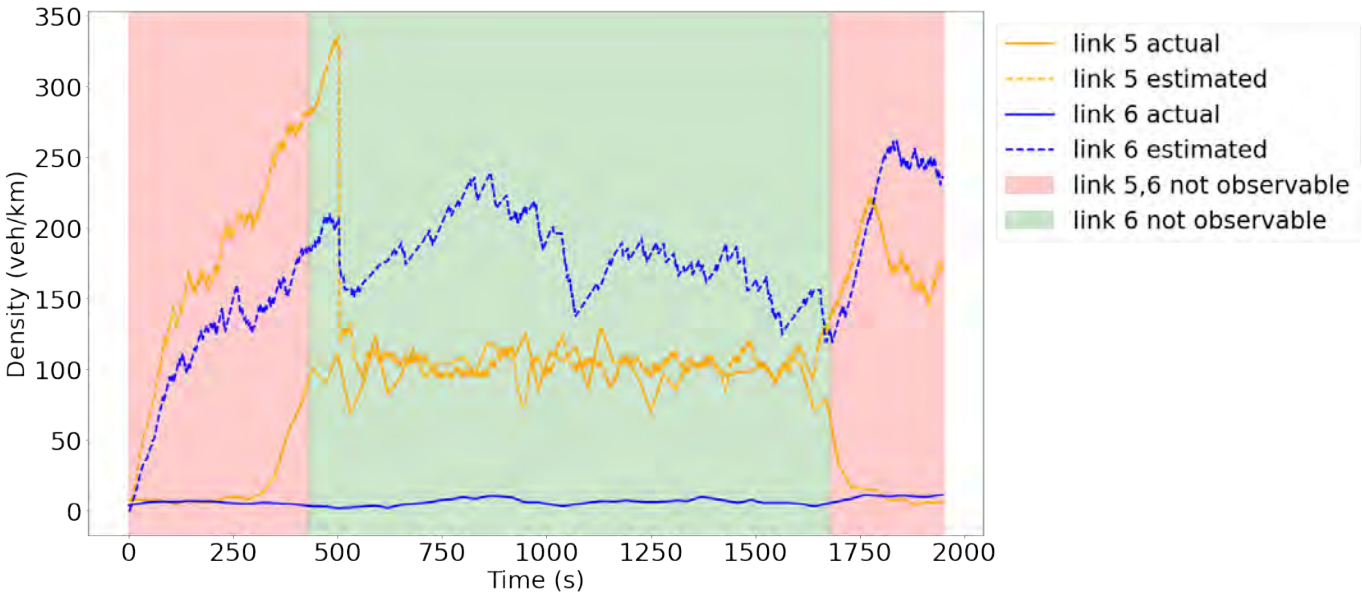


Figure 7: Estimation with sensor located on link 1,2,3,4

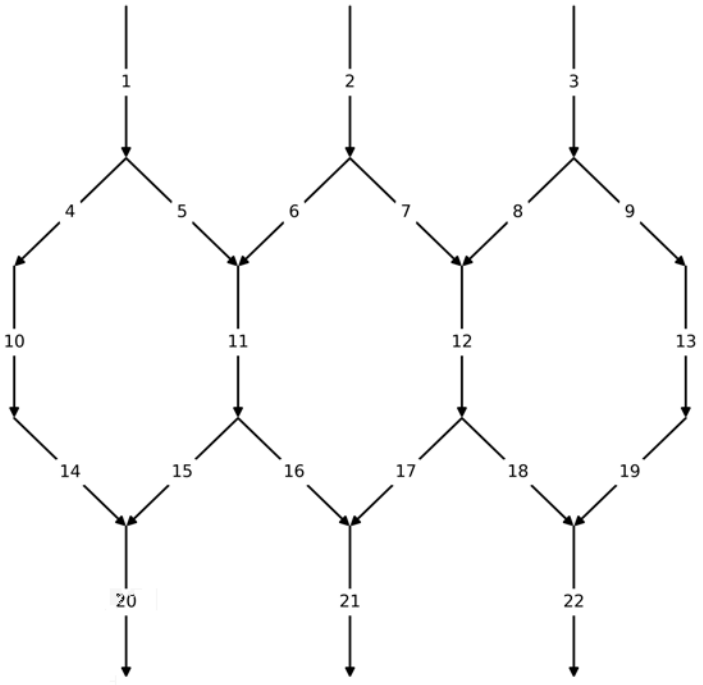


Figure 8: Numerical example: a 22-link network

3.2 A fabricated 22-link network

Now let us show a numerical example using the network used in Rinaldi (2018) (see Figure 8). The experiment is set up as follows. The traffic is simulated based on Eqs. (16,17). We acknowledge that simply adding a Gaussian process noise to the LQM indeed leads to an oversimplified stochastic traffic flow model. For more sophisticated stochastic traffic modeling, interested readers can refer to Jabari and Liu (2013) and Sumalee et al. (2011). Our purpose here is not to develop a stochastic traffic model, but rather to use one to generate traffic ground truth that deviates from the LQM used in our analysis.

