Submitted to *Transportation Science*
manuscript (Please, provide the manuscript number!)

Estimating erratic measurement errors in network-wide traffic flow via virtual balance sensors

Zhenjie Zheng
Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hong Kong, China, zzj17.zheng@polyu.edu.hk

Zhengli Wang*
School of Management and Engineering, Nanjing University, Nanjing 210093, China, zhlwang@nju.edu.cn

Hao Fu
School of Systems Science, Beijing Jiaotong University, Beijing 100091, China, haof1@bjtu.edu.cn

Wei Ma*
Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hong Kong, China, wei.w.ma@polyu.edu.hk

Large-scale traffic flow data are collected by numerous sensors for managing and operating transport systems. However, various measurement errors exist in the sensor data and their distributions or structures are usually not known in the real world, which diminishes the reliability of the collected data and impairs the performance of smart mobility applications. Such irregular error is referred to as the erratic measurement error and has not been well investigated in existing studies. In this research, we propose to estimate the erratic measurement errors in networked traffic flow data. Different from existing studies that mainly focus on measurement errors with known distributions or structures, we allow the distributions and structures of measurement errors to be unknown except that measurement errors occur based on a Poisson process. By exploiting the flow balance law, we first introduce the concept of virtual balance sensors and develop a mixed integer non-linear programming model to simultaneously estimate sensor error probabilities and recover traffic flow. Under suitable assumptions, the complex integrated problem can be equivalently viewed as an estimate-then-optimize problem: first, estimation using machine learning (ML) methods, and then optimization with mathematical programming. When the assumptions fail in more realistic scenarios, we further develop a smart estimate-then-optimize (SEO) framework that embeds the optimization model into ML training loops to solve the problem. Compared to the two-stage method, the SEO framework ensures that the optimization process can recognize and compensate for inaccurate estimations caused by ML methods, which can produce more reliable results. Finally, we conduct numerical experiments using both synthetic and real-world examples under various scenarios. Results demonstrate the effectiveness of our decomposition approach and the superiority of the SEO framework.

Key words: network-wide traffic flow; data correction; flow balance law; mixed integer non-linear programming; smart estimate-then-optimize

*Corresponding authors

1. Introduction

With the advancement of sensing and communication technologies, massive traffic flow data are collected on an unprecedented scale by traffic monitoring sensors. However, measurement errors are ubiquitous in the sensor monitoring systems due to unexpected events and other stochastic externalities (Zheng and Su 2016, Yang, Yang, and Fan 2019). Such errors affect the quality of sensor data and bring challenges to traffic monitoring and management, which is well accepted in existing studies. For example, Rajagopal and Varaiya (2007) report that about 30% of the traffic sensors in California Performance Measurements cannot work properly. Duran and Earleywine (2013) find that irregular noise signals exist in sensor data (Nistor and Buda 2016). In addition, the outliers caused by external events are also reported in a wide range of sensor data (Duran and Earleywine 2013, Hu and Work 2020, Feng et al. 2022). Therefore, it is of great importance to estimate the measurement errors and recover the traffic data.

In practical scenarios, various measurement errors may coexist in sensor data and their distributions or structures are usually unknown in prior (Zheng and Su 2016, Ariannezhad and Wu 2020), which results in erratic measurement errors. Estimating such erratic measurement errors is more challenging than dealing with measurement errors with known distributions or structures. By utilizing the known information, statistical-based methods can be employed to effectively model the errors and recover the traffic data (Wang and Papageorgiou 2005, Yang, Yang, and Fan 2019). However, for erratic measurement errors with unknown distributions or structures, the applicability of these methods is limited, and how to tackle such errors remains an open question. Therefore, this paper focuses on estimating erratic measurement errors from the networked traffic flow data.

In literature, most existing studies on measurement errors of sensor networks focus on the errors with known distributions or structures, such as Gaussian errors or sparse outliers (Zheng and Su 2016), but a few pay attention to the erratic measurement errors. Early research assumes a zero-mean Gaussian error and employs Kalman filter methods to estimate the true data (Sun, Muñoz, and Horowitz 2003, Wang and Papageorgiou 2005). Later, wavelet-based approaches are developed to eliminate the Gaussian errors and sparse outliers (Boto-Giralda et al. 2010, Zheng et al. 2011). Recently, techniques grounded in compressed sensing and robust principal component analysis have been developed to distinguish between sparse errors and true data (Zheng and Su 2016, Hu and Work 2020, Feng et al. 2022). Moreover, Yang, Yang, and Fan (2019) develop a generalized method of moments (GMM) based approach to determine the parameters of systematic errors and Gaussian errors in networked sensor data, which can successfully estimate the systematic and random errors under certain assumptions. However, these methods typically require known distributions or sparse structures of measurement errors, which limits their applications in the real world. To address this issue, some studies (Vanajakshi and Rilett 2004, Kikuchi, Mangalpally, and Gupta 2006, Chen, Chootinan, and Recker 2009, Yin et al. 2017) utilize the flow balance law to estimate the erratic measurement errors. However, all these approaches handle the errors in a deterministic way and can hardly estimate the error probabilities of sensors. Additionally, the data recovery in these studies relies on strong assumptions about data structures. For example, Yin et al. (2017) require that the erroneous data can be inferred from the correct data based on the flow conservation. There are also related studies that develop machine learning (ML)-based denoising methods, such as BM3D (Danielyan, Katkovnik, and Egiazarian 2011), noise2noise (Lehtinen et al. 2018), and conditional generative adversarial network (Yi and Babyn 2018) to estimate and reduce the measurement errors in sensor data. However, these methods rely solely on collected data and lack the capability to leverage intrinsic traffic domain knowledge through optimization analysis. Table 1 summarizes the focus of existing studies.

Table 1 Summary of existing studies in the literature.

Papers	Method	Error types		Domain knowledge	Probabilistic	Integration of ML and MP
		EK	EE			
Sun, Muñoz, and Horowitz (2003)	Kalman filter	✓			✓	
Vanajakshi and Rilett (2004)	Non-linear optimization	✓	✓	✓		
Wang and Papageorgiou (2005)	Kalman filter	✓			✓	
Kikuchi, Mangalpally, and Gupta (2006)	Fuzzy optimization	✓	✓	✓		
Chen, Chootinan, and Recker (2009)	Norm method		✓	✓		
Boto-Giralda et al. (2010)	Wavelet analysis	✓			✓	
Zheng et al. (2011)	Wavelet analysis	✓			✓	
Danielyan, Katkovnik, and Egiazarian (2011)	BM3D	✓			✓	
Zheng and Su (2016)	Compressed sensing	✓		✓		
Yin et al. (2017)	ℓ_1 minimization		✓	✓		
Yi and Babyn (2018)	cGAN	✓				
Lehtinen et al. (2018)	noise2noise	✓				
Yang, Yang, and Fan (2019)	GMM	✓		✓	✓	
Hu and Work (2020)	RPCA	✓		✓		
Feng et al. (2022)	RPCA	✓		✓		
Our framework	MINLP and SEO	✓	✓	✓	✓	✓

BM3D: Block-matching and 3D filtering; cGAN: Conditional generative adversarial network; GMM: Generalized method of moments; RPCA: Robust principal component analysis; MINLP: Mixed integer non-linear programming; MP: Mathematical programming; SEO:Smart estimate-then-optimize

EK: Errors with known distributions or structures; EE: Erratic errors with unknown distributions or structures.

According to the aforementioned review of existing studies, there are three main challenges when we estimate the erratic measurement errors in the networked traffic flow data. First, various measurement errors may coexist in sensor data (Zheng and Su 2016), which complicates the separation and analysis of these errors. To address this issue, a unified approach should be developed to describe the occurrences of these erratic errors. Second, the distributions of erratic measurement errors are typically unknown in the real world, which means limited information can be obtained about these errors. To compensate for the lack of information, prior traffic domain knowledge, such as traffic flow conservation and global low-rank structure of traffic data (Chen, He, and Sun 2019, Yang, Yang, and Fan 2019), can be utilized. Third, the challenge also arises from the unknown positions or structures of measurement errors, which leads to a mixture of correct and erroneous data (Yin et al. 2017). Based on the limited information, it is challenging to accurately identify the erroneous data in a deterministic manner. To tackle this challenge, probabilistic modeling and ML techniques can be employed to estimate the error probability of each sensor rather than striving for accurate identification. Once the error probability is determined, the most probable erroneous data can be identified with the maximum likelihood.

In this paper, we advocate the novel concept of virtual balance sensors (VBS), which are virtually installed at each node of the road network. The VBS is a binary sensor that outputs 1 when the measured inflow does not equal the measured outflow at the node, *i.e.*, flow balance law does not hold due to measurement errors, and outputs 0 otherwise. This implies that the VBS can be observed directly with flow data as illustrated in Figure 1. Instead of directly modeling the measurement error of each link, we propose to model the error distribution of the VBS, which could help us estimate the erratic measurement error without knowing its specific distribution. For example, when a link is measured, we do not know whether there is an error in the measurement; in contrast, if the VBS of a node outputs 1, we can confidently tell that there is a measurement error in either the inflow or outflow of the node, since the probability measure of two continuous random errors equating to each other is zero.

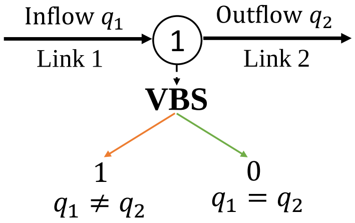


Figure 1 Detection of measurement errors using VBS.

Utilizing the concept of VBS, a methodological framework is developed to estimate the erratic measurement errors in traffic flow data on road networks as illustrated in Figure 2. We allow the distributions of measurement errors to be unknown and model the erratic errors in a probabilistic manner. The input includes the flow data collected by sensors and the network structure. Then, the proposed framework outputs the estimated sensor error probabilities and recovered data based on the integration of ML and optimization analysis. Specifically, we first introduce the virtual balance sensor to indicate whether the flow balance law is violated at a given node. Using the data from virtual sensors, we then simultaneously estimate sensor error probabilities and recover traffic flow using maximum likelihood within a mixed-integer nonlinear programming model. Under suitable assumptions, the complex integrated problem can be equivalently decomposed into two simpler subproblems: first, estimation using ML methods, and then optimization using mathematical programming. This estimate-then-optimize paradigm has been widely applied in transportation and logistics (He, Liu, and Shen 2022, Tang and Khalil 2022, Sadana et al. 2024). When the assumptions fail in more realistic scenarios, we employ a smart estimate-then-optimize (SEO) framework (Liu and Grigas 2021, Elmachtoub and Grigas 2022, Zhang et al. 2024) in the literature, which integrates ML-based estimations with optimization analysis grounded in traffic domain knowledge. Compared to the two-stage method, the SEO framework significantly reduces the risk of suboptimal decisions that might arise from inaccurate estimations caused by ML methods, which can produce more robust results. Additionally, the performance of our approach and the bound of recovery errors are analyzed. Finally, numerical experiments are conducted on the Nguyen-Dupuis network and Sha Tin network in Hong Kong under various scenarios. Results demonstrate the decomposition approach is effective and the SEO framework can achieve better performance than the two-stage method.

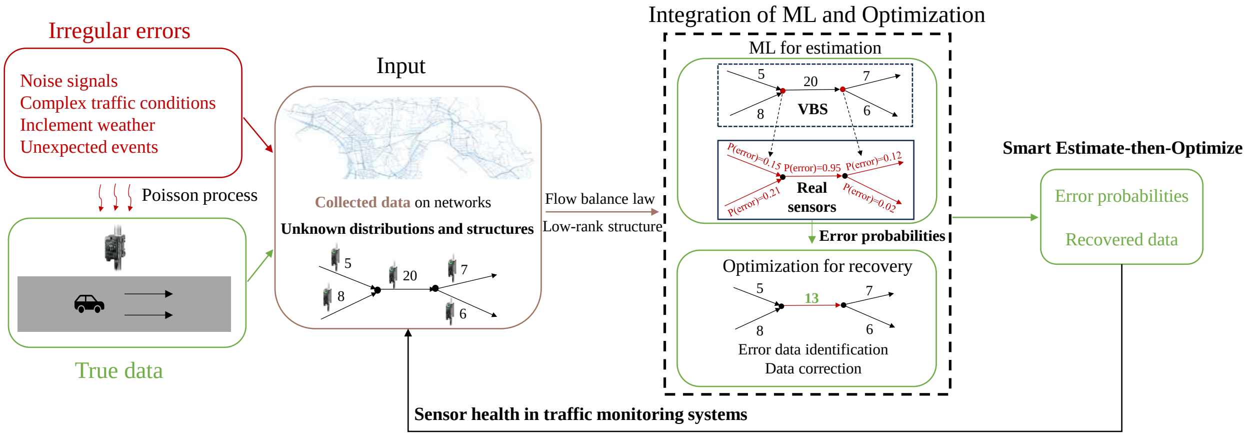


Figure 2 **Estimating the erratic measurement errors in sensor monitoring systems on road networks.**

We summarize the main contributions of this research as follows:

- We introduce the concept of virtual balance sensors, on top of which a MINLP model is formulated to simultaneously estimate sensor error probabilities and recover traffic flow data with the maximum likelihood;
- Under suitable assumptions, the complex integrated problem is equivalently decomposed into two simpler subproblems and solved using a two-stage method: ML for estimation, followed by mathematical programming for optimization;

- When the assumptions fail in more realistic scenarios, we develop an SEO framework that integrates estimation with optimization by embedding the optimization model into ML training loops to solve the integrated problem, which can recognize and compensate for inaccurate ML estimation; and
- Numerical experiments conducted on the Nguyen-Dupuis network and the Sha Tin county network in Hong Kong demonstrate the effectiveness of the decomposition approach and the superiority of the SEO framework across various scenarios.

The rest of the paper is organized as follows. In Section 2, we introduce our problem setting. In Section 3, we introduce the concept of virtual balance sensors and elaborate on the proposed MINLP model. We also demonstrate that the model can be solved using a two-stage method. The solution method is presented in Section 4. Furthermore, an SEO framework is developed to solve the problem under more realistic scenarios in Section 5. The performance of our approach is evaluated in Section 6 using the Nguyen-Dupuis network and Sha Tin county network in Hong Kong under various scenarios. Lastly, conclusions and future research directions are discussed in Section 7.

2. Problem statement

In this research, we focus on estimating the erratic measurement errors in traffic flow data collected by sensors on networks. The sensors installed on links are indexed as $1,\cdots,e,\cdots,E$. The time intervals are sequentially indexed as $1,\cdots,t,\cdots,T$. For a specific sensor e during time interval t, the collected traffic flow is denoted by $q_{e,t}$. However, the collected data $q_{e,t}$ may deviate from the true traffic flow, denoted as $\bar{q}_{e,t}$, due to the potential erratic measurement errors. Mathematically, the relationship between collected data $q_{e,t}$ and true data $\bar{q}_{e,t}$ is formulated as follows:

$$q_{e,t} = \bar{q}_{e,t} + \xi_{e,t}, \tag{1}$$

where $\xi_{e,t}$ is the erratic measurement error with unknown distributions and structures. Furthermore, the error probability of sensor e during time interval t is denoted by $p_{e,t}$. Our objective is to estimate $p_{e,t}$ and recover $\bar{q}_{e,t}$ using the collected data $q_{e,t}$. The notations used in this paper are summarized in Appendix A.

Essentially, we introduce an integrated estimation and optimization problem that aims to minimize flow recovery error (i.e., optimization) with uncertain sensor error probabilities (i.e., estimation) from collected data. Intuitively, simultaneous estimation of $p_{e,t}$ and recovery of $\bar{q}_{e,t}$ is a challenging task due to the limited information available regarding measurement errors. However, this paper demonstrates that the problem can be resolved by strategically exploiting the flow balance law, leveraging the low-rank structure of traffic data, and learning the causes and generation mechanism of errors through ML techniques and optimization analysis based on traffic domain knowledge.

3. Model

In this section, we introduce our methodological framework. Specifically, we first use the Poisson process to model the occurrence of erratic measurement errors in Section 3.1. We then introduce a novel concept of virtual balance sensors (VBS) to monitor the occurrence of error at each intersection in Section 3.2. With the data from these VBS, we formulate a MINLP model to estimate the error probability $p_{e,t}$ and recover the true data $\bar{q}_{e,t}$ in Section 3.3. Section 3.4 analyzes the properties and challenges in solving the model. Section 3.5 presents the reformulated model. Finally, Section 3.6 demonstrates that the reformulated model can be equivalently decomposed and solved using a two-stage estimate-then-optimize method under suitable assumptions.

3.1. Occurrences of erratic measurement errors

In the field of reliability engineering, it is well accepted that sensor errors or failures are caused by a series of random events. This also applies to traffic sensors installed on urban road networks. Typically, these random events originate from various exogenous factors, such as sensor features, unexpected noise signals, complex traffic conditions, inclement weather, and so on, which can be modeled using the Poisson process (Ariannezhad and Wu 2020, Feng et al. 2022).

Following similar approaches in reliability engineering, we employ the Poisson process to model the erratic measurement errors in traffic sensor data. To validate this hypothesis, we use real flow data in Hong Kong to conduct a hypothesis test (see Appendix B for details), and the results demonstrate that the occurrences of erratic measurement errors follow Poisson processes. Specifically, if there is no random event, we say that the sensor works properly without errors. Conversely, if more than one random event occurs, we say that measurement errors occur during the data collection.

The error probability of sensor e during time interval t, that is, $p_{e,t}$, can be formulated using the Poisson process:

$$p_{e,t} = \sum_{n=1}^{+\infty} exp(-\lambda_{e,t}\eta)\frac{(\lambda_{e,t}\eta)^n}{n!} = 1 - exp(-\lambda_{e,t}\eta), \tag{2}$$

where n is the number of arrivals in the interval, η is the length of a time interval in which we average the traffic flow, and $\lambda_{e,t}$ is the arrival rate of the Poisson process associated with sensor e during time interval t. The basic assumption of our approach is that the arrival rate of sensor failure is invariant among all the time intervals, that is, $\lambda_{e,t} = \lambda_e$, $\forall 1 \leq t \leq T$. Similar assumptions have been adopted in the existing studies and can be justified by restricting the estimation time horizon to ensure an appropriate time duration (Yera et al. 2021, Yang, Yang, and Fan 2019). As a result, the parameters governing error generation are considered fixed throughout the estimation time horizon, that is, $p_{e,t} = p_e,\ \forall 1 \leq t \leq T$. Without loss of generality, we set the time interval η to 1 and Equation (2) can be reformulated as follows:

$$p_e = \underset{\forall 1 \leq t \leq T}{p_{e,t}} = \sum_{n=1}^{+\infty} exp(-\lambda_{e,t}) \frac{(\lambda_{e,t})^n}{n!} = 1 - exp(-\lambda_e). \tag{3}$$

When the occurrences of measurement errors do not follow the Poisson process, the proposed framework remains effective. In such cases, other mathematical techniques or functions can be adopted to characterize p_e (see Section 5 for details).

3.2. Virtual balance sensors

We introduce the virtual balance sensors (VBS) as a novel and pivotal element in our framework to integrate the flow balance law. These VBS are a set of virtual sensors installed at nodes (vertices) of the network, which are indexed as $1, \cdots, v, \cdots V$. Unlike the real sensors that monitor the traffic flow on links, the VBS are used to monitor whether the flow balance law is violated at the node. Let $\delta_{v,t}$ denote the data recorded by the VBS installed at node v during time interval t. It is formulated as follows:

$$\delta_{v,t} = \begin{cases} 1, & if \quad \sum_{e \in E^+(v)} q_{e,t} - \sum_{e \in E^-(v)} q_{e,t} \neq 0; \\ 0, & \text{otherwise}, \end{cases} \tag{4}$$

where $E^+(v)$ and $E^-(v)$ are the sets of entering links and exiting links of node v, respectively.

