Optimal sensor placement and density estimation in large-scale traffic networks

Martin Rodriguez-Vega

UNIVERSITÉ GRENOBLE ALPES

THÈSE

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE ALPES

Spécialité : **Automatique - productique**

Arrêté ministériel : 7 août 2006

Présentée par
Martin RODRIGUEZ VEGA

Thèse dirigée par **Carlos CANUDAS DE WIT** et
codirigée par **Hassen FOURATI**

préparée au sein du
laboratoire Grenoble Images Parole Signal Automatique (GIPSA)
dans **l'école doctorale Eléctronique Eléctrotechnique Automatique & Traitement du signal (EEATS)**

Optimal sensor placement and density estimation in large-scale traffic networks

Thèse soutenue publiquement le **09 avril 2021**,
devant le jury composé de:

Gildas BESANÇON
Professeur des Universités, Grenoble INP, Président du jury
Antonella FERRARA
Professeure des Universités, Università di Pavia, Rapportrice
Bernhard FRIEDRICH
Directeur de recherche, Technische Universität Braunschweig, Rapporteur
Francesco VITI
Professeur associé, Université du Luxembourg, Examinateur
Hassen FOURATI
Professeur associé, GIPSA-Lab, Co-encadrant de thèse (Invité)

UNIVERSITÉ DE GRENOBLE ALPES

ÉCOLE DOCTORALE EEATS

Eléctronique, Eléctrotechnique, Automatique, Traitement du Signal

T H È S E

pour obtenir le titre de

docteur en sciences

de l'Université de Grenoble

Mention : AUTOMATIQUE

Présentée et soutenue par

Martin RODRIGUEZ VEGA

Optimal sensor placement and density estimation in large-scale traffic networks

Thèse dirigée par Carlos CANUDAS-DE-WIT

préparée au laboratoire Grenoble Images Parole Signal Automatique (GIPSA-Lab)

soutenue le 09/04/2021

Jury :

Rapporteurs :	Antonella FERRARA	-	Università di Pavia
	Bernhard FRIEDRICH	-	Technische Universität Braunschweig
Co-encadrant :	Hassen FOURATI	-	GIPSA-Lab
Président :	Gildas BESANÇON	-	Grenoble INP
Examinateur :	Francesco VITI	-	Université du Luxembourg

Résumé — Ce travail de doctorat est effectué dans le cadre du projet Scale-FreeBack et financé par le Conseil de Recherche Européen. Son objectif général est de poser de nouvelles bases pour une théorie traitant des réseaux physiques complexes à dimension arbitraire. Un cas particulier est celui des systèmes de transport intelligents, qui sont confrontés à des défis importants pour empêcher l'apparition de congestions. Les contributions du travail de doctorat se concentrent sur la phase de surveillance de la gestion du trafic. Nous traitons trois problèmes principaux : la localisation des capteurs dans le cadre de contraintes budgétaires, l'estimation de la densité du trafic en utilisant des sources de données hétérogènes, et l'estimation d'un état de trafic moyen pour une région d'un réseau. Tout d'abord, le problème du placement optimal des capteurs est examiné. Notre contribution consiste en analyser deux technologies de capteurs, l'une qui mesure le flux absolu dans une route, et l'autre qui estime les paramètres d'intersection ici appelés turning ratios, qui sont une mesure relative du comportement des conducteurs en matière de choix d'itinéraire. Nous trouvons comment choisir les emplacements pour chaque type de technologie de telle sorte qu'un nombre minimal de capteurs soit nécessaire et que le flux de trafic puisse être calculé pour chaque route du réseau. Le deuxième problème concerne la reconstruction du flux et de la densité à l'aide de sources d'information hétérogènes. En plus des capteurs fixes analysés dans le premier problème, une autre source de données disponible est le *Floating Car Data* (FCD), qui regroupe les trajectoires des véhicules individuels, avec un taux de pénétration inconnu et généralement agrégées en raison de la réglementation sur la protection de la vie privée. Nous analysons comment intégrer ces sources de données afin de pouvoir estimer la densité et le débit de chaque route du réseau, à la fois statiques et dynamiques en fonction de la quantité d'informations disponibles. Pour le troisième problème, nous considérons l'estimation de la densité agrégée d'une zone urbaine. Cela est intéressant lorsque la densité de chaque route individuelle d'une zone n'est pas nécessaire, ou lorsque la puissance de calcul est limitée. La convergence d'un estimateur pour la densité moyenne de la zone a été analysée, mais il a été constaté qu'en général, l'estimateur ne converge pas. Pour résoudre ce problème, nous proposons une méthode qui calcule une représentation virtuelle du même réseau physique où chaque route est divisée en un certain nombre de cellules, de sorte que l'estimateur du système virtuel converge. Dans certaines conditions, nous montrons que la différence entre les moyennes réelles et virtuelles est faible. L'efficacité de nos contributions a été testée en utilisant des données simulées et réelles. La simulation consiste en une application du célèbre logiciel de trafic microscopique Aimsun, où la dynamique des véhicules individuels est calculée dans un réseau réel modélisé. Les données réelles sont obtenues à partir de capteurs situés dans le centre-ville de la ville de Grenoble.

Mots clés : Reseaux de trafic à grande échelle, Estimation de flux et densité, Placement des capteurs

Abstract — This PhD thesis is done in the context of the ERC Advanced Grant project Scale-FreeBack. Its overall aim is to set new foundations for a theory dealing with complex physical networks with arbitrary dimension. One particular case is intelligent transportation systems and their challenges to prevent the occurrence of congestions. The contributions of the PhD work are mainly related to the monitoring of large-scale traffic network states. We deal with three main problems: the location of sensors under budget constraints, the estimation of traffic density using heterogeneous data sources, and the estimation of an aggregated state for a region of an urban traffic network. First, the optimal sensor placement problem is considered. Our contribution consists in the analysis of two sensor technologies, one that measures the absolute flow in a road, and another that estimates intersection parameters called turning ratios, which are a relative measure of route choice behavior of drivers. We propose an approach to choose the locations for each type of technology such that a minimal number of total sensors are required, and that traffic flow can be estimated for each road of the network. The second problem considers the estimation of flow and density using heterogeneous sources of information. In addition to the fixed flow sensors used in the first problem, another available data source is the so called Floating Car Data (FCD) which provides the trajectories of individual vehicles, albeit at an unknown penetration rate and generally aggregated due to privacy regulations. We analyze how to fuse these data sources to be able to estimate the density and flow of every road in the network, for the static and dynamical cases depending on the amount of available information. For the third problem, we consider the estimation of the aggregated density of an urban network. This is of interest when the density of every individual road of a zone is not required to be known, or when computational power is limited. The convergence of an estimator for the average density of the zone was analyzed, but it was found that in general, the estimator does not converge. To solve this problem, we propose a method to calculate a virtual representation of the same underlying physical network where each road is divided into a number of cells, such that the estimator for the virtual system converges. Under certain conditions, we show that the difference between the real and virtual averages is small. The effectiveness of our contributions were tested using simulated and real data. In the first case, the simulation consists of an application of the well known microscopic traffic software Aimsun, where the dynamics of individual vehicles are calculated in a modeled real network. In the second case, real data is obtained from sensors located in the downtown area of the city of Grenoble and collected using the Grenoble Traffic Lab(GTL).

Keywords: Large scale traffic networks, Density and flow estimation, Sensor location

Grenoble Images Parole Signal Automatique (GIPSA-Lab)
11 Rue des Mathématiques, 38400 Saint-Martin-d'Hères

Résumé

Le développement des réseaux de transport est l'un des principaux enjeux de l'expansion des villes modernes. À mesure que les zones urbaines s'étendent et se peuplent, les embouteillages deviennent plus fréquents et plus coûteux. Les Systèmes de Transport Intelligents (Intelligent Transportation Systems - ITS) font référence à l'utilisation de technologies et d'applications avancées pour offrir des solutions innovantes aux problèmes de congestion du trafic. Certaines techniques étudiées par les ITS consistent en la régulation du trafic (mesure de la vitesse, tarification, fourniture d'informations, contrôle du cycle des feux de circulation) et la planification stratégique des transports. L'Estimation de l'Etat du Trafic (Traffic State Estimation - TSE) est utile pour fournir des informations vitales à l'application efficace et précise des stratégies de contrôle du trafic routier. Le TSE consiste à utiliser des données de trafic partiellement accessibles et bruitées pour déduire la valeur de variables de trafic, telles que le débit, la densité, la vitesse, ou le temps de parcours.

Cette thèse traite du TSE dans le cas de réseaux de trafic urbain à grande échelle. À cette fin, des méthodes sont proposées pour résoudre deux problèmes principaux : comment trouver la localisation optimale des capteurs, et comment incorporer des sources de données hétérogènes dans une approche d'estimation du débit et de la densité. Dans le contexte de l'état de l'art, la plupart des travaux TSE ne considèrent que le cas où les autoroutes, et les travaux traitant des réseaux généraux nécessitent souvent des opérations complexes inadaptées aux très grands systèmes ou supposent l'utilisation de données très riches qui pourraient ne pas être disponibles pour la plupart des applications. Par conséquent, nos principales contributions consistent à proposer des méthodes efficaces pour la localisation des capteurs et l'estimation de l'état en utilisant des sources de données couramment utilisées.

Les principaux chapitres de cette thèse sont résumés ci-dessous.

Estimation de la densité et du débit en régime stationnaire

Ce chapitre traite le problème de l'estimation du débit et de la densité dans les réseaux urbains dans le cas de régimes stationnaires. Ceci est fait en deux étapes. Dans la première étape, nous examinons le problème de la minimisation du nombre de capteurs nécessaires pour récupérer complètement le débit de véhicules. Nous considérons deux technologies de capteurs possibles : l'une qui permet de mesurer les Rapport de Virage (Turning Ratio - TR) à une intersection donnée et l'autre qui mesure directement le débit dans une route. La deuxième étape consiste à utiliser les mesures des capteurs localisés dans la première étape en plus des Données des Véhicules Flottantes (Floating Car Data - FCD) pour cestimer le débit et la densité de chaque route du réseau.

Pour résoudre l'étape de localisation des capteurs, la condition de régime stationnaire a été utilisée pour écrire un ensemble de contraintes linéaires utilisant uniquement la conservation du

débit à chaque intersection, en tenant compte de la présence ou de l'absence d'informations sur le TR. Ainsi, le problème de localisation des capteurs pouvait être résolu en réduisant l'espace nul de la matrice résultante. Nous avons résolu ce problème en trouvant la solution de deux problèmes indépendants : localiser de manière optimale un nombre donné de capteurs TR, puis, en utilisant ce résultat comme entrée, localiser les capteurs de débit. La méthode proposée trouve l'emplacement du nombre minimal de ces capteurs en temps quasi-linéaire, en associant l'espace nul de la matrice à l'espace des cycles du graphe sous-jacent, de sorte que des techniques efficaces de parcours de graphe peuvent être utilisées. La complexité des algorithmes proposés s'est révélée beaucoup plus efficace que les méthodes précédentes qui reposent sur des calculs algébriques.

En utilisant les informations des capteurs installés, l'étape d'estimation a été formulée comme un problème d'optimisation quadratique, pour calculer simultanément le débit et la densité dans chaque route en utilisant les données des capteurs et les vitesses routières moyennes obtenues à partir du FCD. La fonction de coût est constituée de la différence entre les débits mesurés et estimés, et de la distance entre les estimations de densité et les valeurs prédites à l'aide du Diagramme Fondamental (Fundamental Diagram - FD).

Les approches proposées ont été testées à l'aide de simulations microscopiques dans le réseau réel de Grenoble. Les résultats de l'estimation se sont révélés très proches de la vérité de terrain, avec des erreurs relatives médianes inférieures à 10% pour l'estimation du débit et de la densité. Cependant, l'exigence de conditions stationnaires limite l'applicabilité des méthodes.

Estimation dynamique de la densité et du débit

Dans ce chapitre, nous avons proposé une méthode pour estimer l'évolution dynamique du débit et de la densité dans les réseaux à grande échelle de circulation urbaine en utilisant des sources de données hétérogènes : capteurs de comptage, capteurs d'identification des véhicules et FCD. Nous avons discuté de la mise en œuvre de ces types de données dans des scénarios de cas réels tels que des taux d'échantillonnage et des intervalles de temps variables, et de la manière de les prendre en compte dans la mise en œuvre de l'estimateur.

La méthode nécessite la connaissance de peu de paramètres, car il n'y a pas de modélisation explicite des débits de sortie en termes de la densité, donc aucune FD n'est nécessaire. L'estimation repose sur la loi de conservation pour chaque route et l'équation hydrodynamique : le débit sortant de chaque route est calculé comme le produit entre la densité estimée et la vitesse moyenne donnée par la FCD.

Cependant, la connaissance des TR pour chaque intersection est nécessaire. Pour simplifier cette exigence, nous supposons qu'un nombre limité de capteurs de TR peut être installé et que pour les autres intersections, ils peuvent être estimés en utilisant une méthode heuristique basée sur le structure du réseau. Le schéma de localisation des capteurs prend en entrée les valeurs estimées des TR et estime ensuite l'erreur dans la reconstruction de la densité

introduit par les déviations des valeurs des TR par rapport aux valeurs réelles. Pour chaque intersection, un poids de sensibilité est calculé comme l'énergie totale de l'erreur due à une déviation donnée des paramètres pour chacune des routes entrantes. Ensuite, les capteurs sont placés dans les intersections ayant le poids de sensibilité le plus élevé.

Nous avons testé les méthodes proposées en utilisant des simulations microscopiques sur une sélection de la ville de Grenoble. Les résultats montrent que l'estimateur est performant, puisqu'il peut estimer la trajectoire globale des états du trafic, avec de nombreuses routes présentant peu d'erreurs. Les méthodes proposées sont flexibles, car elles peuvent être appliquées à différents coûts. En outre, l'algorithme permet l'inclusion de nouvelles données, de sorte qu'avec le temps, les informations précédentes peuvent être utilisées pour améliorer l'emplacement des capteurs de TR afin d'augmenter les performances de l'estimateur.

Estimation de la densité moyenne

Dans ce chapitre, nous considérons l'estimation de la moyenne de toutes les densités routières dans un réseau de trafic. Nous avons analysé la convergence d'un observateur unidimensionnel en boucle ouverte pour la densité moyenne d'un réseau, dans le cadre d'une dynamique de circulation linéaire telle que le régime d'écoulement libre. En général, de tels observateurs ne convergent pas, mais nous proposons une méthode pour obtenir une représentation virtuelle du réseau avec les mêmes propriétés physiques de sorte que l'observateur converge. Cette nouvelle représentation est construite en divisant chaque route du réseau original en un certain nombre de cellules de longueurs spécifiques. Nous montrons qu'il est possible de trouver une représentation virtuelle du réseau utilisant des divisions routières inhomogènes de telle sorte que l'observateur proposé converge vers la densité moyenne du système virtuel.

Bien que la densité moyenne du système original et celle du système virtuel ne soient pas nécessairement égales, nous montrons qu'en considérant des réseaux à grande échelle, la différence entre ces deux variables est presque toujours faible, et donc que la méthode peut donner une approximation utile de la valeur d'intérêt.

Les méthodes ont été testées en utilisant des traces simulées de véhicules individuels dans le cas réel de la ville de Grenoble, France. La densité moyenne estimée s'est avérée proche des valeurs réelles, avec une erreur RMS de 10%.

Application à la ville de Grenoble

Ce chapitre présente le cas d'étude du centre-ville de Grenoble et valide les méthodologies décrites au chapitre 3 pour l'estimation dynamique du débit et de la densité de chaque route d'un réseau, et l'estimation de la densité moyenne de l'ensemble du réseau décrite au chapitre 4, en utilisant des données réelles.

La section considérée du réseau de circulation de Grenoble a été équipée de capteurs collectant des informations sur le débit de circulation et les vitesses dans les limites du domaine, ainsi que les TR pour un nombre sélectionné d'intersections. Ces informations sont disponibles sur la plateforme expérimentale GTL-Ville, qui fournit également des données et des analyses de trafic en temps réel.

Pour calculer les TR des intersections qui n'étaient pas équipées de capteurs, nous avons utilisé la Classification Fonctionnelle des Routes (Functional Road Classification - FRC), qui attribue à chaque route un indice correspondant à sa fonction (par exemple, s'il s'agit d'une route principale ou d'une simple route dans une zone résidentielle). Un poids a été attribué à chaque indice FRC, et les TR pour chaque intersection ont été supposés être proportionnels au poids FRC des routes sortantes. Les poids ont été calibrés à l'aide d'un problème d'optimisation de correspondance des débit en utilisant les données des capteurs d'entrée et de sortie aux limites du réseau.

Pour la validation de l'estimation dynamique de la densité dans chaque route, les algorithmes décrits ont été mis en œuvre. Un ensemble de capteurs situés à l'intérieur du réseau a été utilisé pour la validation croisée. Les estimations ont ensuite été comparées aux valeurs réels fournies par ces capteurs. Bien que le problème du TSE dans les réseaux à grande échelle soit difficile, les résultats obtenus sont encourageants car le débit estimé pour les routes individuelles est très proche des données réels fournies par les capteurs. Pour plus de la moitié des emplacements de validation, les trajectoires moyennes présentent une erreur inférieure à 20%, et dans tous les cas, les erreurs sont inférieures à 45%. Nous identifions comme principale source d'erreur l'incertitude des valeurs des TR, due à l'utilisation de la méthode heuristique basée sur les données FRC. Cependant, ceci peut être amélioré à l'avenir en effectuant plus de campagnes de mesure, ainsi les résultats d'estimation dans l'application réelle devraient s'améliorer de manière significative.

En ce qui concerne l'estimation de la densité moyenne, les résultats montrent que pendant une grande partie de la journée, les estimations et les valeurs réels sont très proches, avec une erreur moyenne de 8%. Cependant, il a été mis en évidence que lorsque les hypothèses sous-jacentes de la dynamique linéaire ne sont pas satisfaites, les trajectoires des densités réelles et estimées divergent, augmentant l'erreur à presque 20%.

Conclusions et perspectives

Bien que les méthodes décrites dans ce travail de recherche aient fourni de bons résultats initiaux, il reste des problèmes et des questions ouvertes qui peuvent apporter des améliorations significatives à la précision des estimations, et qui peuvent servir de base à de futures recherches.

Une voie claire est l'amélioration des méthodes d'estimation des TR. Nous avons montré que l'utilisation du FRC de chaque route pour estimer ces paramètres peut conduire à des estimations utiles de la propagation des débit dans le réseau. Néanmoins, l'analyse que nous

avons effectuée pour ce modèle n'est en aucun cas exhaustive, et la prise en compte d'autres paramètres tels que la géométrie du réseau (par exemple, l'angle de raccordement entre les routes) peut conduire à des résultats plus précis. Pour ce faire, des mesures supplémentaires des TR dans le réseau sont nécessaires, ce qui permettra d'améliorer l'estimation dynamique du débit et de la densité. Une autre amélioration possible consiste à envisager des approches basées sur les chemins plutôt que sur les intersections, car celles-ci prennent en compte non seulement la structure du réseau mais aussi des informations supplémentaires relatives aux itinéraires possibles entre les origines et les destinations.

Une importante question ouverte est de savoir comment traiter la congestion dans le cas de l'estimateur de la densité moyenne. Ce problème n'est pas trivial, car il concerne entre autres l'étude de la détectabilité et de l'observabilité des systèmes non linéaires, pour lesquels les outils théoriques sont limités. La principale difficulté concernant cette extension est que le suivi de la congestion nécessite de stocker l'état de chaque route, ce qui va à l'encontre de l'objectif d'avoir un estimateur à faible dimension. Néanmoins, une étude plus approfondie pourrait permettre de développer de nouveaux outils théoriques capables de fournir un descripteur plus général de la dynamique moyenne du trafic.

Dans la continuité de cette thèse, l'un des objectifs principaux est de valoriser les données réelles de la ville de Grenoble et les approches d'estimation proposées pour améliorer la plate-forme GTL-Ville. Par exemple, cela peut être fait en utilisant les débit et les densités estimés comme entrée pour les méthodes qui prédisent la consommation d'énergie et les émissions des véhicules qui sont générées par les utilisateurs du réseau. Ces informations sont cruciales pour le développement de la gestion écologique du trafic, qui élabore des stratégies visant à améliorer la durabilité des réseaux de transport.

Contents

List of Figures

List of Tables

List of acronyms

AE	Absolute Error
AVI	Automatic Vehicle-Identifier
BT	Bluetooth
CDF	Cumulative Distribution Function
CTM	Cell Transmission Model
DFS	Depth First Search
DTA	Dynamic Traffic Assignment
FCD	Floating Car Data
FD	Fundamental Diagram
FRC	Functional Road Classification
GTL	Grenoble Traffic Lab
ITS	Intelligent Transportation System
LWR	Lighthill Whitham Richards model
ME	Mean Error
MFD	Macroscopic Fundamental Diagram
NSLP	Network Sensor Location Problem
O/D	Origin Destination
RAE	Relative Absolute Error
RME	Relative Mean Error
TA	Traffic Assignment
TR	Turning Ratio
TSE	Traffic State Estimation

List of symbols

$\mathcal{N}$	Set of intersections.
$\mathcal{C}$	Set of source and sink nodes of traffic.
$\mathcal{E}$	Set of roads.
$\mathcal{E}_{\text{in}}, \mathcal{E}_{\text{out}}$	Sets of boundary incoming and outgoing roads, respectively.
$\mathcal{E}_{\text{net}}$	Set of internal roads.
$\mathcal{I}(k), \mathcal{O}(k)$	Sets of incoming and outgoing roads to/from intersection k, respectively.
$\mathcal{T}$	Set of possible turns between roads.
$\mathcal{B}$	Set of intersections equipped with Bluetooth sensors.
$\mathcal{R}$	Set of intersections with turning ratio measurements.
$\mathcal{U}$	Set of unmeasured intersections.
$\mathcal{S}$	Set of roads with flow sensors.
ρ_i	Density of road i.
$\varphi_i^{in}, \varphi_i^{out}$	Incoming and outgoing flow to/from road i, respectively.
v_i	Space-mean speed of road i.
$\boldsymbol{\rho}$	Vector of road densities.
$\boldsymbol{\varphi}^{\text{in}}, \boldsymbol{\varphi}^{\text{out}}$	Vectors of road inflows and outflows, respectively.
$\mathbf{v}$	Vector of road mean speeds.
$\boldsymbol{\varphi}^{\text{mea}}$	Vector of measured flows.
$\boldsymbol{\varphi}^{\text{ext}}$	Vector of external input demands.
$\mathbf{v}^{\text{FCD}}$	Vector of road speeds measured via floating car data.
$\mathbf{y}$	Output vector.
ℓ_i	Length of road i.
$r_{i,j}$	Turning ratio from road i to road j.
R	Turning ratio matrix.
V	Diagonal road speed matrix.
L	Diagonal road length matrix.

$\mathcal{V}_i$	Set of vehicle indexes inside of road i.
ν_α	Velocity of vehicle α.
ρ_{av}	Average density of a traffic network.
$\delta_i^{(k)}$	Length of the k-th cell of road i.
$\mathbf{n}$	Vector of number of cells per road, division vector.
Γ_i	Number of lanes of road i.
Φ	Flow-density fundamental diagram.
Ψ	Speed-density fundamental diagram.
ρ^c	Critical density.
ρ^{max}	Jam density.
φ^{max}	Road capacity.
w	Congested wave propagation speed.
$\deg^{\text{out}}(k)$	Out-degree of node k.
O	Big O notation for computational complexity.
$\mathbb{I}$	Identity matrix.
$\mathbf{u}_i$	Standard basis vector for coordinate i.

Chapter 1

Introduction

The development of transportation networks is one of the main issues in the evolution of modern cities. As urban areas become larger and more populated, traffic congestions occur more frequently with higher levels of cost in todays life. Intelligent Transportation Systems (ITS) refers to the use of advanced technologies and applications to offer innovative solutions to the problems of traffic congestion. Some techniques under study by ITS consist on traffic control (ramp metering, pricing, information provision, traffic light cycle control), and strategic transportation planning. Traffic State Estimation (TSE) is one of the main ingredients required to provide vital information to the efficient and accurate application of traffic control strategies. TSE refers to the use of partially observed and noisy traffic data to infer the value of traffic variables, such as flow, density, velocity, traveling time, and others [Seo+17].

There are three main components to the development of any TSE method: a traffic flow model, the use of input data as partial observation of the traffic state, and a state estimation approach [Seo+17]; [Ros+20].

1.1 Traffic flow modeling

1.1.1 Single roads

To model traffic dynamics, the main two types of model are microscopic and macroscopic.

1.1.1.1 Microscopic models

Microscopic models describe the trajectory of each individual vehicle. They are based on the assumption that each driver is affected only by the state of the vehicle directly in front [New02], and thus are also known as car-following-models [WK+15]. Each trajectory is described by means of an Ordinary Differential Equation (ODE). Consider a vehicle indexed by α, whose position at time t is $s_\alpha(t)$, and its velocity $\nu_\alpha(t)$. Its trajectory is described by

$$\begin{aligned} \dot{s}_\alpha(t) &= \nu_\alpha(t) \\ \dot{\nu}_\alpha(t) &= a(s_\alpha - s_{\alpha-1}, \nu_\alpha, \nu_{\alpha-1}) \end{aligned} \tag{1.1}$$

where $\alpha - 1$ is the indexing of the vehicle in front, and a is called the acceleration function. This function is designed to provide more realistic description of real traffic phenomena, such

as reaction time, desired speed profile, breaking time, etc. For instance, the Optimal Velocity Model (OVM) [Ban+95] models the acceleration function as

$$a(\Delta s, \nu_\alpha) = \frac{\nu_{opt}(\Delta s) - \nu_\alpha}{\tau} \tag{1.2}$$

where $\Delta s = s_\alpha - s_{\alpha-1}$, and ν_{opt} is called the optimal-velocity function. This model assumes that each vehicle has a desired velocity that will be attained on a time scale given by the parameter τ. This function can be chosen arbitrarily under some constraints [TK13], but a simple choice proposed in [Sug99] is

$$\nu_{opt}(\Delta s) = \tanh(\Delta s - 2) + \tanh(2) \tag{1.3}$$

Due to their nature, microscopic models are ill-suited to describe general traffic conditions, or to serve for estimation or control techniques. Instead, microscopic models are well-suited to study the effects of single vehicles in traffic streams, the interactions between heterogeneous traffic compositions, describing human behavior, and to provide realistic simulations that serve as validation data for other methods.

1.1.1.2 Macroscopic models

Macroscopic models describe traffic as a continuous, similarly as the description of fluid dynamics. The main variables are the density, flow, and velocity. The traffic density $\rho(x,t)$ at time t around a point x, refers to the number of vehicles per unit length, and has units of vehicles per kilometer (veh/km). Traffic flow $\varphi(x,t)$ refers to the number of vehicles that traverse some location x per unit time, and has units of vehicles per hour (veh/h). The velocity $v(x,t)$ refers to the average speed of vehicles, with common units kilometers per hour (km/h).

The basis for any macroscopic traffic model is the conservation law of vehicles, which is expressed by means of a Partial Differential Equation (PDE)

$$\frac{\partial}{\partial t}\rho(x,t) + \frac{\partial}{\partial x}\varphi(x,t) = 0. \tag{1.4}$$

Furthermore, the tree variables are related by the hydrodynamic relation

$$\varphi(x,t) = v(x,t)\rho(x,y). \tag{1.5}$$

The development of traffic models started with [Gre+34], who manually collected data from a road stretch during a period of time, measuring the speed of each vehicle and the inter-vehicle spacing (from which density can be obtained). By plotting this data in a speed-density diagram, the authors observed an inverse correlation between these variables. They proposed a linear function of the velocity in terms of the density, which is also known as the Greenshield's Fundamental Diagram (FD) (see Fig. 1.1). With this idea, [LW55] and [Ric56], independently proposed the Lighthill-Whitham-Richards (LWR) model, which assumes that the speed (and flow) are functions of the traffic density, only. Thus,

$$\varphi(x,t) = \Phi(\rho(x,t)) \quad , \quad v(x,t) = \Psi(\rho(x,t)) \tag{1.6}$$

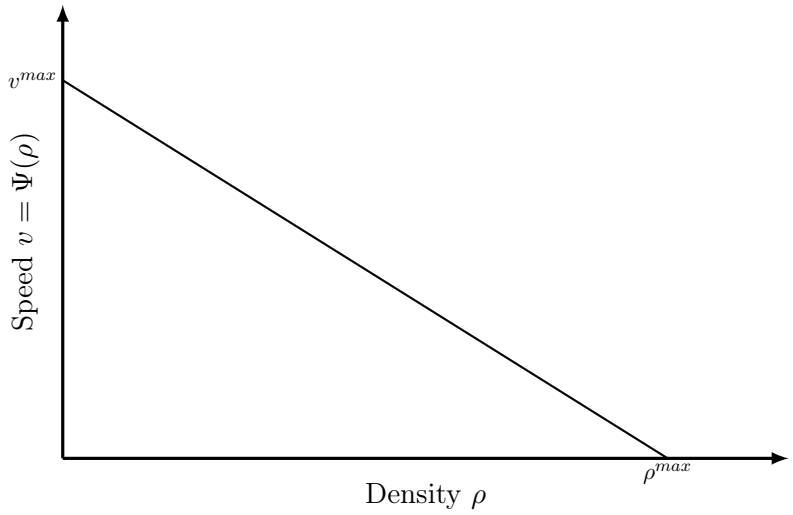


Figure 1.1: Greenshield's speed-density fundamental diagram.

where $\Phi(\rho)$ and $\Psi(\rho)$ are known as the flow-density and speed-density FD, respectively. This allows to rewrite the conservation law as an equation of only the density

$$\frac{\partial}{\partial t}\rho(x,t) + \frac{\partial}{\partial x}\Phi(\rho(x,t)) = 0. \tag{1.7}$$

Thanks to (1.5), Φ and Ψ can be obtained from one another. For the rest of this document, unless specified differently, the FD refers exclusively to the flow-density function Φ. The LWR can be used with any choice of FD, and since the first proposal of the Greenshield's FD, many other options have been proposed to better match traffic data [Tra11]. Nevertheless, from empirical observations, the FD should have the following properties [Car+12]:

- $\Phi(0) = 0$, and there is a value ρ^{max} such that $\Phi(\rho^{max}) = 0$. ρ^{max} is known as the jam density.
- Φ is convex[1].
- The free-flow speed $v^{max} = \dot{\Phi}(0)$ is a finite positive number, and is the maximum vehicle speed.
- The congested wave speed $w = \dot{\Phi}(\rho^{max})$ is a finite negative number.
- The FD has a unique maximum value φ^{max}, also known as the capacity.

The LWR is arguably the most commonly used model for traffic research applications. Its discretized version was first proposed in [Dag94], and is known as the Cell Transmission Model

[1]Some authors allow the FD to be non-convex near the jam density, but it is still required that Φ is unimodal.

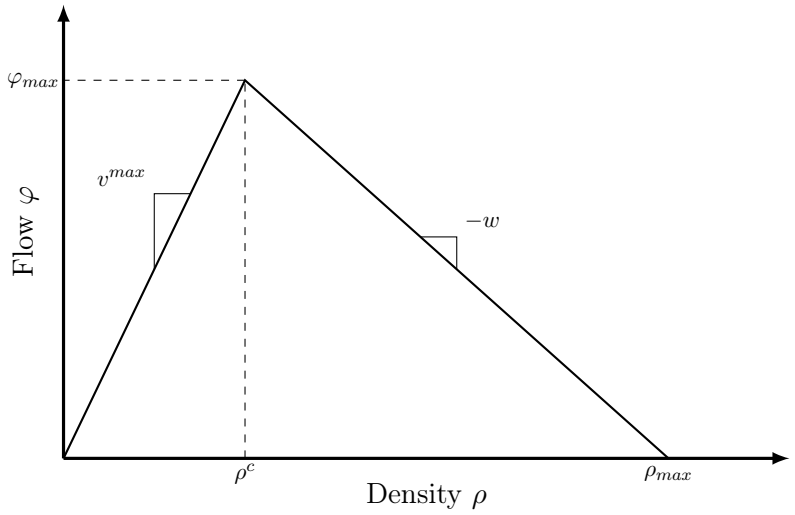


Figure 1.2: Triangular fundamental diagram.