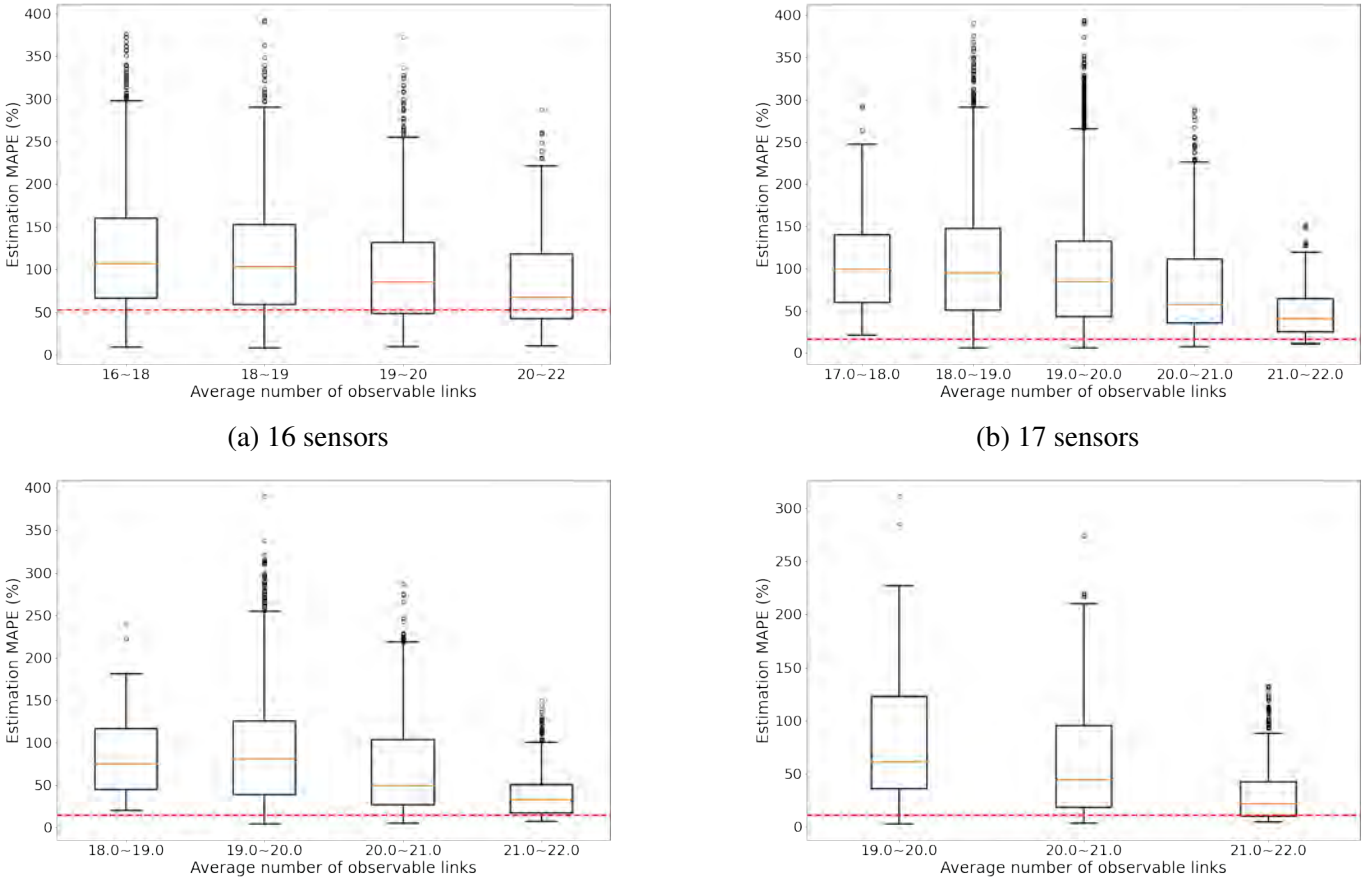
We intentionally controlled boundary flows to generate 585 different congestion modes. Links 10 and 11 are set to be bottlenecks. Modes are weighted based on the duration of their occurrence. Details of the parameter setup of the experiment can be found at `https://github.com/Xinyue-H/Sensor-Location-Optimization`.

First, let us explore the performance of different sensor location strategies in terms of both observability and estimation quality. The total number of sensors takes a value between 16 and 19. The optimal sensor location solution for each p is obtained by solving the optimization problem in (14). The optimal sensor location solutions and their resulted average observabilities and MAPE are reported in the Table 2. To generate different sensor location strategies for comparison purpose, we enumerated all sensor location combinations for p =17, 18, or 19. For $p = 16$, we sampled 6000 possible sensor location combinations, 1500 in each bin. Figure 9 reports the estimation errors from different sensor location strategies that are categorized based on their average observability. Each box plot shows the max, 75% quartile, median, 25% quartile, and the min MAPE resulted from various sensor location solutions within the corresponding

observability bin[3]. The red dashed line corresponds to the optimal sensor location solution. Using $p = 16$ as an example, our optimal sensor location leads to an average MAPE around 54%.

Table 2: Optimal sensor placement solutions

Sensor Num	Sensor location	Observability	MAPE (%)
16	(1, 2, 3, 5, 6, 7, 8, 9, 14, 15, 16, 17, 18, 19, 20, 21)	21.52	53.69
17	(1, 2, 3, 5, 6, 7, 8, 9, 14, 15, 16, 17, 18, 19, 20, 21, 22)	21.74	16.97
18	(1, 2, 3, 5, 6, 7, 8, 9, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22)	21.90	14.86
19	(1, 2, 3, 5, 6, 7, 8, 9, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22)	21.96	11.41