Similar to the real sensors, the error probability of VBS v can also be derived. According to the flow balance law, it is expected that the flow entering node v should be equal to that exiting node v, that is, $\delta_{v,t} \equiv 0$. When $\delta_{v,t} = 1$, it indicates that error occurs at node v, that is, VBS v during time t. When $\delta_{v,t} = 0$, it indicates that no error occurs, since the probability measure of several continuous random errors equating to each other is zero. Let $\mathcal{p}_v$ denote the error probability of VBS v, which can be directly calculated by $\delta_{v,t}$:

$$\mathcal{p}_v = \frac{\sum_{t=1}^{T} \delta_{v,t}}{T}. \tag{5}$$

However, due to minor disturbances in traffic conditions or other stochastic externalities, the flow balance law may not be satisfied and therefore the estimated $\delta_{v,t}$ is biased. To address this issue, we can modify Equation (4) to allow the existence of minor disturbances, which is widely adopted in the existing studies (Zheng and Su 2016, Yin et al. 2017). This modification has only a slight impact on the estimation accuracy, as demonstrated in the numerical experiments in Section 6.1.4.

3.3. Multi-objective mixed integer non-linear programming model

Based on the concept of VBS, we first demonstrate how to simultaneously estimate sensor error probabilities and recover traffic flow data using a mixed-integer nonlinear programming model.

3.3.1. Decision variables Let $\hat{q}_{e,t}$ be the decision variable representing the recovered traffic flow of sensor e during time t. When there is no measurement error, $\hat{q}_{e,t}$ equals to the collected data, that is, $\hat{q}_{e,t} = q_{e,t}$. When measurement errors occur, we use a decision variable $z_{e,t}$ to replace the erroneous data, that is, $\hat{q}_{e,t} = z_{e,t}$. We also introduce a binary decision variable $\gamma_{e,t}$ to indicate the occurrence of measurement errors:

$$\gamma_{e,t} = \begin{cases} 1, & \text{if measurement errors occur in sensor } e \text{ during time interval } t; \\ 0, & \text{otherwise.} \end{cases}$$

The relationship of these decision variables can be expressed as follows:

$$\hat{q}_{e,t} = (1-\gamma_{e,t})q_{e,t} + \gamma_{e,t}z_{e,t}. \tag{6}$$

In summary, the decision variables are $\hat{q}_{e,t}, \gamma_{e,t}, z_{e,t}$ and p_e.

3.3.2. Objective function Our objective function is composed of two components. The primary objective is to recover traffic flow data with the maximum likelihood of sensor errors, while the second objective is to minimize the rank of the recovered matrix.

In order to infer the most probable scenario for the occurrences of measurement errors, we first aim to minimize the corresponding negative log-likelihood. Given the collected data $q_{e,t}$ from sensor e during time interval t, the errors occur with a probability p_e. We can use a Bernoulli distribution with a parameter p_e to characterize whether the collected data $q_{e,t}$ is corrupted by the errors. The corresponding likelihood is formulated as follows:

$$l_{e,t} = (p_e)^{\gamma_{e,t}}(1-p_e)^{1-\gamma_{e,t}}. \tag{7}$$

For the data from all sensors within the estimation time horizon, the corresponding likelihood is formulated as follows:

$$\mathcal{L} = \prod_{e=1}^{E}\prod_{t=1}^{T}(p_e)^{\gamma_{e,t}}(1-p_e)^{1-\gamma_{e,t}}. \tag{8}$$

To minimize the negative log-likelihood of the collected data, we take the logarithm of $\mathcal{L}$, which results in the primary objective function $\mathcal{F}_1$:

$$\min\ \mathcal{F}_1 = -\log(\mathcal{L}) = -\sum_{e=1}^{E}\sum_{t=1}^{T}(\gamma_{e,t}\log(p_e) + (1-\gamma_{e,t})\log(1-p_e)). \tag{9}$$

When $\mathcal{F}_1$ is minimized, we can identify the occurrences of measurement errors with the maximum likelihood and recover traffic data by the flow balance law. However, it is worth noting that the flow balance law alone cannot effectively recover the flow when multiple links connected to the same node are erroneous. To address this issue, other prior information should be utilized.

In the existing studies (Liang, Zhao, and Sun 2022, Feng et al. 2022), it has been widely recognized that the traffic flow data has a global low-rank structure, which can be employed to impute the missing values or remove the measurement errors. By leveraging the low-rank structure, numerous matrix completion or tensor decomposition methods are developed and have achieved satisfactory performance in recovering traffic data (Bae et al. 2018, Liang, Zhao, and Sun 2022). Following the same logic, our second objective $\mathcal{F}_2$ is to minimize the rank of the recovered matrix $\hat{\boldsymbol{Q}}$:

$$\min\ \mathcal{F}_2 = rank(\hat{\boldsymbol{Q}}). \tag{10}$$

As a result, the multi-objective function is formulated as $\mathcal{F} = \{\mathcal{F}_1, \mathcal{F}_2\}$.

3.3.3. Constraints We optimize the multi-objective function $\mathcal{F}$ subject to the following constraints:

$$\delta_{v,t} \le \sum_{e \in E(v)} \gamma_{e,t} \le |E(v)|_{\#} \delta_{v,t}, \quad \forall 1 \le v \le V, 1 \le t \le T; \tag{11a}$$

$$\sum_{e \in E^+(v)} \hat{q}_{e,t} - \sum_{e \in E^-(v)} \hat{q}_{e,t} = 0, \quad 1 \le v \le V; \tag{11b}$$

$$\hat{q}_{e,t} = (1 - \gamma_{e,t}) q_{e,t} + z_{e,t}, \quad 1 \le e \le E, 1 \le t \le T; \tag{11c}$$

$$z_{e,t} \le \gamma_{e,t} \Theta, \quad 1 \le e \le E, 1 \le t \le T; \tag{11d}$$

$$p_e T = \sum_{t=1}^{T} \gamma_{e,t}, \quad 1 \le e \le E; \tag{11e}$$

$$\hat{q}_{e,t} \ge 0,\ z_{e,t} \ge 0,\ \gamma_{e,t} \in \{0,1\},\ p_e \ge 0, \quad 1 \le e \le E, 1 \le t \le T. \tag{11f}$$

$E(v)$ is the set of links connecting to the VBS v, and $|E(v)|_{\#}$ is the cardinality of $E(v)$. For VBS v during time interval t, if errors are detected (i.e., $\delta_{v,t} = 1$), then Constraints (11a) guarantee that there are errors in the sensors associated with VBS v (i.e., $\sum_{e \in E(v)} \gamma_{e,t} \ge 1$). Conversely, if the error is not detected by VBS v during time interval t (i.e., $\delta_{v,t} = 0$), then Constraints (11a) guarantee that there are no errors in the sensors associated with VBS v (i.e., $\sum_{e \in E(v)} \gamma_{e,t} = 0$). Constraints (11b) ensure that recovered traffic flow $\hat{q}_{e,t}$ satisfy the flow balance law. Constraints (11c) and (11d) express the relationship of decision variables characterized by Equation (6). Since Equation (6) is non-linear, we use Constraints (11c) and (11d) to convert the original constraints into two sets of linear constraints, where Θ is a large positive number. Constraints (11e) ensure that the estimated error probability p_e aligns with the number of identified errors $\sum_{t=1}^{T} \gamma_{e,t}$, when T is sufficiently large. Moreover, the values of decision variables are restricted by Constraints (11f).

It is worth noting that Constraints (11e) needs to be modified when T is a finite number. This is because $\sum_{t=1}^{T} \gamma_{e,t}$ is discrete while $p_e T$ is continuous, which makes the model infeasible. To address this issue in practical applications, we replace Constraints (11e) with the following constraints:

$$\sum_{t=1}^{T} \gamma_{e,t} - 1 \le p_e T \le \sum_{t=1}^{T} \gamma_{e,t} + 1, \quad \forall 1 \le e \le E. \tag{12}$$

Moreover, the correct flow data may also violate the flow balance law due to the minor disturbances of traffic conditions as discussed in Section 3.2:

$$\left|\sum_{e\in E^{+}(v)} \bar{q}_{e,t} - \sum_{e\in E^{-}(v)} \bar{q}_{e,t}\right| = \epsilon_{v,t}, \tag{13}$$

where $\epsilon_{v,t}$ is a random variable with a relatively small scale compared to the traffic flow. To address this issue, we can relax Constraints (11b) in practical applications as follows:

$$\left|\sum_{e\in E^{+}(v)} \hat{q}_{e,t} - \sum_{e\in E^{-}(v)} \hat{q}_{e,t}\right| \le \sigma_{\epsilon}, \tag{14}$$

where σ_ϵ is the standard deviation of $\epsilon_{v,t}$. Similar formulations have been adopted in the existing studies (Zheng and Su 2016, Yin et al. 2017).

3.4. Model analysis

Based on the discussions in the above sections, the formulation for estimating the sensor error probabilities and recovering the network-wide traffic flow can be summarized as follows:

$$\begin{aligned} \min_{\hat{q}_{e,t},\gamma_{e,t},z_{e,t},p_e} \quad & \mathcal{F} \\ \text{s.t.} \quad & \text{Constraints (11a) - (11f).} \end{aligned} \tag{15a}$$

In this section, we demonstrate that the integrated problem cannot be resolved solely through mathematical programming. Actually, the proposed optimization model encounters two significant challenges in estimating erratic errors: biased estimation and computational inefficiency.

3.4.1. Biased estimation The error probability p_e estimated by model (15) may deviate from the true error probability. Taking the simple road network in Figure 1 as an example, we assume that an error is detected by the VBS during time $t=1$. According to Constraints (11a), we know that there are errors in either sensor 1 or 2. We traverse each feasible solution and find two optimal numerical solutions for the primary objection function $\mathcal{F}_1$: (1) $\gamma_{1,1}=1,\gamma_{2,1}=0,p_1=1,p_2=0$; and (2) $\gamma_{1,1}=0,\gamma_{2,1}=1,p_1=0,p_2=1$. It is found that both solutions assign a zero error probability to one of the sensors. For any given value of T, similar results are produced by the model, which leads to a biased estimation of p_e. Note that $p_e=0$ is not included in the domain of $p_e\log(p_e)$ and therefore we say these solutions are numerical solutions. To facilitate the following analysis, we define $p\log(p)=0\ for\ p=0$. This definition is reasonable as $\lim_{p\to 0} p\log(p)=0$.

In the following discussion, we will theoretically demonstrate the biases induced by model (15). We first analyze the primary objective function $\mathcal{F}_1$ in Proposition 1.

Proposition 1. *The primary objective function $\mathcal{F}_1$ of model (15) is a concave function for $0 < p_e < 1, \forall 1 \leq e \leq E$.*

Proof See Appendix C.1. □

Additionally, we further simplify model (15) to enhance its solvability in Proposition 2. Without loss of generality, we use the network in Figure 1 to conduct the analysis for simplicity, while the propositions can be extended to general networks.

Proposition 2. *If T is sufficiently large, for the network shown in Figure 1, then the optimal p_e for model (15) can be determined by solving the following optimization model.*

$$\min_{p_1,p_2} -T\sum_{e=1,2}\left[p_e\log(p_e)+(1-p_e)\log(1-p_e)\right]$$

$$\text{s.t.}\quad p_1+p_2 \geq \frac{\sum_{t=1}^{T}\delta_{1,t}}{T}; \tag{16a}$$

$$0 \leq p_1 \leq \frac{\sum_{t=1}^{T}\delta_{1,t}}{T}; \tag{16b}$$

$$0 \leq p_2 \leq \frac{\sum_{t=1}^{T}\delta_{1,t}}{T}. \tag{16c}$$

Proof See Appendix C.2. □

Proposition (2) establishes that solving model (16) yields the optimal p_e for model (15). By leveraging Propositions 1 and 2, we can derive the closed-form optimal solutions the simplified network in Figure 1, and hence the biases can be demonstrated.

Proposition 3. *The optimal values p_e for model (15) on the network of Figure 1 can be either $p_1 = \frac{\sum_{t=1}^{T}\delta_{1,t}}{T}, p_2 = 0$ or $p_1 = 0, p_2 = \frac{\sum_{t=1}^{T}\delta_{1,t}}{T}$ when $\frac{\sum_{t=1}^{T}\delta_{1,t}}{T} < 1$.*

Proof See Appendix C.3. □

According to Proposition 3, one can see that the optimal solutions for Figure 1 always allocate a zero error probability to one of the sensors, which is not consistent with reality and leads to biased estimations of the error probabilities. For the general networks, these propositions still apply under similar scenarios.

The direct cause for the biased results is that the optimal solution for minimizing a concave function typically lies at the boundaries of the feasible region. However, the fundamental reason is the limited information available about erratic errors. For model (15), we attempt to estimate the error probabilities p_e of all sensors using the data collected by VBS v. Although VBS v can detect the occurrences of errors, it does not tell us which sensors are erroneous. Such limited information makes it difficult to obtain an accurate estimation. In view of this, the optimization model (15) requires reformulation to accurately estimate the sensor error probabilities.

3.4.2. Computational efficiency Based on Proposition 1, it is established that model (15) is designed to minimize a concave function $\mathcal{F}_1$. This characteristic introduces complexity in identifying a global minimum, as such functions inherently tend to have a downward opening shape that can make global optimization challenging. Consequently, model (15) is a non-convex MINLP with the combinatorial explosion of binary decision variables, which is NP-hard. This indicates that the solution time increases exponentially in proportion to the growth in input size. For most problems of interest, especially for road networks that represent the real cities, model (15) is difficult to be optimized for state-of-the-art solvers in a reasonable time due to the TE binary decision variables in the constraints. As a result, there is a necessity to reformulate the model or develop efficient algorithms that can simplify the solving process.

3.5. Model reformulation

In this section, we use ML techniques to reformulate and simplify the MINLP model by exploiting the error generation mechanism.

3.5.1. Existing methods for addressing the biased estimation To accurately estimate the erratic errors, existing studies introduce supplementary prior information or assumptions into their models, which can be broadly categorized into three types. First, some studies (Wall and Dailey 2003, Kim et al. 2019, Yang, Yang, and Fan 2019) assume that the network is equipped with well-calibrated sensors, which implies that their error probabilities p_e are known to be 0 in advance and therefore reduces the estimated parameters. Second, studies such as Yang, Yang, and Fan (2019) propose that sensors may follow latent error patterns and sensors of the same pattern share parameters. By utilizing these patterns, the estimation of error distributions is effectively simplified, enhancing both the efficiency and accuracy of the estimation. Third, Yin et al. (2017) put forth the assumption that erroneous data exhibit a special sparse structure, which allows for the direct identification and correction of measurement errors using the flow balance law. In summary, existing studies propose various approaches to reduce the number of estimated parameters, which can accurately estimate p_e under certain assumptions. However, these models or assumptions neglect the nature of the underlying process that generates the errors, which limits their applications.

3.5.2. Incorporating error generation mechanism into the model In this research, we provide a new perspective to estimate p_e by naturally exploring the causes and generation mechanism of erratic errors. As discussed in Section 3.1, the erratic measurement errors are caused by a series of exogenous variables $\boldsymbol{x_e}$, which include factors like sensor features, traffic conditions, weather conditions, and so on. This implies that the value of the arrival rate λ_e is determined by $\boldsymbol{x_e}$, which is represented as:

$$\lambda_e = -\log(1-p_e) = \chi(\boldsymbol{x_e}), \tag{17}$$

where $\boldsymbol{x_e}$ is the vector of exogenous variables. χ can be different ML methods, such as linear regression, neural networks, Gaussian process, etc. To facilitate the analysis of model properties, we simply use a linear function to characterize the relationship between λ_e and $\boldsymbol{x_e}$ in this paper, as represented by $\chi(\boldsymbol{x_e}) = \boldsymbol{\beta}\boldsymbol{x_e}^T$, where $\boldsymbol{\beta}$ is the coefficient vector of these variables.

When the relationship between λ_e and $\boldsymbol{x_e}$ is not linear, or the relationship is non-linear and unknown, we can still use other mathematical techniques or ML methods, such as polynomial regression or neural networks to estimate the coefficient vector $\boldsymbol{\beta}$, as detailed in Section 5.

Moreover, we can introduce other prior assumptions, such as latent error patterns, error distributions, or specific error structures, to estimate p_e. However, we argue that our assumption more accurately reflects the real world scenarios in modeling the erratic errors. This is based on the fact that such errors would not appear out of nowhere, but are instead induced by latent factors.

3.5.3. Lagrangian relaxation of the tight constraints

By applying Equation (17), we can formulate the following new constraints:

$$p_e = 1 - \exp(-\lambda_e) = 1 - \exp(-\chi(\boldsymbol{x_e})). \tag{18}$$

However, incorporating the new Constraints (18) into model (15) may result in infeasibility even with the linear regression model. Let $\boldsymbol{X}$ denote the coefficient matrix, where each row of $\boldsymbol{X}$ is given by $\sum_{e\in E(v)} \boldsymbol{x_e}^T$. Let $\boldsymbol{y}$ denote the vector corresponding to $-log(1-\rlap{\,/}p_v)$. We demonstrate such infeasibility with $\boldsymbol{X}$ and $\boldsymbol{y}$ by Proposition 4 under Assumption 1.

Assumption 1. *The occurrences of errors follow the Poisson process and the VBS can detect the occurrences of errors, that is, Equations (2) through (5) hold.*

Proposition 4. *Given Assumption 1 and $\chi(\boldsymbol{x_e}) = \boldsymbol{\beta}\boldsymbol{x_e}^T$ hold, the system of equations $\boldsymbol{y} = \boldsymbol{\beta}\boldsymbol{X}$ can be derived and therefore model (15) with Constraints (18) is infeasible when $rank(\boldsymbol{X}) < rank([\boldsymbol{X}, \boldsymbol{y}])$.*

Proof See Appendix C.4. □

Proposition 4 demonstrates that the inclusion of latent factors $\boldsymbol{x_e}$ in the optimization model may lead to infeasibility due to the implicit tight constraints $\boldsymbol{y} = \boldsymbol{\beta}\boldsymbol{X}$. Essentially, the infeasibility arises due to the neglected random error terms accounting for unobserved effects in Equation (17), which leads to an inconsistent and overdetermined system. To address this issue, we use the Lagrangian relaxation to transfer the tight constraints into the objective function, which represents a penalty. A solution to the relaxed problem is an approximate solution to the original problem. In order to bring the relaxed solution closer to the original optimal solution, the weight of the transferred objective function should be a big positive number $\mathcal{H}$. As a result, the new optimization model is formulated as follows:

$$\min_{\hat{q}_{e,t},\gamma_{e,t},z_{e,t},p_e} \quad \left[\mathcal{H}\|\boldsymbol{y}-\chi(\boldsymbol{X})\|_2^2, \mathcal{F}_1, \mathcal{F}_2\right]$$

$$\text{s.t.} \quad \delta_{v,t} \le \sum_{e\in E(v)} \gamma_{e,t} \le |E(v)|_{\#}\delta_{v,t}, \quad \forall 1\le v\le V, 1\le t\le T; \tag{19a}$$

$$\sum_{e\in E^+(v)} \hat{q}_{e,t} - \sum_{e\in E^-(v)} \hat{q}_{e,t} = 0, \quad \forall 1\le v\le V; \tag{19b}$$

$$\hat{q}_{e,t} = (1-\gamma_{e,t})q_{e,t} + z_{e,t}, \quad \forall 1\le e\le E, 1\le t\le T; \tag{19c}$$

$$z_{e,t} \le \gamma_{e,t}\Theta, \quad \forall 1\le e\le E, 1\le t\le T; \tag{19d}$$

$$\sum_{t=1}^{T}\gamma_{e,t} - 1 \le p_e T \le \sum_{t=1}^{T}\gamma_{e,t} + 1, \quad \forall 1\le e\le E; \tag{19e}$$

$$\hat{q}_{e,t}\ge 0,\ z_{e,t}\ge 0,\ \gamma_{e,t}\in\{0,1\},\ p_e\ge 0, \quad \forall 1\le e\le E, 1\le t\le T. \tag{19f}$$

3.6. Model properties

As discussed above, we can make use of the squared norm of random errors $\|\boldsymbol{y}-\chi(\boldsymbol{X})\|_2^2$ to account for the unobserved effects on $\boldsymbol{y}$. In light of this, it can be demonstrated that model (19) yields an unbiased estimation of p_e under suitable conditions even when χ is linear regression.