(CTM). In this model, a road is divided into an arbitrary number of cells of equal length. The density of each cell is computed as

$$\rho_i[k+1] = \rho_i[k] + \frac{\Delta t}{\ell_i}(\varphi_{i-1}[k] - \varphi_i[k]) \tag{1.8}$$

where $\rho_i[k]$ is the density of cell i at time step k, $\varphi_i[k]$ is the flow between cells i and $i+1$, ℓ_i is the cell length, and Δt is the discretization time. For every cell and every time step, the CTM computes the Demand D_i, which is the maximum flow that exit from cell i, and the supply S_i, which is the maximum flow that can enter the cell. These values are obtained by using a particular FD with a triangular shape (see Fig. 1.2), such that

$$D_i = \min(v^{max}\rho, \varphi^{max}) \tag{1.9}$$

$$S_i = \min(w(\rho^{max} - \rho), \varphi^{max}) \tag{1.10}$$

The inter-cell flow is then computed as

$$\varphi_i[k] = \min(D_i[k], S_{i+1}[k]) \tag{1.11}$$

First-order models such as the LWR and the CTM are nevertheless unable to capture some real traffic phenomena such as stop-and-go wave and capacity-drop. This is because vehicle speed is instantaneously adjusted according to the FD by changes in the density. As a solution to these issues [AR00], second-order models are proposed as an extension. In addition to the conservation law, a second equation is added that models the smooth change in the vehicle velocity. A well known second order model is the Aw-Rascle-Zhang (ARZ) [AR00]; [Zha02],

$$\begin{aligned} \frac{\partial}{\partial t}\rho + \frac{\partial}{\partial x}(v\rho) &= 0 \\ \frac{\partial}{\partial t}(v + p(\rho)) + v\frac{\partial}{\partial x}(v + p(\rho)) &= 0 \end{aligned} \tag{1.12}$$

where $p(\rho)$ is an increasing function of ρ and is called the pressure.

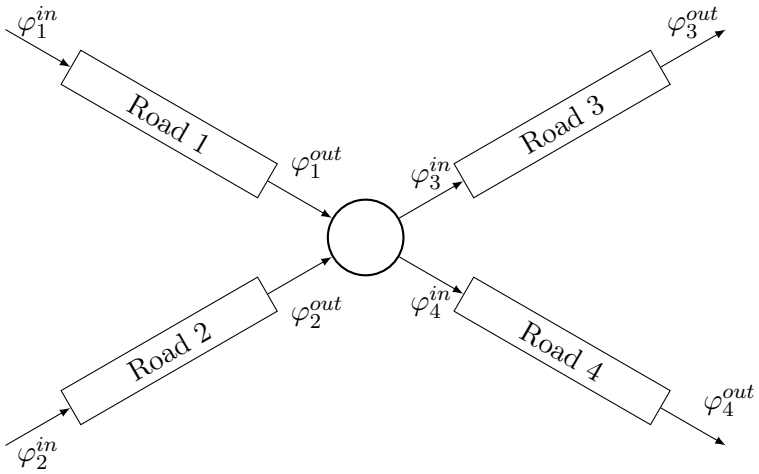


Figure 1.3: Flow exchange at an intersection.

1.1.2 Network models

Urban traffic networks consist of a collection of roads which are connected by intersections, as shown in Fig. 1.3. Each road section follows a dynamical model as described in Section 1.1.1. In addition to this, network models require to specify how flow behaves at each intersection. This is referred to as a junction model. To simplify the notation, define

$$\varphi_i^{in}(t) = \varphi_i(0, t) \tag{1.13}$$

and

$$\varphi_i^{out}(t) = \varphi_i(\ell_i, t) \tag{1.14}$$

as the inflow and outflow of the i-th road, respectively.

The first component of junction models is vehicle routing, which is the problem distributing the incoming flows to an intersection to the respective outgoing roads according to user preferences. One commonly used technique for this problem is the use of intersection parameters called turning ratios. A Turning Ratio (TR) $r_{i,j}$ is the fraction of the flow in road i, φ_i^{out} that turns to road j at an intersection.

Another common solution to this problem is Traffic Assignment (TA). Usually, these models take as input the Origin-Destination (OD) matrices, and then calculate the optimal routes from each origin to each destination, according to a specific cost function such as the traveling-time. Each OD flow is then divided proportionally into the calculated routes. Modern approaches periodically recalculate the optimal routes based on the current state of the network, which is known as the Dynamic Traffic Assignment (DTA).

The second component of junction models is the calculation of the road outflows and inflows given the densities of each road. One of the most commonly used junctions model is the flow-maximizing model, which was proposed in [Dag95] as an extension of the CTM to networks. This model assumes that drivers cooperate such that the total flow that can be

delivered from an intersection is maximized subject to the downstream road supply constraints. For the example in Fig. 1.3 this can written as

$$\begin{aligned} \max \quad & \varphi_1^{out} + \varphi_2^{out} + \varphi_3^{in} + \varphi_4^{in} \\ \text{subject to} \quad & \varphi_1^{out} \leq D_1(\rho_1) \\ & \varphi_2^{out} \leq D_2(\rho_2) \\ & \varphi_3^{in} \leq S_3(\rho_3) \\ & \varphi_4^{in} \leq S_4(\rho_4) \\ & \varphi_3^{in} = r_{1,3}\varphi_1^{out} + r_{2,3}\varphi_2^{out} \\ & \varphi_4^{in} = r_{1,4}\varphi_1^{out} + r_{2,4}\varphi_2^{out} \end{aligned} \tag{1.15}$$

The flow-maximizing model is very commonly used in the analysis of traffic networks. However, it has been found that this model generally overestimates the flows that would be produced in real scenarios, as it is unable to describe vehicle decelerations when approaching intersections [LHP17]. Other models have been proposed that incorporate other phenomena such as traffic light cycles [Jab16]. Nevertheless, realistic junction models remain an open problem for the treatment of traffic networks.

Region-based models

The described network models usually require the estimation of many parameters such as the TRs and OD matrices, and the resulting model can be very high-dimensional for large traffic networks as it must account for the density and flows of each road section. As an alternative to this issues, region-based models (also called reservoir models) consider the partition of the network into regions which are analyzed as whole entities, where the states now correspond to the average density and the average flow of the entire region. [GD08] provided experimental evidence of a Macroscopic Fundamental Diagram (MFD), which is a relationship between the average density of a region and its outgoing flow. Since then, the MFD has received increased attention and extended to include more complex scenarios. For instance, [AZG17] proposed an extension of the MFD to include heterogeneous traffic composition such as the mixture of cars and buses, and [ML19] analyzed how to use the MFD framework to handle congestion propagation due to internal and external trips. One of the main issues of the MFD is that changes in the average region density generate instantaneous changes in the flow production, which can in some cases imply the existence of vehicles with infinite speed. To solve this issue, MFD formulations that account for delays have been proposed in [PL20]. Region-based models have been used most commonly in the context of perimeter control [AG13], in the solution of the DTA problem in large networks [Hua+20], and in the estimation of network traveling times [Sae+20].

1.2 Input data

Broadly speaking, traffic data can be collected using stationary sensors and Floating Car Data (FCD).

1.2.1 Stationary sensors

This category includes technologies such as inductive-loop detectors, microwave and infrared radars, and traffic cameras [KMG06]. As their name indicates, they are located in a fixed position on a traffic road or intersection, and provide information of the vehicles that traverse such particular location.

Variables provided by stationary sensors include

- Number of vehicles per unit time
- Occupancy, i.e., percentage of time that the sensor detects a vehicle.
- Vehicle speed.
- Vehicle length or classification.

However, depending on the particular technology, some variables might not be available, or have different error tolerances. For example, single induction-loops can provide an accurate flow value, but cannot measure vehicle speeds directly. On the other hand, radars can measure the speed of vehicles directly with high precision, but are unable to detect stopped vehicles, and can present count omissions when two vehicles have the same speed. Traffic cameras can provide very accurate and detailed information about the traffic flow, but are usually very expensive to maintain, require sophisticated image processing techniques for large-scale applications, and are affected by environmental factors such as rain or fog [KMG06].

A limitation of most stationary sensors is that they only provide information about a specific location, and thus cannot easily be used to obtain network-level information such as traveling times, Origin-Destination (OD) demands, or vehicle route choice. Nevertheless, an exception to this limitation are the Automatic Vehicle-Identifier (AVI) sensors, such as license-plate readers. When a network of this technology is available, multiple detections at different places and times of the same vehicle is possible [GM18].

In addition to this traditional sensors, new traffic data sources have become available due to widespread use of WiFi and Bluetooth (BT) capable devices. Recent works as [BC13] and [AjSB13] have analyzed the use of these technologies as AVI sensors. This can be done by installing devices in certain location, which then constantly search for other capable devices and log the individual Media Access Control (MAC) address. Compared to the traditional license-plate readers, BT/WiFi AVI have the advantage of being cheaper, and can cover a wide area without requiring line-of-sight [BC13], but have the disadvantage of having an unknown

penetration rate. Nevertheless, this new technology can be shown to be a valuable addition to complement sensing networks [BQC15].

Network sensor location problem

Stationary sensors are traditionally the most common source of traffic data, specially for the case of inductive-loops [TK13]; [Seo+17]. Nevertheless, due to their installation and maintenance cost [KMG06], only a limited number of sensors can be installed in a traffic network. Because of the crucial role of this data, the number and location of such sensors must be selected optimally to minimize the budget, while providing the information required for a given application. This is known as the Network Sensor Location Problem (NSLP) [GM12].

In the literature, different approaches have been considered for the determination of the minimum number of sensors according to the degree of knowledge of the network and the available sensing technologies. [BCR01] defined the region of influence of flow sensors located at intersections, and with the knowledge of TRs in the network, the authors were able to provide necessary conditions for flow observability and some heuristic bounds on the number of sensors. [BCG06] performed a deeper analysis and determined that the sensor location problem is NP-complete.

Methods developed by [Cas+08] and [HPC09] formulated the problem by enumerating all possible paths of the network to construct a matrix of constraints to give bounds on the required number of sensors to estimate the flow everywhere. Furthermore, the authors analyzed the effect of different network topologies on these bounds. Additionally, if there is available information about unused paths, [Cas+13] and [Cas+14] provided extensions to reduce the number of paths that need to be counted. [HL14] and [Fu+16] used these methods with not only stationary flow sensors, but also AVI sensors.

A different alternative was presented by [Ng12], who considered the conservation of flow equations at intersections instead of path flows. This method does not require the enumeration of paths and is able to calculate the number of sensors as a function of the number of nodes and edges in a graph. Additionally, they proposed an algorithm for flow sensor location. This method was improved by [He13], who provided an efficient algorithm based on the topology of the network and graph theory. However, as this approach uses few assumptions about the network (only topology is used), the number of sensors may be large for real applications [Vit+14].

More robust approaches to this problem have also been studied. [He13], [Vit+14] and [RV17] proposed methods to assess and locate sensors for partial observability, i.e. only some of the flows of the network are estimated. Other works such as [Xu+16] and [LWK16] considered the problem of noise in the measurements, and proposed a trade-off between the number of sensors and the robustness of the flow estimation to noise. More complete reviews about models and methods used for flow estimation and sensor location can be found in [GM12] and [Cas+15].

1.2.2 Floating Car Data

FCD refers to information provided by a subset of vehicles in the network, which periodically communicate with a centralized station. The most common type of FCD corresponds to GPS coordinates, from which it is possible to reconstruct the trajectories of individual vehicles via map-matching [TK13]. In other cases, the named probe or connected vehicles, are equipped with additional sensors, and in addition to their GPS coordinates, provide data about their environment, such as the distance and velocities of surrounding vehicles [BLRP16]; [SKA15a]. In most applications however, FCD is aggregated such that only the average speed in certain road sections is available [Seo+17]. Nevertheless, FCD have received increasing attention, and many of current TSE works make use of a combination of stationary sensors and FCD [Seo+17].

1.3 Estimation approaches

Most of the TSE approaches proposed in the literature were developed initially for the case of one-dimensional roads, such as Highways and Freeways [FSS18]. A very common approach is the use of the Extended Kalman Filter (EKF). For instance, [TI07] proposes the use of the EKF coupled with the CTM to estimate the density of sections of highways, using data from stationary sensors located at certain sections of the road. This approach is based on linearizing the CTM around a current state, and then compare the measurements with the predictions of the linearized model. Later, [HB10] made use of the increase of availability of GPS data to complement the stationary sensor data. In this formulation, a Lagrangian model was used as input to the EKF. More recently, [SW14]; [SKA15b] show how information from connected vehicles can be provided to a KF approach to improve highway state estimation. Another common TSE technique is the use of Luenberger-like observers, as shown in [AI+04], which are based on switching between the free-flow and congested states, and then apply an estimate correction based on measurements. This technique has been successfully applied to real cases in [WOK12], and in a decentralized manner in [Viv+15] and more recently in [GCZ17]. In [Bra+17], the authors proposed a control scheme integrated with a Luenberger-like observer using the Asymmetric CTM, proposed in [Mun+03]. Other techniques such as Particle Filters (PF) have also been used as in [MBH07] which showed better results than KF based alternatives, and more recently in [KTK21] in conjunction with data from connected vehicles. PF techniques are shown to have an accurate estimation, albeit with an elevated computational cost. A more comprehensive review of TSE techniques can be found in [Seo+17].

More and more, the need to study urban traffic networks is increasing in the last years for TSE issues. Some methods have been developed such as the Unscented Kalman Filter (UKF) in [PN06], and Bayesian probabilistic model-based Expectation–Maximization Extended Kalman Filter (EM-EKF) in [GQZ17]. Nevertheless, in comparison to highways, urban networks still have not received as much attention, [Seo+17]. This is due in part to the complexity of modeling traffic dynamics with intersections, the related entry and exit flows, and the lack of sufficient sensor data. [He13] provides a method to estimate the equilibrium

flow for each road from sensor data using graph-based algorithms. In [LWK16], a method to estimate the dynamical evolution of the density of each road is proposed by using both stationary sensor measurements and FCD. This is done by first giving a flow estimate for each road as if the network is in stead-state, and then use the speed-density FD to provide density pseudo-measurements. This estimates are then the input to a Luenberger-like observer. In [Lad+18], FCD is used to determine if each road is in free-flow or congested regime, and then uses the flow-density FD and stationary sensor data to provide density estimates to each road in the traffic network. More recently, [Ros+20] proposes a data-based density estimation method that does not require the use of a FD. Instead, data from connected vehicles is used to estimate the outflow of each road, and then uses the density conservation law to estimate the traffic state.

1.4 Problematics and contributions

This works deals with the TSE problem for urban networks, using heterogeneous sources of data such as stationary flow sensors, FCD, and AVIs. We analyze three different problematics in this context.

First, we consider the problem of flow and density estimation for each road of an urban network under steady-state conditions. In this context, we propose a method to identify the minimum number and optimal locations of flow and TR sensors. Although the NSLP has received considerable attention as seen in Section 1.2.1, these works consider only the location of flow sensors, assuming either full knowledge of TRs, or complete lack thereof. By considering the different information provided by both sensor types, we identify the cost trade-off between them which can reduce the total number of sensors. The method only relies on the knowledge of the network topology, and does not require any information about route choices of O/D paths which may be hard to obtain. Furthermore, we provide an efficient algorithm that calculate the sensor locations with quasi-linear complexity[2], which is a contribution compared to other works in the literature and can be applied to large networks. Furthermore, we provide a method to use the calculated sensor locations with additional FCD to estimate the flow and density for every road in the network. These results are presented in Chapter2.

The second problem considers the dynamic evolution of the density and flow of every road in the urban network. Our main contribution is the proposal of a data-based TSE method for general urban networks that does not require the use of a FD. This method considers also heterogeneous data sources from flow sensors, TR sensors, and FCD. For the dynamic case, TR information is needed for every intersection. However, this requirement is relaxed by assuming a priori estimates of these parameters. To reduce the error introduced by this assumption, we propose a method to identify the optimal location of a small number of TR sensors by measuring the error sensitivity of each intersection to errors in the TR values. These results are presented in Chapter 4.

[2]Quasi-linear complexity refers to a number of computations of the order $n \log(n)$, where n is the number of nodes.

For the third problem, Chapter 4 considers the estimation of the average density of an area of a traffic network, without the need to know the density of each individual road. This allows the use of a low-dimensional observer, greatly simplifying the computational cost. This information can then be used as an input to the MFD modeling, in perimeter control applications. Other average-density estimation approaches such as the ones in [AG13] and [AG13] use limited density measurements in few roads as a proxy for the entire region. We find that this methods have questionable precision, which incurs in errors in the MFD inputs. Recent advances in network theory focus on the development of low-dimensional observers for linear combinations of the system's state, as in [SII17] and [NWK19]. We propose to explore the applicability of such theory to traffic networks, when sensors are only present at the boundaries. Due to the complexity of modeling traffic dynamics with intersections, we assume that traffic networks follow linear dynamics and focus at first on free-flow regime. This work remains important for region-based estimators where free-flow conditions are satisfied (interconnected residential neighborhoods or regions of cities with low/moderate traffic for examples). Moreover, if the region does not satisfy the linear dynamics condition, the proposed estimator can still provide bounds on the average density. Even though the underlying methods assume free-flow, simulations with congested regimes are carried out to model the resulting error, and to provide working ranges where the estimator is applicable under mixed regimes. Finally, such proposed theory can prepare for future work focusing on the same problem but under high congestion.

Finally, Chapter 5 presents a study case applied to the downtown of the city of Grenoble in France. This uses real data from traffic sensors located at the boundaries of a selected region of the city, whose locations where decided using the methods developed in the previous chapters. This data is then used to evaluate the performance of the proposed estimation approaches.

1.5 Publications

1.5.1 Journals

- Martin Rodriguez-Vega, Carlos Canudas-de-Wit and Hassen Fourati. "Location of turning ratio and flow sensors for flow reconstruction in large traffic networks". *Transportation Research Part B: Methodological*, 121 (March 2019) pp. 21-40.

1.5.2 Working papers

- Martin Rodriguez-Vega, Carlos Canudas-de-Wit and Hassen Fourati. "Dynamic density and flow reconstruction in large-scale urban networks using heterogeneous data sources". Under review in *Transportation Research Part C: Emerging technologies*. (2021).

- Martin Rodriguez-Vega, Carlos Canudas-de-Wit and Hassen Fourati. "Average density estimation for urban traffic networks: application to the Grenoble network". Under review (3rd round) in *Transportation Research Part B: Methodological*. (2021).

1.5.3 International conferences

- Martin Rodriguez-Vega, Carlos Canudas-de-Wit and Hassen Fourati. "Flow and density estimation in Grenoble using real data". *7th International conference on Time Series and Forecasting (ITISE)*, Meloneras, Spain, July 2021.
- Martin Rodriguez-Vega, Carlos Canudas-de-Wit and Hassen Fourati. "Urban network traffic state estimation using a data-based approach". *16th IFAC Symposium on Control in Transportation Systems (CTS)*, Lille, France, June 2021.
- Martin Rodriguez-Vega, Carlos Canudas-de-Wit and Hassen Fourati. "Average density detectability in traffic networks using virtual road divisions". *IFAC World Congress 2020*, Berlin, Germany, July 2020, pp. 17300-17305.
- Martin Rodriguez-Vega, Carlos Canudas-de-Wit and Hassen Fourati. "Joint flow and density reconstruction in large traffic networks using partial turning ratio information". *57th IEEE Conference on Decision and Control (CDC)*, Miami, USA, December 2018, pp. 205-210.
- Andres Ladino, Carlos Canudas-de-Wit, Alain Kibangou, Hassen Fourati and Martin Rodriguez-Vega. "Density and flow reconstruction in urban traffic networks using heterogeneous data sources". *European Control Conference (ECC)*, Limassol, Cyprus, June 2018, pp. 1679-1684.

Chapter 2

Density and flow estimation in steady-state

Contents

2.1 Overview

This chapter treats the problem of flow and density estimation in urban networks for the case of steady-state regime. This is done in two stages. In the first stage, we examine the problem of minimizing the number of sensors needed to completely recover the vehicular flow. We consider two possible sensor technologies: one that allows the measurement of turning ratios (TRs) at a given intersection and the other that directly measures the flow in a road. We propose a method that finds the location of the minimum number of such sensors in quasi-linear time. The second stage consists of using the measurements from the sensors located in the first stage in addition with FCD to calculate estimates for the flow and density of each road of the network. The proposed methods are evaluated via microscopic vehicle simulations, using the real network of the city of Grenoble.

2.2 Model

We represent traffic networks by means of a directed graph. The nodes of the graph are partitioned in two disjoint sets $\mathcal{C}$ and $\mathcal{N}$: $\mathcal{C}$ corresponds to source and sink nodes of the network and $\mathcal{N} = \{1, 2, \ldots, n_{\mathcal{N}}\}$ represents intersections which are not able to generate or store vehicular flow. The edges $\mathcal{E} = \{1, 2, \ldots, n_{\mathcal{E}}\}$ represent the set of roads of the network. Denote $\mathcal{I}(k)$ as the set of incoming edges to some node k and $\mathcal{O}(k)$ as the set of outgoing edges from k. We define a partition of $\mathcal{E}$ in three sets: $\mathcal{E}_{\text{in}} = \bigcup_{k\in\mathcal{C}} \mathcal{O}(k)$ are the boundary incoming roads, $\mathcal{E}_{\text{out}} = \bigcup_{k\in\mathcal{C}} \mathcal{I}(k)$ are the boundary outgoing roads, and $\mathcal{E}_{\text{net}} = \mathcal{E} \setminus (\mathcal{E}_{\text{in}} \cup \mathcal{E}_{\text{out}})$ are the internal roads of the network.

Definition 2.1. A feasible traffic network is a directed graph $\{\mathcal{C}\cup\mathcal{N}, \mathcal{E}\}$ such that the following conditions are met:

- Every edge is part of a path that starts with an edge from $\mathcal{E}_{\text{in}}$ and ends with an edge from $\mathcal{E}_{\text{out}}$.
- The graph contains no self loops.
- $\forall k \in \mathcal{N}$ we have $\mathcal{I}(k) \cap \mathcal{E}_{\text{out}} = \varnothing$ and $\mathcal{O}(k) \cap \mathcal{E}_{\text{in}} = \varnothing$.
- There is no production or storage of vehicles in the nodes in $\mathcal{N}$.

2.2.1 Single road model

The traffic state of the network refers to the values of the density, flow and velocity which are defined for each road. Denote by ρ_i the density of road i. From the conservation law, we have

$$\frac{\mathrm{d}}{\mathrm{d}t}\rho_i(t) = \frac{1}{\ell_i}(\varphi^{\text{in}}(t) - \varphi^{\text{out}}(t)) \tag{2.1}$$

where φ^{in} is the incoming (or upstream) flow and φ^{out} is the outgoing (or downstream) flow. This chapter is limited to the case when the entire network is in steady-state. This is formally true only when external boundary flows are constant and the network dynamics reach equilibrium, or it is approximately true when the external boundary flows slowly evolve in time. This conditions establishes that

$$\rho_i(t) = \rho_i \quad \varphi_i^{\text{in}}(t) = \varphi_i^{\text{in}} \quad \varphi_i^{\text{out}}(t) = \varphi_i^{\text{out}} \quad \forall i \in \mathcal{E} \quad \forall t \tag{2.2}$$

This in turn implies,

$$0 = \varphi_i^{\text{in}} - \varphi_i^{\text{out}} \tag{2.3}$$

so each road $i \in \mathcal{E}$ is characterized by a unique vehicular flow

$$\varphi_i = \varphi^{\text{in}} = \varphi^{\text{out}} \tag{2.4}$$

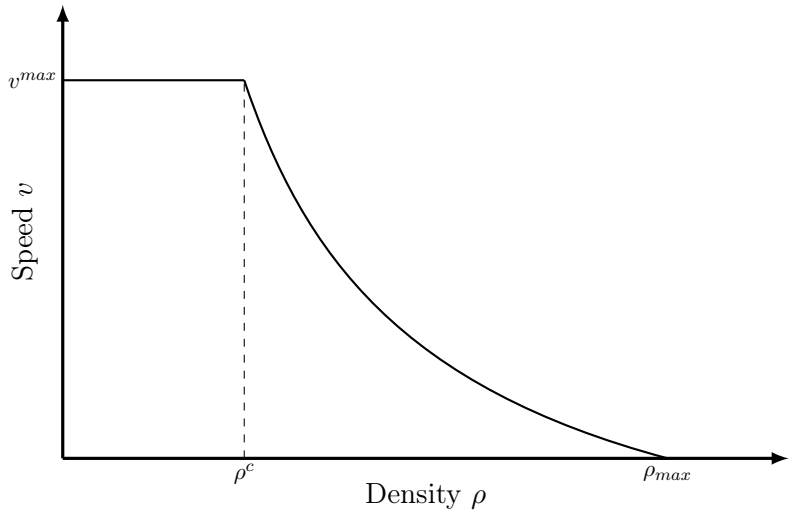


Figure 2.1: Speed-density relation from the triangular fundamental diagram.

Furthermore, we adopt the triangular fundamental diagram to model the relationship between density and flow in equilibrium,

$$\varphi_i = \Phi(\rho_i) = \begin{cases} v_i^{max}\rho_i & \text{if} \quad 0 \leq \rho_i \leq \rho_i^c \\ \\ w(\rho^{max} - \rho_i) & \text{if} \quad \rho^c \leq \rho_i \leq \rho_i^{max} \end{cases} \tag{2.5}$$

as shown in Fig. 1.2. This FD has as parameters the free-flow speed v_i^{max}, the congestion-wave speed w_i, the jam density ρ^{max}, the critical density ρ_i^c, and the maximum road capacity φ_i^{max}. Note that these parameters are related to each other, and knowledge of just three is sufficient to calculate the rest. Taking the relationship $\varphi_i = v_i\rho_i$ where v_i is the space-mean speed of road i, the speed-density fundamental diagram is given by

$$v_i = \Psi(\rho_i) = \begin{cases} v_i^{max} & \text{if} \quad 0 \leq \rho_i \leq \rho_i^c \\ \\ w_i \left(\dfrac{\rho_i^{max}}{\rho_i} - 1 \right) & \text{if} \quad \rho_i^c \leq \rho_i \leq \rho_i^{max} \end{cases} \tag{2.6}$$

If a road has a density higher than its critical value, it is said to be in congested regime. Otherwise, the road is said to be in free-flow.

2.2.2 Intersection model

At the network level, the conservation law imposes constraints in the flow distribution of the incoming and outgoing roads in each intersection. Consider an internal intersection $k \in \mathcal{N}$, such that it is neither a source or sink of traffic flow. Therefore, flow conservation requires that the total incoming flow must be equal to the total outgoing one,

$$\sum_{j \in \mathcal{O}(k)} \varphi_j - \sum_{i \in \mathcal{I}(k)} \varphi_i = 0. \tag{2.7}$$

However, suppose that for an intersection $k \in \mathcal{N}$ information is available about the turning ratios. A Turning Ratio (TR) $r_{i,j}$ is defined as the proportion of traffic flow in road $i \in \mathcal{E}$ that turns to road $j \in \mathcal{E}$ at an intersection $k \in \mathcal{N}$ such that $i \in \mathcal{I}(k)$ and $j \in \mathcal{O}(k)$. With this information, the flow in each outgoing road from an intersection can be written as a linear combination of the flows in the incoming roads,

$$\varphi_j - \sum_{i \in \mathcal{I}(k)} r_{i,j}\varphi_i = 0 \quad , \quad \forall j \in \mathcal{O}(k). \tag{2.8}$$

Note that joining (2.7) and (2.8) implies $\sum_j r_{i,j} = 1$.

2.2.3 Measurements and linear constraints

This chapter considers that two data sources are available: flow sensors in single roads and TR sensors in intersections. The presence of these sensors in particular locations provides different levels of information, which are represented as constraints in the flow distribution.

It can be seen that (2.8) can only be used when the TRs are known. Let $\mathcal{R} \subseteq \mathcal{N}$ be the set of intersections equipped with TR sensors. For each of the outgoing edges of the intersections in $\mathcal{R}$, an equation following (2.8) can be written. Define $A(\mathcal{R})$ as the matrix that collects the resulting equations for all intersections in $\mathcal{R}$:

$$A(\mathcal{R})_{i,j} = \begin{cases} 1 & \text{if} \quad e_i = j \\ -r_{j,e_i} & \text{else} \end{cases} , \tag{2.9}$$

where e_i is the i-th element of the ordered set $\bigcup_{k \in \mathcal{R}} \mathcal{O}(k)$. As an edge can only belong to one source node, it is clear that the sets $\mathcal{O}(k)$ for all $k \in \mathcal{N}$ are mutually disjoint. Therefore, the number of rows of $A(\mathcal{R})$ is $|\bigcup_{k \in \mathcal{R}} \mathcal{O}(k)| = \sum_{k \in \mathcal{R}} \deg^{\text{out}}(k)$, where $\deg^{\text{out}}(k)$ is the out-degree (number of outgoing edges) of node k.

Let $\mathcal{U} = \mathcal{N} \setminus \mathcal{R}$ be the set of unmeasured intersections. Following a similar discussion to $\mathcal{R}$, each of the intersections in $\mathcal{U}$ has assigned an equation similar to (2.7). Define $B(\mathcal{U})$ as the matrix that collects the resulting equations for each $k \in \mathcal{U}$:

$$B(\mathcal{U})_{i,j} = \begin{cases} 1 & \text{if} \quad j \in \mathcal{O}(k_i) \\ -1 & \text{if} \quad j \in \mathcal{I}(k_i) \\ 0 & \text{else} \end{cases} , \tag{2.10}$$

where k_i is the i-th element of the ordered set $\mathcal{U}$. Check that $B(\mathcal{U})$ has size $(n_{\mathcal{N}} - n_{\mathcal{R}}) \times n_{\mathcal{E}}$ with $n_{\mathcal{R}} = |\mathcal{R}|$.

To represent the location of flow sensors, let $\mathcal{S} \subseteq \mathcal{E}$ be the set of roads with cardinality $|\mathcal{S}| = n_s$ equipped with flow sensors. We define $C(\mathcal{S}) \in \{0,1\}^{n_s \times n_{\mathcal{E}}}$ such that $C(\mathcal{S})_{i,j} = 1$ if the i-th sensor is located in the j-th road. Note that we can write $C(\mathcal{S}) = [\mathbf{u}_{e_1}\ \mathbf{u}_{e_2}\ \cdots\ \mathbf{u}_{e_{n_{\mathcal{S}}}}]^T$, where e_i is the i-th element of the set $\mathcal{S}$, and $\mathbf{u}_i$ is the standard basis vector in the i-th coordinate . We assume that the elements of $\mathcal{S}$ are distinct, and so rank $C(\mathcal{S}) = n_{\mathcal{S}}$.

Using these matrices, we can write the linear constraints as

$$\begin{bmatrix} L(\mathcal{R}) \\ C(\mathcal{S}) \end{bmatrix} \boldsymbol{\varphi} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\varphi}_m \end{bmatrix}, \tag{2.11}$$

where $L(\mathcal{R}) = [A(\mathcal{R})^T \; B(\mathcal{U})^T]^T$ and $\boldsymbol{\varphi}_m$ are the measurements given by the flow sensors.

Example 2.1. Consider the traffic network shown in Figure 2.2. Denote $\mathcal{C} = \{v^+, v^-\}$ where

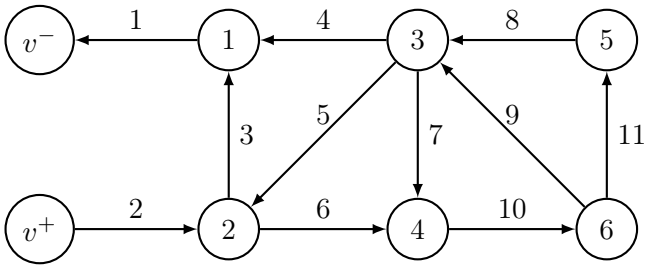


Figure 2.2: Example traffic network.

v^+ is a source node and v^- is a sink node. Additionally, the intersections are denoted by $\mathcal{N} = \{1, 2, \ldots, 6\}$ and the roads by $\mathcal{E} = \{1, 2, \ldots, 11\}$. Assume that TR sensors are located at intersection $\mathcal{R} = \{2, 3\}$, and thus $\mathcal{U} = \{1, 4, 5, 6\}$. Applying the previous description, the linear constraint matrices are

$$A(\mathcal{R}) = \begin{array}{c} \\ \varphi_3 \\ \varphi_6 \\ \varphi_4 \\ \varphi_5 \\ \varphi_7 \end{array} \begin{array}{c} \begin{array}{ccccccccccc} \varphi_1 & \varphi_2 & \varphi_3 & \varphi_4 & \varphi_5 & \varphi_6 & \varphi_7 & \varphi_8 & \varphi_9 & \varphi_{10} & \varphi_{11} \end{array} \\ \begin{bmatrix} 0 & -r_{23} & 1 & 0 & -r_{53} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -r_{26} & 0 & 0 & -r_{56} & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -r_{84} & -r_{94} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -r_{85} & -r_{95} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -r_{87} & -r_{97} & 0 & 0 \end{bmatrix} \end{array},$$

$$B(\mathcal{U}) = \begin{array}{c} \\ 1 \\ 4 \\ 5 \\ 6 \end{array} \begin{array}{c} \begin{array}{ccccccccccc} \varphi_1 & \varphi_2 & \varphi_3 & \varphi_4 & \varphi_5 & \varphi_6 & \varphi_7 & \varphi_8 & \varphi_9 & \varphi_{10} & \varphi_{11} \end{array} \\ \begin{bmatrix} 1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 \end{bmatrix} \end{array}.$$

The first two rows of $A(\mathcal{R})$ correspond to the outgoing edges of intersection 2 (roads 3 and 6), and the last three rows are related to the outgoing edges of intersection 3 (roads 4, 5 and

7). Similarly, each of the rows of $B(\mathcal{U})$ are associated with intersections 1, 4, 5 and 6. Now consider that sensors are located at edges $\mathcal{S} = \{1, 9\}$. The resulting sensing matrix is

$$C(\mathcal{S}) = \begin{array}{c} \begin{array}{ccccccccccc} \varphi_1 & \varphi_2 & \varphi_3 & \varphi_4 & \varphi_5 & \varphi_6 & \varphi_7 & \varphi_8 & \varphi_9 & \varphi_{10} & \varphi_{11} \end{array} \\ \left[\begin{array}{ccccccccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{array}\right] \end{array}.$$

Note that the first row of $C(\mathcal{S})$ is $\mathbf{u}_1^T$ and the second one is $\mathbf{u}_9^T$.