(a) 16 sensors

(b) 17 sensors

(c) 18 sensors

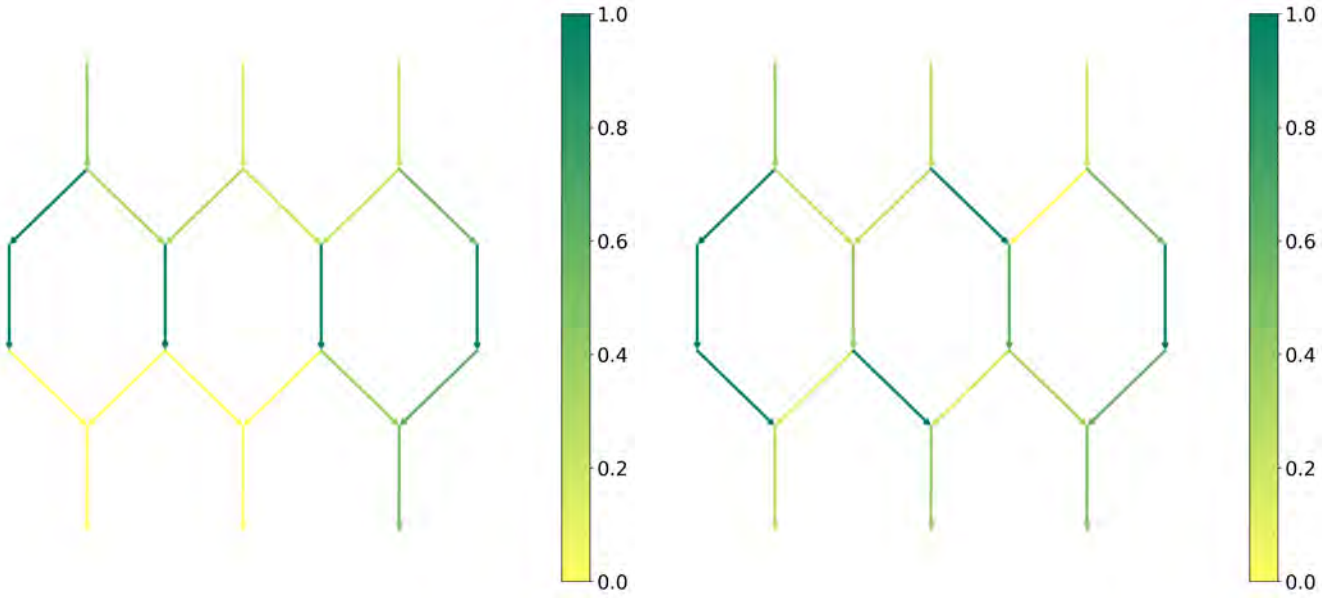
(d) 19 sensors

Figure 9: Impact of observability on estimation quality

Based on the results in Figure 9, we have the following findings:

- The optimal sensor location solutions from our max-observability sensor location model perform consistently well in terms of traffic state estimation measured by MAPE.
- Overall, sensor placement with better observability tends to result in improved estimation quality, indicated by lower MAPE values.

[3]For the case of 16 sensors, since there are very few sensor location strategies resulted in an observability between 16-17 and 21-22, the bins are merged with 17-18 and 20-21, respectively.



(a) Naive solution based on path-covering rule (b) Optimal sensor location solution

Figure 10: Comparison of observability resulted from two sensor location solutions

- Sensor placement with high observability is associated with a reduced risk of extreme inaccuracies in estimation.

These results indicate that maximizing observability across various traffic conditions should be a key consideration in the sensor placement problem due to its demonstrated significance in dynamic traffic state estimation, which is a key for real-time traffic surveillance.

In the experiment above, the number of sensors is relatively large. Readers might wonder whether the concept of obervability still matters in case one has only a small number of sensors to deploy. Now, let us show an example of placing only 4 sensors in the 22-link network. By solving the optimization problem in (14) with $p = 4$ and the 585 modes being incorporated, we identify the optimal sensor locations as links 7, 13, 14, 16, which leads to 11.76 observable links on average. This max-observability sensor location strategy is compared with a naive choice of placing sensors on links 10, 11, 12 and 13 to cover all four paths connecting the three OD pairs. The naive choice leads to 8.89 observable links on average. Figure 10 provides more details of the comparison of the two sensor location choices. Darker green indicates better observability. If a link is observable all-time, the corresponding value should be 1. This example shows that even with very few sensors to work with, strategically placing them still makes a significant difference.

Finally, let us show an example where sensor location solutions to ensure structural observability vs. exact observability may differ. Note that in this example the parameters of some links are intentional altered (to set to be identical) to create symmetry, with the intention to generate a case where structural observability differs from observability. Since exact observability requires knowledge of the exact values of parameters, we will implement the experiment in the "most likely" mode (the one that occurs the most during the simulation). The inference diagram and the necessary sensors are shown in Figure 11. For structural observability, our Graphical Approach concludes only the source nodes (link 1, 2, 3, 6, 9, 20, 21) have to be observed, as shown in green. However, for exact observability considering the link weights, extra sensors must be put on link 15 and link 17 (colored in blue). In link pairs (14, 15) and (16, 17), the individual links within a pair cannot be distinguished due to symmetry with only the upstream being

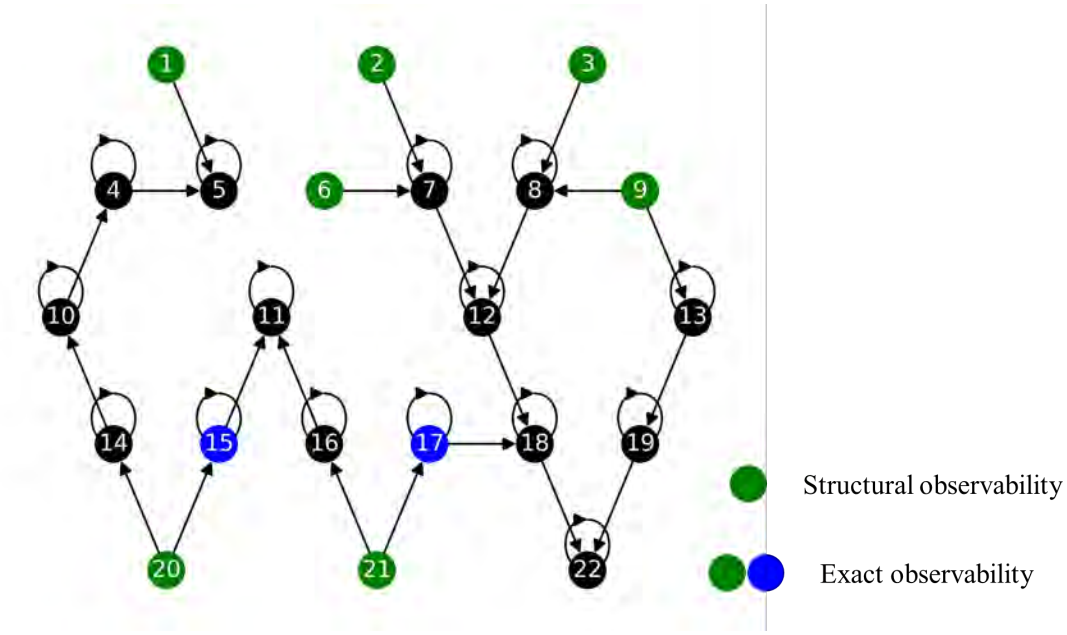


Figure 11: Sensor placement for exact observability for one of the modes

observed. As a result, extra sensors are needed on at least one individual component within a pair.

3.3 A real-world case study on I80 East with Google Maps data

In this section, we present a case study conducted using real traffic data collected on the I80 East Davis - Sacramento highway corridor, which experiences regular afternoon peak congestion. This case study is included to demonstrate how one might implement our model in practice, including how to prepare information regarding traffic modes and their weights that is required in the sensor location model.