Assumption 2. *Let $\boldsymbol{\varepsilon}$ be the vector of the random error term. $\boldsymbol{y} = \boldsymbol{\beta}\boldsymbol{X} + \boldsymbol{\varepsilon}$ satisfies the general assumptions of the linear regression (e.g., linear relationship, independent variables, normal distributions) (Hayashi 2011).*

Proposition 5. *Under Assumptions 1 and 2, the optimal $\boldsymbol{\beta}$ for model (19) can be determined by solving the linear regression when T is sufficiently large. The optimal $\boldsymbol{\beta}$ and corresponding p_e are unbiased.*

Proof See Appendix C.5. □

Proposition 5 demonstrates the $\hat{p}_e$ produced by model (19) is optimal and unbiased under Assumptions 1 and 2. As a result, the value of the transferred objective function is 0 and the primary objection function $\mathcal{F}_1$ is a constant term (see Equation (9)). In other words, the model (19) can be equivalently decomposed and solved with a two-stage method: first, estimate the sensor error probabilities p_e, and then solve the simplified optimization model. When the relationship between λ_e and $\boldsymbol{x_e}$ is non-linear and unknown, neural networks can also be employed to estimate p_e, as detailed in Section 5. In fact, the effectiveness of model (19) mainly comes from the use of latent factors, which significantly reduce the number of estimated parameters, regardless of the specific form of the function.

We then explore the performance of model (19) in recovering traffic data. We show that our model is capable of achieving either an exact or stable recovery of the traffic flow data under certain assumptions.

Assumption 3. *The true flow matrix $\boldsymbol{Q}$ satisfies the general assumptions of matrix completion (i.e., low rank, uniformly distributed across all entries, at least $rank(\boldsymbol{Q})$ observations per column/row) (Candes and Plan 2010).*

Proposition 6. [*Exact recovery*] *Under Assumptions 1 through 3, model (19) can exactly recover the true traffic data when all erroneous data are correctly identified with the maximum likelihood (i.e., $\gamma_{e,t} = 1, \forall q_{e,t} \neq \bar{q}_{e,t}$ and $\gamma_{e,t} = 0, \forall q_{e,t} = \bar{q}_{e,t}$).*

Proof See Appendix C.6. □

Proposition 7. [*Stable recovery*] *When partial erroneous data are not correctly identified, our model ensures that the recovery error $\|\hat{\boldsymbol{Q}} - \boldsymbol{Q}\|_2^2 \leq 4\sqrt{\frac{(3-r)min(E,T)}{1-r}}U + 2U$, where $U = \sum_{e,t\in\mathcal{Z}}(\xi_{e,t})^2$, $\mathcal{Z} = \{(e,t)|\gamma_{e,t} = 0, q_{e,t} \neq \bar{q}_{e,t}\}$, and $r = \frac{card(\left\{\xi_{e,t}|\xi_{e,t}\neq 0, 1\leq e\leq E, 1\leq t\leq T\right\})}{E*T}$.*

Proof See Appendix C.7. □

Proposition 6 reveals that the key to the accurate recovery of traffic flow data is to ensure a minimum of $rank(\boldsymbol{Q})$ correct observations for each column/row, that is, $\sum_{t=1}^{T}\gamma_{e,t} \leq T - rank(Q)$ or $\sum_{e=1}^{E}\gamma_{e,t} \leq T - rank(Q)$. This implies that we can appropriately relax the constraints to achieve better performance by estimating $rank(Q)$ in prior. Moreover, it also suggests that traffic data can still be successfully recovered even when some correct data are mistakenly identified as erroneous, owing to the global low-rank structure. In contrast, the unidentified erroneous data would mislead the reconstruction process and yield unexpected results. Proposition 7 demonstrates that the model performance remains reliable even in the presence of minor stochastic disturbances within the traffic data. These propositions help us to design effective algorithms for solving the optimization model.

4. Solution method

In this section, we first estimate sensor error probabilities p_e based on Proposition 5, which simplifies model (19) to a convex MINLP. Subsequently, the generalized benders decomposition algorithm is used to solve the simplified model. Moreover, solution strategies applicable to large-scale problems are discussed.

4.1. Toward a single convex objective function

Proposition 5 demonstrates the optimal p_e for model (19) can be determined by minimizing the transferred objection function. With the determined error probability $\hat{p}_e$, both the transferred objective function and primary objection function $\mathcal{F}_1$ of model (19) simplify to constants. Therefore, the original multi-objective function is transformed into a single objective function $\mathcal{F}_2$: $rank(\hat{\boldsymbol{Q}})$. To efficiently solve the model, we replace the $rank(\hat{\boldsymbol{Q}})$ with the nuclear norm of the matrix $rank(\hat{\boldsymbol{Q}})$

to transform the non-convex objective into a convex one, which has been widely adopted in the existing studies (Chen, He, and Sun 2019, Feng et al. 2022). Consequently, this leads to a single convex objective function: $\mathcal{F}_2 \approx \|\hat{\boldsymbol{Q}}\|_*$.

4.2. Generalized benders decomposition

Although the model with the new objective is a convex mixed integer programming model, it remains challenging with current state-of-the-art commercial solvers. To tackle this challenge, we employ the Generalized Benders Decomposition (GBD), an efficient approach for addressing nonlinear mixed integer optimization problems, to solve the model. Specifically, the GBD first decomposes the complex optimization problem into a primal problem and a master problem. Then, the primal and master problems are solved iteratively. Meanwhile, the solution of one problem would provide information or constraints to the other problem, until the optimal solution is found (see Appendix D for details).

For large-scale problems, the GBD algorithm may not efficiently solve the optimization model. The reason is that the master problem, an integer programming model, is still intractable. To address this issue, we propose a more efficient algorithm that can be used to solve large-scale problems. Specifically, the algorithm first treats $\gamma_{e,t}$ as continuous variables, thereby the model is transformed into a convex optimization problem that can be solved by the alternating direction method of multipliers (ADMM) method. Then, the algorithm generates solutions for the original model based on the continuous $\gamma_{e,t}$. The basic idea is to generate solutions that maximize $\sum_{e=1}^{E}\sum_{t=1}^{T}\gamma_{e,t}$ while satisfying the constraints. This is because the misidentification of correct data as erroneous marginally affects the data recovery performance due to the global low-rank structure of traffic flow, as discussed in Section 3.6. Moreover, we can incorporate prior domain knowledge into the algorithm. For example, if some sensors are recently calibrated, we can assign 0 to the corresponding $\gamma_{e,t}$. Although the algorithm may fail to find the global optimum, the experiments in Section 6 show empirically that it produces good results. The detailed procedure of the algorithm is shown in Appendix D.

5. Smart estimate-then-optimize framework for erratic errors

In Section 3, we prove that the integrated estimation and optimization problem formulated by model (19) can be equivalently decomposed and solved using a two-stage method. However, this equivalent decomposition relies on two key assumptions: (1) The occurrences of erratic errors follow a Poisson process; and (2) The relationship between the error arrival rate λ_e and features $\boldsymbol{x_e}$ is linear. When these assumptions fail in more realistic scenarios, it is difficult to obtain unbiased estimates of p_e, which may result in suboptimal performance of the optimization model. Consequently, a simple two-stage method cannot well address the integrated problem. To overcome this challenge, we propose a smart estimate-then-optimize framework in this section.

5.1. Basic idea of smart estimate-then-optimize

In practice, the integration of estimation and optimization is usually used for parameter estimation and informed decision making (Liu and Grigas 2021, Elmachtoub and Grigas 2022). For example, many mobility applications aim to minimize a traveler's commute time from origin to destination by solving a shortest-path optimization model, where the travel time on each road is unknown and need to be estimated. Typically, ML tools are used to develop an estimation model that estimates these unknown parameters for the optimization model. These estimated parameters then complete the specification of the optimization model, which is subsequently solved to make final decisions.

Intuitively, such an integrated problem can be solved by separating the estimation and optimization into two stages: first, training an accurate estimation model, and then solving the optimization model using the estimated parameters. However, perfect estimations without errors are impracticable, and estimation errors do not appropriately measure the quality of decisions. Consequently, as highlighted in many existing studies (Kotary, Fioretto, and Van Hentenryck 2021, Tang and Khalil 2022), directly training an estimation model based solely on estimation errors tends to produce worse decisions than incorporating decision errors into the training process.

To address this challenge, an SEO framework that combines ML techniques and optimization analysis is proposed as a transformative paradigm for data-driven decision making, as shown in Figure 3. This framework integrates estimation into optimization and takes into account its impact on the final decision. In contrast to the two-stage methods, the SEO framework embeds an optimization layer within the ML training loop, which ensures that the optimization process is informed by ML estimations and compensates for potential inaccuracies. As a result, the integration can lead to more accurate and reliable decisions (Elmachtoub and Grigas 2022, Tang and Khalil 2022). More importantly, even when the estimation problem is challenging, the final optimization model can still perform well. Recent studies on SEO (Liu and Grigas 2021, Zhang et al. 2024) have also shown remarkable performance in solving a variety of integrated problems.

5.2. SEO formulation for erratic errors

In our work, we develop an SEO framework to integrate the estimation of sensor error probabilities into data recovery optimization. We estimate sensor error probabilities using neural networks, and embed the optimization model into the neural network training loops as an optimization layer. In particular, the SEO framework involves the unsupervised estimation of sensor error probabilities, which can be effectively addressed using observations from VBS. Moreover, the traffic domain knowledge discussed in Section 3 is preserved within the SEO framework.

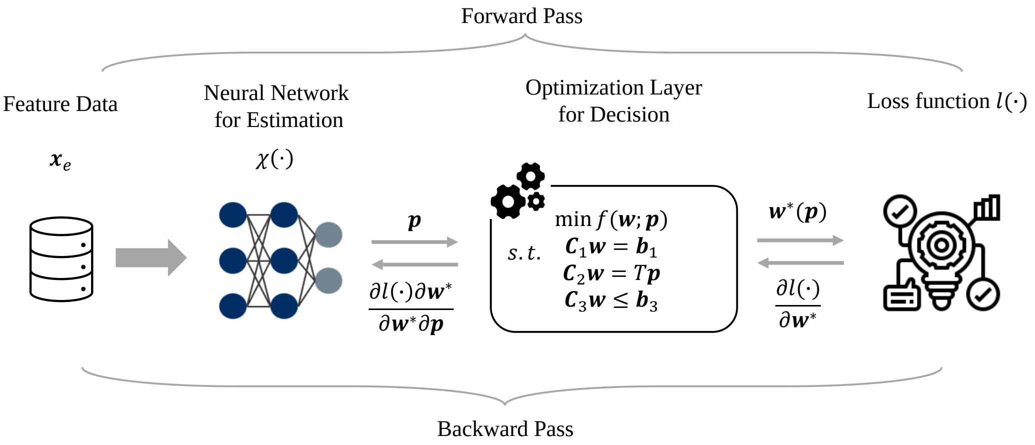

Forward Pass
Feature Data
$\boldsymbol{x}_e$
Neural Network for Estimation
$\chi(\cdot)$
$\boldsymbol{p}$
$\frac{\partial l(\cdot)\partial \boldsymbol{w}^*}{\partial \boldsymbol{w}^*\partial \boldsymbol{p}}$
Optimization Layer for Decision
$\min f(\boldsymbol{w};\boldsymbol{p})$
$s.t.$ $\boldsymbol{C}_1\boldsymbol{w} = \boldsymbol{b}_1$
$\boldsymbol{C}_2\boldsymbol{w} = T\boldsymbol{p}$
$\boldsymbol{C}_3\boldsymbol{w} \leq \boldsymbol{b}_3$
$\boldsymbol{w}^*(\boldsymbol{p})$
$\frac{\partial l(\cdot)}{\partial \boldsymbol{w}^*}$
Loss function $l(\cdot)$
Backward Pass

Figure 3 **Illustration of the SEO framework.**

5.2.1. An integrated neural network Within the SEO framework, the reliance on the Poisson process and linear assumptions is eliminated. Instead, we model the sensor error probabilities p_e as being directly influenced by the features $\boldsymbol{x}_e$, which can be formulated as follows:

$$p_e = \chi(\boldsymbol{x}_e), \tag{20}$$

where $\chi(\cdot)$ denotes a variety of neural networks. Then, the estimated sensor error probabilities p_e serve as input parameters of the data recovery model. This results in a parameterized optimization model, which can be summarized in matrix form as follows:

$$\min_{\boldsymbol{w}} \;\; f(\boldsymbol{w};\boldsymbol{p})$$

$$\text{s.t.} \quad \boldsymbol{C}_1\boldsymbol{w} = \boldsymbol{b}_1; \;\; \boldsymbol{C}_2\boldsymbol{w} = \boldsymbol{p}T; \;\; \boldsymbol{C}_3\boldsymbol{w} \leq \boldsymbol{b}_3; \tag{21a}$$

$$\boldsymbol{w} = [\hat{\boldsymbol{q}}, \boldsymbol{\gamma}, \boldsymbol{z}]; \tag{21b}$$

$$\hat{q}_{e,t} \geq 0, \;\; z_{e,t} \geq 0, \;\; \gamma_{e,t} \in \{0,1\}, \quad \forall 1 \leq e \leq E, 1 \leq t \leq T, \tag{21c}$$

where $\boldsymbol{w}$ is the vector of decision variables. The vector $\boldsymbol{p}$ is composed of sensor error probabilities p_e, which are learned from feature data $\boldsymbol{x}_e$ using neural networks. The function $f(\boldsymbol{w};\boldsymbol{p})$ represents the nuclear norm of the recovered flow matrix with parameter $\boldsymbol{p}$. The matrices $\boldsymbol{C}_1$, $\boldsymbol{C}_2$, and $\boldsymbol{C}_3$ are coefficient matrices in the constraints, and $\boldsymbol{b}_1$ and $\boldsymbol{b}_3$ are known constants.

From the perspective of SEO, the optimization model (21) is viewed as a special neural network layer that is directly connected to $\chi(\boldsymbol{x}_e)$, as illustrated in Figure 3. During the forward pass, the neural networks first estimate the sensor error probabilities $\boldsymbol{p}$ from feature data $\boldsymbol{x}_e$. The estimated $\boldsymbol{p}$ then completes the specification of the optimization model, which is solved using the tractable algorithm in Appendix D. The solution of model (21) becomes the output of the optimization layer, which indicates that the value of $f(\boldsymbol{w};\boldsymbol{p})$ varies with respect to $\boldsymbol{p}$ or the parameters of the neural networks. In this way, the estimation and optimization problems are successfully integrated into the neural networks.

5.2.2. Derivatives of solutions with respect to estimated parameters Training the SEO framework involves a backward pass based on the solution of the optimization model. It is required to compute the derivative of the solutions with respect to the input parameter $\boldsymbol{p}$. For model (21), the derivatives cannot be directly obtained due to the presence of integer decision variables. We adopt the continuous relaxation of the integer variables, which can achieve comparable performance (Tang and Khalil 2022, Zhang et al. 2024). After relaxation, model (21) becomes a convex optimization problem, allowing the derivatives to be obtained by differentiating its KKT conditions. The Lagrangian of the relaxed model is formulated as follows:

$$\mathscr{L}(x,\boldsymbol{\nu}_1,\boldsymbol{\nu}_2,\boldsymbol{\nu}_3) = f(\hat{Q};\boldsymbol{p}) + \boldsymbol{\nu}_1^\top(\boldsymbol{C}_1 x - \boldsymbol{b_1}) + \boldsymbol{\nu}_2^\top(\boldsymbol{C}_2\boldsymbol{w} - T\boldsymbol{p}) + \boldsymbol{\nu}_3^\top(\boldsymbol{C_3 w} - \boldsymbol{b_3}), \tag{22}$$

where $\boldsymbol{\nu}_1$, $\boldsymbol{\nu}_2$, and $\boldsymbol{\nu}_3 \geq 0$ are the vectors of Lagrange multipliers. The KKT conditions for stationarity, primal feasibility, and complementary slackness are formulated as follows:

$$\nabla_{\boldsymbol{w}}\mathscr{L}(\boldsymbol{w}^*,\boldsymbol{\nu}_1^*,\boldsymbol{\nu}_2^*,\boldsymbol{\nu}_3^*) = \nabla_{\boldsymbol{w}} f(\boldsymbol{w}^*) + \boldsymbol{C}_1^\top\boldsymbol{\nu}_1^* + \boldsymbol{C}_2^\top\boldsymbol{\nu}_2^* + \boldsymbol{C}_3^\top\boldsymbol{\nu}_3^* = 0; \tag{23}$$

$$\boldsymbol{C}_1\boldsymbol{w}^* = \boldsymbol{b_1}, \quad \boldsymbol{C}_2\boldsymbol{w}^* = \boldsymbol{p}T; \tag{24}$$

$$D(\boldsymbol{\nu}_3^*)(\boldsymbol{C}_3\boldsymbol{w}^* - \boldsymbol{b_3}) = 0; \tag{25}$$

where $D(\cdot)$ creates a diagonal matrix from a vector, $\boldsymbol{w}^*, \boldsymbol{\nu}_1^*, \boldsymbol{\nu}_2^*$ and $\boldsymbol{\nu}_3^*$ are the optimal solutions of model (21). By taking the differentials of these conditions with respect to $\boldsymbol{p}$, we can obtain a system of linear equations:

$$\begin{bmatrix} \nabla^2 f(\boldsymbol{w}^*) & \boldsymbol{C}_1^\top & \boldsymbol{C}_2^\top & \boldsymbol{C}_3^\top \\ \boldsymbol{C}_1 & 0 & 0 & 0 \\ \boldsymbol{C}_2 & 0 & 0 & 0 \\ D(\boldsymbol{\nu}_3^*)\boldsymbol{C}_3 & 0 & 0 & D(\boldsymbol{C}_3\boldsymbol{w}^* - \boldsymbol{b}_3) \end{bmatrix} \begin{bmatrix} \frac{\partial \boldsymbol{w}^*}{\partial \boldsymbol{p}} \\ \frac{\partial \boldsymbol{\nu}_1^*}{\partial \boldsymbol{p}} \\ \frac{\partial \boldsymbol{\nu}_2^*}{\partial \boldsymbol{p}} \\ \frac{\partial \boldsymbol{\nu}_3^*}{\partial \boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ T\boldsymbol{I} \\ 0 \end{bmatrix}. \tag{26}$$

Solving the linear equations provides the derivatives $\frac{\partial \boldsymbol{w}^*}{\partial \boldsymbol{p}}$, which can then be used to update the parameters of neural networks during the backward pass.

5.2.3. Loss function It is worth noting that our approach for estimating sensor error probabilities is fundamentally unsupervised, given that ground truth data are typically unavailable in the real world. However, as previously mentioned, the known error probabilities of VBS, denoted as $\boldsymbol{\varrho}_v$, can support for the estimation of p_e. Consequently, our loss function is designed not only to minimize the objective function of model (19) but also to incorporate a penalty term that addresses discrepancies between the observed and estimated VBS error probabilities. Let $\hat{\boldsymbol{\varrho}}_v$ denote the VBS error probabilities derived by neural networks. The loss function ℓ is formulated as follows:

$$\ell = f(\boldsymbol{w};\boldsymbol{p}) + \Theta(\hat{\boldsymbol{\varrho}}_v - \boldsymbol{\varrho}_v)^2. \tag{27}$$

To summarize, the forward and backward pass of the SEO framework are outlined as follows:

Algorithm 1 Forward and backward pass for SEO

```
Require: Feature data $\boldsymbol{x}_e$ and observed flow $q_{e,t}$
Initialize neural network parameters $\theta$ for estimator $\chi(\boldsymbol{x}_e;\theta)$.
for epochs do
    Estimate the sensor error probabilities $\boldsymbol{p}$ with $\chi(\boldsymbol{x}_e;\theta)$.
    Solve model (21) with parameter $\boldsymbol{p}$ to obtain the optimal solution $\boldsymbol{w}^*(\boldsymbol{p})$.
    Compute the loss $\ell$ with Equation (27).
    Compute the gradient $\frac{\partial \ell}{\partial \boldsymbol{p}} = \frac{\partial \ell}{\partial \boldsymbol{w}^*}\frac{\partial \boldsymbol{w}^*}{\partial \boldsymbol{p}}$ by solving Equation (26).
    Update parameters $\theta$ for estimator $\chi(\boldsymbol{x}_e;\theta)$ with the computed gradient.
end for
Return $\boldsymbol{p}$ and $\boldsymbol{w}^*$.
```

6. Numerical experiments

In this section, we validate the proposed model using both Nguyen-Dupuis network and a real-world Sha Tin county network in Hong Kong.