As a specific case, let the TRs be such that the input flows 2, 5, 8 and 9 are distributed evenly to their corresponding destinations. It is straightforward to check that

$$\operatorname{rank}\left(\left[\begin{array}{c} A(\mathcal{R}) \\ B(\mathcal{U}) \\ C(\mathcal{S}) \end{array}\right]\right) = 11.$$

Therefore, this matrix is invertible and flow estimation can be done. □

It is clear that the system (2.11) has a solution when the matrix in the left hand side has rank equal to the number of columns (the number of roads in the network $n_{\mathcal{E}}$). Therefore, the condition for full road flow observability is

$$\operatorname{rank}\left(\left[\begin{array}{c} L(\mathcal{R}) \\ C(\mathcal{S}) \end{array}\right]\right) = n_{\mathcal{E}}. \tag{2.12}$$

2.3 Number and location of flow and TR sensors

Our objective is to find the minimum number of flow and TR sensors and their positions. For the moment, we will consider the scenario where the number of TR sensors $|\mathcal{R}| = n_{\mathcal{R}}$ is given and the number of flow sensors $|\mathcal{S}|$ is minimized. Later, in Section 2.3.3, we will expand the results to the general case where both $|\mathcal{R}|$ and $|\mathcal{S}|$ are the decision variables.

The problem formulation is to find the optimal sets $\mathcal{R}^*$ and $\mathcal{S}^*$ that solve

$$\begin{aligned} \mathcal{R}^*, \mathcal{S}^* \quad = \quad & \underset{\mathcal{R},\mathcal{S}}{\operatorname{argmin}} \quad |\mathcal{S}| \\ & \text{subject to} \quad \operatorname{rank}\left(\left[\begin{array}{c} L(\mathcal{R}) \\ C(\mathcal{S}) \end{array}\right]\right) = n_{\mathcal{E}} \\ & \qquad\qquad |\mathcal{R}| = n_{\mathcal{R}} \end{aligned} . \tag{2.13}$$

Proposition 2.1. *For any $n_{\mathcal{R}} \in \{0, 1, \ldots, n_{\mathcal{N}}\}$, problem (2.13) has at least one feasible solution.*

Proof. Consider $\mathcal{S} = \mathcal{E}$ for which $C(\mathcal{E}) = \mathbb{I}_{n_\mathcal{E}}$. It is straightforward that rank $\left(\begin{bmatrix} L(\mathcal{R}) \\ C(\mathcal{E}) \end{bmatrix} \right) = n_\mathcal{E}$ for matrix $L(\mathcal{R})$. This selection of sets $\mathcal{R}, \mathcal{S}$ satisfies the constraints, hence the feasible set is non empty. □

It can be seen that this problem is combinatorial. Using a brute force approach, i.e. calculate the cost for every possible combination and selecting the optimal one, would require $O((n_\mathcal{N}/n_\mathcal{R})^{n_\mathcal{R}}\, 2^{n_\mathcal{E}})$ operations, which is expensive to solve. Because of this, we consider a partition of this problem into two problems with a lower degree of complexity. The first consists in independently calculating the optimal set of TR sensors as

$$\begin{array}{rcll} \mathcal{R}^* & = & \underset{\mathcal{R}}{\text{argmax}} & \text{rank } L(\mathcal{R}) \\ & & \text{subject to} & |\mathcal{R}| = n_\mathcal{R} \end{array} . \tag{2.14}$$

Once we find the set $\mathcal{R}^*$, the second problem consists in finding any set $\mathcal{S}^*$ that satisfies the following conditions:

$$\begin{array}{ll} \text{find any} & \mathcal{S}^* \subseteq \mathcal{E} \\ \text{such that} & \text{rank } C(\mathcal{S}^*) = n_\mathcal{E} - \text{rank } L(\mathcal{R}^*) \\ & \text{rank} \left(\begin{bmatrix} L(\mathcal{R}^*) \\ C(\mathcal{S}^*) \end{bmatrix} \right) = \text{rank } C(\mathcal{S}^*) + \text{rank } L(\mathcal{R}^*) \end{array} . \tag{2.15}$$

Next, we show that solving problems (2.14) and (2.15) is equivalent to solving problem (2.13).

Lemma 2.1. *For any given set $\mathcal{R}^* \subseteq \mathcal{N}$, problem* (2.15) *has at least one solution.*

Proof. From the definition of $L(\mathcal{R}^*)$, it can be seen that

$$0 < \text{rank } L(\mathcal{R}^*) \leq \sum_{k \in \mathcal{N}} \deg^{\text{out}}(k) < n_\mathcal{E}.$$

Consider $\mathcal{S} = \mathcal{E}$, then $C(\mathcal{E}) = \mathbb{I}$, and rank $C(\mathcal{E}) = n_\mathcal{E}$. It follows that rank $C(\mathcal{E}) +$ rank $L(\mathcal{R}^*) > n_\mathcal{E}$. However, rank $\left(\begin{bmatrix} L(\mathcal{R}^*) \\ C(\mathcal{E}) \end{bmatrix} \right) = n_\mathcal{E}$, and some rows of $C(\mathcal{E})$ are linear combinations of the other rows. We can find a solution with the following iterative process:

1. Initialize $\mathcal{S} = \mathcal{E}$.
2. While rank $L(\mathcal{R}^*) +$ rank $C(\mathcal{S}) > n_\mathcal{E}$
   - 2.1 Find $s \in \mathcal{S}$ such that the corresponding row of $C(\mathcal{S})$ is a linear combination of the other rows of the matrix.
   - 2.2 Assign $\mathcal{S} \leftarrow \mathcal{S} \setminus \{s\}$

3. Define $\mathcal{S}^* \leftarrow \mathcal{S}$

As the process only removes redundant rows, the rank of the augmented matrix is still equal to $n_{\mathcal{E}}$. Additionally, the algorithm only ends when rank $L(\mathcal{R}^*)$ + rank $C(\mathcal{S}) = n_{\mathcal{E}}$. Therefore, for a given $\mathcal{R}^*$, it is always possible to find $\mathcal{S}^*$ which satisfies the constraints. □

Lemma 2.2. *Let $\mathcal{R}^*, \mathcal{S}^*$ be an optimal solution to problem* (2.13)*. Then, the rows of matrices $L(\mathcal{R}^*)$ and $C(\mathcal{S}^*)$ are linearly independent.*

Proof. We proceed by contradiction. Assume that the rows of $C(\mathcal{S}^*)$ are not linearly independent to the rows of $L(\mathcal{R}^*)$. There exists at least one row of $C(\mathcal{S}^*)$, $\mathbf{u}_s^T$ such that $s \in \mathcal{S}^*$, which is a linear combination of the rows of $\begin{bmatrix} L(\mathcal{R}^*) \\ C(\mathcal{S}^*) \end{bmatrix}$.

Define a new set $\mathcal{S}' = \mathcal{S}^* \setminus \{s\}$. Matrix $C(\mathcal{S}')$ is the same as $C(\mathcal{S}^*)$ but with row $\mathbf{u}_s^T$ removed. It is evident that rank $\left(\begin{bmatrix} L(\mathcal{R}^*) \\ C(\mathcal{S}') \end{bmatrix} \right) = n_{\mathcal{E}}$ and the constraints are still satisfied. Additionally, $|\mathcal{S}'| = n_{\mathcal{S}}^* - 1 < |\mathcal{S}^*|$, making $\mathcal{S}^*$ non optimal, which is a contradiction. □

Theorem 2.1
$\mathcal{R}^*, \mathcal{S}^*$ is a solution to problem (2.13) *if and only if $\mathcal{R}^*$ is a solution to problem* (2.14) *and $\mathcal{S}^*$ is a solution to problem* (2.15)*.*

Proof. Recall, by construction, rank $C(\mathcal{S}) = |\mathcal{S}|$ for any $\mathcal{S} \subseteq \mathcal{E}$.

First, we prove implication. Assume that $\mathcal{R}^*, \mathcal{S}^*$ is a solution to problem (2.13). From Lemma 2.2 and the constraints of problem (2.13), it is evident that $\mathcal{S}^*$ is a solution to problem (2.15). By way of contradiction, assume there exists $\mathcal{R}'$, $|\mathcal{R}'| = n_{\mathcal{R}}$ such that rank $L(\mathcal{R}')$ > rank $L(\mathcal{R}^*)$. Because of Lemma 2.1, we can find $\mathcal{S}'$ satisfying problem (2.15). It is easy to check that the pair $\mathcal{R}', \mathcal{S}'$ lie in the feasible region of problem (2.13), and that $|\mathcal{S}'| < |\mathcal{S}^*|$. This implies that $\mathcal{S}^*$ is not an optimal solution. This is a contradiction, then this $\mathcal{R}'$ cannot exist and therefore, rank $L(\mathcal{R}^*) \geq$ rank $L(\mathcal{R})$ for any $\mathcal{R}$, $|\mathcal{R}| = n_{\mathcal{R}}$, therefore $\mathcal{R}^*$ is a solution to (2.14).

Now we proceed to necessity. Assume that $\mathcal{R}^*$ is a solution to problem (2.14) and $\mathcal{S}^*$ is a solution to problem (2.15), then $|\mathcal{S}^*| = n_{\mathcal{E}} -$ rank $L(\mathcal{R}^*)$. Consider that another pair $\mathcal{R}', \mathcal{S}'$ is an optimal solution to problem (2.13), therefore $|\mathcal{S}'| \leq |\mathcal{S}^*|$. By construction, it must be that rank $L(\mathcal{R}^*) \geq$ rank $L(\mathcal{R}')$, and using Lemma 2.2, rank $L(\mathcal{R}^*) \geq n_{\mathcal{E}} - |\mathcal{S}'|$. This implies that $|\mathcal{S}^*| \leq |\mathcal{S}'|$, and thus, $|\mathcal{S}^*| = |\mathcal{S}'|$, and the pair $\mathcal{R}^*, \mathcal{S}^*$ is also an optimal solution to problem (2.13). □

2.3.1 Optimal location of turning ratio sensors

Theorem 2.2 (later in this section) allows to write the rank of $L(\mathcal{R})$ for any $\mathcal{R}$ as a function of the number of intersections in the network and their out-degrees, i.e.

$$\text{rank } L(\mathcal{R}) = n_{\mathcal{N}} - n_{\mathcal{R}} + \sum_{k \in \mathcal{R}} \deg^{\text{out}}(k). \tag{2.16}$$

Using (2.16), problem (2.14) can be rewritten as

$$\begin{array}{rcll} \mathcal{R}^* & = & \underset{\mathcal{R}}{\text{argmax}} & \sum_{k \in \mathcal{R}} \deg^{\text{out}}(k) \\ & & \text{subject to} & |\mathcal{R}| = n_{\mathcal{R}} \end{array}. \tag{2.17}$$

To solve this problem, the following algorithm is proposed:

Algorithm 2.1. *Location of TR sensors*

Inputs: Directed graph $\{\mathcal{C} \cup \mathcal{N}, \mathcal{E}\}$ *and number of TR sensors* $n_{\mathcal{R}}$.

Output: Set of intersections $\mathcal{R}^*$.

1. *Calculate the vector of out-degrees:* $\mathbf{d}(k) \leftarrow deg^{out}(k)$, $\forall k \in \mathcal{N}$.
2. *Sort* $\mathbf{d}$ *from highest to lowest and return the sorting vector* $\boldsymbol{\lambda}$*, i.e.* $\boldsymbol{\lambda}(1)$ *is the index of the highest element of* $\mathbf{d}$*,* $\boldsymbol{\lambda}(2)$ *is the index of the second highest, and so on.*
3. *Construct* $\mathcal{R}^* \leftarrow \{\boldsymbol{\lambda}(1), \boldsymbol{\lambda}(2), \ldots, \boldsymbol{\lambda}(n_{\mathcal{R}})\}$. □

Remark: Several solutions are possible depending on the multiplicity of the out-degrees of the nodes of the network, however, all these solutions are optimal.

Example 2.2. Consider the traffic network shown in Figure 2.2. Furthermore, assume that $n_{\mathcal{R}} = 2$. By applying Algorithm 2.1 to these inputs we obtain:

1. The vector of out-degrees is $\mathbf{d} = [1 \quad 2 \quad 3 \quad 1 \quad 1 \quad 2]^T$.
2. After sorting $\mathbf{d}$, the resulting sorting vector is $\boldsymbol{\lambda} = [3 \quad 2 \quad 6 \quad 1 \quad 4 \quad 5]^T$.
3. The output is found by selecting the first two elements of $\boldsymbol{\lambda}$: $\mathcal{R}^* = \{2, 3\}$.

Note however that there are several nodes with the same out-degree. Because of this, the set $\{3, 6\}$ would also be a solution to problem (2.17). □

The following propositions show that problem (2.14) can be converted into problem (2.17).

Lemma 2.3. *For any feasible traffic network* $\{\mathcal{C} \cup \mathcal{N}, \mathcal{E}\}$ *(see Definition 2.1) and any set of intersections* $\mathcal{R} \subseteq \mathcal{N}$*,* $A(\mathcal{R})$ *is full row rank.*

Proof. Without loss of generality assume that the ordering of the elements of $\mathcal{E}$ is such that the smaller indexes correspond to $\mathcal{E}_{\text{in}}$, followed by $\mathcal{E}_{\text{net}}$ and ending with $\mathcal{E}_{\text{out}}$. Denote $|\mathcal{E}_{\text{in}}| = n_{in}$ and $|\mathcal{E}_{\text{out}}| = n_{out}$.

For now, let $\mathcal{R} = \mathcal{N}$. Using the proposed indexing, we can write

$$A(\mathcal{N})_{i,j} = \begin{cases} 1 & \text{if} \quad i = j + n_{in} \\ -r_{j,i+n_{in}} & \text{else} \end{cases},$$

for $i = 1, 2, \ldots, n_{\mathcal{E}} - n_{in}$ and $j = 1, 2, \ldots, n_{\mathcal{E}}$.

We can split $A(\mathcal{N}) = [X \quad Y]$ where Y is a square matrix of size $n_{\mathcal{E}} - n_{in}$ and can be written as $Y = \mathbb{I} - R^T$ with $R_{i,j} = r_{i+n_{in},j+n_{in}}$ for $i, j \in \{1, 2, \ldots, n_{\mathcal{E}} - n_{in}\}$. Matrix R has the following properties:

1. The diagonal entries of R are zero as $r_{i,i} = 0$.
2. All roads in $\mathcal{E}_{\text{out}}$ have no downstream neighbors, then $r_{i,j} = 0$, $\forall i \in \mathcal{E}_{\text{out}}$. Thus, the last n_{out} rows of R are zero. Nevertheless, by flow conservation, all the other rows of R sum to 1.
3. For every row i of R there is a sequence of nonzero elements of R of the form $R_{i,i_1}, R_{i_1,i_2}, \ldots, R_{i_q,j}$ such that $j \in \mathcal{E}_{\text{out}}$.

Properties 1 and 2 imply that Y^T is weakly diagonally dominant matrix, i.e. $Y_{i,i} \geq |\sum_{j \neq i} Y_{i,j}|$, where the strict inequality is true only for the last n_{out} rows. Nevertheless, property 3 states that for every row i, there is a sequence of nonzero elements that connect row i to one of the last n_{out} rows. Because of this, Y^T is a special type of matrix called *weakly chained diagonally dominant*, which is known to be non-singular [SC74]. Thus all rows of $A(\mathcal{N})$ form a linearly independent set.

For any arbitrary $\mathcal{R} \subseteq \mathcal{N}$, $A(\mathcal{R})$ is just a reduced version of $A(\mathcal{N})$ with some of its rows removed. As the rows of $A(\mathcal{N})$ are linearly independent, it is straightforward that $A(\mathcal{R})$ is full row rank as well. □

Lemma 2.4. *For any feasible traffic network* $\{\mathcal{C} \cup \mathcal{N}, \mathcal{E}\}$ *(see Definition 2.1) and any set of intersections* $\mathcal{U} \subseteq \mathcal{N}$*,* $B(\mathcal{U})$ *is full row rank.*

Proof. Initially assume that $\mathcal{U} = \mathcal{N}$. By construction, each column of $B(\mathcal{N}) \in \{-1, 0, 1\}^{n_{\mathcal{N}} \times n_{\mathcal{E}}}$ has only 2 non-zero entries which sum to 0. The only exceptions are columns corresponding to indexes $\mathcal{E}_{\text{in}}$ which have only one non-zero element equal to -1, and columns corresponding to indexes $\mathcal{E}_{\text{out}}$ with one non-zero entry equal to 1. Note that $B(\mathcal{N})$ is the incidence matrix of $\{\mathcal{N}, \mathcal{E}\}$.

We proceed by contradiction. Assume that $B(\mathcal{N})$ is not full rank. This implies that there exist two sets $\mathcal{U}_1 \subset \mathcal{N}$ and $\mathcal{U}_2 \subset \mathcal{N}$ such that $\sum_{k \in \mathcal{U}_1} B(\mathcal{N})_{k,j} = -\sum_{k \in \mathcal{U}_2} B(\mathcal{N})_{k,j}$, $\forall j$, which is equivalent to $\bigcup_{k \in \mathcal{U}_1} \mathcal{O}(k) = \bigcup_{k \in \mathcal{U}_2} I(k)$ and $\bigcup_{k \in \mathcal{U}_1} \mathcal{I}(k) = \bigcup_{k \in \mathcal{U}_2} O(k)$. This means that for

every node in $\mathcal{U}_1$, any downstream neighbor must be a member of $\mathcal{U}_1$ or $\mathcal{U}_2$. Subsequently, any downstream chain of nodes must be contained in one of these two sets. As every edge is part of path ending in an element of $\mathcal{E}_{\text{out}}$, the previous statements imply that there exists a node $k \in \mathcal{N}$ such that $\mathcal{I}(k) \cap \mathcal{E}_{\text{out}} \neq \varnothing$, which is a contradiction. Hence, the rows of $B(\mathcal{N})$ form a linearly independent set.

Following a similar discussion as at the end of the proof of Lemma 2.3, we note that for $\mathcal{U} \subseteq \mathcal{N}$, the corresponding $B(\mathcal{U})$ implies only the removal of rows from $B(\mathcal{N})$, so the remaining rows are still linearly independent, and thus $B(\mathcal{U})$ is full row rank. □

Theorem 2.2
For any feasible network $\{\mathcal{C} \cup \mathcal{N}, \mathcal{E}\}$, and sets $\mathcal{R} \subseteq \mathcal{N}$ and $\mathcal{U} = \mathcal{N} \setminus \mathcal{R}$, it holds that rank $L(\mathcal{R})$ = rank $A(\mathcal{R})$ + rank $B(\mathcal{U})$.

Proof. For an arbitrary $k \in \mathcal{N}$, we can check that $\mathbb{1}^T A(\{k\}) = B(\{k\})$. This is due to the fact that $\sum_i r_{i,j} = 1$ for all j, and that $r_{i,j} = 0$ if j is not a downstream neighbor of i. Conversely, $\mathbb{1}^T A(\{k\})$ and $B(\{p\})$ share no non-zero entries for $k \neq p$, because $\mathcal{O}(k) \cap \mathcal{O}(p) = \varnothing$ and $\mathcal{I}(k) \cap \mathcal{I}(p) = \varnothing$ if $k \neq p$. Thus, for an arbitrary set $\mathcal{K} \subseteq \mathcal{N}$, each row of $B(\mathcal{K})$ can be obtained by the combination of unique rows of $A(\mathcal{K})$. As $\mathcal{R} \cap \mathcal{U} = \varnothing$, no row of $B(\mathcal{U})$ can be written as a combination of rows of $A(\mathcal{R})$, so both matrices have linearly independent rows. □

2.3.2 Optimal location of flow sensors

In this section we discuss an efficient solution to problem (2.15). The following Corollary provides a way to calculate the optimal number of flow sensors n_s^*.

Corollary 2.1. *For any feasible traffic network $\{\mathcal{C} \cup \mathcal{N}, \mathcal{E}\}$ and any set of intersections $\mathcal{R}^* \subseteq \mathcal{N}$ with cardinality $n_{\mathcal{R}}$, the minimum number of flow sensors required to infer all flows in the network is*

$$n_s^* = n_{\mathcal{E}} - n_{\mathcal{N}} + n_{\mathcal{R}} - \sum_{k \in \mathcal{R}^*} deg^{out}(k). \tag{2.18}$$

Proof. Let $\mathcal{S}^*$ be an optimal solution to (2.15), then $n_{\mathcal{S}}^* = |\mathcal{S}^*| = \text{rank } C(\mathcal{S}^*) = n_{\mathcal{E}} - \text{rank } L(\mathcal{R}^*)$. Thus, the corollary follows directly from Theorem 2.2. □

To solve the sensor location problem we require a method that locates the number of sensors indicated in Corollary 2.1 in such a way that the rows of $\mathcal{C}(\mathcal{S}^*)$ and $L(\mathcal{R}^*)$ are linearly independent.

Because of the efficiency and simplicity of graph-based approaches such as the one of [He13], we decided to expand these techniques to include partial information of TRs. We propose Algorithm 2.2 which makes use of the topological structure of the traffic network and the information given by the set $\mathcal{R}^*$. The algorithm creates spanning trees of the input graph

by using the well known Depth First Search (DFS), which is a well known graph traversal algorithm.

Algorithm 2.2. *Location of flow sensors*

Inputs: Directed graph $\{\mathcal{C} \cup \mathcal{N}, \mathcal{E}\}$ *and set* $\mathcal{R}^* \subseteq \mathcal{N}$.

Output: Set of measured edges $\mathcal{S}^*$.

1. *Replace all nodes in* $\mathcal{C}$ *with a single node* v_0 *such that* $\mathcal{E}_{in} = \mathcal{O}(v_0)$ *and* $\mathcal{E}_{out} = \mathcal{I}(v_0)$.
2. *Initialize* $\mathcal{E}_R \leftarrow \varnothing$ *and* $\mathcal{E}' \leftarrow \mathcal{E}$. *For each* $k \in \mathcal{R}^*$:
   - 2.1 *Find* $\{e_1, e_2, \ldots, e_q\} = \mathcal{O}(k)$.
   - 2.2 *Assign* $\mathcal{E}_R \leftarrow \{e_2, \ldots, e_q\}$.
   - 2.3 *Assign* $\mathcal{E}' \leftarrow \mathcal{E}' \setminus \mathcal{E}_R$.
3. *Ignoring edge direction, perform a DFS over* $\{\mathcal{N} \cup \{v_0\}, \mathcal{E}'\}$ *starting at* v_0. *Denote* $\mathcal{N}_T$ *and* $\mathcal{E}_T$ *as the visited nodes and edges, respectively.*
4. *While* $\mathcal{R}^* \setminus \mathcal{N}_T \neq \varnothing$:
   - 4.1 *For each* $k \in \mathcal{R}^* \setminus \mathcal{N}_T$:
     - 4.1.1 *Construct* $\mathcal{M}_k = \{m \in \mathcal{N} \mid \exists j \in \mathcal{O}(k) \cap \mathcal{I}(m) \cap \mathcal{E}_R\}$.
     - 4.1.2 *If* $\exists m \in \mathcal{M}_k \cap \mathcal{N}_T$:
       - 4.1.2.1 *Find* $j \in \mathcal{O}(k) \cap \mathcal{I}(m)$.
       - 4.1.2.2 *Return* k *and* j. *Exit loop.*
   - 4.2 *Find* $e_1 \in \mathcal{O}(k) \cap \mathcal{E}'$.
   - 4.3 *Assign* $\mathcal{E}' \leftarrow \mathcal{E}' \setminus \{e_1\}$
   - 4.4 *Assign* $\mathcal{E}_R \leftarrow (\mathcal{E}_R \setminus \{j\}) \cup \{e_1\}$
   - 4.5 *Ignoring edge direction, perform a DFS over* $\{\mathcal{N} \cup v_0, \mathcal{E}'\}$ *starting at* k. *Denote* $\mathcal{N}_k$ *and* $\mathcal{E}_k$ *the visited nodes and edges, respectively.*
   - 4.6 *Assign* $\mathcal{N}_T \leftarrow \mathcal{N}_T \cup \mathcal{N}_k$.
   - 4.7 *Assign* $\mathcal{E}_T \leftarrow \mathcal{E}_T \cup \mathcal{E}_k \cup \{j\}$.
5. *Assign* $\mathcal{S}^* = \mathcal{E}' \setminus \mathcal{E}_T$.

Remark: For the case $\mathcal{R}^* = \varnothing$, only steps 1, 3 and 5 are performed, and our algorithm becomes the same as the one presented in [He13].

The first step of the algorithm aggregates the sources and sinks into a single node. The resulting node will satisfy the flow conservation equations and will make the graph strongly

connected. Step 2 removes all but one of the outgoing edges for each of the nodes in $\mathcal{R}^*$. Then, step 3 constructs a tree from the remaining edges. However, because of the removal of edges in step 2, the resulting graph may become disconnected, then the DFS algorithm may not reach all of the nodes in the graph. Step 4 redoes the removal of outgoing edges from $\mathcal{R}^*$ such that the resulting graph is connected, and finishes the construction of a spanning tree of the original graph. Finally, flow sensors are located in edges which are not included in the spanning tree or the removed edges.

Remark: The removal of links in step 2 and the construction of spanning trees using the DFS algorithm are not unique, and alternative node and edge indexing might yield different sensor configuration. Nevertheless, all multiple solutions provide the same number of sensors, and are thus equally optimal.

Example 2.3. Consider the traffic network shown in Figure 2.2. Let $\mathcal{R}^* = \{2, 3\}$ as obtained in Example 2.2. The optimal location of flow sensors is given by Algorithm 2.2: the application of step 1 will generate the graph shown in Figure 2.3.

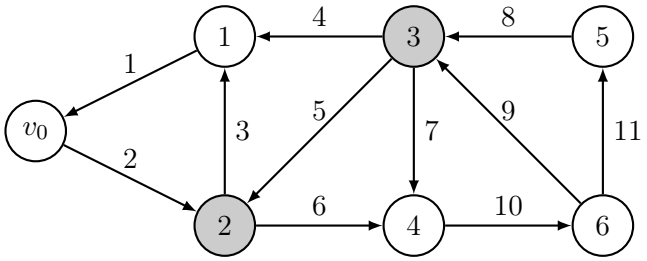


Figure 2.3: Graph after Step 1 of Algorithm 2.2.

During step 2, the algorithm iterates over the elements of $\mathcal{R}^*$. For intersection 2, it can be seen that $\mathcal{O}(2) = \{3, 6\}$, where edge 6 is arbitrarily selected and added to set $\mathcal{E}_R$. Similarly for node 3, $\mathcal{O}(3) = \{4, 5, 7\}$, and edges 4 and 5 are arbitrarily selected and added to $\mathcal{E}_R$. Then, $\mathcal{E}_R = \{4, 5, 6\}$ is removed from the graph. The resulting graph is shown in Figure 2.4.

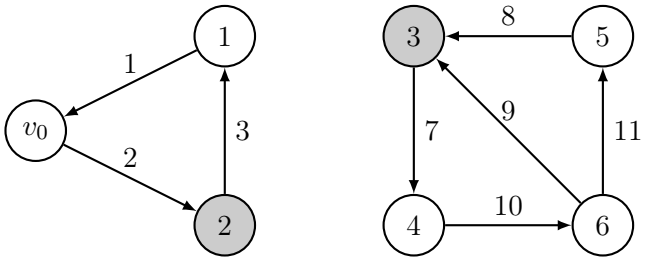


Figure 2.4: Graph after Step 2 of Algorithm 2.2.

Next, step 3 performs a DFS staring at v_0 yielding $\mathcal{N}_T = \{v_0, 1, 2\}$ and $\mathcal{E}_T = \{2, 3\}$. However, it can be seen that the graph of Figure 2.4 is disconnected and that intersection 3 does not belong to the visited nodes $\mathcal{N}_T$. Thus, we proceed with step 4, where we search for

a node in $\mathcal{R}^*$ that has an outgoing edge that connects to one of the nodes in $\mathcal{N}_T$, i.e. node 3. Recall that in step 2, edge 7 was arbitrarily selected to remain in the graph. We now remove edge 7 from the graph and add it to $\mathcal{E}_R$. Then, we perform a DFS staring from node 3, adding the visited nodes $\mathcal{N}_3 = \{3, 4, 5, 6\}$ and edges $\mathcal{E}_3 = \{8, 10, 11\}$ to $\mathcal{N}_T$ and $\mathcal{E}_T$. Finally, edge 4 is removed from $\mathcal{E}_R$ and added to $\mathcal{E}_T$, hence reconnecting the disconnected components (see Figure 2.5).

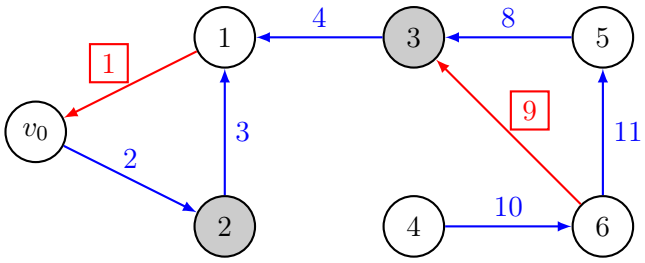


Figure 2.5: Graph after Step 4 of Algorithm 2.2.

After these steps, the visited nodes are $\mathcal{N}_T = \{v_0, 1, 2, 3, 4, 5, 6\}$ and the visited edges are $\mathcal{E}_T = \{2, 3, 4, 8, 10, 11\}$. Figure 2.5 shows the resulting graph, where the edges belonging to $\mathcal{E}_T$ are shown in blue and the remaining edges are marked in red. Note that the subgraph $\{\mathcal{N}_T, \mathcal{E}_T\}$ contains all the nodes, is connected, and has no cycles. Thus, it is a spanning tree of the original input. Finally, sensors are located in the remaining edges $\mathcal{S}^* = \{1, 9\}$, shown with boxes in the figure. □

In the Appendix A.1 we show that the output of Algorithm 2.2 is indeed a solution to problem (2.15).

2.3.3 Sensor cost minimization

Until now, we considered the case where the number of TR sensors was fixed and the number of flow sensors was minimized. Nevertheless, another case of interest might be when the number of both sensors are to be decided such that another variable (e.g. installation cost) is the decision criteria. This can be formulated as,

$$\begin{aligned} \underset{\mathcal{R},\mathcal{S}}{\text{argmin}} \quad & c_{\mathcal{S}}|\mathcal{S}| + c_{\mathcal{R}}|\mathcal{R}| \\ \text{subject to} \quad & \operatorname{rank}\left(\begin{bmatrix} L(\mathcal{R}) \\ C(\mathcal{S}) \end{bmatrix}\right) = n_{\mathcal{E}} \end{aligned}, \tag{2.19}$$

where $c_{\mathcal{R}}, c_{\mathcal{S}}$ are the overall costs of the TR and flow sensors, respectively. For the rest of this section, we will write $|\mathcal{S}| = n_{\mathcal{S}}$ and $|\mathcal{R}| = n_{\mathcal{R}}$ to simplify notation.

In the previous sections it was shown that for any fixed $n_{\mathcal{R}}$ the minimum number of flow sensors corresponds to

$$n_{\mathcal{S}} = n_{\mathcal{E}} - n_{\mathcal{N}} + n_{\mathcal{R}} - \sum_{k \in \mathcal{R}} \deg^{\text{out}}(k).$$

Without loss of generality, assume a permutation of the node indexing such that $\deg^{\text{out}}(k) \geq \deg^{\text{out}}(k+1)$ for all $k \in \mathcal{N}$. This can be achieved by the sorting operation used in Algorithm 2.1. With this indexing, we define

$$f(k) = \sum_{m=1}^{k} \deg^{\text{out}}(m) \quad , \quad k \in \mathcal{N}. \tag{2.20}$$

To solve (2.19), first find the value of $n_{\mathcal{R}}$ that satisfies

$$\begin{array}{ll} \underset{n_{\mathcal{R}}}{\text{argmin}} & c_{\mathcal{S}} n_{\mathcal{S}} + c_{\mathcal{R}} n_{\mathcal{R}} \\ \text{subject to} & n_{\mathcal{S}} = n_{\mathcal{E}} - n_{\mathcal{N}} + n_{\mathcal{R}} - f(n_{\mathcal{R}}) \\ & 0 \leq n_{\mathcal{R}} \leq n_{\mathcal{N}} \end{array} , \tag{2.21}$$

and then use this number as an input to Algorithms 2.1 and 2.2. It is straightforward to check that this procedure provides a solution to (2.19): for any $n_{\mathcal{R}}$, it was shown that the algorithms generate sets $\mathcal{S}$ and $\mathcal{R}$ that satisfies the constraints of (2.19) and (2.21).