The highway network consists of 21 mainstream links and 11 on- and off-ramps that form merging and diverging junctions, as depicted in Figure 12. The 10.6-mile long mainstream is segmented into links that are separated using the orange pins on the map. We collected real-time travel speed information through the Google Maps API every 10 minutes from 2 PM to 7 PM, Monday through Thursday, throughout the months of April to June 2023. The speed profiles for three days in June are visualized in Figure 13. We observe a recurring pattern of congestion propagation every day and the data serves as the prior knowledge of the various traffic conditions. The weight factor w_k is obtained based on the occurrence frequency of the conditions. From all the traffic modes, we choose the ones that occurred more than 10 times. A total of 47 different modes are selected, which takes approximately 44% of all-time occurrence. Note that even though we have link speed information as ground truth, it does not directly translate to link density information as there is not a one-to-one relationship between speed and density under the uncongested region. Therefore, the question of observing link density is not trivial.

We solve the optimization problem in (14) to obtain the best sensor location solutions under various p. As an illustration, the optimal sensor locations for $p = 3, 6, 9$ are plotted in Figure 14. Links 4 and 7 are of highest priority since there is usually free-flow traffic before link 4 and congested traffic after link 7. Off-ramps are also prioritized since their information is not captured by the mainstream when the main congestion source is the downstream mainstream segment.