6.1. Numerical experiments using the Nguyen-Dupuis network

6.1.1. Experimental setup We first use the same modified Nguyen-Dupuis network in Yang, Yang, and Fan (2019) to validate our proposed framework. Compared with the classic Nguyen-Dupuis network, the modified network is bidirectional and presents a more complicated topology structure as shown in Figure 4. Specifically, the network is comprised of 6 origin/destination nodes (denoted by double circles), along with 19 intermediate nodes and 50 links. We also employ the same method in existing studies (Yin et al. 2017, Yang, Yang, and Fan 2019) to generate the samples of road traffic by randomizing flows using the traffic assignment algorithm.

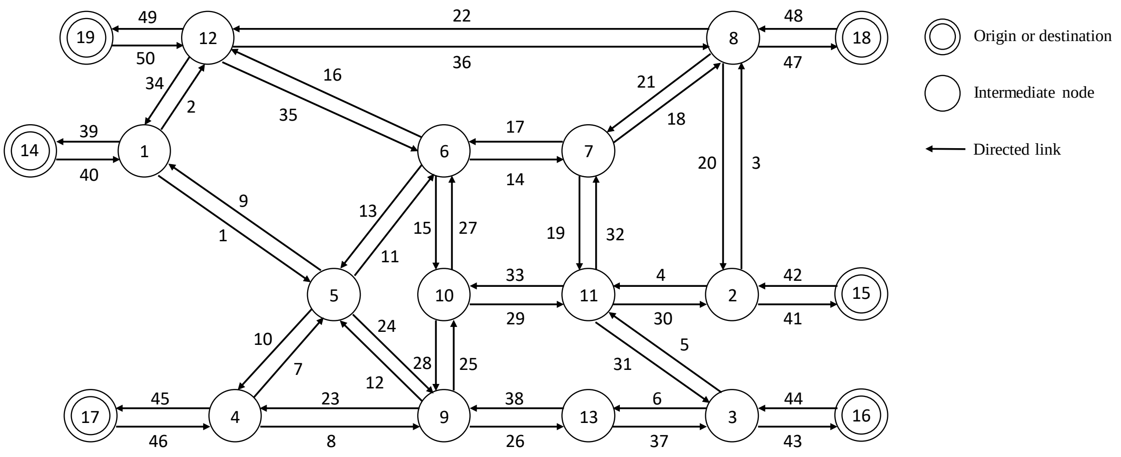


Figure 4 Illustration of the modified Nguyen-Dupuis network.

Following the similar experimental setup established in the existing studies (Yin et al. 2017, Yang, Yang, and Fan 2019), the sensors are deployed on all links. Due to the impact of exogenous factors and random variations, erratic measurement errors may occur during the data collection. In this paper, we use four categories of exogenous features, encompassing a total of 8 explanatory variables to characterize the arrival rate of measurement errors. Table 2 provides a summary of these variables and their respective coefficients, indicating the effects on the arrival rate λ_e. For a given variable, the positive coefficient implies that as the value of the explanatory variable increases, the mean of the arrival rates also tends to increase. Consequently, the corresponding sensor error probability would increase. Similarly, for the negative coefficient, as the value of the explanatory variable increases, the mean of the arrival rates tends to decrease, leading to a lower sensor error probability.

We use the traffic flow data with erratic measurement errors as the input of the proposed framework. This implies that the distributions and structures of the measurement errors are unknown in the experiment. Moreover, the erratic measurement errors indicate that multiple error distributions, such as Gaussian distribution, Uniform distribution, Exponential distribution, and Gamma distribution coexist in the traffic data (see Appendix E for details). The error probabilities of sensors are determined by the exogenous variables in Table 2 using linear regression. We randomly assign the values of these explanatory variables for all sensors. The corresponding error probabilities are summarized in Table 3. Given the error probabilities, we adopt the similar method in Zheng and Su (2016) to determine the magnitudes of erratic measurement errors, which are set to $[1,2,3]\sigma$, where σ is the standard deviation of the collected data. As a result, the collected data with erratic measurement errors can be obtained.

Table 2 True and estimated coefficients of the explanatory variables.

Categories	Variables	Types	True coefficients	Estimated coefficients	MAE
Sensor features	installation time (months)	continuous	0.0100	0.0090	0.0010
	sensitivity (%)	continuous	0.0100	0.0101	0.0001
Link features	link length (km)	continuous	0.0050	0.0056	0.0006
	lane number	discrete	0.0010	0.0006	0.0004
	speed limit (km/h)	continuous	-0.0010	-0.0010	0.0000
Traffic condition	traffic index	continuous	-0.0040	-0.0051	0.0011
Weather conditions	foggy or rainy	dummy	0.1000	0.1045	0.0045
	sunny	dummy	\	\	\

6.1.2. Experimental results The input to the proposed framework is the collected traffic flow with erratic measurement errors and exogenous variables. Note that the coefficients of exogenous variables are not known in advance. The framework then outputs the estimated error probabilities of sensors and recovered traffic flow. To be specific, we first use the approach introduced in

Section 3.5.3 to estimate the error probabilities of real sensors. The performance of our approach is evaluated with the mean absolute error (MAE) and mean absolute percent error (MAPE):

$$MAE = |\hat{p}_e - p_e|, \qquad MAPE = \frac{|\hat{p}_e - p_e|}{p_e} \times 100\%, \tag{28}$$

where $\hat{p}_e$ and p_e are the estimated and true error probability of sensor e, respectively. We summarize the results in Table 3, where the estimated and true error probabilities of all sensors are reported. The average MAE and MAPE of the estimated error probabilities are 0.0061 and 0.0325, respectively. We also visualize the estimated error probabilities in Figure 5. The results demonstrate the satisfactory performance of our approach, aligning with the theoretical analysis presented in Section 3.6. In addition, we provide the estimated coefficients of exogenous variables in Table 2.

Table 3 True and estimated error probabilities (EP) of sensors.

Sensor	True EP	Estimated EP	MAE	MAPE	Sensor	True EP	Estimated EP	MAE	MAPE
1	0.1485	0.1451	0.0034	0.0230	26	0.3849	0.3803	0.0046	0.0119
2	0.0531	0.0504	0.0027	0.0505	27	0.1697	0.1654	0.0042	0.0249
3	0.1531	0.1512	0.0019	0.0123	28	0.2368	0.2387	0.0019	0.0079
4	0.1234	0.1182	0.0052	0.0422	29	0.1840	0.1805	0.0035	0.0190
5	0.1146	0.1119	0.0028	0.0241	30	0.1637	0.1615	0.0022	0.0137
6	0.1654	0.1603	0.0051	0.0311	31	0.1785	0.1756	0.0029	0.0165
7	0.1487	0.1465	0.0022	0.0145	32	0.1475	0.1427	0.0048	0.0324
8	0.3373	0.3332	0.0041	0.0121	33	0.1529	0.1525	0.0004	0.0027
9	0.0884	0.0881	0.0003	0.0035	34	0.0606	0.0591	0.0014	0.0238
10	0.0485	0.0449	0.0036	0.0742	35	0.0879	0.0814	0.0065	0.0744
11	0.1870	0.1837	0.0033	0.0177	36	0.1022	0.0962	0.0061	0.0593
12	0.2340	0.2331	0.0009	0.0040	37	0.0496	0.0457	0.0039	0.0795
13	0.0724	0.0714	0.0010	0.0136	38	0.3699	0.3634	0.0065	0.0177
14	0.1121	0.1096	0.0025	0.0226	39	0.0827	0.0828	0.0001	0.0007
15	0.4202	0.4173	0.0028	0.0068	40	0.1824	0.1784	0.0040	0.0220
16	0.2433	0.2418	0.0015	0.0062	41	0.5257	0.5162	0.0095	0.0182
17	0.1682	0.1641	0.0041	0.0244	42	0.1498	0.1450	0.0048	0.0320
18	0.0692	0.0676	0.0016	0.0236	43	0.0724	0.0703	0.0020	0.0279
19	0.1672	0.1603	0.0069	0.0415	44	0.2923	0.2920	0.0003	0.0010
20	0.1711	0.1678	0.0033	0.0192	45	0.3102	0.3032	0.0070	0.0227
21	0.0756	0.0699	0.0057	0.0754	46	0.2937	0.2890	0.0047	0.0162
22	0.0761	0.0709	0.0052	0.0684	47	0.1355	0.1341	0.0014	0.0103
23	0.3416	0.3363	0.0053	0.0155	48	0.1802	0.1744	0.0058	0.0320
24	0.2069	0.2008	0.0060	0.0291	49	0.1251	0.1216	0.0035	0.0280
25	0.1377	0.1345	0.0032	0.0232	50	0.1963	0.1974	0.0011	0.0056
Average MAE: 0.0036					Average MAPE: 0.0256				

The data recovery results of the optimization model are summarized in Table 4. Additionally, we visualize the observed flow and recovered flow in Figure 6 for the error magnitude σ. Figure 6(a) compares the observed flow, recovered flow against the true flow of all sensors on the network. Figure 6(b) shows the average observed flow, recovered flow, and true flow of all sensors among the estimation time horizon. According to these results, we find that the residual errors post-optimization are notably less than the initial errors in the data, which demonstrates the effectiveness of our approach in recovering the traffic flow data.

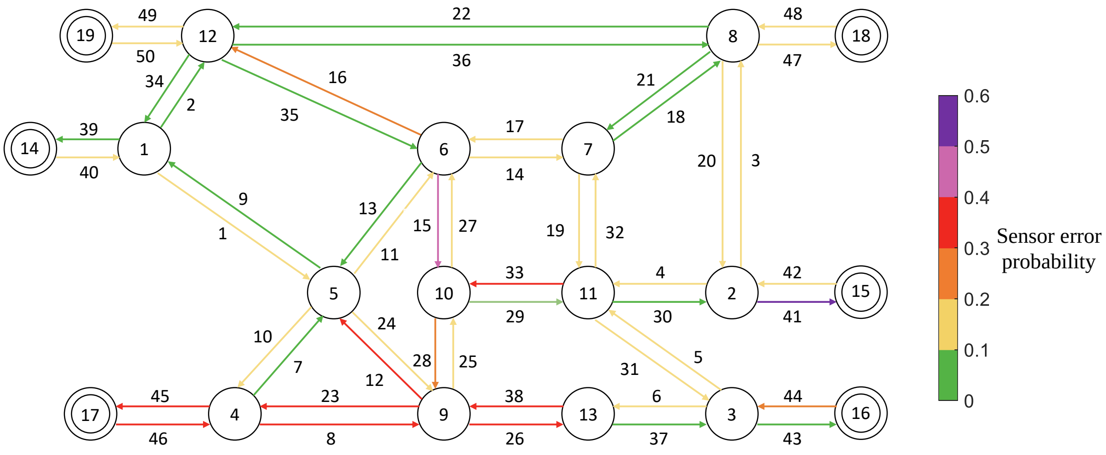
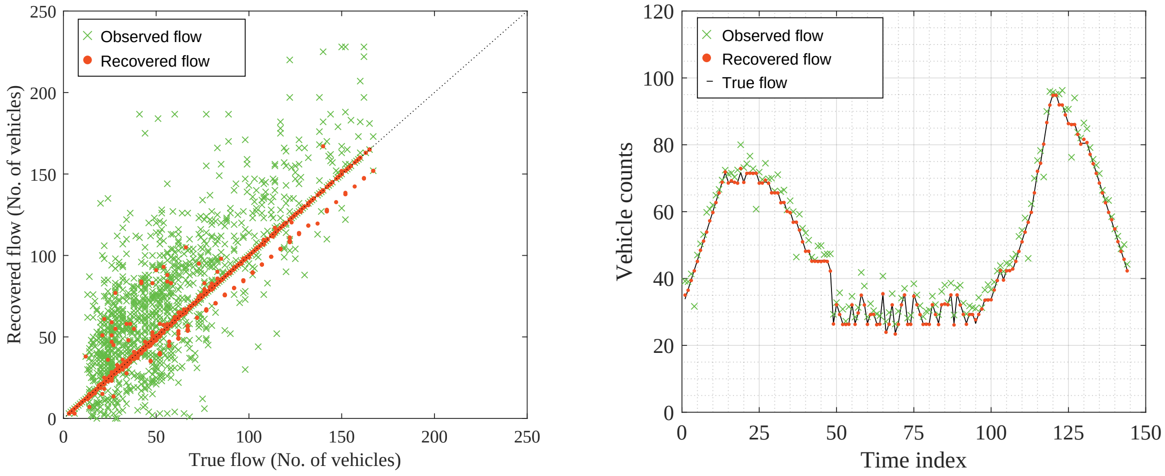


Figure 5 Estimated error probabilities on the modified Nguyen-Dupuis network.

Table 4 Data recovery on the modified Nguyen-Dupuis network.

Magnitudes of errors	Initial errors in the data		Residual errors after optimization		Reduction in MAE/MAPE
	MAE	MAPE	MAE	MAPE	
σ	4.1431	0.0986	0.4538	0.0088	3.6893/0.0898
2σ	9.0439	0.2521	3.7679	0.1172	5.2760/0.1349
3σ	12.9835	0.3570	4.6752	0.1505	8.3083/0.2065
Average	8.8715	0.2444	3.1812	0.1003	5.6904/0.1441



(a) Recovered traffic flow of all sensors.

(b) Recovered traffic flow in different time intervals.

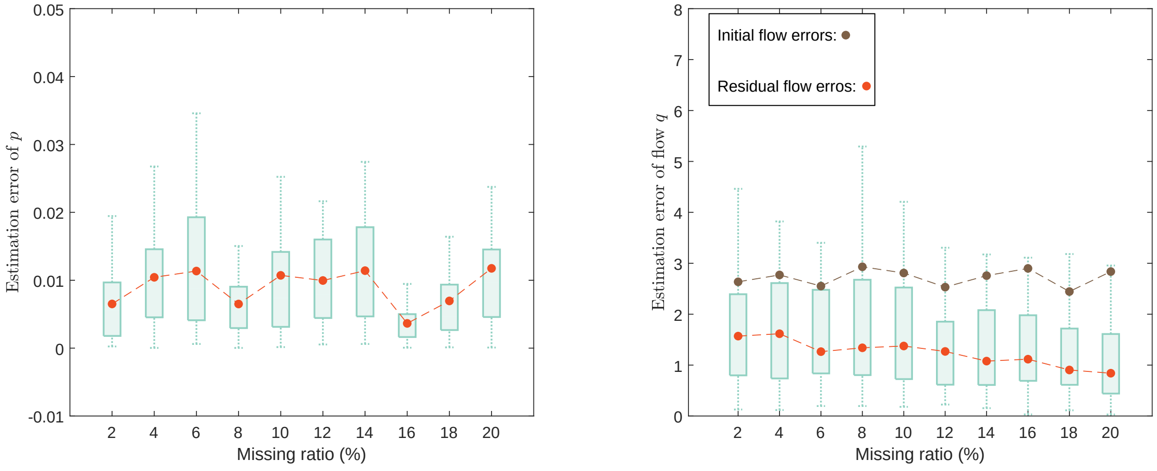
Figure 6 Recovered traffic flow on the modified Nguyen-Dupuis network for the error magnitude σ.

6.1.3. Data missing scenario In the above experiments, we use the traffic flow data with full observation as the input. However, the missing values may exist in the collected data. As discussed in Section 1, we can take the missing data as a special case of erratic measurement errors with known positions. Under this setting, we test the performance of the proposed framework. Let M_r

be the missing ratio in the erroneous data:

$$M_r = \frac{N_m}{N_e}, \tag{29}$$

where N_m and N_e are the number of missing data and erroneous data, respectively.



(a) Estimation error of p.

(b) Estimation error of q.

Figure 7 Estimation error of p and q under the data missing scenario.

Using the incomplete data with measurement errors as the input, the proposed framework can still output the estimations of sensor error probability p and traffic flow q. To handle the missing values, we add a new constraint:

$$\gamma_{e,t} = 1, \quad \forall (e,t) \in \Phi_m, \tag{30}$$

where Φ_m indicates the set of missing data. Figure 7(a) presents the average absolute estimation error of p as the missing ratio increases from 2% to 20%. We can see that there is no significant change in the average estimation error of p, which is in line with our expectations. This is because the missing data can always be correctly identified by Equation (30) and therefore the sensor error probability p can be accurately estimated. The box plots also provide an overview of the estimation errors for all sensors. We can see that the majority of the estimation errors lie in the interval $[0, 0.02]$.

Figure 7(b) compares the estimation errors of flow against the initial flow errors. An interesting result is that the average estimation error of flow gradually decreases with the increase of the missing ratio. This is not surprising because the positions of missing values are known in advance. Consequently, the missing data provide more information to the data recovery model when compared to the erratic measurement errors with unknown positions. As the missing ratio increases,

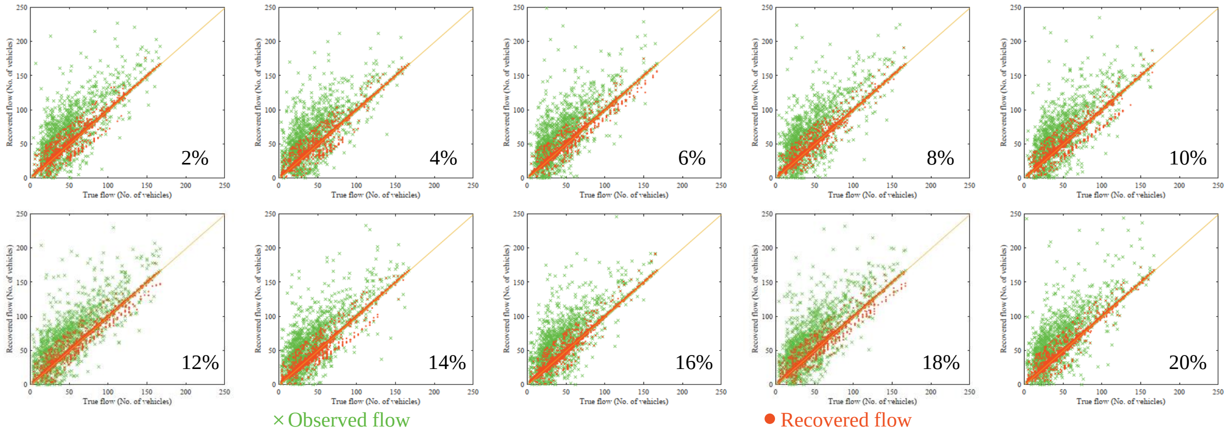


Figure 8 **Observed flow and recovered flow under the data missing scenario.**

more information is provided and therefore improves the performance of data recovery. Such results also validate our assertion that the missing data can be viewed as a special case of erratic measurement errors. Additionally, we present the observed flow and recovered flow for each missing ratio in Figure 8, which demonstrates the effectiveness of our optimization model under the data missing scenario.

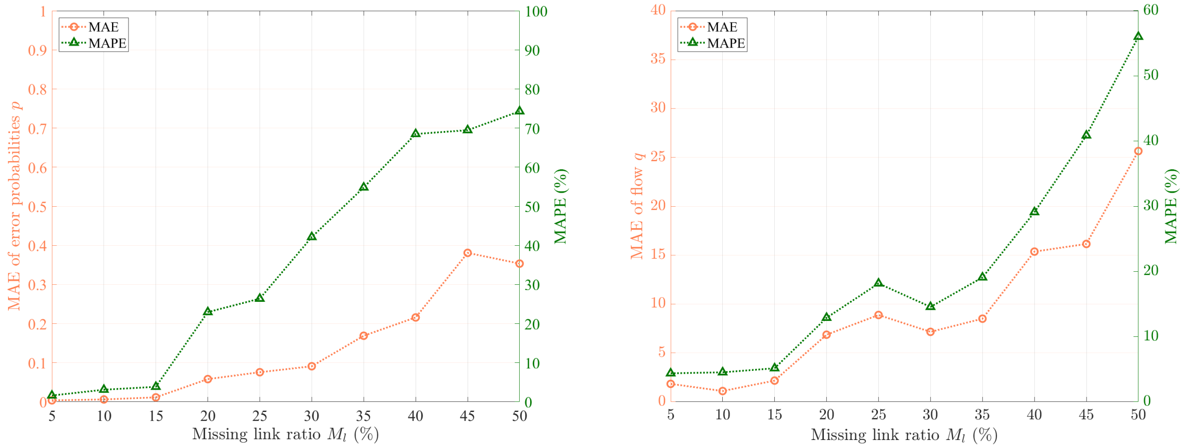
We have also examined the model performance when entire data from certain links are missing. We randomly select some links where all data are unobserved and generate erratic measurement errors for other links. The missing link ratio M_l is defined as follows:

$$M_l = \frac{Number\ of\ unobserved\ links}{|E|_\#} \times 100\%. \tag{31}$$

The model is evaluated on the modified Nguyen-Dupuis network across a range of M_l values. Figures 9(a) and 9(b) illustrate the results of sensor error probabilities and recovery errors of traffic flow, respectively. It can be found that the proposed model remains effective when the value of M_l is small (i.e., $M_l \leq 15\%$). However, the model may not work effectively with higher M_l (i.e., $M_l \geq 20\%$) values. This occurs because the flow balance law and low-rank completion methods become ineffective when critical links in the network are unobserved. Consequently, the observed correct data are insufficient for accurate estimation and recovery.