Problem (2.21) can be simplified to

$$\begin{array}{ll} \underset{n_{\mathcal{R}}}{\text{argmin}} & (c_{\mathcal{S}} + c_{\mathcal{R}}) n_{\mathcal{R}} - c_{\mathcal{S}} f(n_{\mathcal{R}}) \\ \text{subject to} & 0 \leq n_{\mathcal{R}} \leq n_{\mathcal{N}} \end{array} . \tag{2.22}$$

Due to the choice of node indexing, $\deg^{\text{out}}(k)$ is non-increasing, and therefore, $f(k)$ is concave. The cost function in (2.22) is then convex, so gradient-based approaches are guaranteed to work. After straightforward manipulations we find that the optimal $n_{\mathcal{R}}$ is such that

$$\deg^{\text{out}}(n_{\mathcal{R}}) \geq \frac{c_{\mathcal{S}} + c_{\mathcal{R}}}{c_{\mathcal{S}}} \geq \deg^{\text{out}}(n_{\mathcal{R}} + 1). \tag{2.23}$$

This problem is easily solved by performing a simple search and returning the corresponding index. For the case $(c_{\mathcal{S}} + c_{\mathcal{R}})/c_{\mathcal{S}} > \deg^{\text{out}}(1)$, then $n_{\mathcal{R}} = 0$. On another hand, if this ratio is equal to 1, we let $n_{\mathcal{R}}$ to be the largest integer for which $\deg^{\text{out}}(n_{\mathcal{R}}) = 2$. This is because locating TR sensors in nodes with only one outbound edge will not provide any useful information.

Note that when $(c_{\mathcal{S}} + c_{\mathcal{R}})/c_{\mathcal{S}}$ is an integer, there may be multiple equally optimal solutions to the problem.

2.3.4 Computational complexity

To solve the original problem (2.13), it was shown that it could be split into two simpler problems (2.14) and (2.15), for which the solutions are given using Algorithms 2.1 and 2.2 respectively. Therefore, the total computational complexity of the proposed method is the sum of the complexities of these two algorithms.

For Algorithm 2.1, the only required operation is the sorting of the vector of out-degrees, which is known to have complexity $O(n_{\mathcal{N}} \log n_{\mathcal{N}})$. For Algorithm 2.2, it can be seen that steps

3 and 4 contain the more complex operations, as they require to perform graph traversing and to search for particular structures. It was discussed that step 2 of Algorithm 2.2 could generate a disconnected graph with q connected components. As each of the subgraphs must contains at least one node with an outgoing edge the previously connected to another subgraph, it must be that $q \leq n_{\mathcal{R}}$. For each of the subgraphs $\{\mathcal{N}_i, \mathcal{E}_i\}$, $i = 0, \ldots, q-1$, a DFS is performed (Steps 3 and 4). As each of this searches has complexity $O(|\mathcal{N}_i| + |\mathcal{E}_i|)$, the total complexity is

$$\sum_{i=0}^{q-1} O\left(|\mathcal{N}_i| + |\mathcal{E}_i|\right) = O\left(\sum_{i=0}^{q-1} |\mathcal{N}_i| + \sum_{i=0}^{q-1} |\mathcal{E}_i|\right) = O(n_{\mathcal{N}} + n_{\mathcal{E}}).$$

On the other hand, step 4 requires to find for every subgraph an edge that connects to the main graph. This search operation has an average complexity of $O(\log n_{\mathcal{N}})$ that must be repeated at most $n_{\mathcal{R}}$ times.

By taking into account all the operations from both algorithms, we obtain a total complexity of

$$O(n_{\mathcal{N}} \log n_{\mathcal{N}} + n_{\mathcal{E}}).$$

This represents a significant improvement when compared to other works in the literature such as [HPC09] and [Ng12] that rely on Gaussian elimination, and have computational complexity of $O(n_{\mathcal{N}} n_{\mathcal{E}}^2)$.

2.4 Flow and density estimation approach

Using the sensor location described in the previous section, it is possible to use the TR and flow measurements to obtain a unique solution to (2.11). Nevertheless, this is not enough to uniquely determine the density in each road. This can be seen from the fundamental diagram as shown in Fig. 2.6: for a given value of flow for some road $i \in \mathcal{E}$, two possible densities can be associated, the density in free-flow ρ_i^f and the density in congestion ρ_i^c.

To discern between these two densities, information about the speed in each road is required. Suppose that speed from floating car data is available everywhere. Note from (2.6) that speed data can be used to identify if a road is congested or in free-flow: if $v_i = v_i^{max}$ the road is in free-flow, and if $v_i < v_i^{max}$ the road is in congestion. Once a road is classified in one of these regimes, an affine mapping between flow and density can be established as $\boldsymbol{\rho} = F(\mathbf{v})\boldsymbol{\varphi} + \mathbf{d}(\mathbf{v})$ where $F(\mathbf{v})$ is a diagonal matrix and $d(\mathbf{v})$ is a vector, and are both defined from the FD as

$$F_{i,i}(\boldsymbol{v}) = \begin{cases} 1/v^{max} & \text{if} \quad v^{max} - v_i < \epsilon \\ -1/w & \text{if} \quad v^{max} - v_i \geq \epsilon \end{cases} \tag{2.24}$$

and

$$d_e(\boldsymbol{v}) = \begin{cases} 0 & \text{if} \quad v^{max} - v_e < \epsilon \\ \rho_{max} & \text{if} \quad v^{max} - v_e \geq \epsilon \end{cases} \tag{2.25}$$

The tunable parameter ϵ takes into account the uncertainty in the speed measurements.

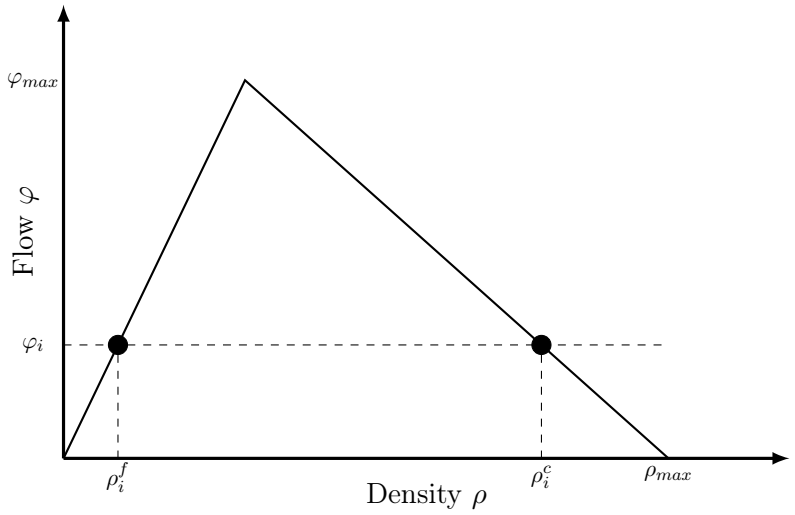


Figure 2.6: Density solutions from the fundamental diagram.

Using the sensor measurements and the FCD for the road speed, the joint estimation of traffic flow and density is formulated as the following optimization problem,

$$\begin{aligned} \min_{\hat{\boldsymbol{\varphi}},\hat{\boldsymbol{\rho}}} \quad & \left\| \begin{bmatrix} F(\mathbf{v}) & -\mathbb{I} \\ C(\mathbb{S}) & \mathbf{0} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\varphi}} \\ \hat{\boldsymbol{\rho}} \end{bmatrix} - \begin{bmatrix} \mathbf{d}(\mathbf{v}) \\ \boldsymbol{\varphi}_m \end{bmatrix} \right\|^2 \\ \text{s.t.} \quad & L(\mathcal{R})\hat{\boldsymbol{\varphi}} = \mathbf{0} \\ & 0 \leq \hat{\boldsymbol{\varphi}} \leq \varphi_{max} \\ & 0 \leq \hat{\boldsymbol{\rho}} \leq \rho_{max} \end{aligned} , \tag{2.26}$$

where $\hat{\boldsymbol{\rho}}$ and $\hat{\boldsymbol{\varphi}}$ are the estimated quantities. This formulation has as objective to minimize the difference between the estimated pair of flow and density, and the flow measurements and the regime of the triangular FD determined by speed data. The constraints impose the conservation of flow at the intersections, and that the variables remain of sensible size. As the cost function is convex and the constraints are linear, this is a convex optimization problem which can be solved using off-the-shelf tools.

2.5 Simulation and validation

In this section, we evaluate the performance of the proposed approach on a real traffic network from the city of Grenoble, France. The experimental platform GTL-Ville[1] developed for the Grenoble city is the study case of the Scale-FreeBack[2] project. A section of the city's downtown was chosen, spanning an approximate area of 1.5 km by 1 km, and can be seen in Fig. 2.7. A graph of the zone was constructed under MATLAB, as shown in Fig. 2.8, which includes information for each road such as its number of lanes, speed limit, and length.

[1] http://gtlville-dev.inrialpes.fr/

[2] Scale-Free control for complex physical network systems, http://scale-freeback.eu.

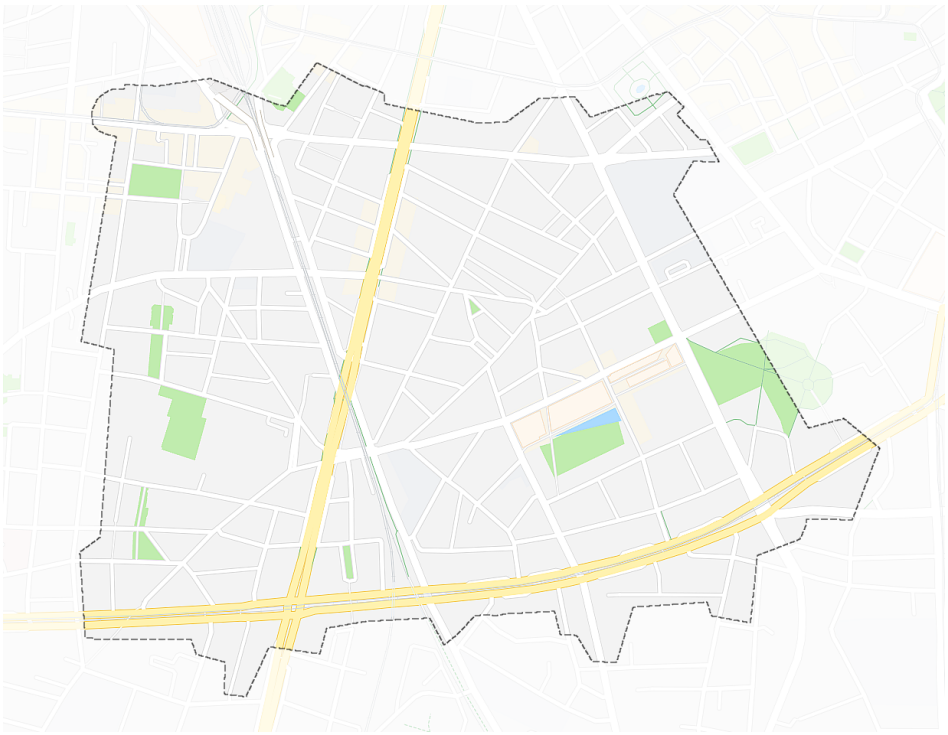


Figure 2.7: Selected area of Grenoble downtown.

Note: The shape of the edges in the figure represent only the connections between roads and intersections, and do not follow the actual geometry of the roads; the arching present in some places is a graphical way to avoid overlapping between bidirectional roads.

2.5.1 Sensor location

For a given value of $n_{\mathcal{R}}$ we solved problems (2.15) and (2.17). The process was done iteratively by using different values of $n_{\mathcal{R}}$. In Fig. 2.9 we show the required number of flow sensors for each value of $n_{\mathcal{R}}$, as well as the theoretical prediction from (2.18). We can observe that the outputs of the algorithm coincide exactly with the predictions from (2.18). For the case $n_{\mathcal{R}} = 0$, a total of 389 flow sensors are required, which represents 48% of all roads. This result is coherent with the reported information by [Ng12] and [Vit+14], who estimated that real networks require the measurement of between 50% and 60% of all roads to estimate all flows. Information provided by TR sensors reduces the number of flow sensors needed. However, the plot shows a piecewise-linear behavior, where the slope of each section is non-increasing as the number of TR sensors increases. Thus, there are diminishing returns in the benefits of including more TR information. From (2.18), the slope of each linear section relates to the out-degree of the intersections being equipped with TR sensors. When $n_{\mathcal{R}} \leq 90$, intersections have 3 or more outgoing roads, hence each TR sensor reduces the number of flow sensors in 2 or more. When $90 < n_{\mathcal{R}} \leq 245$, intersections have 2 outgoing roads, then each TR sensor reduces the number of flow sensors in 1. Lastly, if $n_{\mathcal{R}} > 245$, there are only intersections with

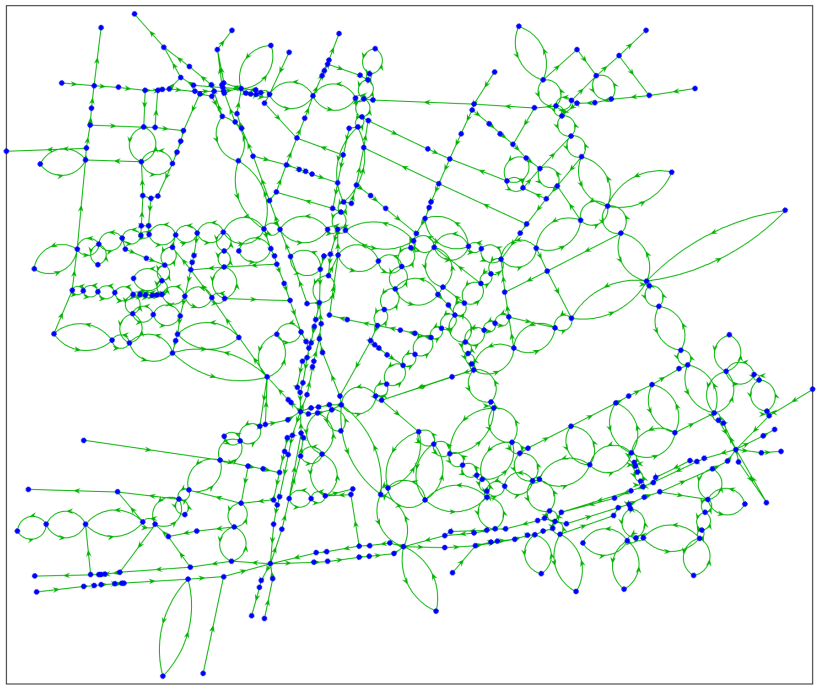


Figure 2.8: Directed graph representing the selected traffic network of Grenoble downtown. The nodes in blue represent intersections and centroids. The edges in green represent roads.

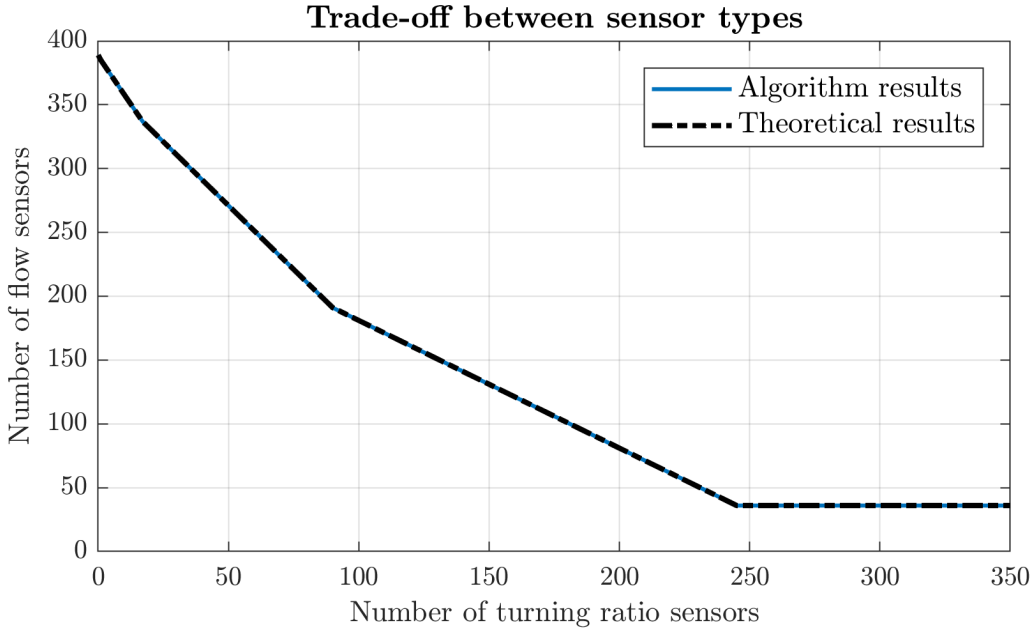


Figure 2.9: Minimum number of flow sensors for a given number of TR sensors.

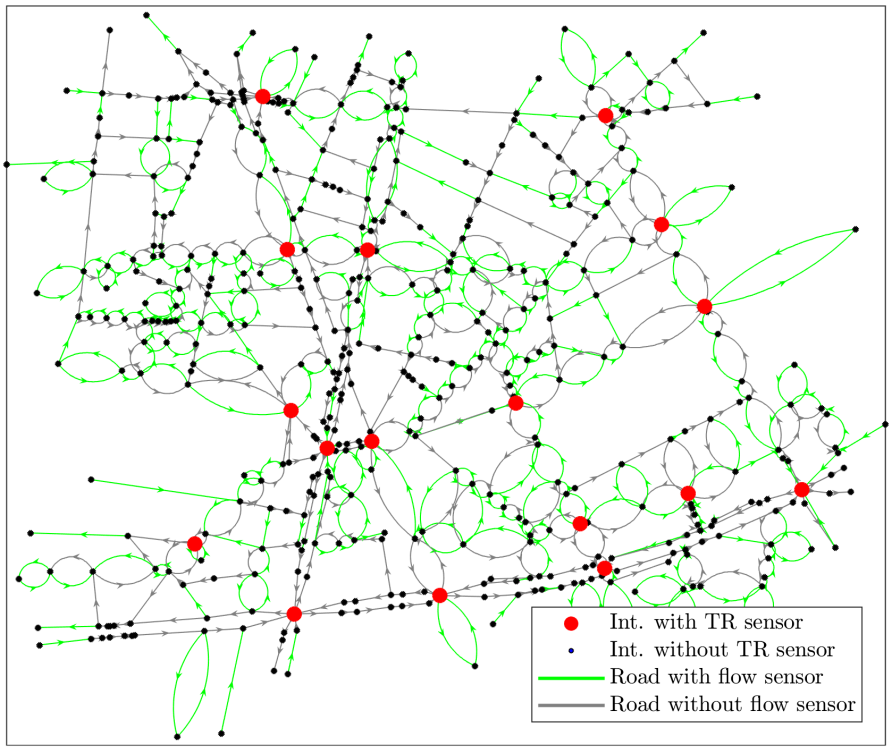


Figure 2.10: Particular configuration of sensors, using 17 TR sensors and 337 flow sensors.

one outgoing road, which provides no benefit. For this last case, the number of required flow sensors is equal to the number of boundary incoming roads.

As a particular example, consider $n_{\mathcal{R}} = 17$, which corresponds to the number of intersections which have 4 or more outgoing roads. The resulting location provided by the algorithms for TR and flow sensors is shown in Fig. 2.10. From (2.23), this choice corresponds to the case when TR sensors are between 2 to 3 times more expensive than flow sensors. In the following sections, we will use this particular configuration to estimate the flow and density.

2.5.2 Flow and density estimation

To evaluate the proposed approach, ground truth values are obtained from simulated traces of individual vehicles using the well-known traffic microsimulator Aimsun [BC05]. This software models the position, speed and acceleration of each vehicle according to the interaction with the other vehicles in each section and intersection.

Aimsun requires as inputs a network geometry, profile of external demands, and definition of the TRs for each intersection. The external demands were set constant in time, in order reach steady-state, with magnitudes obtained from real data collected using already existing sensors in the network and manual data collection. TRs were set arbitrarily to be homogeneously divided to the outgoing roads from each intersection. The initial condition of the

Table 2.1: Fundamental diagram parameters

Variable	Symbol	Fixed value	Min value	Max value
Jam density per lane (veh/km)	ρ^{max}	120	-	-
Critical density per lane (veh/km)	ρ^{c}	36	-	-
Free-flow speed (km/h)	v^{max}	-	10	50
Congested-wave speed (km/h)	w	-	4.29	21.43

network is completely empty, and the simulation time is of 1 hour.

As measurements, data was collected from the intersections and roads as shown in Fig. 2.10. Flow sensors are located at the exit of each selected road, and they count the number of vehicles that passed that location during time intervals of 10 minutes. For TR, the total number of vehicles that made each turn at the simulation end was recorded, and then normalized with the sum of all turns from the same incoming road.

For the validation stage, the flow of each road is recorded in the same way as described for the measurements (vehicles counted at the exit during 10 minute intervals). The density of each road is calculated as the average number of vehicles contained in each road during a time interval of 10 minutes, divided by the road's length and number of lanes.

The estimated flow and density are obtained by implementing (2.26) in MATLAB, using the well known interior-point algorithm [BV04, Chapter 11]. Table 2.1 shows the FD parameters used in estimation. For the jam and critical densities, the values are the same for all roads in the network, as they depend on the minimal distance between vehicles which is a function of cultural driving habits, and the average vehicle length. In the literature, reported values for the jam density range between 100 and 200 veh/km (e.g. [TK13],[KD17],[AI+04]). For the considered scenario, a commonly used value of 120 veh/km for Europe is selected. A ratio of critical to jam density of 0.3 is used to obtain the value of ρ^c [Car+12]. The value for the free-flow speed vary from one road to another, so the minimum and maximum values are shown. In practice, this parameter can be obtained from the FCD by considering the average measured road speeds during periods of low traffic. The congested-wave speed parameters is calculated for each road from the previous parameters.

Figure 2.11 shows the ground truth values obtained directly from Aimsun, and the corresponding estimates. To improve the readability of the figure, roads are indexed to sort in ascending order the ground truth values for each case.. Note that the estimates are dispersed around the real values. This dispersion is caused by several factors. There is uncertainty in the parameters used by the estimator, as the microscopic model used by Aimsun allows for each vehicle to have a stochastic optimal velocity and minimal inter-vehicle headway, which is similar to real driver behavior. Additionally, in the microscopic scale, interactions between vehicles in intersections and accelerations when moving to a different road can cause deviations from an ideal steady-state. Nevertheless, most of the roads show a close agreement with the real values, with the exception of some outliers.

To quantitatively analyze the error between the true and estimated values, we use as a

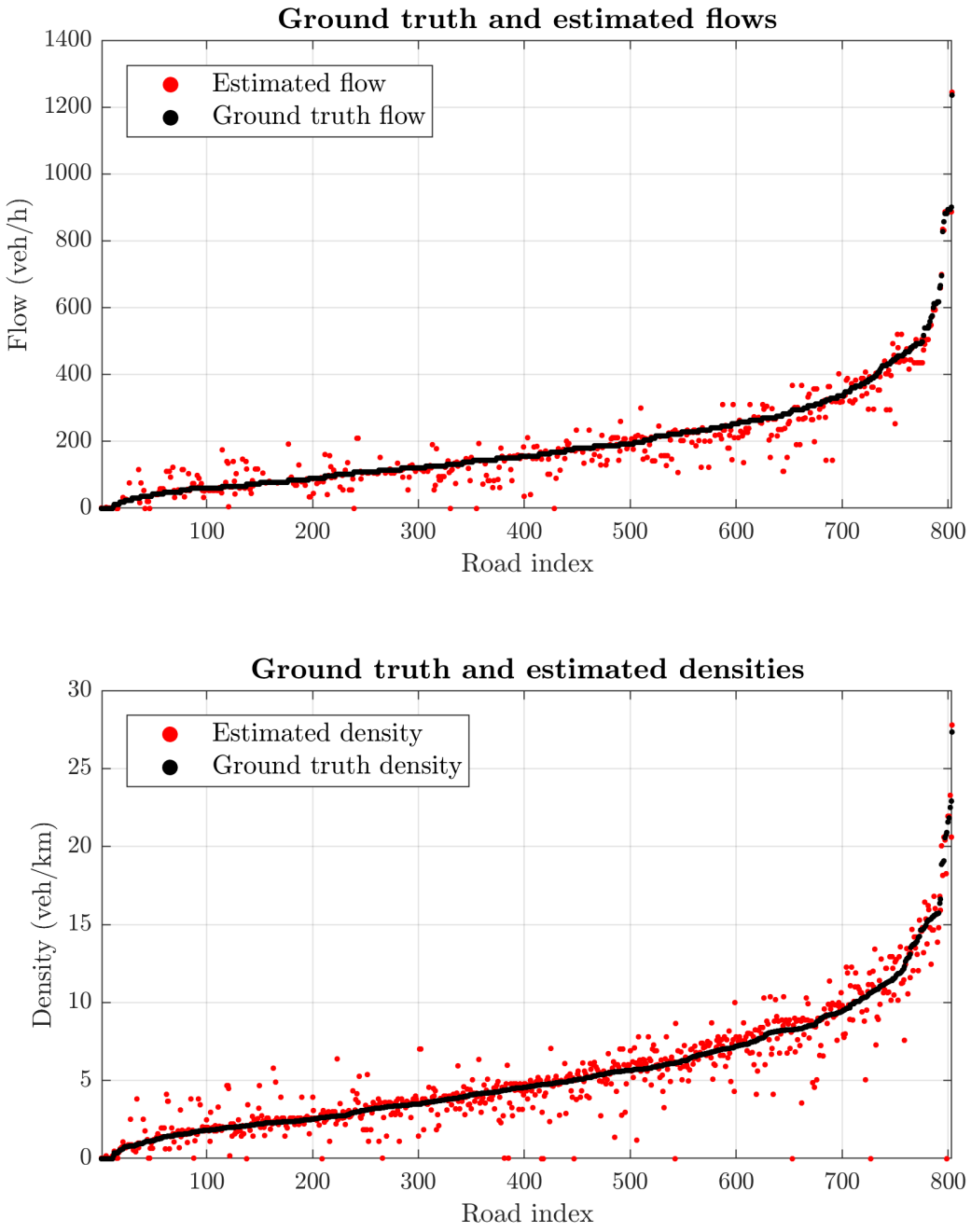


Figure 2.11: Ground truth and estimated values for flow and density in the simulated scenario.

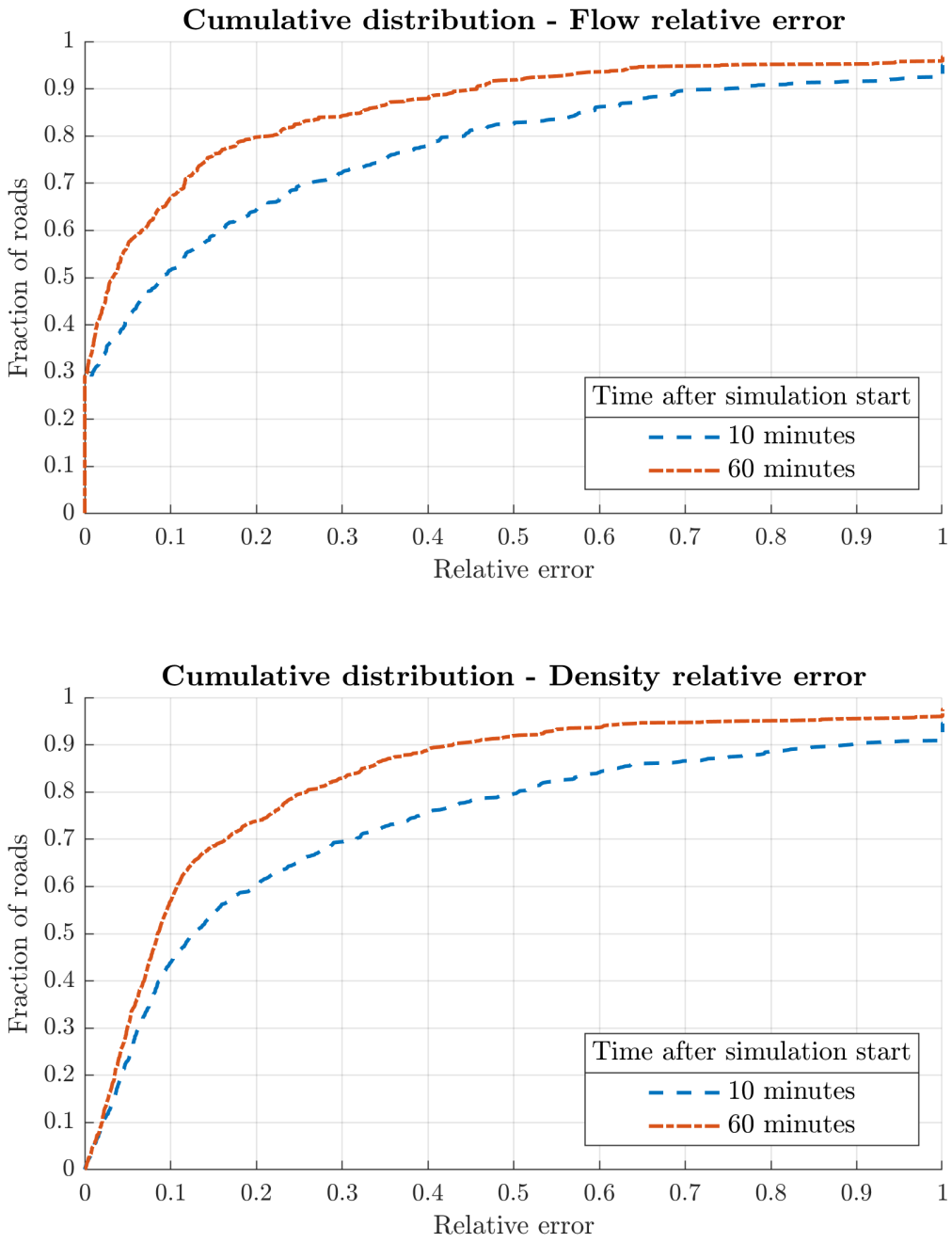


Figure 2.12: Cumulative distribution for the flow and density relative errors. Each point in the curve shows the fraction of roads that have a relative error less than or equal to the abscissa.

metric the relative error (RE). For a given road i, this is defined as

$$\mathrm{RE}_\rho = \frac{|\rho_i - \hat{\rho}_i|}{\rho_i} \tag{2.27}$$

where RE_ρ is a function of density, ρ_i is the true value, and $\hat{\rho}_i$ is the estimated value. The RE_φ for flow is defined analogously. Figure 2.12 shows the cumulative distribution plot for both relative errors. The cumulative distribution function (CDF) is defined as the fraction of roads for which the error is less than or equal to a given value. The figure shows two curves: in blue, the error for the estimates using measurements collected during the first 10 minutes of the simulation, so the steady-state is not yet reached. The red curve shows the results for the estimates using measurements obtained at the end of the simulation (the last 10 minutes from $t = 50$ minutes to $t = 60$ minutes), where steady-state is expected.

For the estimation in steady-state (red curve), we can see that the median error for the flow is about 3%, and for density about 8%. Moreover, 80% of the roads have a error below 20% for flow and below 25% for density. This shows that the proposed approach provides a close estimation for most of the roads in the network for the considered scenario once steady-state is achieved. In contrast, the estimates obtained during the transient (blue curve) show higher errors, evidenced by the horizontal distance between the curves in the figure. Nevertheless, even in these conditions the obtained median errors for flow and density are 10% and 13% respectively, and 60% of the roads present an error below 20% for both variables.

2.6 Concluding remarks

In this work, we solved the network sensor location problem by using two types of sensing technologies. This allows to reduce the number of flow sensors that are originally required. We solved this problem by finding the solution of two independent problems: optimally locating a given number of TR sensors, and then, using this result as an input, we locate flow sensors. These problems were shown to have computationally efficient solutions, as the first one only requires vector sorting, and the second one can be solved by the construction of spanning trees. The methods were shown to give an optimal solution to the original problem. The complexity of the proposed algorithms was shown to be $O(n_\mathcal{N} \log n_\mathcal{N} + n_\mathcal{E})$, which is much more efficient than previous methods that rely on algebraic computations providing complexities of $O(n_\mathcal{N} n_\mathcal{E}^2)$.

The proposed approaches were tested using microscopic simulations in the real network of Grenoble. The estimation results showed to be very close to the ground truth, with median relative errors under 10% for both flow and density estimation. Nevertheless, the number of sensors required to deploy this approach can be unpractical for some applications, even when considering the use of both sensor types. Furthermore, the requirement of steady-state conditions limit the applicability of the methods. In the following chapters we will consider different approaches that do not assume steady-state conditions and that use fewer sensors.

Chapter 3

Dynamic estimation of density and flow

Contents

3.1 Overview

This Chapter proposes a dynamic estimator for the vehicle density of every road section of a large urban traffic network. We assume that a limited number of flow and TR sensors can be installed, and that aggregate floating car data (FCD) are available, such that the space-mean speed of each road can be estimated. We propose a method to locate TR sensors, which takes as input previous low-quality estimates of the turn rates, and then assigns each intersection a weight according to the effect on the total density estimation error caused by perturbations between the a priori and actual TR values. We evaluate the models and estimator for the city of Grenoble, France.