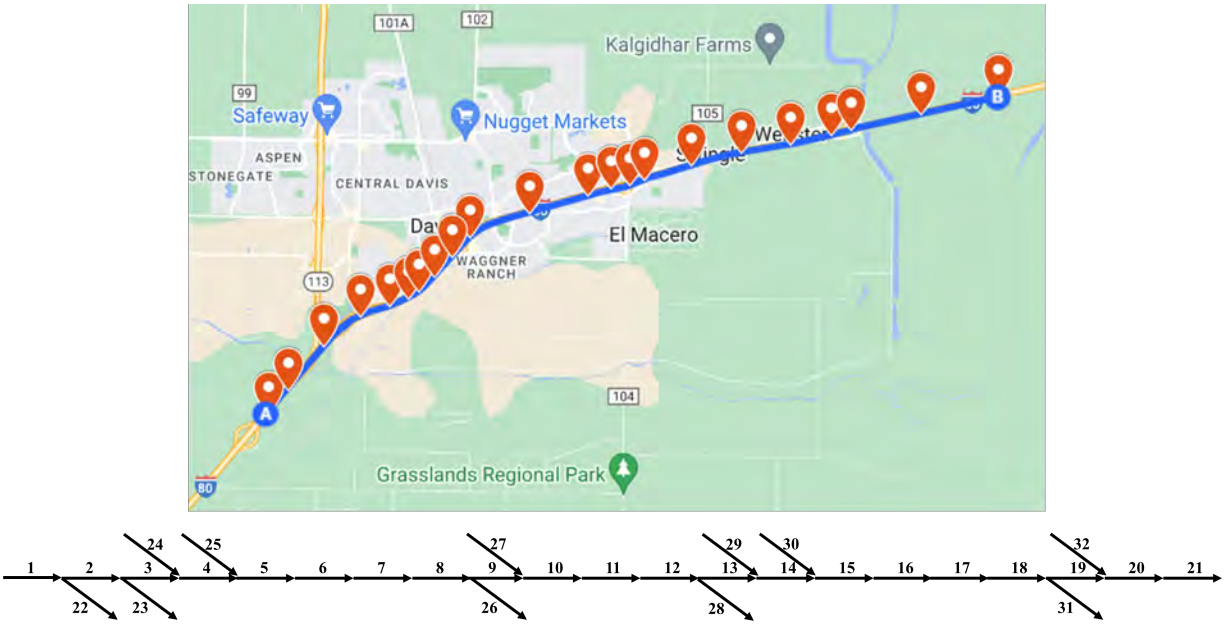


Figure 12: I80 East Davis-Sacramento

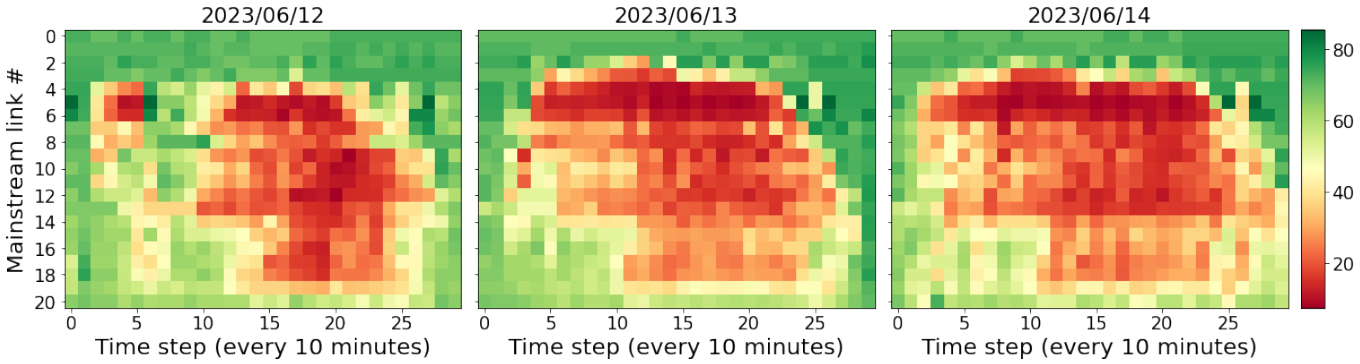


Figure 13: Speed profile of I80 East evolution in time

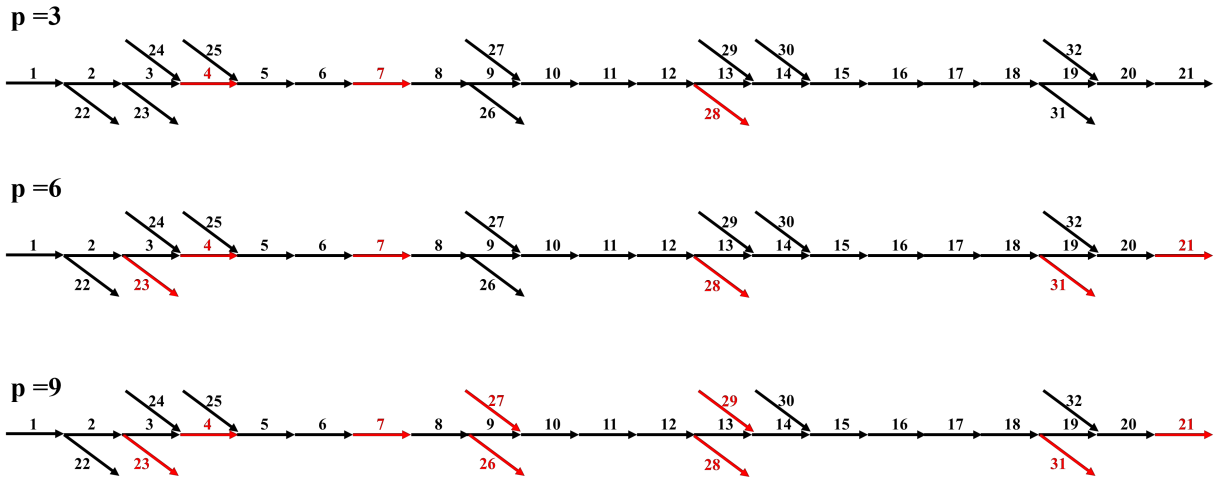


Figure 14: Optimal sensor locations for varying numbers of sensors to be deployed

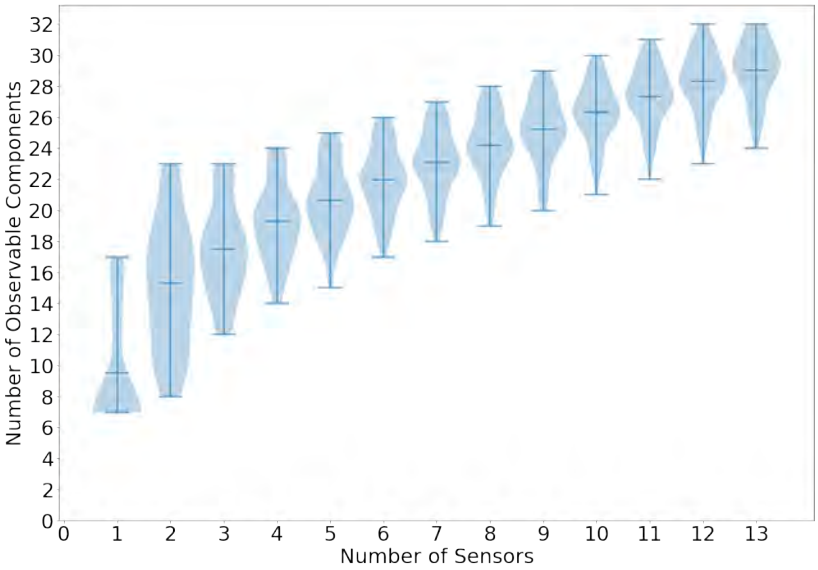


Figure 15: Violin plot for observability under different numbers of sensors to be deployed

Note that while preparing modes as an input to the optimization model (14), we take a quite coarse approach to translate speed to density under the uncongested regime – for an uncongested link (a link where speed is within 90% of its free-flow speed), we use half of its critical density to determine the supply and demand relation between this link and its immediate downstream links. The reason we take such a coarse approach is that before sensors are placed, we are not supposed to have much precise information about the traffic conditions. One can see from the optimal sensor location solutions that even though the information of modes entered to the optimization model is prepared in a quite coarse manner, the sensor placement strategies made good sense. For example, no sensor is suggested for links 1, 2, 3, which is consistent with the fact that they are never congested based on the Google Maps data.

Figure 15 reports the performance of the optimal sensor location strategies (obtained under different p) across various traffic modes. Each violin corresponds to a given p. The shape of violin shows the distribution of the number of observed links across different modes, together with the max, min, and the average number of observability. Based on the results shown in Figure 15, we have the following findings:

- When sensors are strategically placed, they can lead to more inferred information. For example, placing 13 sensors in the 32-link network could lead to inference of link densities of about 28 links on average.
- More sensors could not only improve the average observability, but also reduce the risk of not observing many links (indicated by the narrower observability range and heavier skew towards larger observability as the number of sensors increases).
- The marginal benefit on observability gained from additional sensors slows down as more sensors are deployed, indicating that strategic budgeting may be considered along with strategic location problem.

Notably, when only one sensor can be deployed, the optimal sensor will at most observe 17 links and at least observe 7 links among the 47 different modes. One should be mindful that the knowledge of the exact mode itself contains a significant amount of information. Without such information, a single sensor observing multiple segment downstream may be questionable, particularly considering traffic density dynamics is subject to noise.

4 Discussion

In this study, we have leveraged the algebraic and graphical properties of traffic network dynamics to design optimal sensor deployment solutions to ensure full observability or maximize observability. We have proposed a systematic Algebraic Approach to identify sensor locations for full observability with a minimal number of sensors. The Algebraic Approach remains applicable regardless of the choice of parameter, and thus is an exact observability method. Additionally, by exploiting special features of dynamic traffic systems, we are able to establish new analytical and computational methods beyond what has been reported in the general complex systems literature.

Our sensor location model takes into account a wide range of traffic conditions, and provides an optimal solution that performs well in an average sense. This design philosophy supports the growing interest in moving away from a deterministic mentality and achieving a system design that could cope with broader future uncertainties. It is true that observability analysis about a dynamic system would require some prior knowledge of the system. In the I80 East case study, we showcased that even with coarse information about the modes based on speed information from Google Maps, we are able to identify good sensor location solutions. In a sense, one can consider the preparation of the information about traffic modes equivalent to the exercise of uncertainty modeling when one is constructing a stochastic optimization model. Consistent with the widely adopted practice in the stochastic programming field, the modes and their associated weights/probabilities that are input to the sensor location model do not need to cover the entire distribution of the uncertain traffic condition, rather it is meant to provide a good representation of the important traffic scenarios that are considered significant to the system operator.