6.1.4. Unbalanced flow scenario In the above experiments, we assume that the true flow data conform to the flow balance law. However, such requirement can hardly be fulfilled in real world networks due to disturbances in traffic conditions and other externalities. In view of this, it is essential to examine the performance and robustness of the proposed framework under the unbalanced flow scenario. We use the similar method in Yang, Yang, and Fan (2019) to generate the unbalanced flow data on link e during time interval t:

$$\tilde{q}_{e,t} = \bar{q}_{e,t} + \mu\sqrt{\bar{q}_{e,t}}\epsilon, \tag{32}$$



(a) Estimation error of p.

(b) Recovery error of q.

Figure 9 Model performance in the complete absence of certain links.

where $\bar{q}_{e,t}$ conforms to the flow balance law, $\tilde{q}_{e,t}$ is the adjusted flow that violates the law, μ is the disturbance parameter, and ϵ is a random term that follows the standard normal distribution. The corresponding flow imbalance ratio $r_{v,t}$ at node v during time interval t is defined as follows:

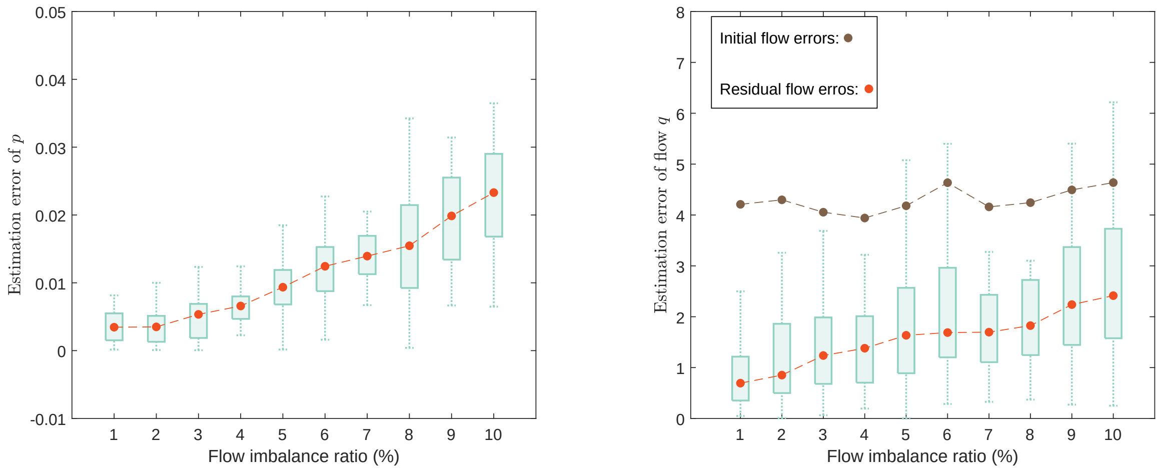
$$r_{v,t} = \frac{2\epsilon}{\sum_{e_1 \in E^+(v)} q_{e_1,t} + \sum_{e_2 \in E^-(v)} q_{e_2,t}}. \tag{33}$$

Using the unbalanced flow as the input, our approach then outputs the estimated error probabilities of sensors and recovered data. Figure 10(a) illustrates the average estimation error of p as the flow imbalance ratio increases from 1% to 10%. We can see that the average estimation error of p gradually increases to 0.02. This is because p_v defined in Equation (5) would be biased under the unbalanced flow scenario, resulting in a biased estimation. However, the box plots in Figure 10(a) indicate that most of the estimation errors fall within the interval $[0, 0.03]$. This suggests that the proposed framework can still produce satisfactory results even when unbalanced flow data are used as the input.

Figure 10(b) illustrates the residual flow errors resulting from our optimization model against the initial flow errors. Despite the gradual increase in the average estimation error of q as the flow imbalance ratio grows, the proposed framework remains effective for reducing the erratic measurement errors in the flow data. In addition, we present the observed flow and recovered flow in Figure 11 for each flow imbalance ratio. These results demonstrate the applicability of our optimization model under the unbalanced flow scenario.

6.2. Evaluation of the SEO framework

As discussed in Section 5, the equivalent decomposition relies on the assumptions of the Poisson process and linear correlation. However, these assumptions may not hold in practice. Therefore,



(a) Estimation error of p.

(b) Estimation error of q.

Figure 10 Estimation error of p and q under the unbalanced flow scenario.

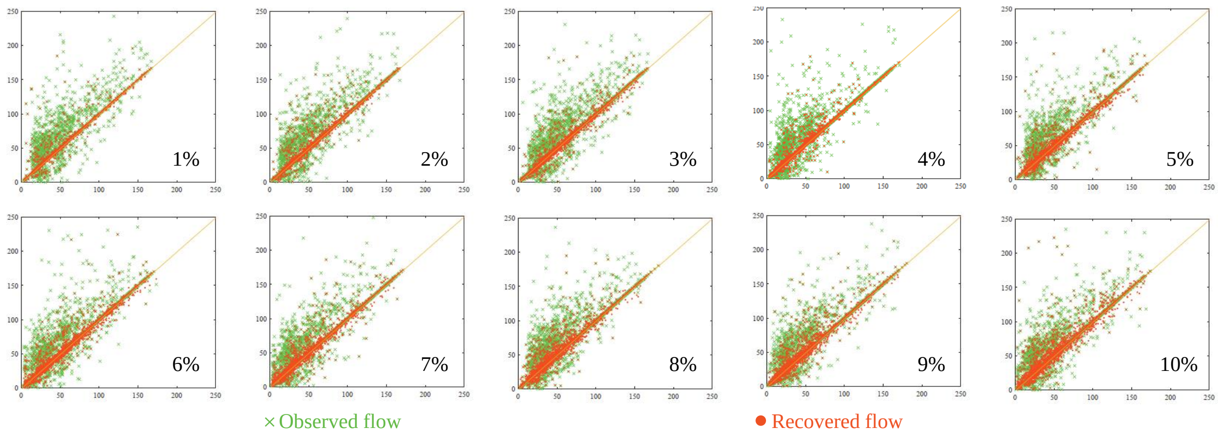


Figure 11 Observed flow and recovered flow under the unbalanced flow scenario.

we propose an SEO framework to address the challenges. In this section, we conduct experiments to evaluate the performance of our SEO framework. Specifically, we employ a non-linear function to describe the relationship between sensor error probabilities p_e and features $\boldsymbol{x}_e$. The specific function is formulated as follows:

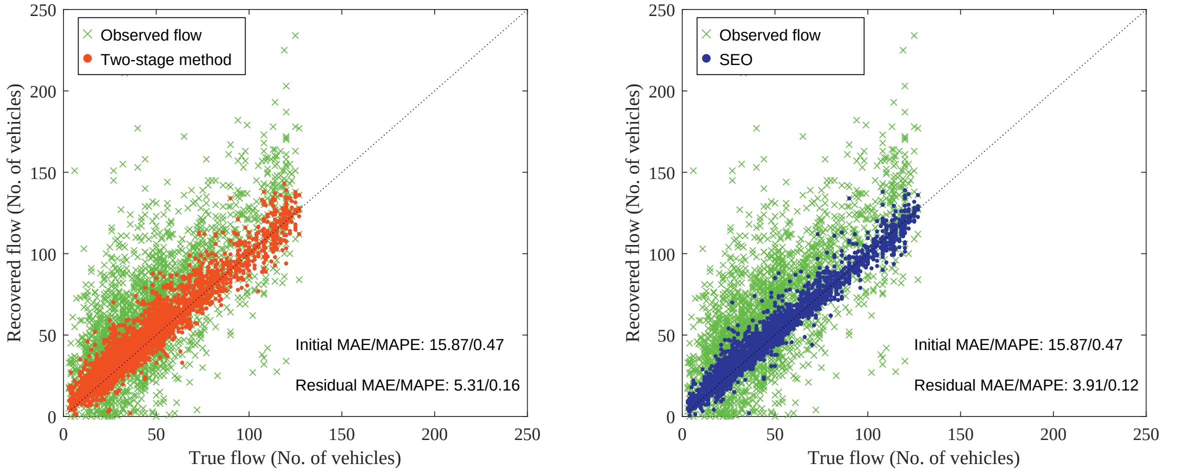
$$p_e = 0.05x_{e,1} + 0.5e^{-x_{e,2}} + 0.005x_{e,3}x_{e,4} - 0.001x_{e,5}^2. \tag{34}$$

In practice, training an estimation model from features is highly challenging, and achieving perfect estimation is infeasible. Therefore, we focus on evaluating the performance of the SEO framework even in the face of significant estimation challenges. To this end, we utilize only feature $x_{e,5}$ to estimate sensor error probabilities p_e in this experiment. We also compare the performance of the SEO framework with that of the two-stage method.

In Table 5, we compare the sensor error probabilities estimated by the SEO framework and the two-stage method when the assumptions do not hold. The two-stage method yields MAE of 0.1450 and MAPE of 0.2504 for the estimated error probabilities, while the SEO framework produces MAE of 0.1239 and MAPE of 0.2457. The inferior estimation accuracy of both methods is expected because we use only one feature to estimate the sensor error probabilities. Furthermore, Figure 12 compares the observed and recovered traffic flow of both methods against the true flow. Specifically, the two-stage method results in MAE of 5.31 and MAPE of 0.16, while the SEO framework achieves MAE of 3.91 and MAPE of 0.12. According to these results, we find that the SEO framework achieves satisfactory performance in estimating sensor error probabilities and recovering traffic flow even when the assumptions do not hold. We also find that the SEO framework can achieve better performance than the two-stage method, which is consistent with existing studies (Tang and Khalil 2022, Elmachtoub and Grigas 2022). Additionally, we observe that the SEO framework also slightly improves the estimation accuracy. This is because the estimation task in our research is unsupervised and therefore the two-stage method cannot achieve the best accuracy. In contrast, the estimation task in the SEO framework is also informed by the objectives of the optimization model, which can enhance the accuracy.

Table 5 True and estimated sensor error probabilities using the two-stage method and SEO framework.

Sensor	TP	Two-stage			SEO			Sensor	TP	Two-stage			SEO		
		EP	MAE	MAPE	EP	MAE	MAPE			EP	MAE	MAPE	EP	MAE	MAPE
1	0.6627	0.5758	0.0870	0.1312	0.7206	0.0578	0.0872	26	0.7849	0.5758	0.2091	0.2664	0.6317	0.1532	0.1952
2	0.7750	0.5762	0.1988	0.2566	0.6331	0.1420	0.1832	27	0.3471	0.5762	0.2290	0.6598	0.6637	0.3165	0.9119
3	0.7018	0.5760	0.1259	0.1793	0.7272	0.0254	0.0362	28	0.7959	0.5754	0.2205	0.2770	0.6525	0.1434	0.1802
4	0.7030	0.5752	0.1278	0.1818	0.7665	0.0635	0.0903	29	0.6488	0.5758	0.0730	0.1125	0.7605	0.1117	0.1722
5	0.5610	0.5745	0.0135	0.0241	0.7174	0.1564	0.2788	30	0.5437	0.5761	0.0324	0.0597	0.6518	0.1082	0.1990
6	0.7669	0.5747	0.1922	0.2506	0.6774	0.0895	0.1167	31	0.4623	0.5756	0.1133	0.2451	0.6843	0.2220	0.4803
7	0.6100	0.5762	0.0338	0.0555	0.6996	0.0896	0.1469	32	0.7185	0.5754	0.1431	0.1992	0.7023	0.0162	0.0226
8	0.7161	0.5763	0.1398	0.1952	0.6774	0.0386	0.0540	33	0.5793	0.5764	0.0029	0.0050	0.6386	0.0593	0.1023
9	0.7796	0.5764	0.2032	0.2607	0.6664	0.1132	0.1452	34	0.6640	0.5745	0.0896	0.1349	0.6781	0.0141	0.0212
10	0.5467	0.5759	0.0292	0.0534	0.6528	0.1061	0.1941	35	0.3243	0.5754	0.2510	0.7740	0.6774	0.3530	1.0884
11	0.7458	0.5757	0.1701	0.2281	0.7124	0.0334	0.0447	36	0.7234	0.5762	0.1473	0.2036	0.6715	0.0519	0.0718
12	0.6833	0.5754	0.1079	0.1579	0.6770	0.0064	0.0093	37	0.7192	0.5760	0.1432	0.1991	0.7252	0.0061	0.0084
13	0.6906	0.5762	0.1144	0.1657	0.7109	0.0203	0.0294	38	0.7454	0.5763	0.1691	0.2269	0.6700	0.0754	0.1011
14	0.7457	0.5756	0.1701	0.2281	0.6839	0.0618	0.0828	39	0.7327	0.5756	0.1570	0.2143	0.6912	0.0415	0.0566
15	0.7952	0.5791	0.2160	0.2717	0.6491	0.1461	0.1838	40	0.7539	0.5763	0.1776	0.2356	0.7493	0.0046	0.0061
16	0.3453	0.5761	0.2307	0.6682	0.6438	0.2985	0.8644	41	0.7614	0.5753	0.1861	0.2444	0.7313	0.0302	0.0396
17	0.3017	0.5758	0.2741	0.9085	0.7126	0.4109	1.3618	42	0.5738	0.5749	0.0011	0.0020	0.6399	0.0661	0.1153
18	0.7909	0.5760	0.2149	0.2717	0.6465	0.1444	0.1826	43	0.7530	0.5764	0.1766	0.2345	0.7118	0.0412	0.0547
19	0.7816	0.5763	0.2052	0.2626	0.6168	0.1647	0.2108	44	0.3835	0.5750	0.1915	0.4992	0.7026	0.3191	0.8320
20	0.7159	0.5760	0.1400	0.1955	0.7206	0.0046	0.0065	45	0.7068	0.5761	0.1307	0.1849	0.6343	0.0725	0.1026
21	0.7629	0.5763	0.1866	0.2446	0.6503	0.1126	0.1476	46	0.7252	0.5760	0.1491	0.2057	0.6519	0.0733	0.1011
22	0.7398	0.5759	0.1639	0.2216	0.6997	0.0401	0.0542	47	0.3948	0.5764	0.1816	0.4599	0.6810	0.2862	0.7250
23	0.7063	0.5756	0.1306	0.1850	0.6463	0.0600	0.0849	48	0.4104	0.5753	0.1649	0.4017	0.6964	0.2860	0.6969
24	0.7424	0.5760	0.1664	0.2241	0.6597	0.0827	0.1114	49	0.5038	0.5751	0.0712	0.1413	0.6678	0.1640	0.3255
25	0.3812	0.5753	0.1941	0.5092	0.6944	0.3132	0.8216	50	0.5749	0.5764	0.0015	0.0026	0.6586	0.0837	0.1456
Average MAE/MAPE of the two-stage method: 0.1450/0.2504								Average MAE/MAPE of the SEO framework:0.1239/0.2457							



(a) Recovered flow produced by the two-stage method.
(b) Recovered flow produced by the SEO framework.

Figure 12 Comparison of recovered traffic flow between the two-stage method and SEO framework.

6.3. Investigation of the sensor health in Hong Kong

To assess the scalability of the proposed framework, we apply it to the Sha Tin county network in Hong Kong (see Figure 13 in Appendix B). This network consists of 272 nodes and 367 links. We use simulation-based methods to generate traffic flow data on the Sha Tin county network. Moreover, we use the proposed algorithm in Appendix D to solve this large-scale problem.

There are 98 loop detectors and 19 closed circuit televisions (CCTV) on the network that can be used to collect the traffic flow. According to the literature (Danczyk, Di, and Liu 2016, Fedorov et al. 2019), the error probabilities of loop detectors range from 3% to 15%, while those of CCTVs fall within the range of 3% to 10%. Based on these parameters, we randomly assign the error probabilities to these sensors using uniform distributions within their respective ranges. However, these sensors are not enough to cover all links and therefore other complementary data, such as traffic speed data can be used to compensate for the lack of sensors. Since the speed data collected by floating cars are available on all links, we can estimate the corresponding flow by exploiting the relationship between traffic speed and flow based on the fundamental diagram. We assume that these estimated flow data are gathered by speed sensors. The error probabilities of these speed sensors are randomly drawn from a uniform distribution between 0% and 50%. Moreover, the magnitude of erratic measurement errors is σ.

Under the above settings, we use the collected traffic flow data with erratic measurement errors as the input. The analysis time period is 3 hours and the flow data are aggregated at 5 minute intervals. Figure 14(a) (see Appendix B) shows that the estimated error probabilities of 367 sensors are exceptionally satisfactory with MAE=0.0024 and MAPE=0.0272. We also compare the observed flow and recovered flow against the true flow on the Sha Tin county network in Figure 14(b) (see Appendix B). We find that the proposed optimization model can significantly reduce the measurement errors in the data. The MAE and MAPE of the recovered flow are 2.32 and 0.03, respectively.

6.4. Limitations in real-world applications

The proposed approaches may face several limitations in real-world applications: (1) Our models are mainly used for estimating erratic measurement errors in flow data. However, due to high installation and maintenance costs, the installed sensors are usually insufficient to cover the entire network. This means large-scale data missing and erratic errors may coexist in real-world applications. When entire data from a large proportion of links are missing in the real world, the models may not perform effectively and therefore requires further investigation in future studies; (2) The computation time of the proposed algorithm (see Appendix D for details) is not efficient for large networks, which may pose challenges for real-time computation; and (3) The proposed two-stage method relies on the assumptions of the Poisson process and linear correlation, which may limit its applicability in practical settings. While the SEO framework effectively addresses these issues, its dependency on network connectivity and flow balance law suggests the need for careful consideration of these factors in complex traffic scenarios. Moreover, the challenges related to compatibility and interoperability with existing traffic monitoring systems, as well as compliance with local regulations, may hinder the effective implementation of the proposed models.

7. Conclusions

In this research, we focus on an integrated estimate-then-optimize problem that aims to simultaneously estimate sensor error probabilities and recover traffic flow from recorded erroneous data and link features. To solve the complex integrated problem, we develop a methodological ML framework that merges ML and optimization analysis grounded in traffic domain knowledge. Specifically, we first introduce the concept of VBS using flow balance laws, on top of which a mixed integer non-linear programming model is formulated. Under suitable assumptions, we then utilize observations of VBS to analyze the properties of the model. Importantly, we prove that the complex integrated problem can be equivalently decomposed and solved with the two-stage method. Compared with existing studies on measurement errors, our framework introduces two key innovations relevant to the ML and transportation domains: (1) ML methods and optimization models are integrated and analyzed within one framework, enabling equivalent decomposition and achieving simultaneous estimation of error probabilities and traffic data recovery; and (2) ML feature engineering techniques are used to mine the information from observations of virtual balance sensors (VBS), significantly reducing unknown parameters and enabling effective estimation of sensor error probabilities.

Furthermore, we develop an SEO framework that tackles the integrated problem when the assumptions fail in more realistic scenarios, which is new to the literature. The SEO framework integrates ML estimation and optimal decision-making by directly accounting for the impact of inaccurate ML estimations within the optimization process. It handles each step of the process, from input data to output decisions, without requiring manual intervention or separate stages for ML estimation and optimization. To achieve this, the optimization model is embedded within the neural network training loop as an optimization layer. The gradients of optimal decisions with respect to the estimated parameters from ML methods are derived to enable backpropagation in ML training. In contrast to the two-stage method, the SEO framework ensures that the optimization process can recognize and compensate for potential ML estimation inaccuracies, thereby enabling more robust and effective decision-making. Numerical experiments are conducted on the Nguyen-Dupuis network and Sha Tin network in Hong Kong under various scenarios. Results demonstrate the decomposition approach is effective and the SEO framework has better performance than the two-stage method.