3.2 Model

We define a traffic network as a directed graph $\mathcal{G} = \{\mathcal{N}, \mathcal{E}\}$ where the nodes $\mathcal{N} \subset \mathbb{N}$ correspond to intersections, and the edges $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$ correspond to road sections. Additionally, for every road section i, there are associated parameters such as the road length ℓ_i, number of lanes Γ_i, and maximum velocity $v_i^{\max}$.

The traffic state refers to the collection of road densities, inflows and outflows for all roads, which we denote with vectors $\boldsymbol{\rho}(t), \boldsymbol{\varphi}^{\text{in}}(t), \boldsymbol{\varphi}^{\text{out}}(t) \in \mathbb{R}^{|\mathcal{E}|}$. Traffic dynamics are governed by the conservation law

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\rho}(t) = L^{-1}(\boldsymbol{\varphi}^{\text{in}}(t) - \boldsymbol{\varphi}^{\text{out}}(t)) \tag{3.1}$$

where L is a diagonal matrix containing the road lengths, i.e., $L = \operatorname{diag}(\boldsymbol{\ell})$.

Intersections are modeled as 0 dimensional points that do not store vehicles. To model the exchange of inflows and outflows of the different roads at the intersections we use the TR parameters. Let $\mathcal{I}(n)$ be the set of incoming roads to some intersection $n \in \mathcal{N}$ and $\mathcal{O}(n)$ be the set of outgoing roads from n. A TR $r_{i,j}$ for $i \in \mathcal{I}(n)$ and $j \in \mathcal{O}(n)$ defines the proportion of vehicles exiting i that enters j. As intersections do not store vehicles, conservation of density implies that

$$\sum_{j \in \mathcal{O}(n)} r_{i,j} = 1 \ , \quad \forall n \in \mathcal{N} \quad \forall i \in \mathcal{E} \setminus \mathcal{E}_{\text{out}} \tag{3.2}$$

where $\mathcal{E}_{\text{out}}$ are outgoing roads at the boundary of the network, hence have no downstream roads. Let $R \in \mathbb{R}^{|\mathcal{E}| \times |\mathcal{E}|}$ be the TR matrix with elements $r_{i,j}$. If there is no connection between roads i, j, then $r_{i,j} = 0$. The input flows of each section can be expressed as a linear combination of the output flows of the preceding sections.

$$\boldsymbol{\varphi}^{\text{in}}(t) = R^{\top}\boldsymbol{\varphi}^{\text{out}}(t) + \boldsymbol{\varphi}^{\text{ext}}(t) \tag{3.3}$$

where $\boldsymbol{\varphi}^{\text{ext}}(t)$ corresponds to the external inflows (also known as the input demands), which are the incoming flow for the roads which are at the boundaries of the network. Combining eqs. (3.1) and (3.3), we have

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\rho}(t) = L^{-1}(R^{\top} - \mathbb{I})\boldsymbol{\varphi}^{\text{out}}(t) + L^{-1}\boldsymbol{\varphi}^{\text{ext}}(t) \tag{3.4}$$

Define $v_i(t)$ as the space-mean speed of road i, that is, the average of the speeds of vehicles inside of road i at time t. Denote $\mathbf{v}(t)$ as the speed vector with elements $v_i(t)$. Using the hydrodynamic relation, we can approximate the outflows as

$$\boldsymbol{\varphi}^{\text{out}}(t) \approx V(t)\boldsymbol{\rho}(t) \tag{3.5}$$

where $V(t) = \operatorname{diag}(\mathbf{v}(t))$. This relation applies accurately when considering very short distances, or when the spatial variations in vehicle speed and density are negligible. We make the following assumption,

Assumption 3.1. The speed and density throughout a road section do not vary significantly in the spatial domain.

Therefore, (3.4) can be rewritten as

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\rho}(t) = L^{-1}(R^\top - \mathbb{I})V(t)\boldsymbol{\rho}(t) + L^{-1}\boldsymbol{\varphi}^{\text{ext}}(t) \tag{3.6}$$

In this paper, we use an open loop observer following the same dynamics of (3.6).To achieve this goal, we require knowledge about the TRs, the road speeds, and the external flows. The measurement of these variables will be examined in Section 3.3. For now, denote $\hat{R}$ as the estimate for the TR matrix, $\hat{V}(t)$ as the estimate for the speed matrix, and $\boldsymbol{\varphi}^{\text{mea}}(t)$ as the flow measurements at the boundary inputs. Thus, the estimator for the system's density is

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\boldsymbol{\rho}}(t) = L^{-1}(\hat{R}^\top - \mathbb{I})\hat{V}(t)\hat{\boldsymbol{\rho}}(t) + L^{-1}\boldsymbol{\varphi}^{\text{mea}}(t) \tag{3.7}$$

where $\hat{\boldsymbol{\rho}}(t)$ is the estimated density. The error associated to this estimate is defined as

$$\mathbf{e}(t) = \boldsymbol{\rho}(t) - \hat{\boldsymbol{\rho}}(t). \tag{3.8}$$

with dynamics,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{e}(t) \;=\; & L^{-1}(R^\top - \mathbb{I})V(t)\mathbf{e}(t) - L^{-1}(R^\top - \mathbb{I})\Omega(t)\hat{\boldsymbol{\rho}}(t) \\ & -L^{-1}\Xi^\top V(t)\hat{\boldsymbol{\rho}}(t) - L^{-1}\Xi^\top\Omega(t)\hat{\boldsymbol{\rho}}(t) + (\boldsymbol{\varphi}^{\text{ext}}(t) - \boldsymbol{\varphi}^{\text{mea}}(t))\end{aligned} \tag{3.9}$$

where $\Xi = \hat{R} - R$ and $\Omega(t) = \hat{V}(t) - V(t)$. The performance of the proposed estimator depends on the stability of (3.9) and how it is affected by the error terms. Consider first the case where all information is known and measurements are ideal, i.e., $\Xi = \mathbf{0}$, $\Omega(t) = \mathbf{0}$, and $\boldsymbol{\varphi}^{\text{ext}}(t) = \boldsymbol{\varphi}^{\text{mea}}(t)$. Theorem 3.1 shows that under some assumptions on the velocity $V(t)$, the estimator is asymptotically stable for any initial condition.

Theorem 3.1
Consider the dynamic system

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{e}(t) = L^{-1}(R^\top - \mathbb{I})V(t)\mathbf{e}(t) \tag{3.10}$$

If $V = diag(\mathbf{v}(t))$ *such that* $\mathbf{v}(t)$ *is continuous, bounded by*

$$\mathbf{0} < v_{min}\mathbf{1} \leq \mathbf{v}(t) \leq v_{max}\mathbf{1}, \tag{3.11}$$

and is continuously differentiable with derivative bounded by

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{v}(t) \leq \frac{v_{min}^2(1-\epsilon)}{\ell_{max}||(\mathbb{I} - R^\top)^{-1}||_\infty}\mathbf{1} \tag{3.12}$$

where $0 < \epsilon < 1$ *is arbitrary, then* (3.10) *is asymptotically stable, i.e.,* $\mathbf{e}(t) \to \mathbf{0}$ *as* $t \to \infty$.

The proof of Theorem 3.1 is shown in Appendix A.2. The restrictions imposed on the velocity are very strict, but they are sufficient conditions which may be conservative, and the system may still be stable when the conditions are not satisfied. This stability result shows that if the measured TRs, velocities and boundary inflows are close to the real values, the estimator can give a close estimation of the network's states. In Section 3.3, we discuss how these variables are measured, and how to locate sensors such the resulting error is minimized.

3.3 Sensor location and input data

The estimator (3.7) requires the knowledge of the TR parameters $r_{i,j}$, the measurement of space-mean speeds $\mathbf{v}(t)$, and some observations of the state variables or incoming signals. We are going to consider the following sources of information:

- Stationary counting sensors. Installed at a certain location in a road, they measure the number of vehicles that pass in front.
- Turning ratio sensors. Installed in all adjacent roads to an intersection, they measure the proportion of vehicles taking any turn.
- FCD. Some of the vehicles in the network report their GPS traces to a remote server. To protect user privacy, the traces are aggregated, but average section velocities are able to be recovered.

As the number of sensors is constrained by budgetary limits, it is important to determine the optimal locations of few sensors, such that the density and flow estimates are close to the real values.

3.3.1 Floating car data

Consider that a fraction of the vehicles in the network are equipped with devices (e.g. a GPS navigator) that periodically report to a centralized server information about the vehicles trajectory, such as its position and velocity at a given time. Define by $\mathcal{V}_i(t)$ the set of vehicles indexes providing FCD that are inside of road i at time t. The total number of vehicles in i cannot be estimated with this information as the penetration rate of vehicles that provides FCD is unknown. However, as the velocity of a vehicle is affected by the velocities of surrounding vehicles, this can be used to estimate the space-mean speed of the section. Let ν_α be the speed of a vehicle indexed by α. We define the aggregated speed for road section i from FCD data by

$$v_i^{\mathrm{FCD}}(t) = \frac{1}{|\mathcal{V}_i(t)|} \sum_{\alpha \in \mathcal{V}_i(t)} \nu_\alpha(t) \tag{3.13}$$

where $|\mathcal{V}_i(t)|$ is the cardinality of $\mathcal{V}_i(t)$, corresponding to the number of vehicles in road i at time t. We assume that FCD speeds are direct measurements of the space-mean road speed, and thus

$$\Omega(t) = \mathbf{0} \tag{3.14}$$

However, due to technical constraints, this information is not provided continuously, but at discrete times denoted by $\{t_0^v\ ,\ t_1^v\ ,\ t_2^v\ ,\ \dots\}$. Thus let $\mathbf{v}^{\mathrm{FCD}}[i]$ be the average of the reported vehicle velocities for all road sections between times t_i^v and t_{i-1}^v. Denote $\hat{V}[i] = \mathrm{diag}(\mathbf{v}^{\mathrm{FCD}}[i])$.

We assume in what follows that the aggregated speeds are available for all roads. In the case where the information is not available for some roads due to a lack of reporting vehicles, we use the value of the maximum speed (or speed limit) $v^{\max}$.

3.3.2 Stationary counting sensors

We assume that flow sensors are located in the boundary inputs of the network, such that the external flows $\boldsymbol{\varphi}^{\text{ext}}(t)$ are measured directly,

$$\boldsymbol{\varphi}^{\text{mea}}(t) = \boldsymbol{\varphi}^{\text{ext}}(t) \tag{3.15}$$

However, in practice the flow measurements are not continuous. Instead, the sensors report the number of vehicles counted during a discrete interval. Let $\{t_0^{\varphi}\ ,\ t_1^{\varphi}\ ,\ t_2^{\varphi}\ ,\ \dots\}$ denote the times at which flow measurements are available. Let $\boldsymbol{\varphi}^{\text{mea}}[m]$ be the measured flow during the interval $[t_{m-1}^{\varphi}, t_m^{\varphi}]$. Then

$$\boldsymbol{\varphi}^{\text{mea}}[m] = \frac{1}{(t_m^{\varphi} - t_{m-1}^{\varphi})} \int_{t_{m-1}^{\varphi}}^{t_m^{\varphi}} \boldsymbol{\varphi}^{\text{ext}}(\tau)\mathrm{d}\tau. \tag{3.16}$$

3.3.3 Turning ratios

The estimator requires the knowledge of the TR parameters for all intersections in the traffic network. This is however unfeasible in practice as it requires to provide every intersection with sensors, which is impossible for networks containing more than a few intersections.

To circumvent this restriction, some authors have proposed the use of heuristic formulas that provide a priori values to the TRs for each intersection, based on few network parameters which are more readily obtainable [Fur90]. As these values are only an approximation, an optimal selection of a reduced number of measurements is then required to complement this initial values. Appendix B discusses an approach to provide a priori values to the TRs based only on the physical properties of the network's roads, which has been validated using data from a Highway near the city of Grenoble.

Suppose that a limited number of sensors is available. The selection of intersections to locate sensors is done to minimize the error in the flow-density estimation caused by deviations between the real and estimated TRs. For this, we propose to identify which intersections generate the highest error given some perturbations in its TRs.

3.3.3.1 Flow estimation error

Let $\hat{r}_{i,j}$ be the a priori value proposed for some TR. Let the deviation between this and the actual value be given by $\xi_{i,j}$ such that,

$$\hat{r}_{i,j} = r_{i,j} + \xi_{i,j} \tag{3.17}$$

The deviations $\xi_{i,j}$ will cause an error in the state estimation. Define $\hat{R}$ and Ξ as matrices with elements $\hat{r}_{i,j}$ and $\xi_{i,j}$, respectively. Note that $\hat{R} = R + \Xi$. To quantify the sensitivity of the TRs in the error, we neglect the effects of time variations of the external flows and velocities, hence $V(t) = V$ and $\boldsymbol{\varphi}^{\text{ext}}(t) = \boldsymbol{\varphi}^{\text{ext}}$.

We want to express the effect of the TR perturbations Ξ in the estimation error $\mathbf{e}(t)$. To simplify the notation, let $M = (\mathbb{I} - R^\top)V$ and $\hat{M} = (\mathbb{I} - \hat{R}^\top)V$, such that $\hat{M} = M - \Xi^\top V$. Under the assumption that section speeds are constant, the dynamics (3.6) and (3.7) become time-invariant, for which a closed-form solution is known,

$$\begin{aligned}\boldsymbol{\rho}(t) &= \mathrm{e}^{-L^{-1}Mt}\boldsymbol{\rho}(0) + \left(\int_0^t \mathrm{e}^{-L^{-1}M\tau}\mathrm{d}\right) L^{-1}\boldsymbol{\varphi}^{\text{ext}} \\ \hat{\boldsymbol{\rho}}(t) &= \mathrm{e}^{-L^{-1}\hat{M}t}\hat{\boldsymbol{\rho}}(0) + \left(\int_0^t \mathrm{e}^{-L^{-1}\hat{M}\tau}\mathrm{d}\right) L^{-1}\boldsymbol{\varphi}^{\text{ext}}\end{aligned} \tag{3.18}$$

Therefore, the error can be written as

$$\mathbf{e}(t) = e^{-L^{-1}Mt}\boldsymbol{\rho}(0) - e^{-L^{-1}\hat{M}t}\hat{\boldsymbol{\rho}}(0) + \int_0^t \left(e^{-L^{-1}M\tau} - e^{-L^{-1}\hat{M}\tau}\right)\mathrm{d}\tau L^{-1}\boldsymbol{\varphi}^{\text{ext}} \tag{3.19}$$

In [RVWF19], we showed that $\mathbb{I} - R^\top$ is an invertible M-matrix, with all eigenvalues having positive real parts. Therefore, it can be shown that $-L^{-1}M$ has eigenvalues with negative-real parts, so it is a stable matrix. Thus, the asymptotic error is

$$\mathbf{e} = \lim_{t\to\infty} \mathbf{e}(t) = (M^{-1} - \hat{M}^{-1})\boldsymbol{\varphi}^{\text{ext}} \tag{3.20}$$

where we have used the facts that the exponential function of a stable matrix goes to zero as time goes to infinity, and that

$$\int_0^\infty e^{At}\mathrm{d}t = -A^{-1} \tag{3.21}$$

for any stable invertible matrix A. The latter fact follows from the property of the exponential function

$$\frac{\mathrm{d}e^{At}}{\mathrm{d}t} = Ae^{At} \tag{3.22}$$

and the fundamental theorem of calculus.

3.3.3.2 Sensitivity of turning ratio deviations

Suppose that the deviations $\xi_{i,j}$ are small, and that we want to calculate the error resulting from small nudges in a single value,

$$\frac{\partial \mathbf{e}}{\partial \xi_{i,j}} = \frac{\partial}{\partial \xi_{i,j}}\left((M^{-1} - \hat{M}^{-1})\boldsymbol{\varphi}^{\text{ext}}\right) = -\left(\frac{\partial}{\partial \xi_{i,j}}\hat{M}^{-1}\right)\boldsymbol{\varphi}^{\text{ext}} \tag{3.23}$$

Consider the following theorem.

Theorem 3.2
([MN19]) Let $A(t)$ be an invertible matrix that depends on a scalar parameter t. Then,

$$\frac{\mathrm{d}A^{-1}}{\mathrm{d}t} = -A^{-1}\frac{\mathrm{d}A}{\mathrm{d}t}A^{-1} \tag{3.24}$$

Using Theorem 3.2, (3.23) can be written as

$$\frac{\partial \mathbf{e}}{\partial \xi_{i,j}} = \hat{M}^{-1}\frac{\partial \Xi^\top}{\partial \xi_{i,j}} V \hat{M}^{-1}\boldsymbol{\varphi}^{\text{ext}} \tag{3.25}$$

Assume that the TR deviations are independent from each other. Thus,

$$\frac{\partial \Xi^\top}{\partial \xi_{i,j}} = \mathbf{u}_j \mathbf{u}_i^\top \tag{3.26}$$

where $\mathbf{u}_i$ is the i-th column of the identity matrix of suitable dimensions. Hence,

$$\frac{\partial \mathbf{e}}{\partial \xi_{i,j}} = \hat{M}^{-1}\mathbf{u}_j \mathbf{u}_i^\top V \hat{M}^{-1}\boldsymbol{\varphi}^{\text{ext}} \tag{3.27}$$

3.3.3.3 Selection of intersections

Consider an intersection $n \in \mathcal{N}$ with incoming roads $\mathcal{I}(n)$ and outgoing roads $\mathcal{O}(n)$. Let $\boldsymbol{\xi}_i = \{\xi_{i,j}\}_{j\in\mathcal{O}(n)}$ for each $i \in \mathcal{I}(n)$. To quantify the effect of the perturbations $\xi_{i,j}$ on the error $\mathbf{e}$ we propose the following procedure:

1. Calculate the Jacobian matrix
$$J_i = \frac{\partial \mathbf{e}}{\partial \boldsymbol{\xi}_i} \tag{3.28}$$
for each incoming road $i \in \mathcal{I}(n)$.

2. Calculate the error energy due to perturbations in the TRs of i using the Frobenius norm of the Jacobian,
$$||J_i||_F^2 = \sum_{k\in\mathcal{E}}\sum_{j\in\mathcal{O}(n)}\left(\frac{\partial e_k}{\partial \xi_{i,j}}\right)^2. \tag{3.29}$$

3. Calculate the intersection weight as the total error energy due to its incoming roads,
$$w_n = \sum_{i\in\mathcal{I}(n)} ||J_i||_F^2 \tag{3.30}$$

4. Locate the available sensors in intersections with the highest values of w_n.

Note that the intersection weight can be simplified to

$$w_n = \sum_{i\in\mathcal{I}(n)}\sum_{j\in\mathcal{O}(n)}\sum_{k\in\mathcal{E}}\left(\sum_{p\in\mathcal{E}}\hat{M}^{-1}_{k,j}\hat{M}^{-1}_{i,p}v_i\varphi_p^{\text{ext}}\right)^2 \tag{3.31}$$

This heuristic procedure is a greedy approach as it assumes that deviations in the TR values are independent from each other.

Once the selected intersections are equipped with sensors, the corresponding elements of the TR matrix can be adjusted with the measured values. One type of sensor capable of measuring this parameter are Bluetooth (BT) readers. The reader detects the presence of a vehicle equipped with a BT device, and reports a count. Counts are matched between incoming and outgoing roads to calculate the percentages of turns during a time interval. As the ratio of vehicles equipped with BT is unknown, these sensors cannot provide absolute flow measurements.

Let $\mathcal{B}$ be the set of intersections with BT readers. These sensors can provide data dynamically, such that the estimated matrix $\hat{R}$ is time dependent. Suppose that ratio data is received at times $\{t_0^R\ ,\ t_1^R\ ,\ t_2^R\ ,\ \dots\}$. Denote $r_{i,j}^{\mathcal{B}}[k]$ as the TR value reported by the sensors using vehicles counts between times t_{k-1}^R and t_k^R. The corresponding matrix of measured TRs is

$$\hat{R}_{i,j}[k] = \begin{cases} r_{i,j}^{\mathcal{B}}[k] & \text{if} \quad \exists n \in \mathcal{B} : i \in \mathcal{I}(n)\ ,\ j \in \mathcal{O}(n) \\ \\ \hat{r}_{i,j} & \text{else} \end{cases} \tag{3.32}$$

3.4 Estimator implementation

To be able to use the estimator described in Section 3.2 in practical applications, it is required to provide a discrete time version of the dynamic equation. Furthermore, as the sampling times for the flow, speed and TR information might be different, we propose an algorithm able to incorporate these measurements to estimate the traffic state.

Define the discrete time density $\boldsymbol{\rho}[k]$ such that it a sampled version of the density at times $t = k\Delta t$. Assuming that Δt is small enough, (3.7) can be discretized as

$$\hat{\boldsymbol{\rho}}[k+1] = \hat{\boldsymbol{\rho}}[k] + \Delta t \left(A[k]\hat{\boldsymbol{\rho}}[k] + L^{-1}\boldsymbol{\varphi}^{\text{mea}}[k] \right) \tag{3.33}$$

where

$$A[k] = L^{-1}(\hat{R}^{\top}[k] - \mathbb{I})\hat{V}[k] \tag{3.34}$$

The discretization step Δt must be chosen such that the estimator is numerically stable. The Courant-Friedrichs-Lewy criterion establishes that a necessary condition for the discretization step is $\Delta t < \min_{\text{i}}(\ell_i / v_i^{max})$, that is, the discretization time is less than the minimum traversal time for all roads in the network. Algorithm 3.1 allows to use the discrete-time estimator with input data provided at arbitrary discrete times.

Algorithm 3.1. *Density estimation*

Inputs:

- *Initial density condition* $\boldsymbol{\rho}[0]$.

- *Discretization time* Δt.
- *Graph* $\mathcal{G}$ *and road lengths* L.
- *Sensor data:*
  - *Aggregate speeds: Update times* t_i^v *for* $i \in \{1, 2, \dots, i^{max}\}$ *and data* $\hat{V}[i]$.
  - *Turning ratios: Update times* t_j^R *for* $j \in \{1, 2, \dots, j^{max}\}$ *and data* $\hat{R}[j]$.
  - *External inflows: Update times* t_m^φ *for* $m \in \{1, 2, \dots, m^{max}\}$ *and data* $\boldsymbol{\varphi}^{\text{mea}}[m]$.

Outputs:

- *Discrete density vector* $\boldsymbol{\rho}[k]$ *for* $k \in \{0, 1, 2, \dots, k^{max}\}$, *where*

$$k^{max} = \lceil \min(t^v_{i^{max}}, t^R_{j^{max}}, t^\varphi_{m^{max}})/\Delta t \rceil.$$

Steps:

1. *Initialize* $(i, j, k, m, T) \leftarrow (1, 1, 0, 1, 0)$.
2. *While* $(i \leq i^{max}) \ \wedge \ (j \leq j^{max}) \ \wedge \ (m \leq m^{max})$
   - 2.1. $T \leftarrow \min(t_i^v, t_j^R, t_m^\varphi)$
   - 2.2. $n \leftarrow \left\lceil \dfrac{T}{\Delta t} \right\rceil$
   - 2.3. *While* $k < n$
     - 2.3.1. $V \leftarrow \hat{V}[i]$
     - 2.3.2. $R \leftarrow \hat{R}^{\mathcal{B}}[j]$
     - 2.3.3. $A \leftarrow L^{-1}(R^\top - \mathbb{I})V$
     - 2.3.4. $\boldsymbol{\rho}[k+1] \leftarrow (\mathbb{I} + \Delta t A)\boldsymbol{\rho}[k] + \Delta t L^{-1}\boldsymbol{\varphi}^{\text{mea}}[m]$
     - 2.3.5. $k \leftarrow k + 1$
   - 2.4. *if* $T \geq t_i^v$ *then* $i \leftarrow i + 1$
   - 2.5. *if* $T \geq t_j^R$ *then* $j \leftarrow j + 1$
   - 2.6. *if* $T \geq t_m^\varphi$ *then* $m \leftarrow m + 1$
3. *Return* $\boldsymbol{\rho}[k]$ $\forall k$.

To deal with the variable input-data times, Algorithm 3.1 uses the sample-and-hold scheme to update the density estimates.

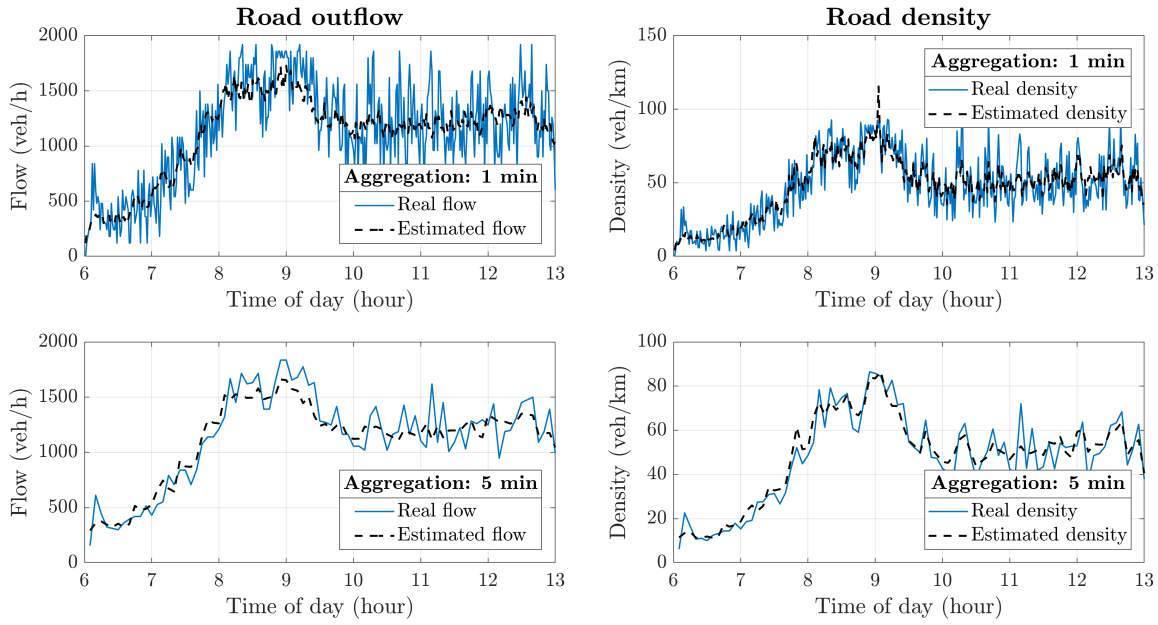


Figure 3.1: Time series for the trajectories of the real and estimated density and flow for one road using different aggregation times. **Top row** uses a window of 1 minute, whereas the **bottom row** uses one of 5 minutes.

3.5 Simulation and validation

To evaluate the performance of the estimator, we considered a zone of downtown Grenoble, France. This corresponds to the same case as shown in Chapter 2 in Fig. 2.7. The ground truth states are obtained via simulated traces of individual vehicles using the well-known traffic microsimulator Aimsun. As input data for Aimsun, we used the TRs as described in Appendix B. As input demands, an arbitrary but realistic time-varying profile was applied to each boundary inflow. As simulation outputs, the position and speed of each vehicle were collected at each time step. From this raw data, the average road speeds and boundary flows are calculated by (3.13) and (3.16). For validation, the ground-truth density for road i is calculated as

$$\rho_i[k] = \frac{1}{\ell_i \tau} \int_{(k-1)\tau}^{k\tau} |\mathcal{V}_i(t)| \mathrm{d}t \tag{3.35}$$

where $\mathcal{V}_i(t)$ is the set of vehicle indexes contained in road i at time t, and τ is the duration of a time window that represents the measurement aggregation time.

3.5.1 Turning ratios known everywhere

In first a simulation scenario, we consider the case where all TRs are known, $\hat{R} = R$. Figure 3.1 shows the evolution of the real and estimated densities for one particular road of the network when using aggregation times of 1 and 5 minutes. Note that for both cases the estimates are

close to the real values. Nevertheless, the real values present a much higher dispersion, which is increases as the aggregation time decreases. This dispersion is due to the stochastic behavior of individual vehicles that are not captured by the macroscopic model. As the aggregation time increases, the individual vehicle decisions are smoothed out, so the estimation error decreases. To quantify the error we consider the following error metrics: *a)* the Mean Error (ME) and Relative Mean Error (RME) defined as

$$\mathrm{ME}_i^{\boldsymbol{\rho}} = \frac{1}{T}\left|\sum_{k=1}^{T}(\rho_i[k] - \hat{\rho}_i[k])\right| \tag{3.36}$$

$$\mathrm{RME}_i^{\boldsymbol{\rho}} = \frac{\mathrm{ME}_i^{\boldsymbol{\rho}}}{\frac{1}{T}\sum_{k=1}^{T}\rho_i[k]} \tag{3.37}$$

where T is the duration of the simulation, which measure the deviation between the mean trajectories of the real and estimated quantities; and *b)* the Absolute Error (AE) and Relative Absolute Error (RAE), defined as

$$\mathrm{AE}_i^{\boldsymbol{\rho}} = \frac{1}{T}\sum_{k=1}^{T}|\rho_i[k] - \hat{\rho}_i[k]| \tag{3.38}$$

$$\mathrm{RAE}_i^{\boldsymbol{\rho}} = \frac{\mathrm{AE}_i^{\boldsymbol{\rho}}}{\frac{1}{T}\sum_{k=1}^{T}\rho_i[k]}, \tag{3.39}$$

which measure the total error between the two trajectories, as it not only consider the difference between the means, but also the high frequency variations of one signal with respect to the other.

Figure 3.2 shows the error Cumulative Distribution Function (CDF) for the relative mean and absolute errors for density. That is, CDF(x) is the percentage of roads that have an error less than or equal to x. The error is presented for three scenarios using varying aggregation times: 1,5, and 10 minutes. Note that for the RME, the three aggregation times exhibit the same behavior, showing that, overall, the mean trajectory of the vehicles is correctly estimated: 90% of the roads present an error lesser than 7%. For the RAE, it is noted that longer aggregation times present a reduced error, as the dispersion of the ground truth values due to the behavior of individual vehicles is smoothed out. As is to be expected, the RAE is greater than the RME, as 90% of the roads present now an error lesser than 18% for the case of an aggregation time of 10 minutes, 25% for the case of 5 minutes, and 52% for the case of 1 minute.

3.5.2 Simulation results with few turning ratio measurements

In the second simulated scenario, a uniform disturbance between -10% and 10% was added to the estimates for the TRs. The random values were used as input to the AIMSUN simulation,

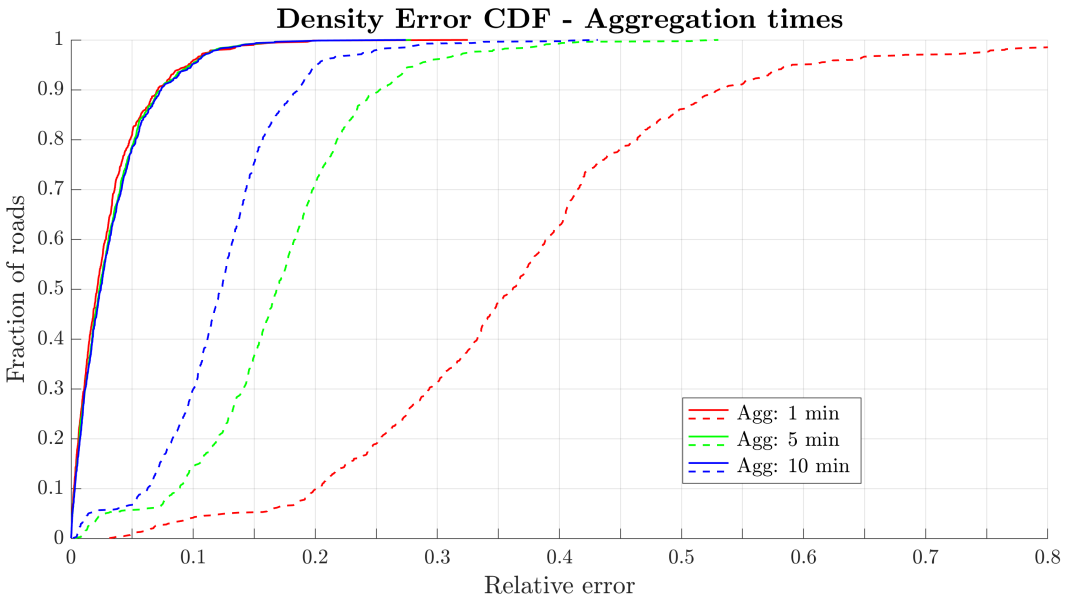


Figure 3.2: Cumulative distribution of the relative mean and absolute errors for density estimation, using different aggregation times. Solid lines correspond to the RME, whereas dashed lines correspond to the RAE.