Despite the progress presented in this paper, the understanding of observability in dynamic traffic systems is still in its preliminary stage and demands much further research. For example, to incorporate the various traffic conditions, either simulations or real data can be used to provide such prior knowledge, as we demonstrated in the numerical example and case study. For future research, one may consider incorporating the simultaneous identification or detection of mode changes in a sensor placement problem. Also, the numerical examples included in this paper only include varying traffic conditions as a source of uncertainty. There could be other sources in a traffic system that might generate uncertainties. For example, an incident might change the shape of the fundamental diagram of involved links. The stochastic programming based sensor location model indeed can handle a wide range of uncertainty scenarios, but to prepare input characterizing those uncertainties is not a trivial task, which itself requires deep domain expertise. In addition, the sensor optimization model (14), which is equivalent to a two-stage stochastic programming model, treats different traffic modes as possible snapshots of uncertain scenarios, despite the dynamic evolution of these modes. If one wishes to model the sequence of the occurrence of various modes caused by dynamic evolution, one may consider a multi-stage stochastic programming framework, where uncertainty evolves over time as a scenario tree. However, such additional modeling flexibility would come at a price of computational difficulty. Another potential direction is to integrate observability and estimation quality in the context of sensor placement problems. Through numerical examples, we have demonstrated a positive correlation between observability and traffic state estimation quality. How to explicitly express estimation quality in the optimal sensor location model, whether through constraints or in the objective function, would be an interesting question to explore. Moreover, the current optimization model aims for an average performance. When estimation quality comes to the picture, there could be many creative ways of expressing a targeted performance, such as in terms of reliability or robustness, depending on one's risk preferences. Finally, our structural observability analyses are based on piecewise linear systems. Even though we are able to show the effectiveness of incorporating structural observability in dynamic systems that deviate from

the piecewise linear model, it would be too naive to assume there could not be more suitable information measures than what we introduced in this paper. Exploring other ways of measuring information richness to guide sensor placement strategies would be a worthy effort for future research.

Acknowledgement

This research is partially supported by grant CMMI 1826874 by the National Science Foundation of the United States.

Appendix A: an example of handling redundancy in the Algebraic Approach

As we mentioned in the Algebraic Approach, the procedure may induce redundant sensors. This is due to the fact that **Step 2** is not aware of the sensor placement result corresponding to different eigenvalues. The numerical example corresponding to Figure 11 in Section 3.2 has the following system matrix A:

$$\begin{bmatrix}
0 & 0 & 0 & 0 & 36 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 36 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 36 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -108 & -18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -18 & 0 & 0 & 0 & 0 & 9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -18 & 0 & 0 & 0 & 9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -18 & 0 & 0 & 0 & 0 & 18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 108 & 0 & 0 & 0 & 0 & 0 & -108 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -108 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -18 & 0 & 0 & 0 & 0 & 0 & 36 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -18 & 0 & 0 & 0 & 0 & 0 & 18 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 108 & 0 & 0 & 0 & -108 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 54 & 0 & 0 & 0 & -108 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 54 & 0 & 0 & 0 & 0 & -108 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -108 & -18 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -18 & 0 & 0 & 0 & 9 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -18 & 0 & 0 & 9 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 108 & 108 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 108 & 108 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -18
\end{bmatrix}$$

If we implement **Step 1-3**, we would get extra sensors needed (link 7, 13) as shown in Figure 16. These two sensors are identified to be redundant. **Step 4** detects and removes the redundant sensors.

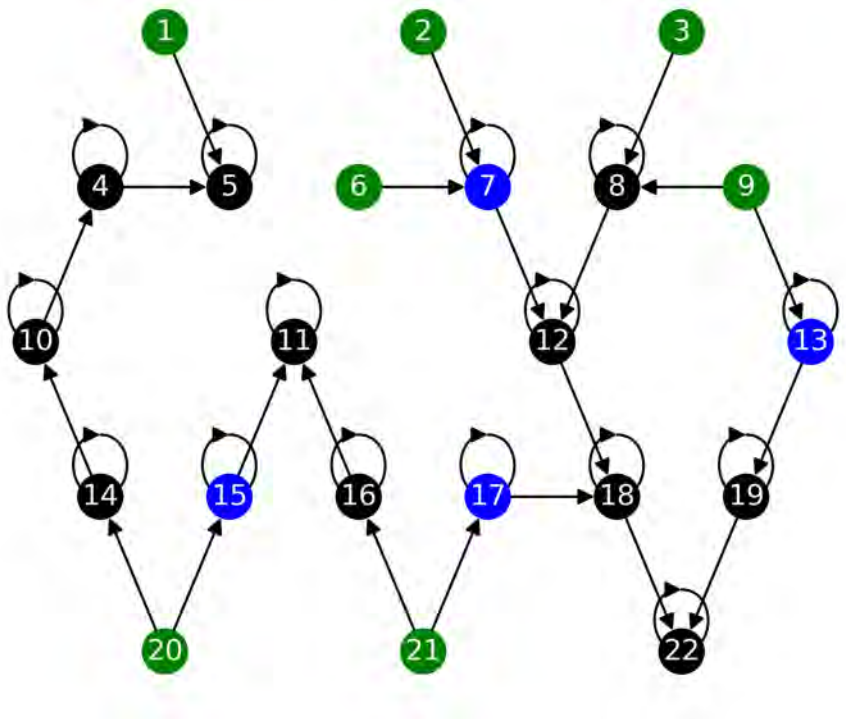


Figure 16: Without Step 4: redundant sensors

Appendix B: an example showing different traffic modes, their corresponding inference diagrams, and incorporation of different modes in the optimal sensor location model