We can obtain valuable information about the real world sensor health conditions in traffic monitoring systems using the proposed framework. Such information includes the estimated sensor error probabilities, which help to assess the sensor health and provide guidance for maintenance or replacement strategies. It also identifies the critical exogenous variables that affect the sensor heath, facilitating tasks related to sensor procurement, sensor installation, and identifying malfunctioning sensors. Moreover, the assessment of sensor data quality contributes to effective data recovery. Although the identified erroneous data and recovered data may deviate from their true values, our approach has fully exploited the available information and endeavors to estimate the most likely values. In addition, there are other types of traffic data collected by sensors, such as speed and density. By leveraging the relationship between these data supported by the fundamental diagram, it is also possible to estimate the erratic measurement errors in these data. It is worth noting that our model can also work in an online fashion with a stream of real-time data. Initially, the real-time data and historical datasets are used as inputs to the model for estimating sensor error probabilities and recover flow data. After that, the real-time data are updated into the historical datasets to prepare for the next incoming data stream. Note that handling real-time streaming data requires efficient data processing and problem-solving methods, which may pose challenges for the current algorithm.

Finally, we outline the possible research directions. First, this paper mainly concentrates on estimating the erratic errors in traffic flow data. When other types of traffic data, such as traffic speed and density, are available, they can also be used to estimate measurement errors. It is expected that correct data, or data with minor disturbances, will have a strong correlation, maintain a low rank, and calibrate the fundamental diagram well. In contrast, erroneous data are likely to be uncorrelated, of high rank, and poorly suited to the fundamental diagram. Based on these observations, we can iteratively identify and correct erroneous data while simultaneously calibrating the fundamental diagram. Second, estimating erratic errors in real-time scenarios requires efficient data processing and problem-solving methods, which may pose challenges for the current algorithm. Therefore, it is of great importance to develop more efficient algorithms that are capable of facilitating the practical applications of our model.

Acknowledgments

The work described in this paper was supported by the National Natural Science Foundation of China (Project No. 72288101, 72101012, 72301023) and a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. PolyU/15206322). The authors thank the reviewer panel for thoughtful comments that substantially improved this study.

References

Ariannezhad A, Wu YJ, 2020 *Large-scale loop detector troubleshooting using clustering and association rule mining. Journal of Transportation Engineering, Part A* 146(7):04020064.

Bae B, Kim H, Lim H, Liu Y, Han LD, Freeze PB, 2018 *Missing data imputation for traffic flow speed using spatio-temporal cokriging. Transportation Research Part C* 88:124–139.

Bertsekas D, 2009 *Convex Optimization Theory*, volume 1 (Athena Scientific).

Boto-Giralda D, Díaz-Pernas FJ, González-Ortega D, Díez-Higuera JF, Antón-Rodríguez M, Martínez-Zarzuela M, Torre-Díez I, 2010 *Wavelet-based denoising for traffic volume time series forecasting with self-organizing neural networks. Computer-Aided Civil and Infrastructure Engineering* 25(7):530–545.

Candes EJ, Plan Y, 2010 *Matrix completion with noise. Proceedings of the IEEE* 98(6):925–936.

Chen A, Chootinan P, Recker W, 2009 *Norm approximation method for handling traffic count inconsistencies in path flow estimator. Transportation Research Part B* 43(8-9):852–872.

Chen X, He Z, Sun L, 2019 *A bayesian tensor decomposition approach for spatiotemporal traffic data imputation. Transportation Research Part C* 98:73–84.

Chiu M, Jackson KR, Kreinin A, 2017 *Correlated multivariate poisson processes and extreme measures. Model Assisted Statistics and Applications* 12(4):369–385.

Danczyk A, Di X, Liu HX, 2016 *A probabilistic optimization model for allocating freeway sensors. Transportation Research Part C* 67:378–398.

Danielyan A, Katkovnik V, Egiazarian K, 2011 *Bm3d frames and variational image deblurring. IEEE Transactions on Image Processing* 21(4):1715–1728.

Duran A, Earleywine M, 2013 *GPS data filtration method for drive cycle analysis applications.* Technical report, National Renewable Energy Lab.(NREL), Golden, CO (United States).

Elmachtoub AN, Grigas P, 2022 *Smart "predict, then optimize". Management Science* 68(1):9–26.

Fedorov A, Nikolskaia K, Ivanov S, Shepelev V, Minbaleev A, 2019 *Traffic flow estimation with data from a video surveillance camera. Journal of Big Data* 6:1–15.

Feng X, Zhang H, Wang C, Zheng H, 2022 *Traffic data recovery from corrupted and incomplete observations via spatial-temporal TRPCA. IEEE Transactions on Intelligent Transportation Systems* .

Geoffrion AM, 1972 *Generalized benders decomposition. Journal of Optimization Theory and Applications* 10:237–260.

Hayashi F, 2011 *Econometrics* (Princeton University Press).

He L, Liu S, Shen ZJM, 2022 *Smart urban transport and logistics: A business analytics perspective. Production and Operations Management* 31(10):3771–3787.

Hu Y, Work DB, 2020 *Robust tensor recovery with fiber outliers for traffic events. ACM Transactions on Knowledge Discovery from Data (TKDD)* 15(1):1–27.

Hu Z, Lam WH, Wong S, Chow AH, Ma W, 2023 *Turning traffic surveillance cameras into intelligent sensors for traffic density estimation. Complex & Intelligent Systems* 1–25.

Kikuchi S, Mangalpally S, Gupta A, 2006 *Method for balancing observed boarding and alighting counts on a transit line. Transportation Research Record* 1971(1):42–50.

Kim D, Shin S, Park D, Kim J, 2019 *Correction of measured traffic volume on expressways based on traffic volume balancing. WIT Transactions on The Built Environment* 182:361–371.

Kotary J, Fioretto F, Van Hentenryck P, 2021 *Learning hard optimization problems: A data generation perspective. Advances in Neural Information Processing Systems* 34:24981–24992.

Lehtinen J, Munkberg J, Hasselgren J, Laine S, Karras T, Aittala M, Aila T, 2018 *Noise2noise: Learning image restoration without clean data. arXiv preprint arXiv:1803.04189* .

Liang Y, Zhao Z, Sun L, 2022 *Memory-augmented dynamic graph convolution networks for traffic data imputation with diverse missing patterns. Transportation Research Part C* 143:103826.

Liu H, Grigas P, 2021 *Risk bounds and calibration for a smart predict-then-optimize method. Advances in Neural Information Processing Systems* 34:22083–22094.

Ma W, Chen GH, 2019 *Missing not at random in matrix completion: The effectiveness of estimating missingness probabilities under a low nuclear norm assumption. Advances in Neural Information Processing Systems* 32.

Nistor S, Buda AS, 2016 *GPS network noise analysis: a case study of data collected over an 18-month period. Journal of Spatial Science* 61(2):427–440.

Rajagopal R, Varaiya PP, 2007 *Health of california's loop detector system.* Technical report.

Sadana U, Chenreddy A, Delage E, Forel A, Frejinger E, Vidal T, 2024 *A survey of contextual optimization methods for decision-making under uncertainty. European Journal of Operational Research* .

Sun X, Muñoz L, Horowitz R, 2003 *Highway traffic state estimation using improved mixture kalman filters for effective ramp metering control. 42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, volume 6, 6333–6338 (IEEE).

Tang B, Khalil EB, 2022 *Pyepo: A pytorch-based end-to-end predict-then-optimize library for linear and integer programming. arXiv preprint arXiv:2206.14234* .

Vanajakshi L, Rilett L, 2004 *Loop detector data diagnostics based on conservation-of-vehicles principle. Transportation Research Record* 1870(1):162–169.

Wall ZR, Dailey DJ, 2003 *Algorithm for detecting and correcting errors in archived traffic data. Transportation Research Record* 1855(1):183–190.

Wang Y, Papageorgiou M, 2005 *Real-time freeway traffic state estimation based on extended kalman filter: a general approach. Transportation Research Part B* 39(2):141–167.

Ward MD, Gleditsch KS, 2018 *Spatial Regression Models*, volume 155 (Sage Publications).

Yang Y, Yang H, Fan Y, 2019 *Networked sensor data error estimation. Transportation Research Part B* 122:20–39.

Yera YG, Lillo RE, Nielsen BF, Ramírez-Cobo P, Ruggeri F, 2021 *A bivariate two-state markov modulated poisson process for failure modeling. Reliability Engineering & System Safety* 208:107318.

Yi X, Babyn P, 2018 *Sharpness-aware low-dose ct denoising using conditional generative adversarial network. Journal of Digital Imaging* 31(5):655–669.

Yin P, Sun Z, Jin WL, Xin J, 2017 *l1-minimization method for link flow correction. Transportation Research Part B* 104:398–408.

Zhang J, Shan E, Wu L, Yin J, Yang L, Gao Z, 2024 *An end-to-end predict-then-optimize clustering method for stochastic assignment problems. IEEE Transactions on Intelligent Transportation Systems* .

Zheng Z, Ahn S, Chen D, Laval J, 2011 *Applications of wavelet transform for analysis of freeway traffic: Bottlenecks, transient traffic, and traffic oscillations. Transportation Research Part B* 45(2):372–384.

Zheng Z, Su D, 2016 *Traffic state estimation through compressed sensing and markov random field. Transportation Research Part B* 91:525–554.

A. Nomenclature

The list of notations used in this paper is shown in Table 6.

Table 6: List of notations.

	Network related variables
e	Index of sensors.
E	Number of sensors.
t	Index of time intervals.
T	Number of time intervals.
VBS	Virtual conservation sensors.
v	Index of VBS.
V	Number of VBS.
$E^+(v)$	Set of sensors associated with entering links of node v.
$E^+(v)$	Set of sensors associated with exiting links of node v.
$E^{(}v)$	Set of sensors associated with node v.
$\|E^{(}v)\|_\#$	Cardinality of set $E(v)$.
	Variables for Poisson process
$\lambda_{e,t}$	Arrival rate of the Poisson process associated with sensor e during time t.
λ_e	Arrival rate of the Poisson process associated with sensor e.
$p_{e,t}$	Error probability of sensor e during time t.
n	Number of arrivals in a time interval.
η	Length of a time interval.
	Variables for the optimization model
$q_{e,t}$	Collected flow data from sensor e during time t.
$\hat{q}_{e,t}$	Recovered flow of sensor e during time t.
$\delta_{v,t}$	Collected data from VBS v during time t.
p_e	Error probability of sensor e.
$\hat{p}_j$	Estimated error probability of sensor e.
$\boldsymbol{p}_v$	Error probability of VBS v.
$\gamma_{e,t}$	Binary decision variable indicating the occurrence of measurement errors.
$z_{e,t}$	Intermediate decision variable used to replace the erroneous data.
$\xi_{e,t}$	Erratic measurement errors associated with $q_{e,t}$.
$l_{e,t}$	Likelihood of the occurrence of errors for sensor e during time t.
$\mathcal{L}$	Likelihood of the occurrence of errors for all sensors within the estimation time horizon.
$\mathcal{F}$	Objective function of the optimization model.
$\mathcal{F}_1$	Primary objective function.
$\mathcal{F}_2$	Second objective function.
$\hat{\boldsymbol{Q}}$	Recovered flow matrix.
$\|\hat{\boldsymbol{Q}}\|_*$	Nuclear norm of $\hat{\boldsymbol{Q}}$.
Θ	A big positive number.
$\boldsymbol{x}_e$	Exogenous variables.
$\boldsymbol{X}$	Matrix of $\boldsymbol{x}_e^T$.
$\chi(\boldsymbol{x}_e)$	A specific machine learning method that maps $\boldsymbol{x}_e$ to λ_e.
$\boldsymbol{\beta}$	Coefficients of $\boldsymbol{x}_e$.

$\hat{\boldsymbol{\beta}}$	Estimated $\boldsymbol{\beta}$.
$\boldsymbol{y}$	Vector of $-log(1-p_v)$.
$\gamma_{e,t}^{k}$	Optimal solution of the master problem at iteration k.
$\hat{q}_{e,t}^{k}$	Optimal solution of the primal problem at iteration k.
λ^{k}	Lagrange multiplier of the primal problem at iteration k.
$\boldsymbol{h}(\hat{q}_{e,t}^{k}, \lambda^{k})$	Lagrange dual function of the primal problem at iteration k.
ς	Convergence threshold of the GBD.
lb	Lower bound of the GBD.
ub	Upper bound of the GBD.
Variables for the SEO framework	
$\mathscr{L}$	Lagrangian function.
$\boldsymbol{w}$	The vector of all decision variables.
$\boldsymbol{C}_n$	The n_{th} coefficient matrix.
$\boldsymbol{b}_n$	The n_{th} constant vector.
$\boldsymbol{p}$	The vector of sensor error probabilities.
$\boldsymbol{\nu}_n$	The n_{th} vector of Lagrange multipliers.
$\boldsymbol{D}(\cdot)$	Diagonal matrix created from a vector.
Variables for evaluating the model	
M_r	Missing ratio in the erroneous data.
N_m	Number of missing data.
N_e	Number of erroneous data.
Φ_m	Set of missing data.
μ	Disturbance parameter.
$\bar{q}_{e,t}$	Ground truth of $q_{e,t}$.
$\tilde{q}_{e,t}$	Adjusted $q_{e,t}$ that violates the flow balance law.
$\mathcal{Z}$	Set of erroneous data that are not identified.
U	Sum of squares of erratic measurement errors in $\mathcal{Z}$.
$r_{v,t}$	Flow imbalance ratio at node v during time t.
Random variables	
ϵ	A random variable representing the minor disturbances of traffic conditions.
$\epsilon_{v,t}$	A random variable representing the minor disturbances of traffic conditions at node v during t.
σ_{ϵ}	Standard deviation of $\epsilon_{v,t}$.
σ	Standard deviation of the collected data.
ε_j	A random variable accounting for unobserved effects.

B. Measurement errors in real sensor data from Hong Kong

In this section, we first present the measurement errors observed in real sensors collected from the Sha Tin network in Hong Kong. The distribution and locations of the sensors are depicted in Figure 13, providing a clear overview of the network coverage. The performance of the proposed decomposition approach in addressing these errors is demonstrated in Figure 14, highlighting its effectiveness in improving data accuracy.

We then provide an example to demonstrate the arrival rate of the measurement errors as illustrated in Figure 15(a). The data are collected on Chatham Road South, Hong Kong SAR on September 24, 2020. The flow data with errors are collected by a sensor. In addition, there is a high resolution camera (1920×1080 pixels per frame) installed on the road (Hu et al. 2023). We use the data recorded by the camera as the ground true flow.

Let $\bar{q}_t$ and q_t denote the true flow and collected flow during time interval t, respectively. Due to the minor disturbances and stochasticities in the real world, the flow data collected by two devices cannot be exactly the same. Therefore, we allow for a minor difference between $\bar{q}_t$ and q_t. Let σ denote the standard deviation of true flow. If $|\bar{q}_t - q_t| \leq \sigma$, we say there is no error. Otherwise, we say the errors occur. Since the Poisson process is defined by stating that the time intervals between two successive events follow the exponential distribution, we can use the Kolmogorov–Smirnov test to conduct the hypothesis testing for exponential distributions. The null hypothesis is that the time interval follows the exponential distribution and the significance level is 0.01. Results demonstrate that the p-value is 0.3768 and therefore we fail to reject that the time intervals follow an exponential distribution. This means that there is no evidence that the occurrences of erratic measurement errors do not follow a Poisson process. Figure 15(b) demonstrates the histogram of time intervals, and the fitted probability density function is drawn over the histogram.

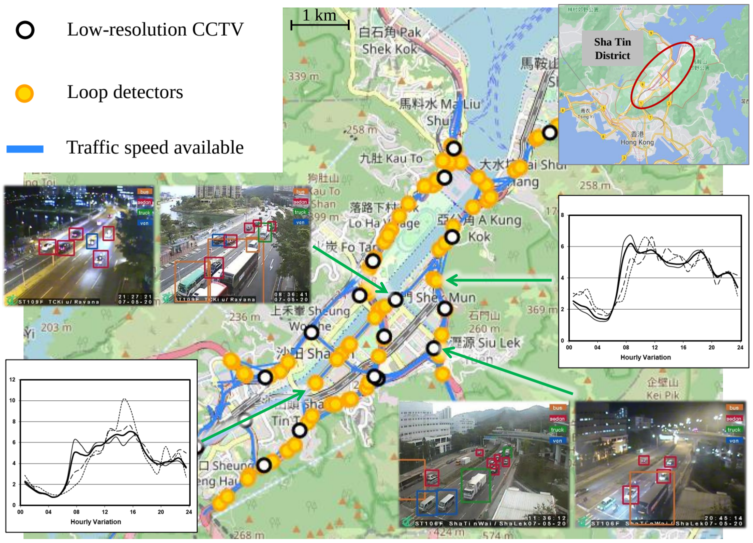


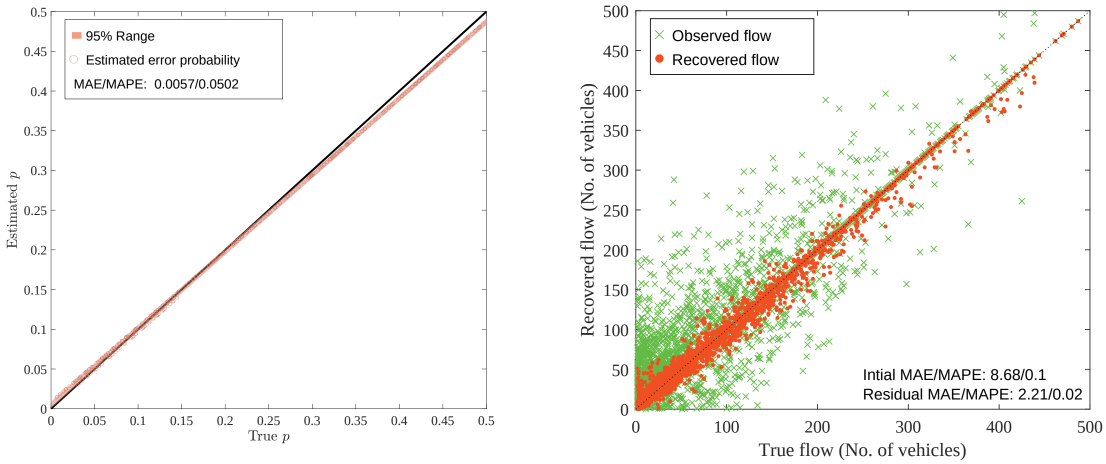
Figure 13 **An overview of the Sha Tin county network and installed sensors in Hong Kong.**

C. Proofs

C.1. Proof of Proposition 1

By applying Constraints (11d), the primary objective function $\mathcal{F}_1$ can be reformulated as follows:

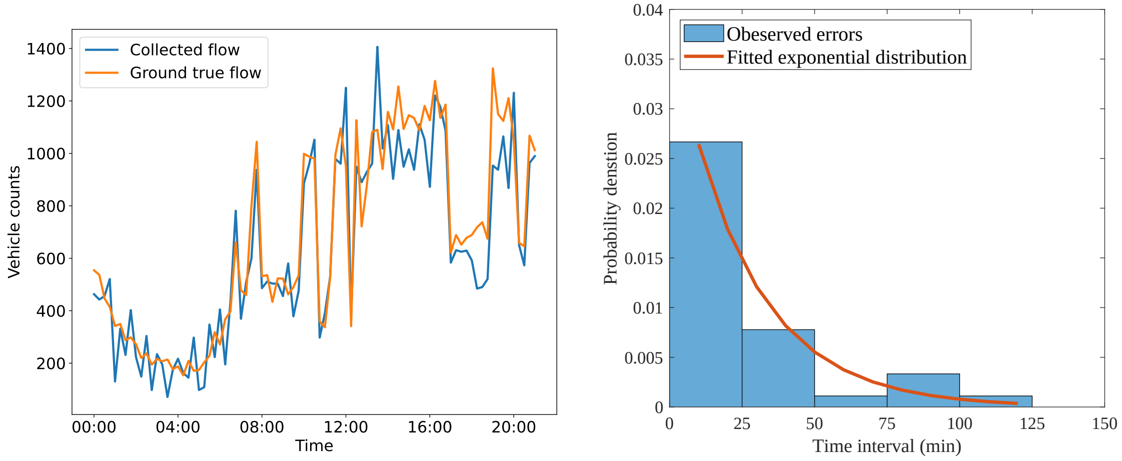
$$\mathcal{F}_1 = -\sum_{e=1}^{E}\sum_{t=1}^{T}(\gamma_{e,t}\log(p_e) + (1-\gamma_{e,t})\log(1-p_e)) = -T\sum_{e=1}^{E}\left[p_e\log(p_e) + (1-p_e)\log(1-p_e)\right]. \quad (35)$$



(a) Estimation result of sensor error probability p on the Sha Tin county network.