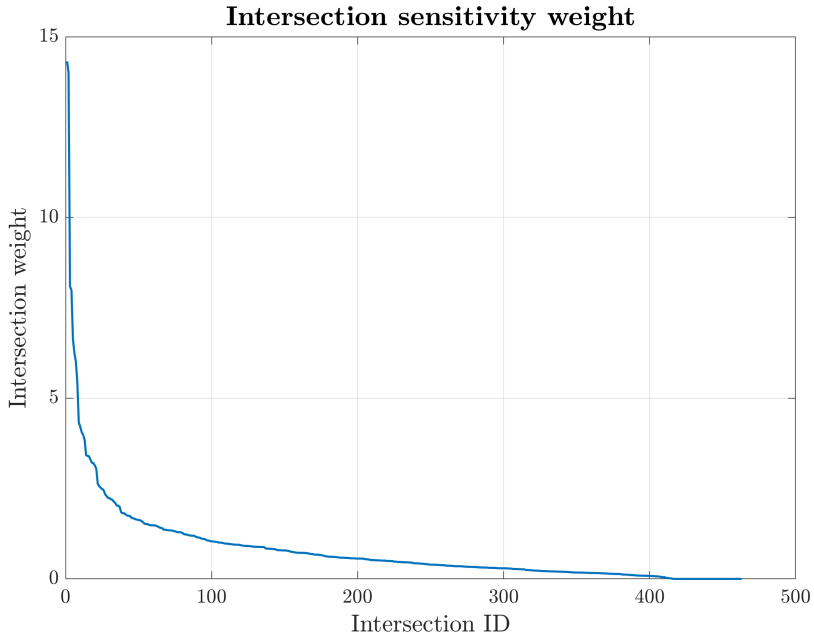


Figure 3.3: Sensitivity weight of each intersection: effect in the estimation error by small perturbations in the TRs. Intersection identifiers were sorted according to the weight.

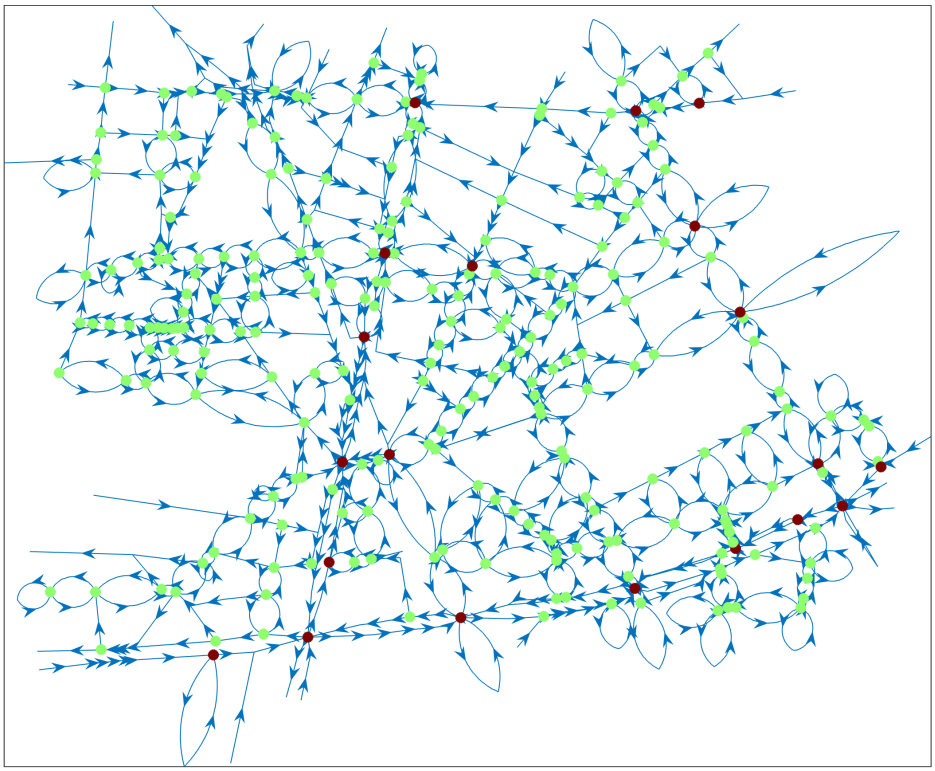


Figure 3.4: Location of TR sensors. Nodes in dark-red are the measured nodes.

but they were unknown for the estimation algorithm. To include TR measurements, the method detailed in Section 3.3.3 was used to measure the sensitivity of each intersection. Figure 3.3 shows the distribution of the intersection sensitivity weight using the initial TR estimates and the maximum demands measured at the boundaries. Note that the sensitivity weight decreases very quickly, so it is expected that diminishing returns will be observed as more sensors are located. Thus, a few number of intersections could provide a value trade-off. As an illustrating example, the 20 intersections with the highest weight were selected such that the TRs are measured directly. This corresponds to approximately 4% of the total number of network's intersections. Fig. 3.4 shows the location of the TR sensors.

The outputs of the simulation were obtained using an aggregation time of 10 minutes. For comparison purposes, the results are also shown for the case where no sensors are located, and where all TRs are known. These correspond to the worst and best case scenarios, respectively. Fig. 3.5 shows the resulting density error CDF. As expected, the case that measures all intersections out-performs the other methods, especially in the estimation of the mean trajectory, as seen in the ME metric. However, this case is generally unfeasible due to practical limitations. Similarly, the no measurements case shows the highest errors and provides an upper-bound to the error. The error obtained using the proposed sensor location scheme (using the intersection weight) presents a significant improvement when compared to the base case with no sensors, as for a given number of roads, the RME increases in about 4%.

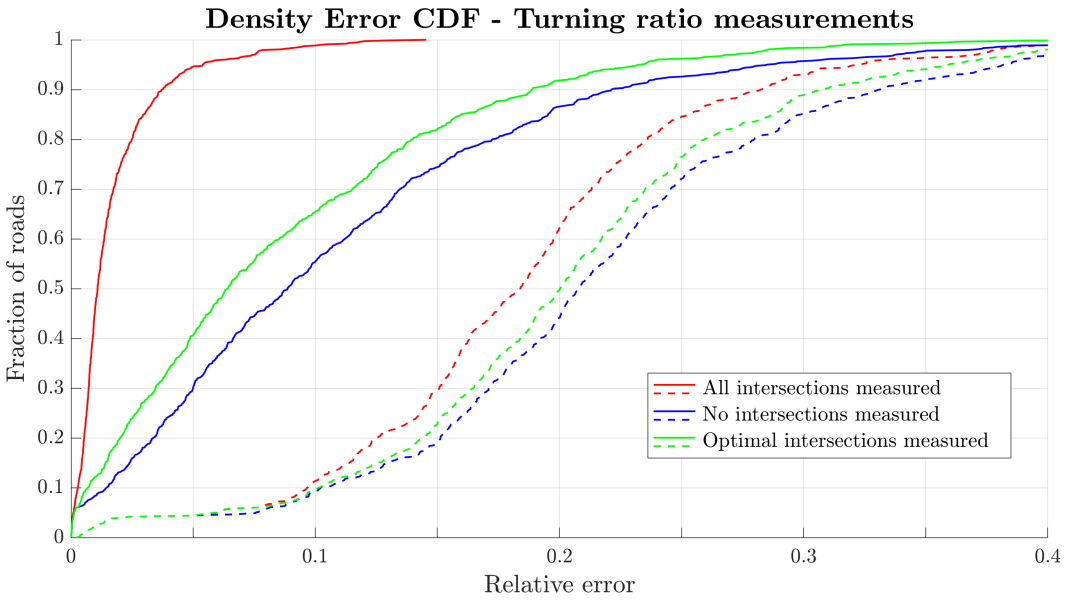


Figure 3.5: Relative error for the density estimation using time varying TRs. Error is reported when measuring all intersections, no intersections, locations using the sensitivity method, random selection of intersections.

3.6 Concluding remarks

In this Chapter, we proposed a method to estimate the dynamical evolution of flow and density in large urban traffic network by using heterogeneous data sources: counting sensors, vehicle identifier sensors, and floating car data. We discussed the implementation of these types of data in real case scenarios such as varying sampling rates and time intervals, and how to take them into account in the estimator implementation.

The method requires little parameters to be known, as there is no explicit modeling of the road outflows in terms of the current density, so no fundamental diagram is required. The estimation relies in a data-based approach based on the FCD. However, the knowledge of the TRs for every intersection is required. To simplify this requirement, we introduce a TR sensor location scheme that takes as input a priori values for these parameters to identify the intersections for which small perturbations in the TR values would generate the highest error.

We tested the proposed methods using microsimulations over a selection of the city of Grenoble. The results show that the estimator performs well, as it is able to estimate the overall trajectory of the traffic states, with many roads presenting little errors. The proposed methods are flexible, as they can be applied for different budgets. In addition, the algorithm allows the inclusion of new data, so with time, the previous information can be used to improve the location of TR sensors to increase the performance of the estimator.

Chapter 4

Average density estimation

Contents

4.1 Overview

In this chapter, we consider the estimation of the average of all road densities in a traffic network. We propose the the use of a reduced-order open-loop observer, and we show that it is possible to find a virtual representation of the network using inhomogeneous road divisions such that the proposed observer converges to the average density of the virtual system.

4.2 Model

Consider the road-graph representation of a traffic network $\mathcal{G} = \{\mathcal{E}, \mathcal{T}, R\}$, where the nodes $\mathcal{E} = \{1, 2, \ldots, p\}$ represent sections of roads, the edges $\mathcal{T} \subset \mathcal{E} \times \mathcal{E}$ represent the possible vehicle transfers (turns) between roads[1], and $R \in \mathbb{R}^{p \times p}$ is the adjacency matrix (TR matrix) whose

[1]Note that this graph differs from those used in Chapters 2 and 3 where the nodes were intersections and the edges roads. The definition used in this Chapter is also referred to as the line-graph or the road-graph.

elements $r_{i,j}$ are the TRs. In addition to $\mathcal{G}$, we include a set of parameters for each road representing physical properties, e.g., a vector of road lengths $\boldsymbol{\ell} \in \mathbb{R}^p$.

As presented in Chapter 3, we consider as state variable the vector of road densities $\boldsymbol{\rho}(t) \in \mathbb{R}^p$. The dynamics of this vector where found to be given by

$$\dot{\boldsymbol{\rho}}(t) = L^{-1}(R^\top - \mathbb{I})\boldsymbol{\varphi}_{out}(t) + L^{-1}B\mathbf{u}(t). \tag{4.1}$$

where $L = \text{diag}(\boldsymbol{\ell})$, $\boldsymbol{\varphi}^{\text{out}}(t)$ is the flow at the downstream end of each road, $B \in \{0,1\}^{p\times q}$ is a selection matrix that indicates the location of boundary inflows, and $\mathbf{u}(t) \in \mathbb{R}^q$ contains the input demands. Assuming that vehicle density and speed do not vary significantly in space, it follows

$$\dot{\boldsymbol{\rho}}(t) = L^{-1}(R^\top - \mathbb{I})V(t)\boldsymbol{\rho}(t) + L^{-1}B\mathbf{u}(t) \tag{4.2}$$

where $V(t) = \text{diag}(\mathbf{v}(t))$ is a matrix containing the space-mean speed of each road. However, depending in the relationship between road density $\boldsymbol{\rho}(t)$ and speed $\mathbf{v}(t)$, this model definition is generally non-linear, or at least, time-dependent. In this Chapter, we will examine the convergence of a one-dimensional estimator for the average density of the whole network. However, due to the limitations of the available tools to analyze the observability and detectability of non-linear systems, we are restricted to the case of linear systems. This is the case when the speed of each road is independent from density and constant in time, $\mathbf{v}(t) = \mathbf{v}$, such that

$$\dot{\boldsymbol{\rho}}(t) = A\boldsymbol{\rho}(t) + L^{-1}B\mathbf{u}(t) \tag{4.3}$$

where $A = L^{-1}(R^\top - \mathbb{I})V$. For instance, the case when all roads are in free-flow, $\boldsymbol{\rho} \leq \boldsymbol{\rho}_c$, satisfies the required condition, but in the congested regime, the road outflow decreases as the density increases, implying the average vehicle speed decreases as well.

Despite this limitation, Section 4.6 evaluates the performance of the estimator under congested scenarios. It was found that for some cases, even multiple congestions in a considered region still provide an error under 10%. Nevertheless, the extension of this work to consider more general cases with congestion is a work in progress.

4.2.1 Average density dynamics

Suppose that sensors are located in a set of nodes $\mathcal{S} \subset \mathcal{N}$ corresponding to the boundaries (inflows and outflows) of the network. Without loss of generality, we index roads such that measured roads have the highest indexes, i.e., $\mathcal{S} = \{p-s+1, \ldots, p\}$ with $q < s < p$. Thus, $\mathbf{y}(t) = C\boldsymbol{\rho}(t)$ where $C = [\mathbf{0}_{s\times m} \quad \mathbb{I}_s]$, and $m = p - s$ is the number of unmeasured nodes.

Consider a partition of the state vector as $\boldsymbol{\rho}(t) = [\boldsymbol{\rho}_1^\top(t) \quad \boldsymbol{\rho}_2^\top(t)]^\top$ such that $\boldsymbol{\rho}_1(t) \in \mathbb{R}^m$ corresponds to the states of the unmeasured nodes (also called internal nodes), and $\boldsymbol{\rho}_2(t) \in \mathbb{R}^s$ to the states of the measured nodes. Note that $\boldsymbol{\rho}_2(t) = \mathbf{y}(t)$. The system matrices are partitioned accordingly,

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}. \tag{4.4}$$

where $A_{11} \in \mathbb{R}^{m\times m}$, $A_{12} \in \mathbb{R}^{m\times s}$, $A_{21} \in \mathbb{R}^{s\times m}$, $A_{22} \in \mathbb{R}^{s\times s}$, and $B_1 \in \mathbb{R}^{m\times q}$ and $B_2 \in \mathbb{R}^{s\times q}$. Analogously, let

$$R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix},\ L = \begin{bmatrix} L_1 & \mathbf{0} \\ \mathbf{0} & L_2 \end{bmatrix},\ V = \begin{bmatrix} V_1 & \mathbf{0} \\ \mathbf{0} & V_2 \end{bmatrix}. \tag{4.5}$$

We aim to estimate the average of the unmeasured states, i.e., $\rho_{av}(t) = \frac{1}{m}\mathbf{1}^\top \boldsymbol{\rho}_1(t)$ without requiring knowledge about the full vector $\boldsymbol{\rho}_1(t)$. Consider a lower-dimensional projected system in which the unmeasured states are aggregated. The average state follows

$$\dot{\rho}_{av}(t) = \frac{1}{m}\mathbf{1}^\top A_{11}\mathbf{1}\rho_{av}(t) + \frac{1}{m}\mathbf{1}^\top A_{12}\boldsymbol{\rho}_2(t) + \frac{1}{m}\mathbf{1}^\top A_{11}\boldsymbol{\sigma}(t) \tag{4.6}$$

where $\boldsymbol{\sigma}(t)$ is the average deviation vector given by $\boldsymbol{\sigma}(t) = \boldsymbol{\rho}_1(t) - \mathbf{1}\rho_{av}(t)$.

4.2.2 Average detectability

Consider the open-loop observer

$$\dot{\hat{\rho}}_{av}(t) = \frac{1}{m}[\mathbf{1}^\top A_{11}\mathbf{1}\hat{\rho}_{av}(t) + \mathbf{1}^\top A_{12}\mathbf{y}(t)] \tag{4.7}$$

which is obtained by following the known dynamics (4.6) ignoring the unknown input $\boldsymbol{\sigma}(t)$.

Definition 4.1 (Average detectability)**.** A system is called average detectable if the open-loop observer (4.7) converges asymptotically, i.e., $\hat{\rho}_{av}(t) \to \rho_{av}(t)$ as $t \to \infty$.

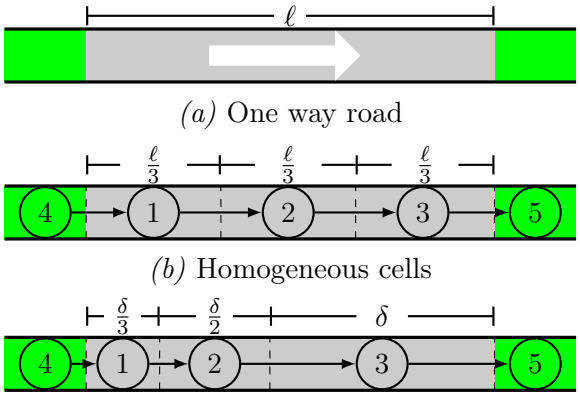
Note that average detectability is a particular case of functional observability, as described in [FHJ10]. Theorem 4.1 states the conditions that a network must satisfy to be average detectable.

Theorem 4.1 ([NWK19])
For systems of the form (4.6)*, the open-loop observer* (4.7) *converges if and only if* $\mathbf{1}^\top A_{11} = -\gamma\mathbf{1}^\top$ *with* $\gamma > 0$.

4.3 Virtual network definition

Theorem 4.1 imposes strict conditions which are not generally satisfied for traffic networks. Nevertheless, we will show that by dividing each road into virtual cells, it is possible to construct a virtual network which is average detectable. We first show some motivating examples which illustrate the need for these virtual partitions in the average detectability problem. Then, we present how these divisions are defined for general networks.

Example 4.1. One way road



(a) One way road

(b) Homogeneous cells

(c) Inhomogeneous cells

Figure 4.1: One way road. Green strips represent sensors located and the upstream and downstream ends. Two different virtual divisions are shown.

Consider a one way road as shown in Fig. 4.1.a. Sensors are located at the upstream and downstream boundaries of the road, represented by green strips in the figure. Let ℓ be the length of road between the sensors, and v be the maximum velocity. We divide this stretch into 3 virtual sections (cells), such that the sum of their lengths is ℓ, and all of them have maximum velocity v. Possible divisions are shown in Figs. 4.1.b and 4.1.c.

First, consider the common approach of considering homogeneous cells as in Fig. 4.1.b, such that cells 1-3 have each length $\ell/3$. The corresponding state matrix is,

$$A = \left[\begin{array}{ccc|cc} -3\ell^{-1} & 0 & 0 & 3\ell^{-1} & 0 \\ 3\ell^{-1} & -3\ell^{-1} & 0 & 0 & 0 \\ 0 & 3\ell^{-1} & -3\ell^{-1} & 0 & 0 \\ \hline 0 & 0 & 0 & -\ell_4^{-1} & 0 \\ 0 & 0 & \ell_5^{-1} & 0 & -\ell_5^{-1} \end{array}\right] v.$$

where ℓ_4 is the length of entry and ℓ_5 is the length of the exit. It can be seen that $\mathbf{1}^\top A_{11} = [0 \quad 0 \quad -3\ell^{-1}]v$. Thus, the condition $\mathbf{1}^\top A_{11} = -\gamma \mathbf{1}^\top$ from Theorem 4.1 is not satisfied, and thus, equal length divisions are not average detectable.

This results seems counterintuitive, as it is known that for one ways roads such as highways, measuring the density of the downstream cell is enough to make the entire system observable and the density of each cell can be known, [BLRP16]. However, we are interested in estimating the average density of all cells directly, without the need to calculate each individual density. Therefore, full observability does not imply average detectability.

Now, consider a division such that cell 3 has length δ, cell 2 has length $\delta/2$ and cell 1 has

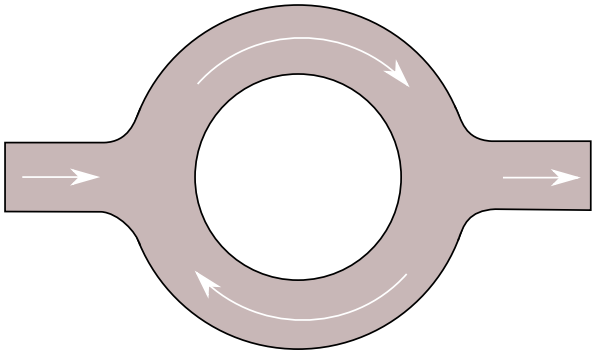


Figure 4.2: Circle road with one entry and one exit.

length $\delta/3$, where $\delta = \frac{6}{11}\ell$ (see Fig. 4.1.c). The corresponding state matrix is

$$A = \left[\begin{array}{ccc|cc} -3\delta^{-1} & 0 & 0 & 3\delta^{-1} & 0 \\ 2\delta^{-1} & -2\delta^{-1} & 0 & 0 & 0 \\ 0 & \delta^{-1} & -\delta^{-1} & 0 & 0 \\ \hline 0 & 0 & 0 & -\ell_4^{-1} & 0 \\ 0 & 0 & \ell_5^{-1} & 0 & -\ell_5^{-1} \end{array}\right] v$$

and thus $\mathbf{1}^\top A_{11} = [-\delta^{-1} \quad -\delta^{-1} \quad -\delta^{-1}]v$. Note that all column sums are equal, and because of Theorem 4.1, this division is average detectable.

Example 4.2. Circle road: networks

Consider a ring road as shown in Fig. 4.2. Suppose that sensors are located at the entry and the exit. Consider the network representation in Fig. 4.3. The green nodes represent sensors in the network boundaries. For simplicity, we no longer index the nodes with sensors as they are not concerned with the average detectability conditions.

Denote by ℓ_1, ℓ_2, v_1, v_2 the lengths and speeds of the top and bottom sections of the circle, respectively. Suppose that both roads have equal speeds, and that units are such that $v_1 = v_2 = 1$. The state matrix of the unmeasured partition is

$$A_{11} = \left[\begin{array}{cc} -\ell_1^{-1} & \ell_1^{-1} \\ r\ell_2^{-1} & -\ell_2^{-1} \end{array}\right].$$

According to Theorem 4.1, to be able to estimate the average density this matrix must satisfy $\mathbf{1}^\top A_{11} = -\gamma \mathbf{1}^\top$, and therefore

$$\begin{array}{rcl} -\ell_1^{-1} + r\ell_2^{-1} & = & -\gamma \\ \ell_1^{-1} - \ell_2^{-1} & = & -\gamma \end{array} \quad \Rightarrow \quad \ell_1 = \frac{2}{r+1}\ell_2.$$

As $r < 1$, the roads cannot be of equal length. Thus, we are interested in finding a way to modify the network, such that physical parameters are conserved (i.e. lengths, velocities and turning ratios), but that the network is average detectable.

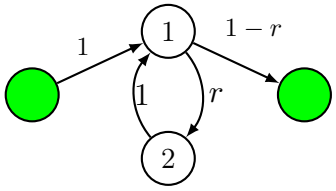


Figure 4.3: Network representation of a circle road.

Let the physical lengths of the roads 1 and 2 be $\ell_1 = \ell_2 = \ell$. Consider a new network where roads 1 and 2 are divided into n_1 and n_2 cells, respectively, as shown in Fig. 4.4. Let cells 1 to n_1 correspond to road 1, and cells n_1+1 to n_1+n_2 correspond to road 2. Furthermore, let δ_i be the length of the i-th cell. The dimension and elements of this network's state matrix, denoted by $A^{(n_1,n_2)}$, depend on the values of n_1, n_2 and the vector of cell lengths $\boldsymbol{\delta}$. The block matrix corresponding to the unmeasured states is

$$A_{11}^{(n_1,n_2)} = \begin{bmatrix} -\delta_1^{-1} & \delta_1^{-1} & 0 & \cdots & 0 & 0 & 0 \\ 0 & -\delta_2^{-1} & \delta_2^{-1} & \cdots & 0 & 0 & 0 \\ \vdots & & \ddots & & & & \vdots \\ 0 & \cdots & & -\delta_{n_1}^{-1} & \delta_{n_1}^{-1} & \cdots & 0 \\ 0 & \cdots & & 0 & -\delta_{n_1+1}^{-1} & \cdots & 0 \\ \vdots & & & & & \ddots & \vdots \\ r\delta_{n_1+n_2}^{-1} & 0 & \cdots & 0 & 0 & \cdots & -\delta_{n_1+n_2}^{-1} \end{bmatrix}.$$

The average detectability condition requires $-\delta_1^{-1} + r\delta_{n_1+n_2}^{-1} = -\gamma$ and $-\delta_i^{-1} + \delta_{i-1}^{-1} = -\gamma$ for $i = 2, 3, \ldots, n_1 + n_2$. Using these equations, we can calculate section lengths as

$$\delta_i = \frac{1}{\left(i + \dfrac{r}{1-r}(n_1+n_2)\right)\gamma}. \tag{4.8}$$

The specific values of n_1 and n_2 must be such that the physical parameters of the network are conserved, this is,

$$\ell = \sum_{i=1}^{n_1} \delta_i = \sum_{i=1}^{n_2} \delta_{n_1+i}. \tag{4.9}$$

By substituting (4.8) into (4.9), we obtain

$$\sum_{i=1}^{n_1} \frac{1}{i + \dfrac{r}{1-r}(n_1+n_2)} = \sum_{i=1}^{n_2} \frac{1}{i + \dfrac{1}{1-r}(n_1+rn_2)}.$$

The values of n_1 and n_2 that satisfy this equation yield a network partition that is average detectable. Note that as the summands on both side of the equation are different, then it must be $n_1 \neq n_2$.

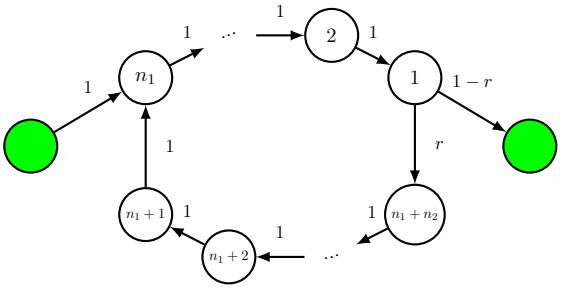


Figure 4.4: Network of a circle road with virtual partitions. Road 1 is divided into n_1 cells, and road 2 into n_2 cells.

From the previous examples it follows that average-detectable virtual networks, i.e. networks constructed using virtual cells, require inhomogeneous divisions of the original roads.

In what follows, we will consider traffic networks as a directed graph with added parameters $\boldsymbol{\ell}$, $\mathbf{v}$. Thus, $\mathcal{G} = \{\mathcal{N}, \mathcal{E}, R, \boldsymbol{\ell}, \mathbf{v}\}$. Additionally, consider a vector $\mathbf{n} \in \mathbb{N}^m$ which specifies the number of virtual cells for each road. For an arbitrary traffic network $\mathcal{G}$ and division vector $\mathbf{n}$ we define the following concepts.

Definition 4.2 (Road division)**.** Consider an arbitrary road $i \in \mathcal{N}$, and n_i the corresponding element of $\mathbf{n}$. A division of road i is a directed path graph whose nodes $\{i^{(1)}, i^{(2)}, \ldots, i^{(n_i)}\}$ are virtual cells of i. The downstream cell is denoted $i^{(1)}$, whereas $i^{(n_i)}$ denotes the upstream cell. Additionally, the length and velocity of the k-th cell of road i are denoted by $\delta_i^{(k)}$ and $v_i^{(k)}$, respectively.

Definition 4.3 (Virtual network)**.** A network $\mathcal{G}^{(\mathbf{n})} = \{\mathcal{N}^{(\mathbf{n})}, \mathcal{E}^{(\mathbf{n})}, R^{(\mathbf{n})}, \boldsymbol{\delta}^{(\mathbf{n})}, \mathbf{v}^{(\mathbf{n})}\}$ is called a virtual network of $\mathcal{G}$ according to $\mathbf{n}$ if its nodes $\mathcal{N}^{(\mathbf{n})}$ correspond to the road divisions of the nodes $\mathcal{N}$, and the adjacency matrix $R^{(\mathbf{n})}$ satisfies $r^{(\mathbf{n})}_{i^{(1)},j^{(n_j)}} = r_{i,j}$ and $r^{(\mathbf{n})}_{i^{(j)},i^{(k)}} = 1$ if $k = j - 1$. Additionally,

$$\boldsymbol{\delta}^{(\mathbf{n})} = [\delta_1^{(1)} \cdots \ \delta_1^{(n_1)} \ \delta_2^{(1)} \cdots \ \delta_2^{(n_2)} \ \cdots \ \delta_m^{(1)} \cdots \ \delta_m^{(n_m)}]^\top$$

and

$$\mathbf{v}^{(\mathbf{n})} = [v_1^{(1)} \cdots \ v_1^{(n_1)} \ v_2^{(1)} \cdots \ v_2^{(n_2)} \ \cdots \ v_m^{(1)} \cdots \ v_m^{(n_m)}]^\top.$$

Definition 4.4 (Admissible virtual network)**.** A virtual network $\mathcal{G}^{(\mathbf{n})}$ of $\mathcal{G}$ is said to be admissible if for every road $i \in \mathcal{N}$, the velocity of any cell is equal to the velocity of the road,

$$v_i^{(k)} = v_i, \tag{4.10}$$

and the sum of cell lengths is equal to the length of the road,

$$\ell_i = \sum_{k=1}^{n_i} \delta_i^{(k)}. \tag{4.11}$$

Given a traffic network $\mathcal{G}$, many possible admissible virtual networks $\mathcal{G}^{(\mathbf{n})}$ can be constructed. Let $\rho_i^{(k)}$ denote the density of cell k of road i, and let $\boldsymbol{\rho}^{(\mathbf{n})}$ be the density vector of

dimension $\mathbf{1}^\top \mathbf{n}$ for the virtual network. Its dynamics are given by

$$\dot{\boldsymbol{\rho}}^{(\mathbf{n})}(t) = A^{(\mathbf{n})}\boldsymbol{\rho}(t) + \mathrm{diag}(\boldsymbol{\delta}^{(\mathbf{n})})^{-1}B^{(\mathbf{n})}\mathbf{u}(t) \tag{4.12}$$

where

$$A^{(\mathbf{n})} = \mathrm{diag}(\boldsymbol{\delta}^{(\mathbf{n})})^{-1}(R^{(\mathbf{n})\top} - \mathbb{I})\mathrm{diag}(\mathbf{v}^{(\mathbf{n})}). \tag{4.13}$$

and $B^{(\mathbf{n})}$ maps the input demands to the upstream cell of the boundary inflows.

Following the same discussion as in sections 4.2.1 and 4.2.2, consider the state partition $\boldsymbol{\rho}^{(\mathbf{n})} = [\boldsymbol{\rho}_1^{(\mathbf{n})\top}\ ,\ \boldsymbol{\rho}_2^{(\mathbf{n})\top}]^\top$. From Theorem 4.1, and as shown in the example, the properties of the measured roads do not affect the average detectability of the system, thus we let the measured nodes to have no partitions so $\boldsymbol{\rho}_2^{(\mathbf{n})} = \boldsymbol{\rho}_2$. Define the average density of the new system as

$$\rho_{av}^{(\mathbf{n})} = \frac{1}{n_{tot}}\boldsymbol{\rho}_2^{(\mathbf{n})} \tag{4.14}$$

where $n_{tot} = \mathbf{1}^\top \mathbf{n}$. Its dynamics are given by

$$\frac{\mathrm{d}}{\mathrm{d}t}\rho_{av}^{(\mathbf{n})}(t) = \frac{1}{n_{tot}}\mathbf{1}^\top A_{11}^{(\mathbf{n})}\mathbf{1}\rho_{av}^{(\mathbf{n})}(t) + \frac{1}{n_{tot}}\mathbf{1}^\top A_{12}^{(\mathbf{n})}\boldsymbol{\rho}_2(t) + \frac{1}{n_{tot}}\mathbf{1}^\top A_{11}^{(\mathbf{n})}\boldsymbol{\sigma}^{(\mathbf{n})}(t) \tag{4.15}$$

and the open-loop observer is given by

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\rho}_{av}^{(\mathbf{n})}(t) = \frac{1}{n_{tot}}\left[\mathbf{1}^\top A_{11}^{(\mathbf{n})}\mathbf{1}\hat{\rho}_{av}^{(\mathbf{n})}(t) + \mathbf{1}^\top A_{12}^{(\mathbf{n})}\mathbf{y}(t)\right] \tag{4.16}$$

The convergence of this observer thus depends on the construction of an admissible virtual graph that satisfies Theorem 4.1.

In the following section we will treat the problem of, for any given traffic network $\mathcal{G}$, finding a division vector $\mathbf{n}$, cell length vector $\boldsymbol{\delta}^{(\mathbf{n})}$, and constant $\gamma > 0$, such that the virtual network $\mathcal{G}^{(\mathbf{n})}$ is admissible and average detectable.

4.4 Estimation approach

The following theorem presents the conditions required for a virtual network to be admissible and average detectable.

Theorem 4.2
Let $\mathcal{G} = \{\mathcal{N}, \mathcal{E}, R, \boldsymbol{\ell}, \mathbf{v}\}$ be a given traffic network. An admissible network $\mathcal{G}^{(\mathbf{n})} = \{\mathcal{N}^{(\mathbf{n})}, \mathcal{E}^{(\mathbf{n})}, R^{(\mathbf{n})}, \boldsymbol{\delta}^{(\mathbf{n})}, \mathbf{v}^{(\mathbf{n})}\}$ is average detectable if and only if there exist $\mathbf{n} \in \mathbb{N}^m$, $\gamma > 0$, and $\boldsymbol{\delta}^{(\mathbf{n})}$ such that

$$\delta_i^{(k)} = \frac{v_i}{(v_i \mathbf{d}_i^\top \mathbf{n} + k)\gamma} \tag{4.17}$$

for all $i = 1, 2, \ldots, m$; $k = 1, 2, \ldots, n_i$; where $\mathbf{d}_i^\top$ is the i-th row of $D = (\mathbb{I} - R_{11})^{-1}R_{11}V_1^{-1}$.