In this section, we illustrate the piecewise linearity of the LQM using the 6-link example in subsection 3.1. Recall that the discretized LQM can be written as Eq. (16). In the SUMO simulation, before 430s, all links are uncongested and the system is operating at free-flow condition (mode 1):

$$\begin{bmatrix} k_1(t+1) \\ k_2(t+1) \\ k_3(t+1) \\ k_4(t+1) \\ k_5(t+1) \\ k_6(t+1) \end{bmatrix} = \begin{bmatrix} 1-v_1\frac{\Delta T}{l_1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1-v_2\frac{\Delta T}{l_2} & 0 & 0 & 0 & 0 \\ v_1\frac{\Delta T}{l_3} & v_2\frac{\Delta T}{l_3} & 1-v_3\frac{\Delta T}{l_3} & 0 & 0 & 0 \\ 0 & 0 & v_3\frac{\Delta T}{l_4} & 1-v_4\frac{\Delta T}{l_4} & 0 & 0 \\ 0 & 0 & 0 & \xi_{4\to5}(t)v_4\frac{\Delta T}{l_5} & 1 & 0 \\ 0 & 0 & 0 & \xi_{4\to6}(t)v_4\frac{\Delta T}{l_6} & 0 & 1 \end{bmatrix} \times \begin{bmatrix} k_1(t) \\ k_2(t) \\ k_3(t) \\ k_4(t) \\ k_5(t) \\ k_6(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} f_1(t)/l_1 \\ f_2(t)/l_2 \\ 0 \\ 0 \\ -g_5(t)/l_5 \\ -g_6(t)/l_6 \end{bmatrix} \tag{18}$$

After 430s, link 5 is congested and becomes the bottleneck. The flow from link 4 to link 5 is now dominated by the supply of link 5, which is $(k_{j,5}-k_5)w_5$. The outflux of link 4 is $g_4 = s_5/\xi_{4\to5}(t)$. The influx of link 6 is $\xi_{4\to6}(t)g_4$. The system now updates under mode 2:

$$\begin{bmatrix} k_1(t+1) \\ k_2(t+1) \\ k_3(t+1) \\ k_4(t+1) \\ k_5(t+1) \\ k_6(t+1) \end{bmatrix} = \begin{bmatrix} 1-v_1\frac{\Delta T}{l_1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1-v_2\frac{\Delta T}{l_2} & 0 & 0 & 0 & 0 \\ v_1\frac{\Delta T}{l_3} & v_2\frac{\Delta T}{l_3} & 1-v_3\frac{\Delta T}{l_3} & 0 & 0 & 0 \\ 0 & 0 & v_3\frac{\Delta T}{l_4} & 1 & \frac{w_5\Delta T}{\xi_{4\to5}(t)l_4} & 0 \\ 0 & 0 & 0 & 0 & 1-\frac{w_5\Delta T}{l_5} & 0 \\ 0 & 0 & 0 & 0 & -\frac{w_5\Delta T}{l_6}\frac{\xi_{4\to6}(t)}{\xi_{4\to5}(t)} & 1 \end{bmatrix} \times \begin{bmatrix} k_1(t) \\ k_2(t) \\ k_3(t) \\ k_4(t) \\ k_5(t) \\ k_6(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\frac{k_{j,5}w_5\Delta T}{\xi_{4\to5}(t)l_4} \\ \frac{k_{j,5}w_5\Delta T}{l_5} \\ \frac{k_{j,5}w_5\Delta T}{l_6}\frac{\xi_{4\to6}(t)}{\xi_{4\to5}(t)} \end{bmatrix} + \begin{bmatrix} f_1(t)/l_1 \\ f_2(t)/l_2 \\ 0 \\ 0 \\ -g_5(t)/l_5 \\ -g_6(t)/l_6 \end{bmatrix} \tag{19}$$

Link 5 becomes observable after 430s, which explains the correction in its estimation. Figure 17 compares the inference diagram for the two modes. Observing links 1,2,3,4 leads to 4 links observable in mode 1 and 5 links observable in mode 2. Note that when we construct the inference diagram, a self-edge

may only be added if the diagonal element of the continuous system matrix is nonzero. Here, since Eqs. (18) and (19) correspond to the discretized system matrix, if the diagonal element is 1 (the corresponding continuous system matrix will have a zero diagonal element), self-edges will not be added.

(a) Mode 1 (b) Mode 2

Figure 17: Inference diagram for mode 1&2

Integer programming formulation Using this example, we also show how the optimization model is formulated. Here we want to locate 4 sensors on the 6-link network to maximize observability, thus we have $n = 6, p = 4$.

$$\text{let } z_i = \begin{cases} 1, \text{if a sensor is put on link } i, \\ 0, \text{otherwise} \end{cases}, x_{k,i} = \begin{cases} 1, \text{if link } i \text{ is observable in mode } k, \\ 0, \text{if link } i \text{ is not observable in mode } k \end{cases}$$

where $i = 1,2,...,6$ and $k = 1,2$ the total number of modes in this case is 2, for simplicity. Let w_1 and w_2 denote the mode frequencies. The observability maximization problem is formulated as:

$$\begin{aligned}
\max \quad & w_1(x_{1,1}+x_{1,2}+x_{1,3}+x_{1,4}+x_{1,5}+x_{1,6}) + w_2(x_{2,1}+x_{2,2}+x_{2,3}+x_{2,4}+x_{2,5}+x_{2,6}) \\
\text{s.t.} \quad & z_1+z_2+z_3+z_4+z_5+z_6 = 4 \\
& \left.\begin{aligned} & x_{1,1} \le x_{1,3}+z_1 \\ & x_{1,2} \le x_{1,3}+z_2 \\ & x_{1,3} \le x_{1,4}+z_3 \\ & x_{1,4} \le z_4 \\ & x_{1,5} \le x_{1,4}+x_{1,6}+z_5 \\ & x_{1,6} \le z_6 \end{aligned}\right\} \textbf{mode 1} \\
& \left.\begin{aligned} & x_{2,1} \le x_{2,3}+z_1 \\ & x_{2,2} \le x_{2,3}+z_2 \\ & x_{2,3} \le x_{2,4}+z_3 \\ & x_{2,4} \le x_{2,5}+x_{2,6}+z_4 \\ & x_{2,5} \le z_5 \\ & x_{2,6} \le z_6 \end{aligned}\right\} \textbf{mode 2} \\
& z_1,z_2,z_3,z_4,z_5,z_6 \in \{0,1\} \\
& x_{k,i} \le 1, \forall k,i
\end{aligned}$$

References

Agarwal, S., Kachroo, P., and Contreras, S. (2015). A dynamic network modeling-based approach for traffic observability problem. *IEEE Transactions on Intelligent Transportation Systems*, 17(4):1168–1178.