(b) Recovered traffic flow on the Sha Tin county network.

Figure 14 Results of the proposed model on the Sha Tin county network.



(a) Collected flow and ground truth.

(b) Histogram of observed errors and fitted distribution.

Figure 15 Illustration of the observed errors in real sensor data on Chatham Road South, Hong Kong.

We prove the concavity of $\mathcal{F}_1$ by deriving its Hessian matrix. The diagonal elements of the Hessian matrix are formulated as follows:

$$\frac{\partial^2 \mathcal{F}_1}{\partial p_e} = -\frac{1}{p_e} - \frac{1}{1-p_e} \leq 0, \quad \forall 1 \leq e \leq E.$$

The off-diagonal elements of the Hessian matrix is formulated as follows:

$$\frac{\partial^2 \mathcal{F}_1}{\partial p_{e_1} \partial p_{e_2}} = 0, \quad \forall 1 \leq e_1 \leq E, \ \forall 1 \leq e_2 \leq E, e_1 \neq e_2.$$

This implies that the Hessian matrix is negative semidefinite. As a result, $\mathcal{F}_1$ is a concave function for $0 < p_e < 1, \forall 1 \leq e \leq E$ (Bertsekas 2009).

C.2. Proof of Proposition 2

Let $\hat{p}_1$ and $\hat{p}_2$ be the optimal solutions of model (16). We first prove that $\mathcal{F}_1(\hat{p}_1, \hat{p}_2)$ establishes the lower bound for $\mathcal{F}_1$ within model (15). Let $\mathcal{P}_o$ and $\mathcal{P}_n$ denote the feasible p_e of models (15) and (16), respectively. It can be proved that $\mathcal{P}_o$ is the subset of $\mathcal{P}_n$ by aggregating Constraints (11a) over the time dimension:

$$\sum_{t=1}^{T} \delta_{1,t} \leq T(p_1 + p_2) \leq 2 \sum_{t=1}^{T} \delta_{1,t}. \tag{36}$$

Constraints (11a) also ensure that $\gamma_{e,t} \leq \delta_{v,t}$, which results in the following equation:

$$p_e = \frac{\sum_{t=1}^{T} \gamma_{e,t}}{T} \leq \frac{\sum_{t=1}^{T} \delta_{1,t}}{T} \qquad \forall 1 \leq e \leq E. \tag{37}$$

This implies that solutions that satisfy Constraints (11a) also meet the constraints of model (16), that is, $\mathcal{P}_o$ is the subset of $\mathcal{P}_n$.

We then prove that feasible solutions of model (15) can be derived from $\hat{p}_1$ and $\hat{p}_2$. This also implies that $\mathcal{F}_1(\hat{p}_1, \hat{p}_2)$ establishes the upper bound for $\mathcal{F}_1$ within model (15). Without loss of generality, we arrange the data from VBS in an ascending order: $\delta_{1,t} \leq \delta_{1,t+1}$, $\forall 1 \leq t \leq T - 1$. Let t_m be the smallest value of t satisfying $\delta_{1,t_m} = 1$. Let $\hat{\gamma}_{1,t}$ and $\hat{\gamma}_{2,t}$ denote the derived solutions. We can derive feasible $\hat{\gamma}_{1,t}$ and $\hat{\gamma}_{2,t}$ as shown in Figure 16. Specifically, we set $\gamma_{1,t} = \gamma_{2,t} = 0$, $\forall 0 \leq t < t_m$; $\gamma_{1,t} = 1$, $\forall t_m \leq t \leq t_m + T\hat{p}_1 - 1$, and $\gamma_{2,t} = 1, \forall T - T\hat{p}_2 + 1 \leq t \leq T + T\hat{p}_1 - 1$. It can be verified that $\hat{\gamma}_{1,t}$ and $\hat{\gamma}_{2,t}$ satisfies all constraints of model (15). For given $\hat{\gamma}_{1,t}$ and $\hat{\gamma}_{2,t}$, model (15) is simplified into a matrix completion problem, ensuring feasibility for the remaining decision variables. Since $\mathcal{F}_1$ is the primary objective function of model (15), the analysis above indicates that $\hat{p}_1$ and $\hat{p}_2$ are the optimal p_e for model (15).

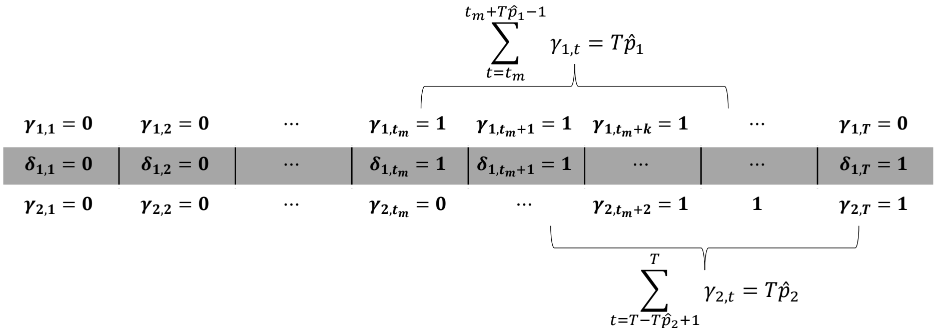


Figure 16 The feasible solution derived from $\hat{p}_1$ and $\hat{p}_2$.

C.3. Proof of Proposition 3

As demonstrated in Proposition 2, the optimal values p_e for model (15) can be determined by solving model (16). The feasible region of model (16) is the dark grey area as shown in Figure 17. According to Proposition 1, the objective function of model (16) is a concave function. Therefore, the corresponding optimal solutions lie in the boundary points (Bertsekas 2009), that is, the three vertexes. When $\frac{\sum_{t=1}^{T}\delta_{1,t}}{T} < 1$, $\mathcal{F}_1(0, \frac{\sum_{t=1}^{T}\delta_{1,t}}{T}) = \mathcal{F}_1(\frac{\sum_{t=1}^{T}\delta_{1,t}}{T}, 0) < \mathcal{F}_1(\frac{\sum_{t=1}^{T}\delta_{1,t}}{T}, \frac{\sum_{t=1}^{T}\delta_{1,t}}{T})$. This implies that the optimal numerical p_e for model (15) are: $p_1 = \frac{\sum_{t=1}^{T}\delta_{1,t}}{T}, p_2 = 0$ or $p_1 = 0, p_2 = \frac{\sum_{t=1}^{T}\delta_{1,t}}{T}$.

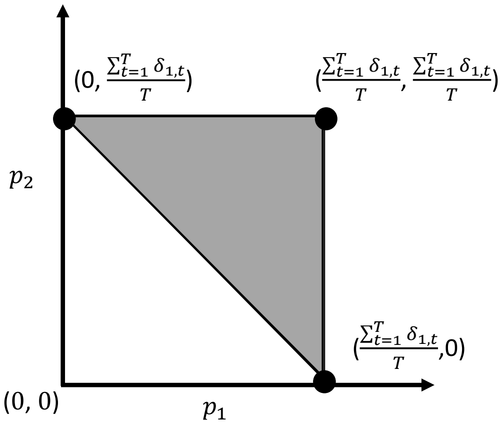

$(0, \frac{\sum_{t=1}^{T}\delta_{1,t}}{T})$
$(\frac{\sum_{t=1}^{T}\delta_{1,t}}{T}, \frac{\sum_{t=1}^{T}\delta_{1,t}}{T})$
p_2
$(\frac{\sum_{t=1}^{T}\delta_{1,t}}{T}, 0)$
(0, 0)
p_1

Figure 17 The feasible region of model (15).

C.4. Proof of Proposition 4

With Constraints (18), the following equation can be derived:

$$-\sum_{e\in E(v)} \log(1-p_e) = \sum_{e\in E(v)} \chi(\boldsymbol{x_e}) = \sum_{e\in E(v)} \lambda_e. \tag{38}$$

Let τ_v be the arrival rate of the Poisson process associated with VBS v. Since the Poisson processes are independent, τ_v is equal to the sum of arrival rates of the sensors associated with VBS v:

$$\tau_v = \sum_{e\in E(v)} \lambda_e. \tag{39}$$

Note that the arrival rate of VBS v can be described by its error probability $\mathcal{p}_v$:

$$\tau_v = -\log(1-\mathcal{p}_v). \tag{40}$$

By applying Equations (38), (39) and (40), the following equation hold:

$$-\log(1-\mathcal{p}_v) = \boldsymbol{\beta} \sum_{e\in E(v)} \boldsymbol{x}_{\boldsymbol{e}}^{T}. \tag{41}$$

This leads to a system of equations $\boldsymbol{\beta X} = \boldsymbol{y}$. When $rank(\boldsymbol{X}) < rank([\boldsymbol{X}, \boldsymbol{y}])$, there is no solution for $\boldsymbol{\beta}$ that can satisfy $\boldsymbol{y} = \boldsymbol{\beta X}$, which implies that model (15) is infeasible.

C.5. Proof of Proposition 5

The dependent variable $\boldsymbol{y}$ can be directly calculated with $\delta_{v,t}$ and $\boldsymbol{y}$ is an unbiased estimation of the true value when T is sufficiently large. This implies that an unbiased estimation of the coefficient vector can be obtained by minimizing the $\|\boldsymbol{y}-\boldsymbol{\beta}\boldsymbol{X}\|_2^2$ using the least squares method (Hayashi 2011), denoted as $\hat{\boldsymbol{\beta}}$. That is, $\hat{\boldsymbol{\beta}}$ is the optimal solution for minimizing the transferred objection function. When $\hat{\boldsymbol{\beta}}$ does not violate other constraints of model (19), $\hat{\boldsymbol{\beta}}$ is the optimal $\boldsymbol{\beta}$ of model (19). Note that $\hat{\boldsymbol{\beta}}$ is unbiased and therefore the estimated error probability $\hat{p}_e = 1 - exp(-\hat{\boldsymbol{\beta}}\boldsymbol{x}_e)$ is unbiased. This implies that the true values of decision variables $\gamma_{e,t}, z_{e,t}$, and $\hat{q}_{e,t}$ satisfy the constraints for given $\hat{p}_e$, which means the model is feasible. This completes the proof.

C.6. Proof of Proposition 6

According to Proposition 5, the optimal p_e for model (19) can be determined. This implies that the transferred objective function $\mathcal{H}\|\boldsymbol{y}-\boldsymbol{\beta}X\|_2^2$ and $\mathcal{F}_1$ of model (19) are constants. When erroneous data are correctly identified with the maximized likelihood, model (19) is simplified into a matrix completion problem:

$$\begin{aligned}
&\min_{\hat{q}_{e,t}} \quad rank(\hat{\boldsymbol{Q}})\\
\text{s.t.} \quad & \hat{q}_{e,t} = q_{e,t}, \quad \forall e,t \in \{(e,t)|\gamma_{e,t}=0\};\\
& \sum_{e\in E^+(v)} \hat{q}_{e,t} - \sum_{e\in E^-(v)} \hat{q}_{e,t} = 0, \quad \forall 1\le v\le V, 1\le t\le T;\\
& \hat{q}_{e,t} \ge 0. \quad \forall 1\le e\le E, 1\le t\le T.
\end{aligned}$$

Under Assumption 3, the true data can be exactly recovered when the rank($\hat{\boldsymbol{Q}}$) is minimized as proved in the existing studies (Candes and Plan 2010, Ma and Chen 2019). □

C.7. Proof of Proposition 7

When the erratic measurement errors are not correctly identified, it means the original problem is transformed into a matrix completion problem with measurement errors. Let $\mathcal{Z}$ denote the set of erroneous data that are not identified, that is, $\mathcal{Z} = \{(e,t)|\gamma_{e,t}=0, q_{e,t}\neq \bar{q}_{e,t}\}$. Let U denote the sum of squares of erratic measurement errors in $\mathcal{Z}$, that is, $U = \sum_{e,t\in\mathcal{U}}(\xi_{e,t})^2$. As proved in the existing studies (Candes and Plan 2010), the recovery error $\|\hat{\boldsymbol{Q}}-\boldsymbol{Q}\|_2^2 \le 4\sqrt{\frac{(3-r)min(E,T)}{1-r}}U + 2U$. □

Proposition 8. *Under Assumption 1, suppose that T is sufficiently large, the optimal p_e for model (15) can be determined by solving the following optimization model.*

$$\min_{p_e} \; -T\sum_{e=1}^{E}[p_e log(p_e) + (1-p_e)log(1-p_e)]$$

$$\text{s.t.} \quad \sum_{e=1}^{E(v)} p_e \ge \frac{\sum_{t=1}^{T}\delta_{v,t}}{T}, \quad \forall 1\le v\le V; \tag{42a}$$

$$0 \leq p_e \leq \frac{\sum_{t=1}^{T} \delta_{v,t}}{T}, \quad \forall 1 \leq v \leq V, \forall e \in E(v); \tag{42b}$$

$$0 \leq p_e \leq 1, \quad \forall 1 \leq e \leq E. \tag{42c}$$

Proof. We prove this proposition by deriving the feasible region of p_e within model (15). The sensor error probability p_e within model (15) is determined by Constraints (11e), that is, $p_e = \frac{\sum_{t=1}^{T} \gamma_{e,t}}{T}$. This implies that the feasible region of p_e can be derived by determining the feasible region of $\sum_{t=1}^{T} \gamma_{e,t}$. The value of $\sum_{t=1}^{T} \gamma_{e,t}$ is only determined by Constraints (11a) because other constraints are not related to the primary objection function within model (15). Therefore, the feasible region of p_e is determined by Constraints (11a). As demonstrated in Proposition 2, feasible region of p_e can be derived by aggregating Constraints (11a) over the time dimension:

$$\sum_{t=1}^{T} \delta_{v,t} \leq \sum_{t=1}^{T} \sum_{e \in E(v)} \gamma_{e,t} \leq |E(v)|_{\#} \delta_{v,t}, \quad \forall 1 \leq v \leq V. \tag{43}$$

Based on Constrains (11e), we can reformulate Equation (43) as follows:

$$\sum_{t=1}^{T} \delta_{v,t} \leq T \sum_{e \in E(v)} p_e \leq |E(v)|_{\#} \delta_{v,t}, \quad \forall 1 \leq v \leq V. \tag{44}$$

Constraints (11a) also ensure that $\gamma_{e,t} \leq \delta_{v,t}$, which results in the following equation:

$$p_e = \frac{\sum_{t=1}^{T} \gamma_{e,t}}{T} \leq \frac{\sum_{t=1}^{T} \delta_{v,t}}{T}, \quad \forall 1 \leq e \leq E. \tag{45}$$

This implies the feasible region of p_e within model (42) is the same as that within (11a). Since the objection function of model (42) is the primary objection function of model (42), we can conclude that the optimal p_e for model (15) can be determined by solving model (42).

D. Algorithms

For the GBD algorithm, the primal problem is formulated as follows:

$$\min_{\hat{q}_{e,t}, z_{e,t}} \|\hat{\boldsymbol{Q}}\|_*$$

$$\text{s.t.} \quad \sum_{e \in E^+(v)} \hat{q}_{e,t} - \sum_{e \in E^-(v)} \hat{q}_{e,t} = 0, \quad 1 \leq v \leq V; \tag{46a}$$

$$\hat{q}_{e,t} = (1 - \gamma_{e,t}^k) q_{e,t} + z_{e,t}, \quad 1 \leq e \leq E, 1 \leq t \leq T; \tag{46b}$$

$$z_{e,t} \leq \gamma_{e,t}^k \Theta, \quad 1 \leq e \leq E, 1 \leq t \leq T; \tag{46c}$$

$$\hat{q}_{e,t} \geq 0, \quad z_{e,t} \geq 0, \quad 1 \leq e \leq E, 1 \leq t \leq T. \tag{46d}$$

$\gamma_{e,t}^k$ is a known solution at iteration k.

The master problem is formulated as follows:

$$\min_{\gamma_{e,t}} \; s$$

$$\text{s.t.} \qquad s \geq \|\hat{\boldsymbol{Q}}^k\|_* + \boldsymbol{\lambda}^k \boldsymbol{h}(\hat{q}^k_{e,t}, \gamma_{e,t}), \quad k \geq 1; \tag{47a}$$

$$\delta_{v,t} \leq \sum_{e \in E(v)} \gamma_{e,t} \leq |E(v)|_{\#} \delta_{v,t}, \quad \forall 1 \leq v \leq V; \tag{47b}$$

$$\sum_{t=1}^{T} \gamma_{e,t} - 1 \leq \hat{p}_e T \leq \sum_{t=1}^{T} \gamma_{e,t} + 1, \quad \forall 1 \leq e \leq E; \tag{47c}$$

$$\gamma_{e,t} \in \{0, 1\}. \tag{47d}$$

where $\|\hat{\boldsymbol{Q}}^k\|_*$ is the optimal objective function value of the primal problem at iteration k. Constraints (47a) are the feasibility cuts derived from the primal problem at iteration k. Specifically, $\hat{q}^k_{e,t}$ and $\boldsymbol{\lambda}^k$ are the optimal solution and Lagrange multipliers of the primal problem at iteration k, respectively. $\boldsymbol{h}(\hat{q}^k_{e,t}, \gamma_{e,t})$ corresponds to Constraints (46a) through (46d) in the primal problem. Since the primal problem is always feasible, we omit the infeasibility cuts of the master problem. Please refer to Geoffrion (1972) for more details. The detailed procedure of GBD is shown in Appendix D. It has been proved in Geoffrion (1972) that the GBD algorithm would terminate in a finite number of iterations for a given convergence threshold $\varsigma \geq 0$.

Algorithm 2 Algorithmic statement of GBD

```
Input:
    The collected flow data: $q_{e,t}$
    The error probability: $p_e$
    An initial guess of $\gamma^1_{e,t}$
    A big positive number: $\Theta$
    Initial lower bound and upper bound: $lb$ and $ub$
    A convergence threshold: $\varsigma = 0.01$
Procedure
$k = 1$
while $(lb - ub)/lb \leq \varsigma$ do
    Solve the primal problem with $\gamma^k_{e,t}$.
    Calculate the Lagrange multiplier $\boldsymbol{\lambda}^k$ and add Constraints (47a) to the
    master problem.
    Update the upper bound: $ub = \min\left\{ub, \|\hat{\boldsymbol{Q}}^k\|_*\right\}$.
    Solve the master problem.
    Update the lower bound: $lb = z^k$.
    $k = k + 1$;
    Update $\gamma^k_{e,t}$ using the optimal solution of the master problem.
end while
```

For the tractable algorithm, we first remove Constraints (19a) through (19d) and transfer Constraints (19e) into the objective function with a predefined weight ω. We then take $\gamma_{e,t}$ as continuous variables, which implies the original problem is transformed into a convex optimization problem similar to the robust principal component analysis. Subsequently, we use the alternating direction method of multipliers (ADMM) algorithm to solve the convex problem. Next, we use the Algorithm 3 to derive a feasible solution for the original problem based on continuous $\gamma_{e,t}$. The basic idea of this algorithm is to generate solutions that maximize the sum $\sum_{e=1}^{E}\sum_{t=1}^{T}\gamma_{e,t}$ by using an expansion factor $E_r \geq 1$. Meanwhile, the algorithm ensures that at least $rank(\boldsymbol{Q})$ elements for each sensor e are identified as the correct data (i.e. $T-\sum_{t=1}^{T}\gamma_{e,t} \geq rank(\boldsymbol{Q})$) by using a pre-estimated parameter r_Q. Furthermore, the algorithm is also designed to satisfy all constraints except the Constraints (19e), which may lead to the misidentification of correct data as erroneous. However, as discussed above, such violation is unlikely to impair the model performance owing to the global low-rank structure of traffic data. We can also incorporate prior domain knowledge into the algorithm. For example, if some sensors are recently calibrated, we can assign 0 to the corresponding $\gamma_{e,t}$.