Proof. Consider an arbitrary cell $i^{(k)}$, with $k \neq 1$ such that its downstream neighbor is cell $i^{(k-1)}$. The column sum of $A_{11}^{(\mathbf{n})}$ corresponding to this cell is

$$-\frac{v_i}{\delta_i^{(k)}} + \frac{v_i}{\delta_i^{(k-1)}} = -\gamma.$$

where we imposed the condition $v_i^{(k)} = v_i$. By induction, we can calculate the length of each cell from $\delta_i^{(1)}$,

$$\frac{1}{\delta_i^{(k)}} = \frac{1}{\delta_i^{(1)}} + \frac{k-1}{v_i}\gamma. \tag{4.18}$$

Cell $i^{(1)}$ has as out-neighbors all cells $j^{(n_j)}$ such that $(i,j) \in \mathcal{E}$. Thus, its corresponding column sum is,

$$-\frac{v_i}{\delta_i^{(1)}} + \sum_{j=1}^{m} \frac{r_{ij} v_i}{\delta_j^{(n_j)}} = -\gamma. \tag{4.19}$$

Define $\boldsymbol{\delta}_{(1)}^{-1} = [1/\delta_1^{(1)} \quad 1/\delta_2^{(1)} \quad \cdots \quad 1/\delta_m^{(1)}]$. By substituting (4.18) into (4.19), we obtain a system of linear equations,

$$(\mathbb{I} - R_{11})\boldsymbol{\delta}_{(1)}^{-1} = \gamma[R_{11}V_1^{-1}\mathbf{n} + (\mathbb{I} - R_{11})V_1^{-1}\mathbf{1}]. \tag{4.20}$$

Thus,

$$\frac{1}{\delta_i^{(1)}} = \gamma\left(\mathbf{d}_i^\top \mathbf{n} + \frac{1}{v_i}\right) \tag{4.21}$$

are the solutions to (4.20) for the downstream cells of each road i. Substitution of (4.21) into (4.18) gives (4.17). □

4.4.1 Approximate solutions

Consider a virtual network whose cell lengths are calculated according to (4.17). Define

$$f_i(\mathbf{n}, \gamma) = \ell_i - \frac{v_i}{\gamma} \sum_{k=1}^{n_i} \frac{1}{v_i \mathbf{d}_i^\top \mathbf{n} + k}. \tag{4.22}$$

such that it corresponds to the error in (4.11), i.e., the error between the sum of cell lengths and the length of road i. Thus, the problem of finding an average detectable and admissible division of a given network is equivalent to finding a vector of integers $\mathbf{n}$ and a constant γ such that $f_i(\mathbf{n}, \gamma) = 0$ for all $i = 1, \ldots m$. However, this is difficult in practice, as it is a combinatorial problem. As a simplification, we can search for solutions that satisfy the constraints approximatively, that is, to find $\mathbf{n}$ and γ such that $|f_i(\mathbf{n}, \gamma)|$ is less than a desired tolerance.

In the following theorems, we propose an alternative system of equations used to calculate $\mathbf{n}$ and γ. To do this, we allow the values of $\mathbf{n}$ to take real (instead of only integer) values.

Then, we approximate the sum in $f_i(\mathbf{n}, \gamma)$ using the continuous functions. This results in a system of equations that is simpler to solve, but that results in approximation error. However, we show that this error is bounded and can be reduced by selecting different values of γ.

Theorem 4.3
Consider any given traffic network $\mathcal{G}$. Let $\mathbf{x} \in \mathbb{R}^m$ and $\gamma > 0$ such that,

$$[(K_\gamma - \mathbb{I})^{-1} K_\gamma - V_1(\mathbb{I} - R_{11})^{-1} V_1^{-1}]\mathbf{x} = \frac{1}{2}\mathbf{1}, \tag{4.23}$$

where $K_\gamma = diag([e^{\gamma \ell_1/v_1} \quad e^{\gamma \ell_2/v_2} \quad \cdots \quad e^{\gamma \ell_m/v_m}])$. Let $\lfloor \cdot \rceil$ denote the nearest integer function. Then, $\mathbf{n} = \lfloor \mathbf{x} \rceil$ and γ satisfy

$$|f_i(\mathbf{n}, \gamma)| \sim O\left((v_i \mathbf{d}_i^\top \mathbf{n} + 1)^{-1}\right) \tag{4.24}$$

for $i = 1, 2, \ldots, m$.

Proof. Let ψ be the digamma function. Its definition and a list of properties can be found in [AS72] . This function satisfies the following identity,

$$\sum_{k=1}^{n} \frac{1}{z+k} = \psi(z+n+1) - \psi(z+1).$$

Therefore, with $z = v_i \mathbf{d}_i^\top \mathbf{n}$, (4.22) can be rewritten as

$$f_i(\mathbf{n}, \gamma) = \ell_i - \frac{v_i}{\gamma}\left[\psi\left(v_i \mathbf{d}_i^\top \mathbf{n} + n_i + 1\right) - \psi\left(v_i \mathbf{d}_i^\top \mathbf{n} + 1\right)\right]. \tag{4.25}$$

Define $\epsilon(z) = \psi(z) - \ln(z - \frac{1}{2})$. It is known that for $z > \frac{1}{2}$, $\epsilon(z)$ is positive and monotonically decreasing. Furthermore, its asymptotic expansion is $\epsilon(z) = \frac{z^{-2}}{24} + \frac{z^{-3}}{24} + \ldots$ as $z \to \infty$. Thus, (4.25) becomes

$$f_i(\mathbf{n}, \gamma) = \ell_i - \frac{v_i}{\gamma}\left[\ln\left(v_i \mathbf{d}_i^\top \mathbf{n} + n_i + \tfrac{1}{2}\right) - \ln\left(v_i \mathbf{d}_i^\top \mathbf{n} + \tfrac{1}{2}\right) + \Delta_i(\mathbf{n})\right], \tag{4.26}$$

where $\Delta_i(\mathbf{n}) = \epsilon\left(v_i \mathbf{d}_i^\top \mathbf{n} + 1\right) - \epsilon\left(v_i \mathbf{d}_i^\top \mathbf{n} + n_i + 1\right)$, is the total error due to this approximation.

Using the Taylor expansion of the logarithm, it can be shown that for any non-negative vector $\mathbf{a}$ and $c > 0$, $\ln\left(\mathbf{a}^\top \lfloor \mathbf{x} \rceil + c\right) - \ln\left(\mathbf{a}^\top \mathbf{x} + c\right)$ is equal to

$$\sum_{k=1}^{\infty} \frac{(-1)^{k+1}}{k}\left[\frac{\mathbf{a}^\top(\lfloor \mathbf{x} \rceil - \mathbf{x})}{\mathbf{a}^\top \mathbf{x} + c}\right]^k \sim O\left(\frac{1}{\mathbf{a}^\top \lfloor \mathbf{x} \rceil + 1}\right).$$

Thus, we can rewrite (4.26) as

$$f_i(\mathbf{n}, \gamma) = \ell_i - \frac{v_i}{\gamma}\left[\ln\left(v_i \mathbf{d}_i^\top \mathbf{x} + x_i + \tfrac{1}{2}\right) - \ln\left(v_i \mathbf{d}_i^\top \mathbf{x} + \tfrac{1}{2}\right) + \Delta_i(\mathbf{n}) + \eta_i(\mathbf{x})\right], \tag{4.27}$$

where $\eta_i(\mathbf{x})$ is the rounding error.

Now, consider the equation

$$0 = \ell_i - \frac{v_i}{\gamma}\left[\ln\left(v_i\mathbf{d}_i^\top\mathbf{x} + x_i + \tfrac{1}{2}\right) - \ln\left(v_i\mathbf{d}_i^\top\mathbf{x} + \tfrac{1}{2}\right)\right] \tag{4.28}$$

Using logarithm identities, this becomes

$$\gamma\frac{\ell_i}{v_i} = \ln\left(\frac{v_i\mathbf{d}_i^\top\mathbf{x} + x_i + \frac{1}{2}}{v_i\mathbf{d}_i^\top\mathbf{x} + \frac{1}{2}}\right),$$

which can be written as $x_i - (e^{\gamma\ell_i/v_i} - 1)v_i\mathbf{d}_i^\top\mathbf{x} = \frac{1}{2}(e^{\gamma\ell_i/v_i} - 1)$. Thus, we obtain a system of m equations,

$$[\mathbb{I} - (K_\gamma - \mathbb{I})V_1 D]\mathbf{x} = \frac{1}{2}(K_\gamma - \mathbb{I})\mathbf{1}. \tag{4.29}$$

Substituting the expression for D into (4.29) and rearranging terms we obtain (4.23), and thus, (4.28) is satisfied for the considered $\mathbf{x}$ and γ. Substituting (4.28) into (4.27), we get

$$|f_i(\mathbf{n}, \gamma)| = \frac{v_i}{\gamma}|\Delta_i(\mathbf{n}) + \eta_i(\mathbf{x})|. \tag{4.30}$$

Note that $|\Delta(\mathbf{n})| < \epsilon(v_i\mathbf{d}_i^\top\mathbf{n} + 1)$, and so $\Delta(\mathbf{n}) \sim O[(v_i\mathbf{d}_i^\top\mathbf{n} + 1)^{-2}]$. Additionally, $\eta_i(\mathbf{x}) \sim O[(v_i\mathbf{d}_i^\top\mathbf{n} + 1)^{-1}]$.

Thus $|f_i(\mathbf{n}, \gamma)| \sim O\left((v_i\mathbf{d}_i^\top\mathbf{n} + 1)^{-1}\right)$, completing the proof. □

Theorem 4.4
There exists γ_{max} such that for every $0 < \gamma < \gamma_{max}$, the solution to (4.23) is positive. Moreover, as γ approaches γ_{max} the magnitude of $\mathbf{x}$ grows arbitrarily large.

Proof. Let $M = (K_\gamma - \mathbb{I})^{-1}K_\gamma - V_1(\mathbb{I} - R_{11})^{-1}V_1^{-1}$. Assume that M is invertible. Using Woodbury's identity[2], we can write M^{-1} as

$$(\mathbb{I} - K_\gamma^{-1}) + (\mathbb{I} - K_\gamma^{-1})V_1K_\gamma(\mathbb{I} - R_{11}K_\gamma)^{-1}V_1^{-1}(\mathbb{I} - K_\gamma^{-1}).$$

Therefore, M is invertible only if $\mathbb{I} - R_{11}K_\gamma$ is invertible.

Let $\lambda(R_{11})$ denote the spectral radius of R_{11}. It can be shown that $\mathbb{I} - R_{11}$ is an invertible M-matrix (see [RVWF19]), and thus, $\lambda(R_{11}) < 1$. For sufficiently small γ, K_γ can be made arbitrarily close to $\mathbb{I}$, such that $\lambda(R_{11}K_\gamma) < 1$.

Let γ_{max} be such that $\lambda(R_{11}K_{\gamma_{max}}) = 1$. Thus, for every $\gamma < \gamma_{max}$, $\mathbb{I} - R_{11}K_\gamma$ is an invertible M-matrix such that

$$(\mathbb{I} - R_{11}K_\gamma)^{-1} = \mathbb{I} + \sum_{k=1}^{\infty}(R_{11}K_\gamma)^k. \tag{4.31}$$

As $\gamma \to \gamma_{max}$, the nonzero elements of $(R_{11}K_\gamma)^k$ increase exponentially. For $\gamma = \gamma_{max}$, the sum diverges and the matrix is not invertible. Finally, for $0 < \gamma < \gamma_{max}$, $(\mathbb{I} - R_{11}K_\gamma)^{-1}$ and $(\mathbb{I} - K_\gamma^{-1})$ are non-negative, which implies that M^{-1} is non-negative. □

[2] $(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}$.

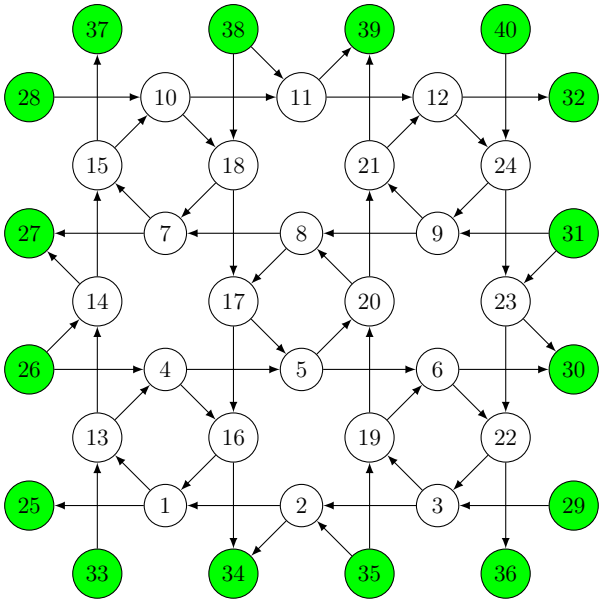


Figure 4.5: 4×4 Manhattan grid (nodes correspond to roads). All TRs are set to 50%. Green nodes symbolize sensor locations.

Example 4.3. Consider the 4 × 4 Manhattan grid network shown in Fig. 4.5. Assume that all roads have the same length of $\ell = 500$m, and the same free-flow velocity of $v = 30$ km·h^{-1}. As all speeds and lengths are equal,

$$K_\gamma = \exp\left(\gamma \frac{\ell}{v}\right) \mathbb{I}, \tag{4.32}$$

and $\gamma_{max} = -(v/\ell) \ln [\lambda(R_{11})]$.

By Theorem 4.4, for any given value of $\gamma \in (0, \gamma_{max})$, the corresponding division vector $\mathbf{n}$ is positive and increases in magnitude as γ increases. This is evidenced in Fig, 4.6, which shows the solutions to (4.23) for different values of γ in the range. The initial choice of 88% is because at this value all elements of $\mathbf{n}$ are greater than or equal to 1.

Using (4.23) to calculate $\mathbf{n}$ induces an error $f_i(\mathbf{n}, \gamma)$ in the admissibility constraint. Consider the total root mean square error (RMSE) for all roads $i = 1, \ldots, m$, as shown in Fig. 4.7. As γ increases, the upper limit of this error decreases, approaching 0 as $\gamma \to \gamma_{max}$. This is because the number of cells per road is also increasing rapidly, so Theorem 4.3 is applicable. In this sense, the lowest error is obtained by choosing γ very close to γ_{max}.

As a specific case, let $\gamma = 0.95\gamma_{max}$, which corresponds to a vector $\mathbf{n}$ with elements 2, 3, and 4, and a RMS error below 3%. The corresponding virtual network is shown in Fig. 4.8.

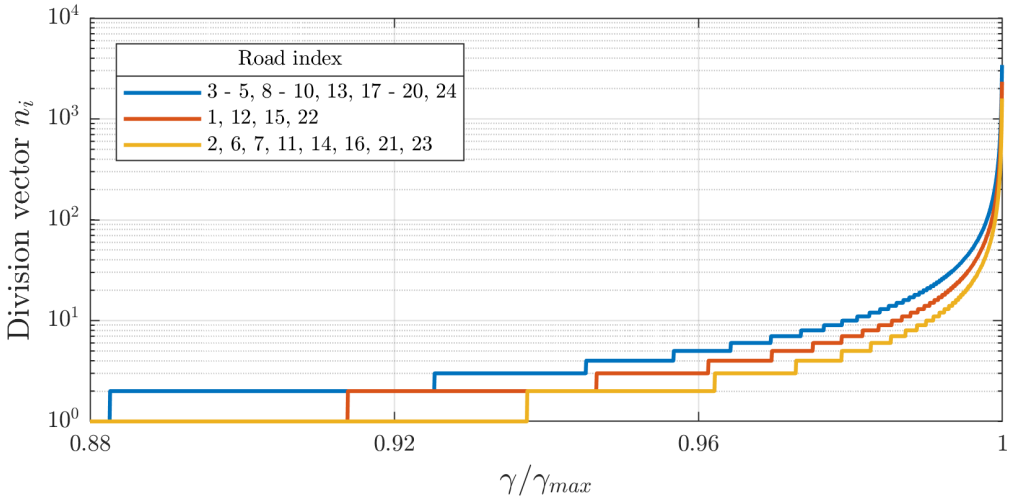


Figure 4.6: Approximate solutions of (4.23) for $\mathbf{n}$ for all unmeasured nodes.

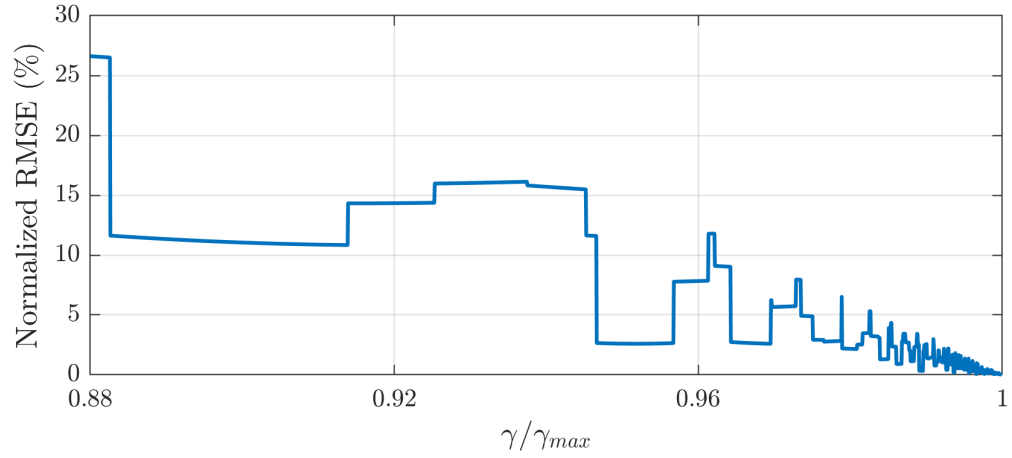


Figure 4.7: Normalized RMSE, $\sqrt{\sum_i f_i(\mathbf{n},\gamma)^2/\ell}$ for different values of γ.

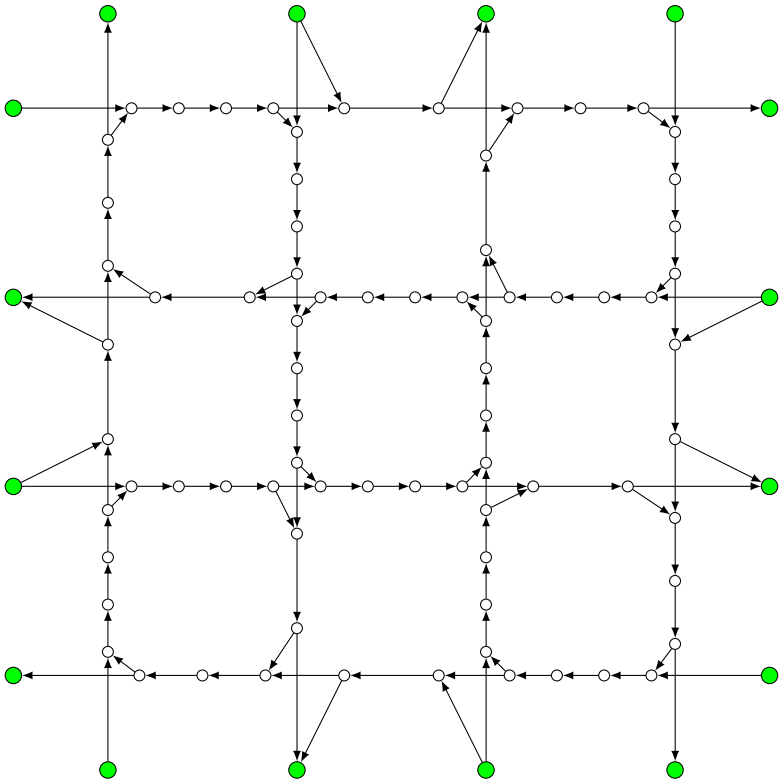


Figure 4.8: Virtual network for a 4×4 Manhattan grid using $\gamma = 0.95\gamma_{max}$.

4.4.2 Observer implementation

Suppose that an average-detectable admissible virtual graph $\mathcal{G}^{(\mathbf{n})}$ is obtained from a given traffic network, such that the open-loop estimator (4.16) converges asymptotically. However, from the previous example it can be seen that virtual networks with low error in the admissibility constraint may require many divisions per road. The dimensions of the matrices $A_{11}^{(\mathbf{n})}$ and $A_{12}^{(\mathbf{n})}$ are proportional to the norm of $\mathbf{n}$, hence, large number of divisions could imply costly calculations to obtain the values required by the estimator. However, the expressions for the gains in the estimator can be simplified such that the matrix $A^{(\mathbf{n})}$ is not explicitly required.

The first term can be simplified by using the fact that the network is average detectable, which requires $\mathbf{1}^\top A_{11}^{(\mathbf{n})} = -\gamma \mathbf{1}^\top$ and thus

$$\frac{1}{n_{tot}} \mathbf{1}^\top A_{11}^{(\mathbf{n})} \mathbf{1} = -\frac{\gamma}{n_{tot}} \mathbf{1}^\top \mathbf{1} = -\gamma.$$

The second term of the estimator can also be simplified. Let

$$\mathbf{b}^\top = \frac{1}{n_{tot}} \mathbf{1}^\top A_{12}^{(\mathbf{n})}$$

be the vector gain for the measurements. From the construction of $A^{(\mathbf{n})}$, it follows,

$$b_{i-n_{tot}} = \frac{1}{n_{tot}} \sum_{j=1}^{m} \sum_{k=1}^{n_j} r_{i,j^{(k)}}^{(\mathbf{n})} \frac{v_i}{\delta_j^{(k)}} \tag{4.33}$$

where the indexes $i = n_{tot} + 1, \ldots, n_{tot} + k$ correspond to the measured boundary nodes in the virtual network. From the definition of $R^{(\mathbf{n})}$, the expression can be simplified to

$$b_{i-n_{tot}} = \frac{1}{n_{tot}} \sum_{j=1}^{m} r_{i,j} \frac{v_i}{\delta_j^{(n_j)}}. \tag{4.34}$$

Substituting (4.17) into (4.34),

$$b_{i-n_{tot}} = \frac{\gamma v_i}{n_{tot}} \sum_{j=1}^{m} r_{i,j} \left(\mathbf{d}_j^\top \mathbf{n} + \frac{n_j}{v_j} \right). \tag{4.35}$$

Using matrix notation, this can be rewritten into

$$\mathbf{b}^\top = \frac{\gamma}{\mathbf{1}^\top \mathbf{n}} \mathbf{n}^\top (D^\top + V_1^{-1}) R_{21}^\top V_2. \tag{4.36}$$

Substituting the definition of D, this simplifies to

$$\mathbf{b}^\top = \frac{\gamma}{\mathbf{1}^\top \mathbf{n}} \mathbf{n}^\top V_1^{-1} (\mathbb{I} - R_{11}^\top)^{-1} R_{21}^\top V_2. \tag{4.37}$$

Hence, the estimator for the average density of the virtual system becomes

$$\dot{\hat{\rho}}_{av}^{(\mathbf{n})}(t) = -\gamma \hat{\rho}_{av}^{(\mathbf{n})}(t) + \frac{\gamma}{\mathbf{n}^\top \mathbf{1}} \mathbf{n}^\top V_1^{-1} (\mathbb{I} - R_{11}^\top)^{-1} R_{21}^\top V_2 \mathbf{y}(t). \tag{4.38}$$

Note that to use the proposed observer only requires the calculation of $\mathbf{n}$, γ, and the topology of the original network. As the virtual network is not required, this provides an efficient way to deploy the estimator even for very high number of divisions.

4.4.3 Calculation of the number of divisions per road

Using Theorems 4.3 and 4.4, we propose the following algorithm to find a vector $\mathbf{n}$ of partitions that yields an admissible network within a given tolerance.

Algorithm 4.1. *Create a virtual network*

Inputs: Traffic network $\mathcal{G} = \{\mathcal{N}, \mathcal{E}, R, \boldsymbol{\ell}, \mathbf{v}\}$. *Tolerance* ϵ.

Output: Vector of divisions $\mathbf{n}$. *Detectability constant* γ.

1. *Initialize range for* γ:

   1.1 *Set* $\gamma_1 \leftarrow 0$.

1.2 Set $\gamma_2 \leftarrow max_i[-\ln(\lambda(R_{11}))v_i/\ell_i]$.

2. Calculate $\mathbf{n}$*:*

2.1 Set $\gamma \leftarrow (\gamma_1 + \gamma_2)/2$.

2.2 Set $\mathbf{x} \leftarrow [(K_\gamma - \mathbb{I})^{-1}K_\gamma - V_1(\mathbb{I} - R_{11})^{-1}V_1^{-1}]^{-1}\mathbf{1}/2$.

2.3 Set $\mathbf{n} \leftarrow \lfloor \mathbf{x} \rceil$.

2.4 If $\exists i \ : \ x_i < 0$

2.4.1 Set $\gamma_2 \leftarrow \gamma$.

2.4.2 Go to Step 2.1.

2.5 If $\exists i \ : \ |f_i(\mathbf{n}, \gamma)| > \epsilon$

2.5.1 Set $\gamma_1 \leftarrow \gamma$.

2.5.2 Go to Step 2.1.

3. Return $\mathbf{n}$*,* γ.

The algorithm is based on an implementation of the well known bisection method [BF85, Chapter 2]. From Theorems 4.3 and 4.4, as γ approaches the maximum value γ_{max}, the error function $f_i(\mathbf{n}, \gamma)$ decreases its magnitude. As γ_{max} is not generally known a priori, the algorithm searches inside a range of possible values until it finds a value of γ for which $f_i(\mathbf{n}, \gamma)$ is less than the given tolerance for each $i = 1, \ldots, m$.

The algorithm explores for values of γ inside of the interval $[\gamma_1, \gamma_2]$, initialized in Step 1. γ_1 is set to zero, which is a trivial constraint from the problem formulation. The expression for the maximum value γ_2 comes from the fact that

$$\lambda(R_{11}K_{\gamma_{max}}) = 1 \tag{4.39}$$

(see proof of Theorem 4.4), and that K_γ is a diagonal matrix.

With this initial limits, the algorithm proceeds by setting γ as the intermediate value between γ_1 and γ_2, and then it is used to calculate a candidate solution $\mathbf{x}$. If $\mathbf{x}$ has negative entries, it means that the current γ is higher than γ_{max}, so the upper limit γ_2 is reduced. Else, the error functions $|f_i(\mathbf{n}, \gamma)|$ are compared against the tolerance. If the tolerance is not met, then the lower limit γ_1 is increased. This process continues until a good-enough solution is reached.

Note that after every step, the exploration range $[\gamma_1, \gamma_2]$ is reduced by half, and therefore, the algorithm will require at most $O(\log_2(1/\epsilon))$ cycles. Additionally, every cycle requires the inversion of a $m \times m$ matrix, each requiring $O(m^3)$ calculations. Therefore, the overall computational cost is $O(m^3 \log_2(1/\epsilon))$.

4.5 Error analysis

In the previous sections we have discussed how to generate a virtual network $\mathcal{G}^{(\mathbf{n})}$ from a given one $\mathcal{G}$. The virtual network keeps the same physical properties as the original one, but its roads have been divided into smaller cells. This transformation allows to have an average detectable network.

Nevertheless, even if the underlying physical network is the same, the dynamics of the real and virtual systems are different, and therefore the trajectories of the corresponding average densities are not identical. Let ρ_{av} be the average density of the given undivided network, and let $\rho_{av}^{(\mathbf{n})}$ be the average density of the divided network (following the methods discussed before). It can be seen that both definitions are not equivalent:

$$\rho_{av} = \frac{1}{m}\sum_{i=1}^{m}\rho_i \tag{4.40}$$

where ρ_i is the density of road i, and,

$$\rho_{av}^{(\mathbf{n})} = \frac{1}{\sum_{i=1}^{m} n_i}\sum_{i=1}^{m}\sum_{k=1}^{n_i}\rho_i^{(k)} \tag{4.41}$$

where $\rho_i^{(k)}$ is the density of cell k of road i, and n_i is the number of divisions of road i. The quantities ρ_i and $\rho_i^{(k)}$ are related by the expression

$$\ell_i\rho_i = \sum_{k=1}^{n_i}\delta_i^{(k)}\rho_i^{(k)} \tag{4.42}$$

which states that the total number of vehicles must be the same in both representations. Figures 4.9 and 4.10 show the trajectories of the average density of the real system ρ_{av}, and the virtual system $\rho_{av}^{(\mathbf{n})}$ for two sample networks in Examples 4.1 and 4.2.

Note that in general, both trajectories are very close, so using the estimator for the virtual system yields a good result for the real one. However, the closeness between these trajectories might depend in the original network parameters, so it is required to establish a way to determine if a particular network might be well represented using the approach described in this paper.

The error between the average density of the original and virtual system can be found to be

$$\rho_{av}^{(\mathbf{n})} - \rho_{av} = \frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{n_i}\left(\frac{1}{n_{av}} - \frac{\delta_i^{(k)}}{\ell_i}\right)\rho_i^{(k)} \tag{4.43}$$

where $n_{av} = \mathbf{1}^\top\mathbf{n}/m$. Note that this expression depends both on the properties of the network division (the cell lengths), and on the state of the virtual network, as the density of each cell is required.

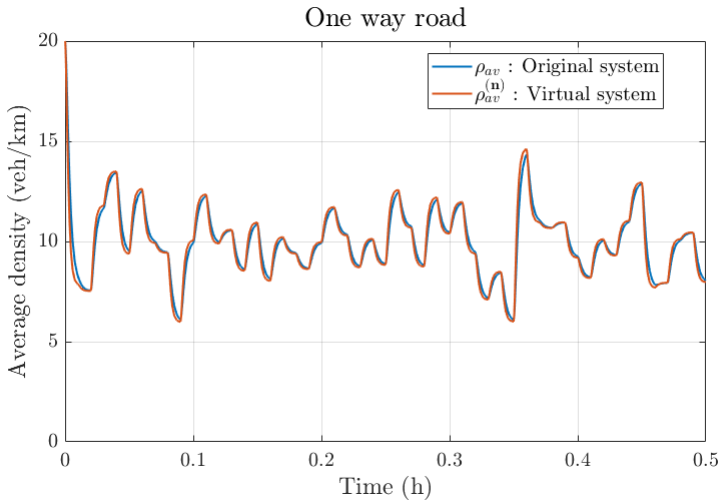


Figure 4.9: Trajectories of the average densities of the original and virtual systems for the one way road from Example 4.1.

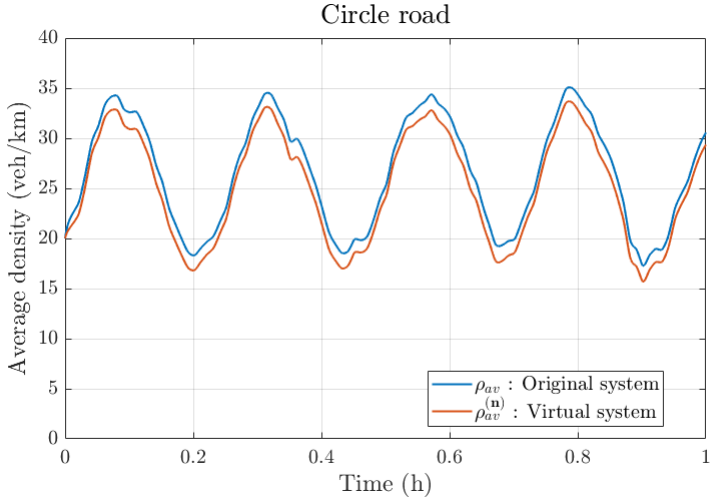


Figure 4.10: Trajectories of the average densities of the original and virtual systems for the circle road from Example 4.2.

4.5.1 Slow varying inputs

Consider the case where the rate of change of the inputs (incoming flows at the boundaries) is small in comparison to the traveling in each road. This can be achieved if the input demands are relatively constant, or in urban scenarios where the distance between intersections is small. Note that this does not imply that the physical dimensions of the entire network is small, just its individual components. This assumption implies that there are small spatial variations in the flow for each road, so that $\boldsymbol{\varphi}_{in}(t) \approx \boldsymbol{\varphi}_{out}(t)$. This in turn implies that the density of each road is approximately constant in space, and thus

$$\rho_i^{(k)} \approx \rho_i \tag{4.44}$$

for each road i and each cell k. It follows that the average density of the virtual system can be approximated as a weighted average of the density of the internal nodes, where the weights are the number of partitions,

$$\rho_{av}^{(\mathbf{n})} \approx \frac{1}{n_{tot}} \mathbf{n}^\top \boldsymbol{\rho}_1. \tag{4.45}$$

where $n_{tot} = \mathbf{1}^\top \mathbf{n}$.Thus, we can simplify the error equation as

$$\rho_{av}^{(\mathbf{n})} - \rho_{av} \approx \frac{1}{m} \sum_{i=1}^{m} \rho_i \left(\frac{n_i}{n_{av}} - 1 \right). \tag{4.46}$$

Note that in this case, the difference between the two trajectories depends only on the densities of the roads (not cells), and the number of cells per road. Using matrix notation, this can be rewritten as

$$\rho_{av}^{(\mathbf{n})} - \rho_{av} \approx \frac{\mathbf{n}^\top \boldsymbol{\sigma}}{\mathbf{n}^\top \mathbf{1}} \tag{4.47}$$

where

$$\boldsymbol{\sigma} = \boldsymbol{\rho} - \bar{\rho}_{av} \mathbf{1}. \tag{4.48}$$

The value of the error depends then on the cell division and the state of the original system, which is unknown. To benchmark the applicability of the proposed method for all possible networks, we can analyze the distribution of the error under the assumption that the deviation $\boldsymbol{\sigma}$ is a random process. We introduce the following theorem,

Theorem 4.5
Consider the metric space $(\mathbb{S}^m, d)$ where $\mathbb{S}^m = \{\mathbf{x} \in \mathbb{R}^m : ||\mathbf{x}|| = 1\}$ is the m-dimensional unit sphere and d is the spherical distance. Consider two independent random vectors $\mathbf{u} \in \mathbb{S}^m$ and $\mathbf{v} \in \mathbb{S}^m$. Then, for any $\epsilon > 0$,

$$P_{\mathbf{u},\mathbf{v}}(|\mathbf{u}^\top \mathbf{v}| < \epsilon) \geq 1 - 2e^{-(m-1)\epsilon^2/2} \tag{4.49}$$

where P is the uniform spherical probability measure.