Bekiaris-Liberis, N., Roncoli, C., and Papageorgiou, M. (2017). Highway traffic state estimation per lane in the presence of connected vehicles. *Transportation research part B: methodological*, 106:1–28.

Carøe, C. C. and Schultz, R. (1999). Dual decomposition in stochastic integer programming. *Operations Research Letters*, 24(1-2):37–45.

Castillo, E., Grande, Z., Calviño, A., Szeto, W. Y., and Lo, H. K. (2015). A state-of-the-art review of the sensor location, flow observability, estimation, and prediction problems in traffic networks. *Journal of Sensors*, 2015.

Collado, R., Papp, D., and Ruszczyński, A. (2012). Scenario decomposition of risk-averse multistage stochastic programming problems. *Annals of Operations Research*, 200(1):147–170. Funding Information: This research was supported by the NSF award CMII-0965689.

Contreras, S., Kachroo, P., and Agarwal, S. (2015). Observability and sensor placement problem on highway segments: A traffic dynamics-based approach. *IEEE Transactions on Intelligent Transportation Systems*, 17(3):848–858.

Daganzo, C. F. (1994). The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4):269–287.

Daganzo, C. F. (1995). The cell transmission model, part ii: network traffic. *Transportation Research Part B: Methodological*, 29(2):79–93.

Gentili, M. and Mirchandani, P. B. (2012). Locating sensors on traffic networks: Models, challenges and research opportunities. *Transportation research part C: emerging technologies*, 24:227–255.

He, S.-x. (2013). A graphical approach to identify sensor locations for link flow inference. *Transportation Research Part B: Methodological*, 51:65–76.

Hu, S.-R., Peeta, S., and Chu, C.-H. (2009). Identification of vehicle sensor locations for link-based network traffic applications. *Transportation Research Part B: Methodological*, 43(8-9):873–894.

Jabari, S. E. and Liu, H. X. (2013). A stochastic model of traffic flow: Gaussian approximation and estimation. *Transportation Research Part B: Methodological*, 47:15–41.

Jin, W. L. (2021). A link queue model of network traffic flow. *Transportation Science*, 55(2):436–455.

Kalman, R. E. (1960). On the general theory of control systems. In *Proceedings First International Conference on Automatic Control, Moscow, USSR*, pages 481–492.

Lighthill, M. J. and Whitham, G. B. (1955). On kinematic waves ii. a theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 229(1178):317–345.

Lin, C.-T. (1974). Structural controllability. *IEEE Transactions on Automatic Control*, 19(3):201–208.

Liu, Y.-Y., Slotine, J.-J., and Barabási, A.-L. (2011). Controllability of complex networks. *nature*, 473(7346):167–173.

Liu, Y.-Y., Slotine, J.-J., and Barabási, A.-L. (2013). Observability of complex systems. *Proceedings of the National Academy of Sciences*, 110(7):2460–2465.

Mousavi, S. S. and Kouvelas, A. (2020). Structural observability of traffic density dynamics on a motorway ring road. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6. IEEE.

Muñoz, L., Sun, X., Horowitz, R., and Alvarez, L. (2003). Traffic density estimation with the cell transmission model. In *Proceedings of the 2003 American Control Conference, 2003.*, volume 5, pages 3750–3755. IEEE.

Ng, M. (2012). Synergistic sensor location for link flow inference without path enumeration: A node-based approach. *Transportation Research Part B: Methodological*, 46(6):781–788.

Nugroho, S. A., Vishnoi, S. C., Taha, A. F., Claudel, C. G., and Banerjee, T. (2021). Where should traffic sensors be placed on highways? *IEEE Transactions on Intelligent Transportation Systems*.

Panda, M., Ngoduy, D., and Vu, H. L. (2019). Multiple model stochastic filtering for traffic density estimation on urban arterials. *Transportation research part B: methodological*, 126:280–306.

Richards, P. I. (1956). Shock waves on the highway. *Operations research*, 4(1):42–51.

Rinaldi, M. (2018). Controllability of transportation networks. *Transportation Research Part B: Methodological*, 118:381–406.

Rockafellar, R. T. and Wets, R. J.-B. (1991). Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of operations research*, 16(1):119–147.

Rostami-Shahrbabaki, M., Safavi, A. A., Papageorgiou, M., Setoodeh, P., and Papamichail, I. (2020). State estimation in urban traffic networks: A two-layer approach. *Transportation Research Part C: Emerging Technologies*, 115:102616.

Ruszczyński, A. and Shapiro, A. (2003). Stochastic programming models. *Handbooks in operations research and management science*, 10:1–64.

Seo, T., Bayen, A. M., Kusakabe, T., and Asakura, Y. (2017). Traffic state estimation on highway: A comprehensive survey. *Annual reviews in control*, 43:128–151.

Sumalee, A., Zhong, R., Pan, T., and Szeto, W. (2011). Stochastic cell transmission model (sctm): A stochastic dynamic traffic model for traffic state surveillance and assignment. *Transportation Research Part B: Methodological*, 45(3):507–533.

Thai, J. and Bayen, A. M. (2014). State estimation for polyhedral hybrid systems and applications to the godunov scheme for highway traffic estimation. *IEEE Transactions on Automatic Control*, 60(2):311–326.

Viti, F., Rinaldi, M., Corman, F., and Tampère, C. M. (2014). Assessing partial observability in network sensor location problems. *Transportation research part B: methodological*, 70:65–89.

Wang, R. and Work, D. B. (2014). Interactive multiple model ensemble kalman filter for traffic estimation and incident detection. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 804–809. IEEE.

Yperman, I. (2007). The link transmission model for dynamic network loading. *Katholieke Universiteit Leuven*, 481:482.

Yuan, Z., Zhao, C., Di, Z., Wang, W.-X., and Lai, Y.-C. (2013). Exact controllability of complex networks. *Nature communications*, 4(1):1–9.

Zhang, L. and Mao, X. (2015). Vehicle density estimation of freeway traffic with unknown boundary demand–supply: an interacting multiple model approach. *IET Control Theory & Applications*, 9(13):1989–1995.