We show the computation time of the proposed algorithm with different network sizes. The computer programs are written in Python, and all the computational tests are performed on a desktop computer with an Intel 4.20 GHz CPU and 256 GB of memory. Additionally, the analysis period is 2 hours divided into 24 intervals. Table 7 summarizes the results. We find that the computation time gradually increases as the network size expands. Additionally, the computer runs out of memory when handling a network with 2000 nodes and 41825 links.

Table 7 Computation time of the proposed algorithm with different network sizes.

Number of nodes	Number of links	Computation time (seconds)	MAE (Vehicle counts)	MAPE
10	13	0.17	0.3894	0.0070
20	52	0.51	0.6942	0.0134
30	103	1.45	1.2141	0.0265
40	179	3.39	0.5752	0.0119
50	317	8.69	1.7811	0.0414
100	490	28.81	1.1058	0.0271
200	693	39.66	1.6954	0.0392
300	1210	360.29	1.4074	0.0348
400	1912	952.50	1.7804	0.0413
500	3017	3722.73	1.6177	0.0358
1000	10796	47994.65	1.3760	0.0319
2000	41825	Out of memory	/	/

E. Generation of measurement errors

In this section, we introduce how to generate erratic measurement errors using the following algorithm.

F. Extension for spatially correlated errors

In the above framework, we mainly focus on the independent erratic measurement errors. However, the correlated errors are inevitable in the real world. For example, the sensors installed on adjacent

```
Algorithm 3 Algorithmic statement of obtaining a local optimal solution
Input:
    Optimal solution for continuous $\gamma_{e,t}$: $\gamma^c_{e,t}$
    Estimated error probability: $\hat{p}_e$
    Expansion factor: $E_r \geq 1$
    Pre-estimated rank of $\boldsymbol{Q}$: $r_Q$
    An empty set: $O$
Procedure
    For each $\delta_{v,t} = 0$, let $\underset{e\in E(v)}{\gamma^c_{e,t}} = 0$ and $O = O \cup (e,t)$.
    For each sensor $e$, sort the numbers in set $\Lambda_e = \left\{\gamma^c_{e,t} | 1 \leq t \leq T, (e,t) \notin O\right\}$
    in descending order.
    For each sensor $e$, let $N_e = min\{E_r * ceil(p_e * T), |\Lambda_j|_{\#} - r_Q\}$.
    Assign the first $N_e$ elements in $\Lambda_e$ to 1 and the rest to 0.
    For each $\delta_{v,t} = 1$ and $\sum_{e\in E(v)} \gamma^c_{e,t} = 0$, randomly select one sensor $e_r$ in set
    $E(v) \bigcap O$ and let $\gamma^c_{e_r,t} = 1$.
    Solve the primal problem with $\gamma^c_{e_r,t}$.
```

```
Algorithm 4 Generation of erratic measurement errors
Input:
    Number of links: $e$
    Number of time intervals: $T$
    Magnitude of measurement errors: $\sigma$
    True flow data on link $e$ during time interval $t$: $q_{e,t}$
    Error probability of sensor: $p_e$
Output: Collected data with erratic measurement errors by sensor $e$
during time $t$: $q_{e,t}$
Procedure
for $e = 1, 2, \cdots E$ do
    for $t = 1, 2, \cdots T$ do
        Generate a random number $\alpha^p \sim Uniform(0,1)$.
        if $\alpha^p \leq p_e$ then
            Generate a random number $\alpha^d \sim Uniform(0,1)$.
            if $\alpha^d \leq 0.25$ then
                $q_{e,t} = \bar{q}_{e,t} + \sigma * Gaussian(0,1)$.    ▷ Gaussian errors.
            else if $\alpha^d \leq 0.5$ then
                $q_{e,t} = \bar{q}_{e,t} + \sigma * Uniform(-1,1)$.    ▷ Uniform errors
            else if $\alpha^d \leq 0.0.75$ then
                $q_{e,t} = \bar{q}_{e,t} + \sigma * Exp(\sigma)$.    ▷ Exponential errors.
            else
                $q_{e,t} = \bar{q}_{e,t} + \sigma * \Gamma(\sqrt{\sigma}, \sqrt{\sigma})$.    ▷ Gamma errors.
            end if
        end if
    end for
end for
if $q_{e,t} < 0$ or $q_{e,t} = \bar{q}_{e,t}$ then
    $q_{e,t} = \bar{q}_{e,t}/2$.    ▷ Non-negative traffic flow with errors.
end if
```

links usually exhibit similar error patterns, indicating that the errors are spatially correlated. This implies that the assumptions in Section 3 no longer hold. As a result, the solution method introduced in Section 4 fails to estimate the sensor error probabilities and recover traffic data successfully. In this section, we employ the common shock model to address these challenges.

F.1. Common shock model for modeling correlated errors

We use the Common Shock Model (CSM) to describe the dependencies of Poisson processes (Chiu, Jackson, and Kreinin 2017). According to the CSM, the correlations between links are caused by common shocks. For example, suppose that sensors 1 and 2 share a common shock $c_{1,2}$ that affects the occurrence of measurement errors. Mathematically, the arrival rates associated with sensors 1 and 2 can be expressed as follows:

$$\begin{aligned} \lambda_1^c &= \lambda_1 + c_{1,2}; \\ \lambda_2^c &= \lambda_2 + c_{1,2}. \end{aligned} \tag{48}$$

where λ_1^c and λ_2^c are the affected arrival rates of sensors 1 and 2, respectively. Moreover, the correlation coefficient $\rho_{1,2}$ between sensors 1 and 2 can be expressed as follows:

$$\rho_{1,2} = \frac{c_{1,2}}{\sqrt{(\lambda_1 + c_{1,2})(\lambda_2 + c_{1,2})}}. \tag{49}$$

In this paper, we consider only the second order correlations. Let ϕ_e denote the set of sensors that are correlated with sensor e. Given a node v, the error probability of VBS v can be formulated based on the CSM:

$$\boldsymbol{p}_v = 1 - exp(\sum_{e \in E^{(v)}} -\lambda_e + \sum_{e \in E(v)} \sum_{\forall e,e' \in \phi_e} -c_{e,e'}). \tag{50}$$

Following the same logic in Section 3.5, we first reformulate $\boldsymbol{p}_v$ for correlated errors as follows:

$$-log(1 - \boldsymbol{p}_v) = \sum_{e \in E^{(v)}} \lambda_e + \sum_{e \in E^{(v)}} \sum_{\forall e,e' \in \phi_e} c_{e,e'}. \tag{51}$$

We then introduce a series of independent variables $\boldsymbol{x}_e$ and common exogenous variables $\boldsymbol{x}_{e,e'}^c$ to characterize the causes of correlated errors:

$$\lambda_e = \boldsymbol{\beta}_1^T \boldsymbol{x}_e; \quad c_{e,e'} = \boldsymbol{\beta}_2^T \boldsymbol{x}_{e,e'}^c \tag{52}$$

Similarly, the coefficient vectors $\boldsymbol{\beta}_1^T$ and $\boldsymbol{\beta}_2^T$ can be estimated by the least squares or maximum likelihood estimation methods as discussed in Section 3.5.3. Finally, the arrival rate λ_e, common shock $c_{e,e'}$, correlation coefficient $\rho_{1,2}$, and sensor error probability p_e can be obtained.

F.2. Optimization model for correlated errors

For the correlated errors, we can use correlated Bernoulli distributions to characterize whether the collected data $q_{e,t}$ is corrupted by the errors. Let $P(\vec{\gamma}_t)$ denote the corresponding joint probability distribution of all sensors during time interval t. According to the correlated Bernoulli distribution, $P(\vec{\gamma}_t)$ can be expressed as follows:

$$P(\vec{\gamma}_t) = P_B(\vec{\lambda}_t)f(\vec{\gamma}_t); \tag{53}$$

$$f(\vec{\gamma}_t) = 1 + \sum_{e_1,e_2\in\Psi} \rho_{e_1,e_2} W_{e_1,t} W_{e_2,t} + \cdots + \sum_{e_1,\dots e_N\in\Psi} \rho_{e_1,\dots e_N} W_{e_1,t} W_{e_2,t}, \cdots, W_{e_N,t}, \tag{54}$$

where $P_B(\vec{\gamma}_t) = \prod_{e=1}^{E} (p_e)^{\gamma_{e,t}} (1-p_e)^{1-\gamma_{e,t}}$, Ψ is the set of correlated errors, $\rho_{e_1,\dots e_N}$ is the correlation coefficient among sensors $e_1 \cdots, e_N$ and $W_{e_n,t} = \frac{\gamma_{e_n,t} - p_{e_n}}{\sqrt{p_{e_n}(1-p_{e_n})}}$. Since we consider only the second order correlations, the corresponding joint probability distribution is formulated as follows:

$$P(\vec{\gamma}_t) = P_B(\vec{\lambda}_t)(1 + \sum_{e_1,e_2\in\Psi} \rho_{e_1,e_2} W_{e_1,t} W_{e_2,t}). \tag{55}$$

The log-likelihood over the estimation time horizon can be expressed as the sum of log-likelihoods for each time interval:

$$log(\mathcal{L}_c) = \sum_{t=1}^{T} \left[\sum_{e=1}^{E} \gamma_{e,t}(log(p_e) - log(1-p_e)) + \kappa_{e_1,e_2} \sum_{e_1,e_2\in\Psi} \gamma_{e_1,t}\gamma_{e_2,t} - p_{e_1}\gamma_{e_2,t} - p_{e_2}\gamma_{e_1,t} \right]; \tag{56}$$

$$\kappa_{e_1,e_2} = \frac{\rho_{e_1,e_2}}{\sqrt{p_{e_1}p_{e_2}(1-p_{e_1})(1-p_{e_2})}}. \tag{57}$$

Using $log(\mathcal{L}_c)$ as the primary objective function, we recover the traffic data with the maximum likelihood. However, it is worth noting that $log(\mathcal{L}_c)$ is a non-linear function due to the term $\gamma_{e_1,t}\gamma_{e_2,t}$, making it challenging to optimize directly. To linearize $log(\mathcal{L}_c)$, we use a new decision variable $\varphi_{e_{1,2},t}$ to replace $\gamma_{e_1,t}\gamma_{e_2,t}$, which results in the following constraints: $\varphi_{e_{1,2},t} \le \gamma_{e_1,t}; \varphi_{e_{1,2},t} \le \gamma_{e_2,t}$; $\varphi_{e_{1,2},t} \ge \gamma_{e_1,t} + \gamma_{e_2,t} - 1$; $\varphi_{e_{1,2},t} \in \{0,1\}$.

Since the error probability p_e and correlation coefficient ρ_{e_1,e_2} are estimated by the common shock model, the multi-objective optimization model in Section 3.3 can be easily extended to address the correlated errors. The primary and second objectives of our model remain as the minimization of the negative log-likelihood $log(\mathcal{L}_c)$ and the nuclear norm of $\hat{\boldsymbol{Q}}$, respectively. For the sake of clarity, let $\boldsymbol{C_1}$, $\boldsymbol{C_2}$, $\boldsymbol{C_3}$ and $\boldsymbol{C_4}$ denote the coefficient matrices in the constraints. The data recovery model for correlated errors can be formulated as follows:

$$\min \ \left[log(\mathcal{L}_c), \|\hat{\boldsymbol{Q}}\|_* \right]$$

$$\boldsymbol{C}_1\hat{\boldsymbol{Q}} + \boldsymbol{C}_2\boldsymbol{Z} + \boldsymbol{C}_3\vec{\gamma} + \boldsymbol{C}_4\vec{\varphi} = 0; \tag{58a}$$

$$\hat{\boldsymbol{Q}} \ge 0, \ \boldsymbol{Z} \ge 0, \ \vec{\gamma} \in \{0,1\}, \ \vec{\varphi} \in \{0,1\}. \tag{58b}$$

F.2.1. Experiments for correlated errors To validate the extended model, we conduct numerical experiments using the data with correlated errors. In this experiment, we mainly focus on the second order correlations. For link e, we simply assume a correlation with its corresponding inverse link e'. For example, links 1 and 9 in Figure 4 are considered correlated. We use the same explanatory variables in Table 3 to estimate sensor error probabilities, with the exception that lane number and speed limit are used as the common exogenous variables to estimate the common shock. This means that links e and e' share the same lane number and speed limit, and the corresponding coefficients are set to 0.003 and 0.001, respectively.

Table 8 Results of the estimated ρ and c with correlated sensors.

Correlated sensors	True ρ	True c	Estimated ρ	Estimated c	MAE/MAPE of ρ	MAE/MAPE of c
1 and 9	0.4191	0.1505	0.4643	0.1607	0.0452/0.1078	0.0102/0.0681
2 and 34	0.4996	0.1540	0.5565	0.1567	0.0569/ 0.1139	0.0027/0.0178
3 and 20	0.3529	0.1278	0.3775	0.1259	0.0245/0.0695	0.0019/0.0146
4 and 30	0.4439	0.1155	0.5241	0.1196	0.0802/ 0.1807	0.0041/0.0354
5 and 31	0.4934	0.1839	0.5653	0.1838	0.0719/0.1457	0.0001/0.0006
6 and 37	0.3443	0.1168	0.3389	0.1130	0.0053/ 0.0155	0.0038/0.0325
7 and 10	0.3373	0.1078	0.3558	0.1025	0.0186/0.0550	0.0054/0.0499
8 and 23	0.2468	0.1645	0.2902	0.1610	0.0434/0.1758	0.0035/0.0214
11 and 13	0.3400	0.1372	0.3307	0.1289	0.0093/0.0273	0.0083/0.0606
12 and 24	0.4762	0.1788	0.5015	0.1859	0.0252/0.0530	0.0071/0.0398
14 and 17	0.3891	0.1257	0.4263	0.1234	0.0372/0.0955	0.0022/0.0178
15 and 27	0.3007	0.1473	0.2891	0.1488	0.0115/0.0384	0.0016/0.0106
16 and 35	0.3287	0.0939	0.3101	0.0860	0.0186/0.0566	0.0078/0.0835
18 and 21	0.4816	0.1631	0.4781	0.1594	0.0036/0.0075	0.0038/0.0231
19 and 32	0.4620	0.1799	0.4626	0.1872	0.0007/0.0014	0.0073/0.0406
22 and 36	0.4380	0.1557	0.4717	0.1507	0.0337/0.0770	0.0051/0.0325
25 and 28	0.5459	0.1694	0.5678	0.1667	0.0219/0.0400	0.0027/0.0157
26 and 38	0.1637	0.1013	0.1943	0.1029	0.0307/0.1873	0.0016/0.0157
29 and 33	0.4064	0.1629	0.4436	0.1754	0.0373/0.0917	0.0124/0.0763
39 and 40	0.3403	0.1308	0.3340	0.1375	0.0062/0.0183	0.0068/0.0518
41 and 42	0.2001	0.1078	0.2139	0.1105	0.0138/0.0689	0.0027/0.0253
43 and 44	0.3925	0.1730	0.3892	0.1872	0.0033/0.0085	0.0142/0.0821
45 and 46	0.2477	0.1537	0.2825	0.1482	0.0349/0.1407	0.0054/0.0353
47 and 48	0.3568	0.1134	0.3667	0.1009	0.0098/0.0276	0.0125/0.1103
49 and 50	0.3318	0.1171	0.3217	0.1133	0.0101/0.0303	0.0038/0.0321
Average MAE/MAPE of ρ: 0.0261/0.0734				Average MAE/MAPE of c: 0.0055/0.0397		

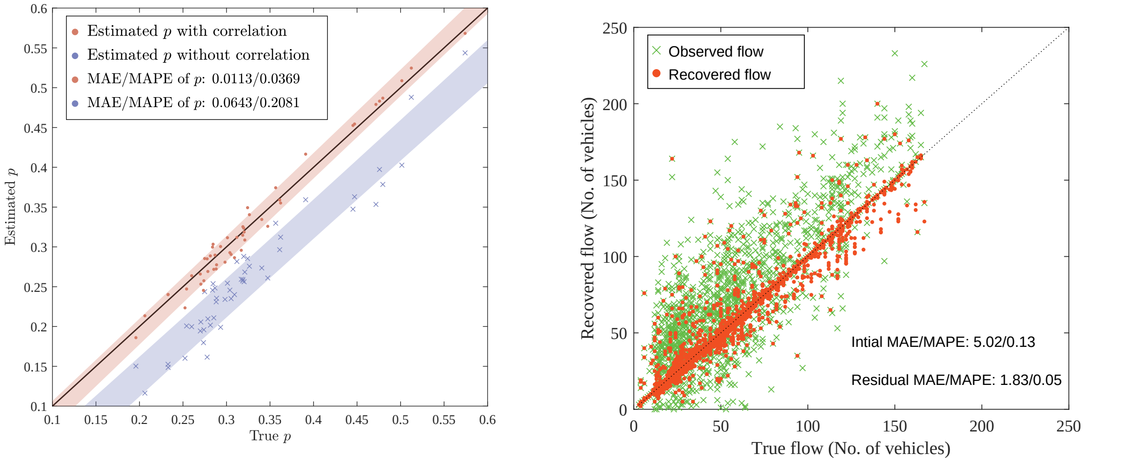
Taking the flow data with correlated measurement errors as the input, the extended model can estimate the correlation ρ between sensors, common shock c, sensor error probability p, and traffic flow q. Table 8 presents the true and estimated correlation ρ and common shock c in detail. We find that the correlation ρ and common shock c can be successfully estimated by the extended model. The average MAE/MAPE of ρ and c are 0.0261/0.0734 and 0.0055/0.0397, respectively. We also compare the estimated error probabilities when considering correlations of sensors with that obtained without considering correlations in Figure 18(a). The shaded region is the 95% probabilistic range of the estimated error probabilities. We find that the model for independent errors cannot deal with the correlated errors and produces a biased estimation of error probabilities.

In contrast, the extended model can well address the correlated errors and provides an accurate estimation of error probabilities. Moreover, we visualize the true flow, observed flow, and recovered flow by the extended model in Figure 18(b). The MAE and MAPE of the recovered flow are 1.83 and 0.05, respectively, which are significantly less than the initial MAE (5.05) and MAPE (0.13). These results imply that the extended model is still effective in estimating the correlated errors in traffic flow data.

To further quantify the spatial correlation of the identified erratic errors, we employ Moran's Index (Ward and Gleditsch 2018) to measure the similarity of error occurrences in neighboring links:

$$Moran\ Index = \frac{|E|_{\#}}{U} \cdot \frac{\sum_{e=1}^{|E|_{\#}} \sum_{e'=1}^{|E|_{\#}} u_{ee'} (\sum_{t=1}^{T} \gamma_{e,t} - \bar{\gamma})(\sum_{t=1}^{T} \gamma_{e',t} - \bar{\gamma})}{\sum_{e=1}^{|E|_{\#}} (\sum_{t=1}^{T} \gamma_{e,t} - \bar{\gamma})^2}, \tag{59}$$

where $u_{ee'}$ is the spatial weight between links e and e'. When links e and e' are connected to the same node, $u_{ee'} = 1$; otherwise, $u_{ee'} = 0$. U is the sum of all spatial weights $u_{ee'}$. $\bar{\gamma}$ is the mean of all $\gamma_{e,t}$. Additionally, we use $z - test$ to determine the statistical significance of the observed spatial correlation. In this experiment, we find that Moran's Index value is 0.1708 and the p-value is 0.0012. The positive Moran's Index indicates a positive spatial correlation, meaning that erratic errors tend to occur in neighboring links within the network. The p-value of 0.0012 suggests that the observed spatial correlation is statistically significant, rather than occurring by random chance.



(a) Estimation of sensor error probability p with correlated errors.

(b) Estimation of traffic flow with correlated errors.

Figure 18 Results of estimated sensor error probabilities and recovered traffic flow with correlated errors.