The proof of the theorem is shown in Annex A.3. This implies that for large dimensions ($m \to \infty$), the probability of the two vectors being ϵ-orthogonal is asymptotically 1.

Because of this, the trajectories of the average density of the original and virtual systems get closer as we consider larger networks. Note that this depends on the assumption $\boldsymbol{\sigma}$ is independent to $\mathbf{n}$. Even though both vectors depend on the properties of the network, this assumption can be justified by the fact that $\boldsymbol{\sigma}$ depends highly on the input demands $\mathbf{u}$, and thus, the orientation of $\boldsymbol{\sigma}$ does not depend on the orientation of $\mathbf{n}$.

4.6 Simulation and validation

To test the methods proposed in this chapter, we will consider the zone of Grenoble downtown as shown in the previous chapters. Figure 4.11 shows the input graph of Grenoble downtown, and the classification of every road into the measured or internal nodes. However, note that some nodes (bottom right) are marked as "ignored". Recall from Theorem 4.3 that the errors in the admissibility condition are of the order

$$|f_i(\mathbf{n}, \gamma)| \sim O\left((v_i \mathbf{d}_i^\top \mathbf{n} + 1)^{-1}\right) \tag{4.50}$$

which imply that as $\mathbf{n}$ increases, the error decreases. Nevertheless, this is only the case when $\mathbf{d}_i^\top$ is non-zero. By construction

$$D = \left(\sum_{i=1}^{\infty} R_{11}^n\right) V_1^{-1}. \tag{4.51}$$

As R_{11} is the adjacency matrix of the unmeasured subgraph, $d_{i,j}$ measures the average distance between roads i and j. Therefore, if road i is part of a strongly connected component, $d_{i,i} > 0$ and the corresponding error can be decreased. Nodes marked as internal in the Figure are part of a strongly connected component, so their admissibility error can be reduced. Nodes marked as ignored are not part of the strongly connected component, so their error is not guaranteed to be reduced arbitrarily. Fortunately, urban networks are usually strongly connected, so the method can be applied in most cases.

To calculate the number of partitions per road, we used Algorithm 4.1, with an error tolerance of 10%. Figure 4.12 shows the resulting division vector: the top part of the figure shows the spatial distribution of road divisions, while the bottom shows the sorted values as a function of road index. Note that this result implies that roads are divided into very wildly varying number of cells, for which the maximum requires around 14500 cells.

Nevertheless, as shown in the previous simulated example, the deployment of the estimator only requires the outputs of the algorithm ($\mathbf{n}$ and γ) and the properties of the original network. Therefore, the very large number of cells in the virtual representation does not lead to a problem in the actual on-line calculations.

To validate the open-loop observer, we used the software Aimsun to produce individual vehicle traces to use as ground truth. To setup the simulation, a time-varying profile was applied to the boundary inflows, using the same procedure described in Chapter 3. The TRs

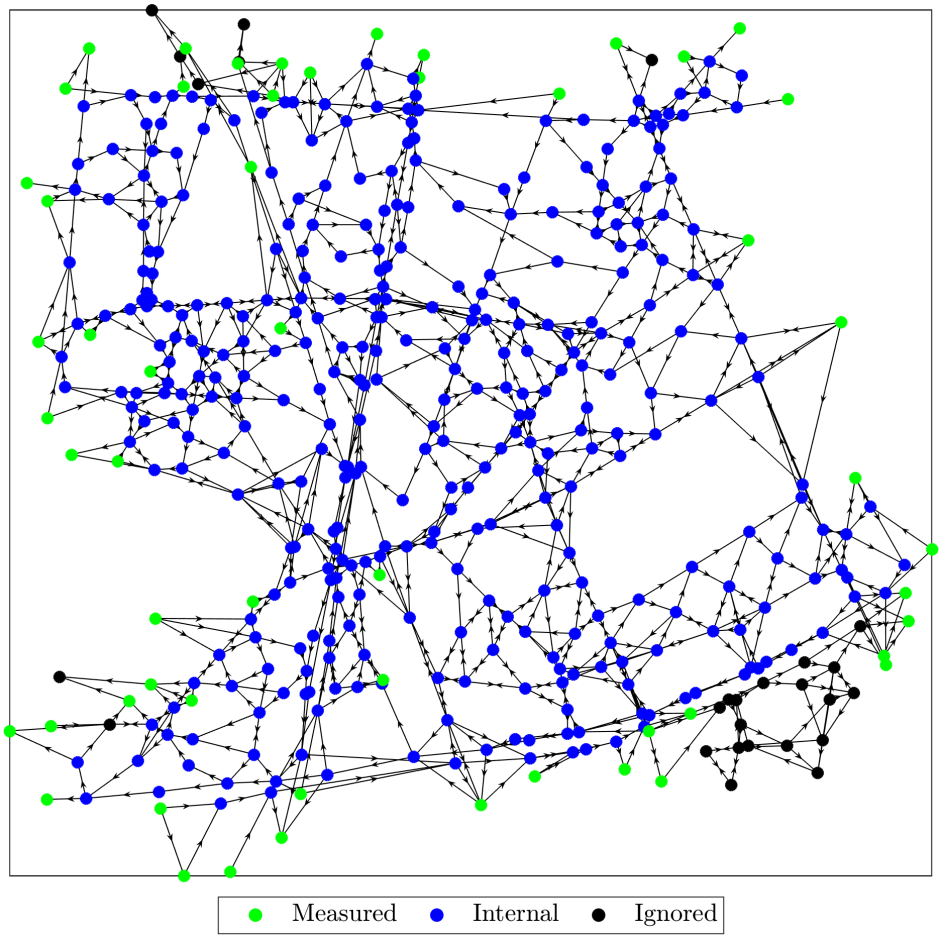


Figure 4.11: Road graph of Grenoble downtown, with road classification for average density estimation.

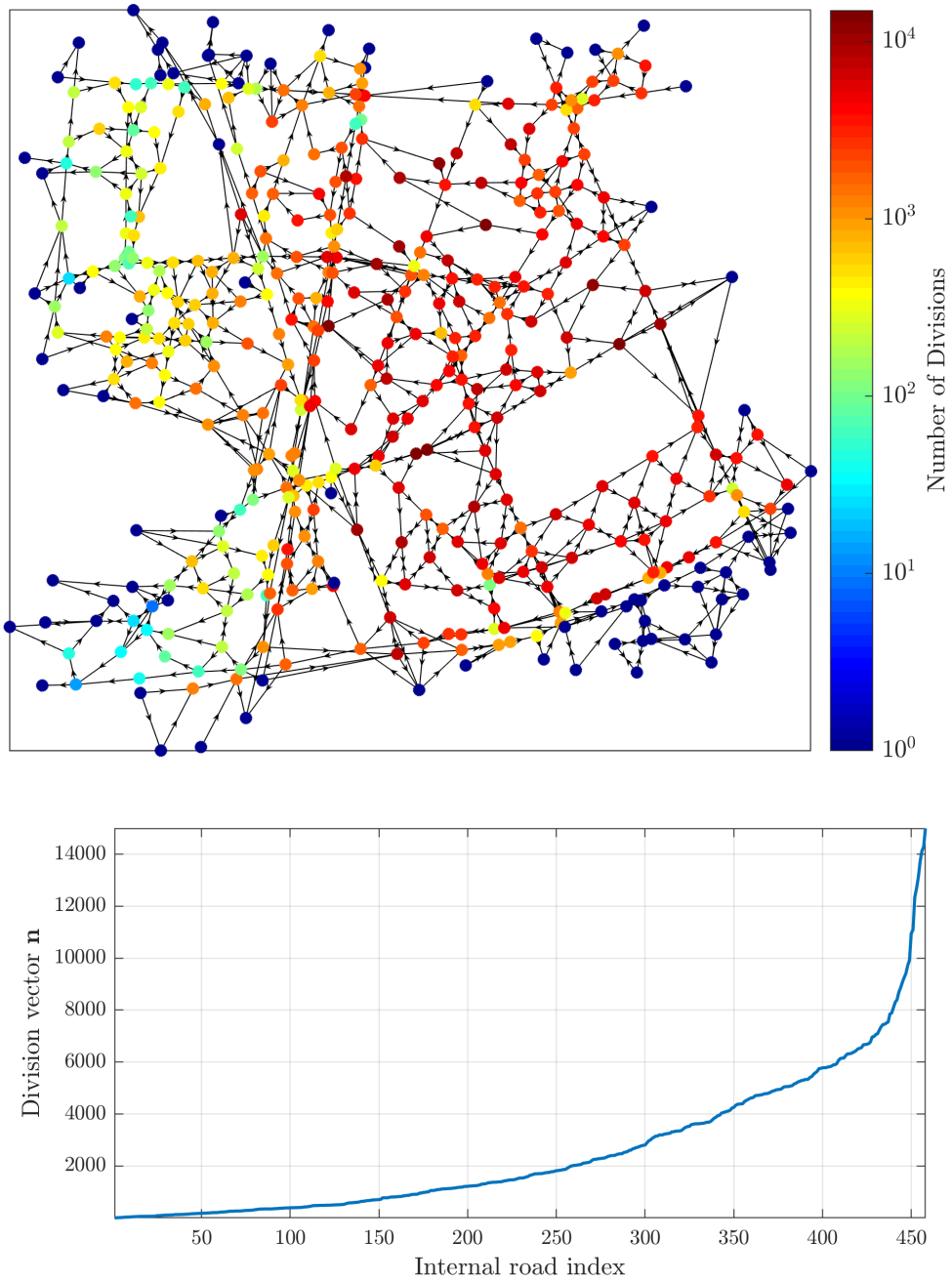


Figure 4.12: **Top:** Network representation of the selected area where the color of each node corresponds to its number of cells. **Bottom:** List of values of the division vector **n** sorted in ascending order.

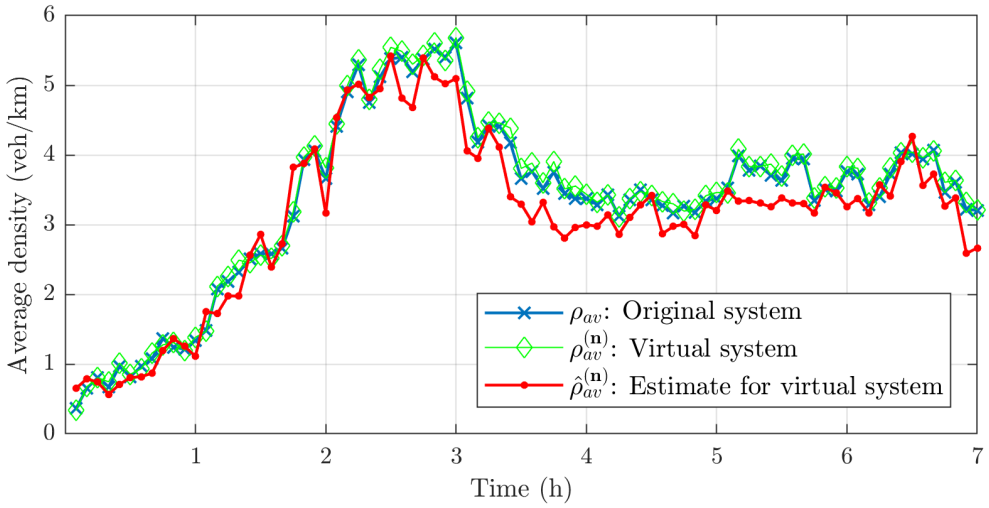


Figure 4.13: Average density trajectories for the Grenoble network.

were arbitrarily set to be distributed homogeneously according to the number of exits from each intersection.

The density of each of the measured nodes were obtained directly and used as input for the estimator (4.16), using the values of $\mathbf{n}$ and γ provided by the algorithm. As ground truth, the density of each of the internal nodes were obtained. We consider two trajectories: the density of the original system $\rho_{av}(t)$ and the density of the virtual system $\rho_{av}^{(\mathbf{n})}(t)$. The former is easily calculated from the outputs of the simulator. Nevertheless, as roads present very high number of cells, it is unfeasible to measure in a significant way the density of each cell. Therefore, we estimate $\rho_{av}^{(\mathbf{n})}$ using the slow-varying inputs assumption,

$$\rho_{av}^{(\mathbf{n})}(t) \approx \frac{1}{n_{tot}} \mathbf{n}^\top \boldsymbol{\rho}_1(t) \tag{4.52}$$

as described in Section 4.5.1. The maximum road traversal time is around 30 seconds, while the inflows are defined constant for an interval of 20 minutes, and the outputs are obtained with an aggregation time of 5 minutes. Therefore, the slow-varying input assumption is applicable. The trajectories of both average densities are shown in Fig. 4.13. From the figure it can be seen that the average densities of the original and the virtual system are almost identical.

The figure also shows the estimated average density trajectory, as obtained with (4.16). Note that the estimates approaches the real trajectories, with a normalized mean squared error of 10%. This error is to be expected, as there is a difference between individual vehicle behavior and the macroscopic model predictions.

4.7 Concluding remarks

In this Chapter, we analyzed the convergence of a one-dimensional open-loop observer for the average density of a traffic network, under linear traffic dynamics such as free-flow regime. In general, such observers do not converge, but we propose a method to obtain a virtual network representation with the same physical properties such that the observer converges. This new representation is constructed by dividing each road in the original network in a number of cells with specific lengths.

Although the average density of the original and virtual systems are not necessarily equal, we show that by considering large networks, the difference between these two variables is almost always small, and thus, the method can give a useful approximation to the value of interest.

The methods were tested using a simulated traces of individual vehicles in the real case of the city of Grenoble, France. The estimated average density was found to be close to the ground truth values, with a RMS error of 10%.

Chapter 5

Application to the city of Grenoble

Contents

5.1 Overview

This Chapter presents the study case of Grenoble downtown and validates the methodologies described in Chapter 3 for the dynamic estimation of the flow and density of each road of a network, and the average-density estimation of the entire network described in Chapter 4, using real data. The considered section of the Grenoble traffic network has been equipped with sensors collecting information about the traffic flow and velocities in the boundary of the domain, and the TRs for a selected number of intersections. First, we describe the available data and how sensor locations were selected using the methods from previous chapters. Then,

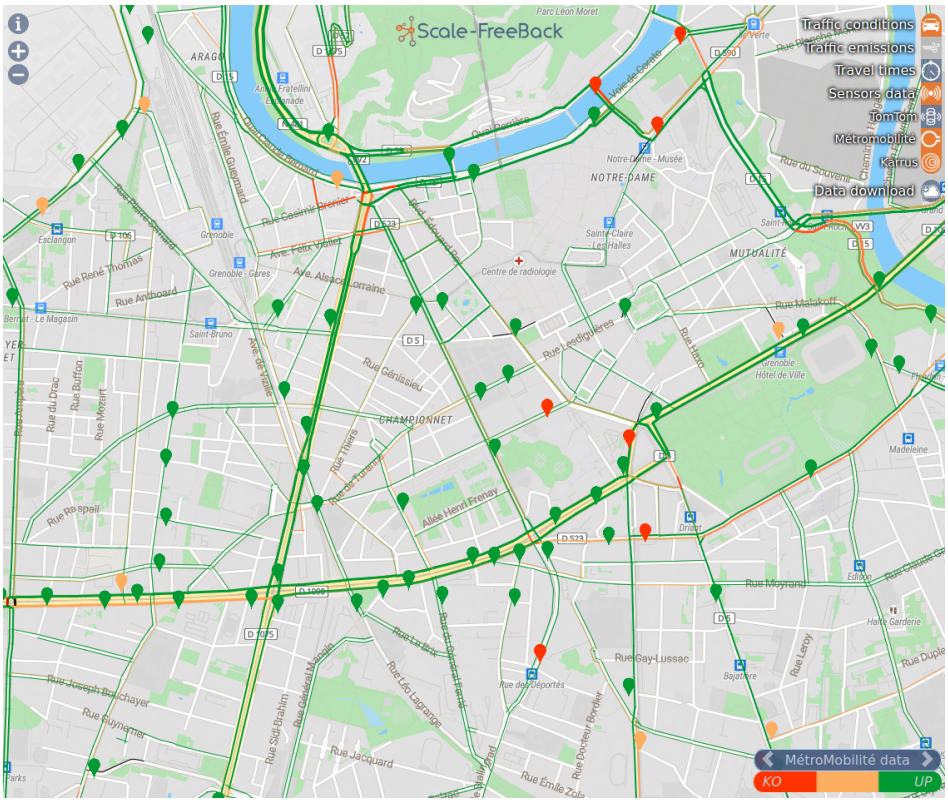


Figure 5.1: Homepage of the GTL-Ville. Points show the location of stationary sensors, and the highlighted roads show the estimated traffic indicators.

we apply the concerned density and flow estimation algorithms, and the results are compared to ground-truth data using a subset of validation sensors.

5.2 Experimental platform

The Grenoble Traffic Lab for urban networks (GTL-Ville)[1] is an experimental platform for real-time collection of traffic data coming from a network of sensors installed in the city of Grenoble. This platform also provides real-time traffic indicators and analysis oriented towards the users of the city, traffic operators, and researchers, which are available for download.

Figure 5.1 shows the homepage of the GTL-Ville. Currently, data statistics can be shown for roads and sensors. Sensor indicators correspond to the data collected from stationary sensors, which includes speed, flow and occupancy. Road indicators correspond to data obtained from FCD provided by TomTom, and include speed, fluidity, and density. For the purposes of this work, a section of the city center was selected to validate the proposed estimation approaches, which spans an area of 1.4 Km by 1 Km. In the following, we describe in depth the available data sources for the selected area.

[1] http://gtlville.inrialpes.fr/

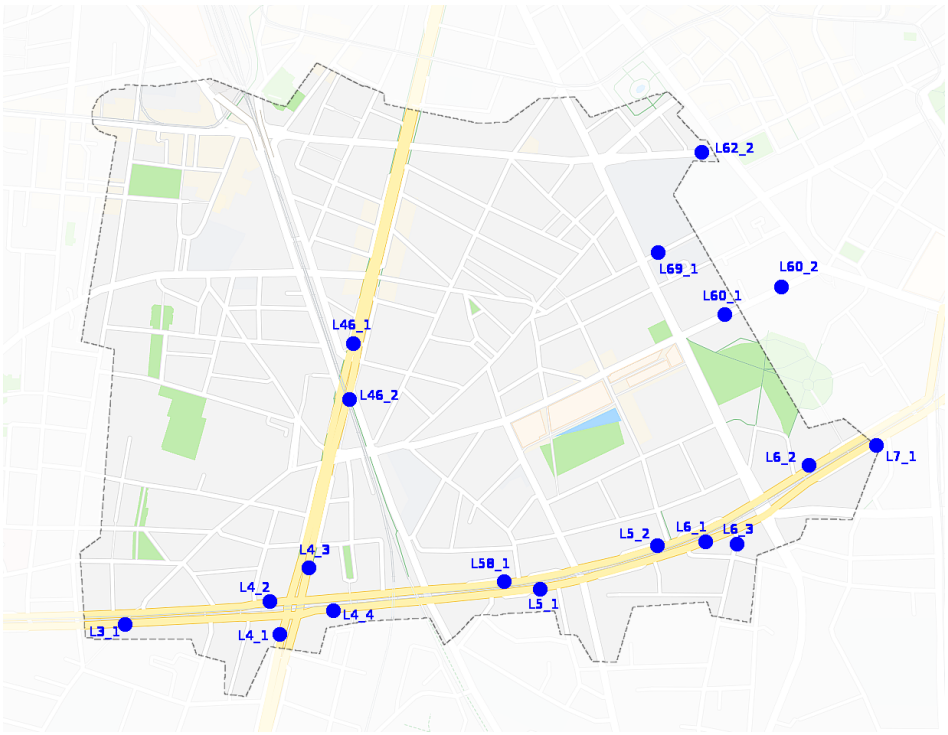


Figure 5.2: Location of single-loop induction sensors in blue, with their corresponding ID in database.

5.2.1 Induction loops

The city of Grenoble is equipped in various locations by induction loop sensors. This technology consists on a coil buried under the pavement that perceives a change in the magnetic induction when a vehicle passes over it. In the section of interest, 18 single-loop sensors are available whose positions are shown in Figure 5.2. These sensors are administered by the city's traffic administration (Metromobilité), which allows the use of this data. Data is received in real time, with an aggregation time of 6 minutes. For each time period, received data is the number of detected vehicles and occupancy.

5.2.2 Radars

To complement the data provided by the induction loops, additional sensors were located as shown in Figure 5.3. These sensors consist of microwave radars, which periodically emit a pulse of radiation and then measure the reflections of these pulses. By measuring the properties of the reflected radiation, this technology can accurately measure the velocity and length of vehicles passing through the specified location, for every lane and direction. These sensors are leased and administered by a subcontractor. An example of a radar installation is shown in Figure 5.4.

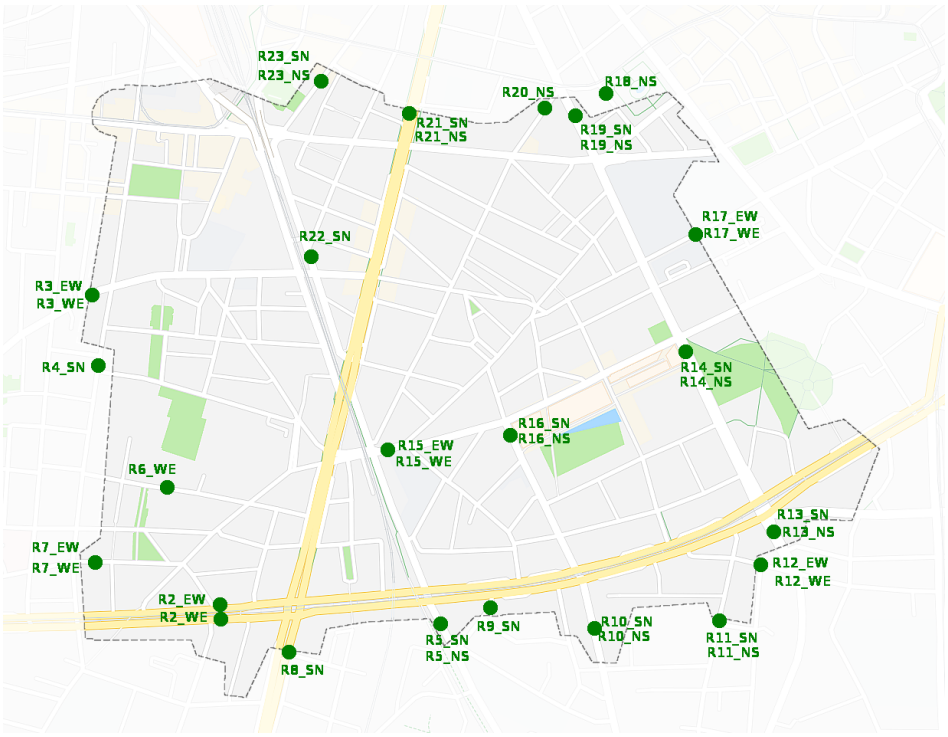


Figure 5.3: Location of radar sensors in green, with their corresponding ID in database.

Figure 5.4: Example of a radar sensor for the measurement of road flow. Courtesy: Karrus.

Table 5.1: Description of the road classes provided by TomTom.

Class	Short description	Long description
0	Highways, Motorways	All roads that are officially assigned as motorways.
1	Major roads of high importance	All roads of high importance, that are part of a connection used for international and national traffic and transport.
2	Other major roads	All roads used to travel between different neighboring regions of a country.
3	Secondary roads	All roads used to travel between different parts of the same region.
4	Local connecting roads	All roads making all settlements accessible or making parts of a settlement accessible.
5	Local roads of high importance	All local roads that are the main connections in a settlement. These are the roads where important through traffic is possible
6	Local roads	All roads used to travel within a part of a settlement or roads of minor connecting importance in a rural area.
7	Local roads of minor importance	All roads that only have a destination function.
8	Other roads	All roads which are not usable by cars.

Data from these sensors are available in real time, with an aggregation time of 1 minute. For each passing vehicle, the sensors save the time of passage, its velocity, length, and lane.

5.2.3 Floating car data

FCD is provided by the driving navigation assistance company TomTom[2]. For each road, the provided information is the Functional Road Classification (FRC), the average speed, and the confidence index. Each of these types of data are described below.

5.2.3.1 Functional Road Classification

The FRC is used to classify roads into homogeneous classes depending on their role in a transportation network [D'A+14]. This classification determines the type of use of each road, for instance, as it differentiates between major roads that experience heavy traffic from a variety of O/D pairs, and minor roads which are inside of a residential area and experience light traffic only. Table 5.1 shows the FRC provided by TomTom and their description[3].

Figure 5.5 shows the FRC of each road of the considered zone of Grenoble. For this area,

[2]https://www.tomtom.com/

[3]Source: https://developer.tomtom.com/traffic-stats/support/faq/what-are-functional-road-classes-frc

Figure 5.5: FRC for the Grenoble network, provided by TomTom.

there are no Motorways so FRC 0 not used. Similarly, there are no roads with FRC 2. All pathways with FRC 8 are ignored as they are not used for the type of traffic under study.

5.2.3.2 Average speed

Average speed data is available for roads that have a FRC between 0 and 6. This information is available in real time, with a frequency rate of 1 minute. However, data is not collected for each individual road, but instead, all considered roads are divided into 120 partitions. Each partition consists of a number roads with the same FRC that forms a straight path, with an average length of 200m. FCD is collected for vehicles contained in the same partition during a time interval of 1 minute, and a speed value is given. Therefore, all roads contained in the same partition have the same speed value. Figure 5.6 shows the map of road partitions for average speed data.

5.2.3.3 Confidence index

Because of internal policies of TomTom, the precise number of vehicles providing the FCD at every time instant is unknown. However, to quantify data quality, each velocity value is supported by an additional confidence index. This value ranges from 0% to 100% and refers to the accuracy of the data. This value depends on the number of recent measurements, the

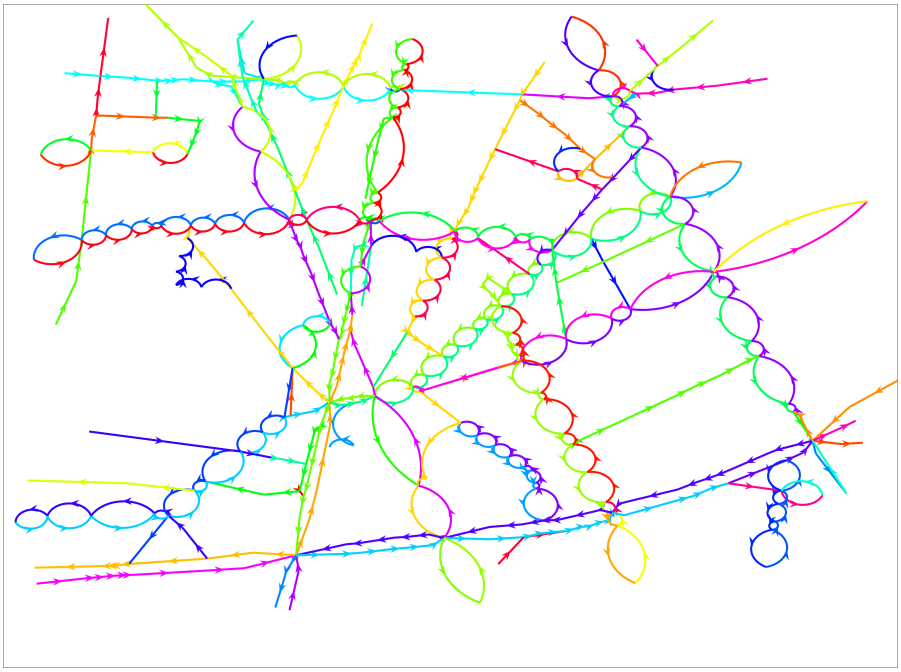


Figure 5.6: Map of road partitions for FCD speed calculation. Adjacent roads of the same color belong to the same partition. Roads with FRC equal to 7 are ignored as no FCD is collected for them.

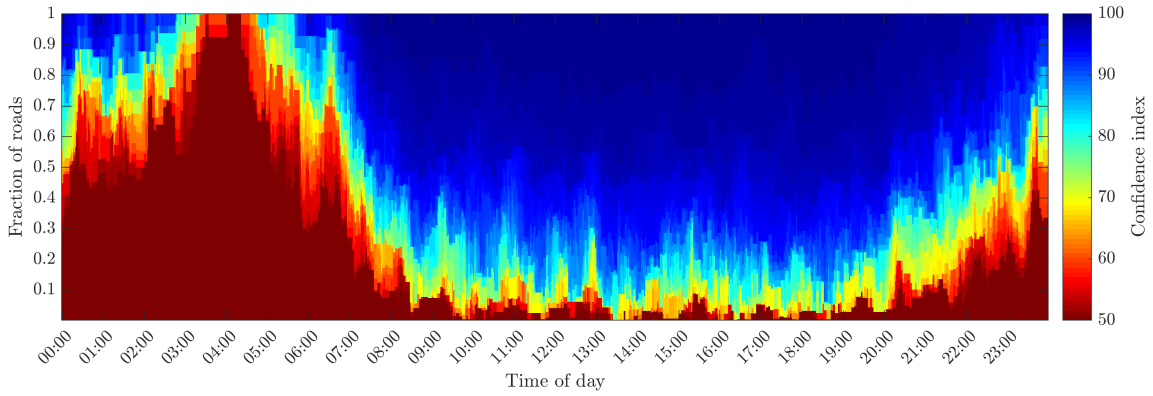


Figure 5.7: Distribution of the confidence index for all roads during December 21, 2020.

standard deviation, and the total volume of measurements. According to TomTom, a value of 100% means that the provided data is the highest quality data, whereas lower values indicate the degree that the data may vary from the actual conditions on the road.

For instance, Figure 5.7 shows the confidence index values for the entire day of December 21, 2020. In the figure, the left-axis shows the fraction of vehicles that have a confidence index less than or equal to the value given by the colorbar. Note that the confidence index is always greater than or equal to 50%. During nighttime, between 00h00 and 5h00, we see that the confidence index is at its lowest values, which can be explained by the low vehicular flow expected during this time, which results in a small number of FCD measurements. As the working hours start around 6h00, the confidence index increases due to an increase in the number of vehicles in the network. During a high portion of the day, between 9h00 and 21h00, the FCD has a high confidence index as 80% of the roads have values greater than 90%.

5.2.4 Bluetooth vehicle identifiers

To obtain direct measurements of TRs, Bluetooth (BT) reader devices were located at the adjacent roads for a selection of intersections. These locations were selected using the method described in Section 3.3.3, which assigns a sensitivity weight to each intersection: errors in the TRs for intersections with a higher weight would generate larger errors in the density estimation. In total, 12 locations were selected as shown in Fig. 5.8, corresponding to the intersections with the highest sensitivity weights in order to reduce the estimation error.

Due to technical and economical constraints, these data is not available in real time. Instead, campaigns of one week duration were organized for each intersection, for a total of 4 campaigns. The timetable for these campaigns is shown in Table 5.2.

To illustrate the data collection process, Fig. 5.9 shows the location of BT devices around one of the chosen intersections. For each sensor pair, the detected vehicle IDs are compared with their corresponding timestamps. Thus, it is possible to assign the origin and destination roads for each of the detected vehicles. This data is aggregated during a time period of 1 hour, so the available information is the number of counts for each of the possible turns in the intersection. From this, TRs are computed as

$$r_{i,j}^{\mathrm{BT}}(t) = \frac{\mathrm{Counts}(i,j,t)}{\sum_{k} \mathrm{Counts}(i,k,t)} \tag{5.1}$$

Table 5.2: Measurement campaigns for the collection of BT data.

Campaign	Intersections	Begin date	End date
1	5, 6, 7	September 21, 2020 - 12h00	September 28, 2020 - 13h00
2	1, 4, 10, 11	September 30, 2020 - 12h00	October 08, 2020 - 8h00
3	3, 9, 12	October 12, 2020 - 13h00	October 21, 2020 - 8h00
4	2, 8	October 21, 2020 - 14h00	October 28, 2020 - 14h00

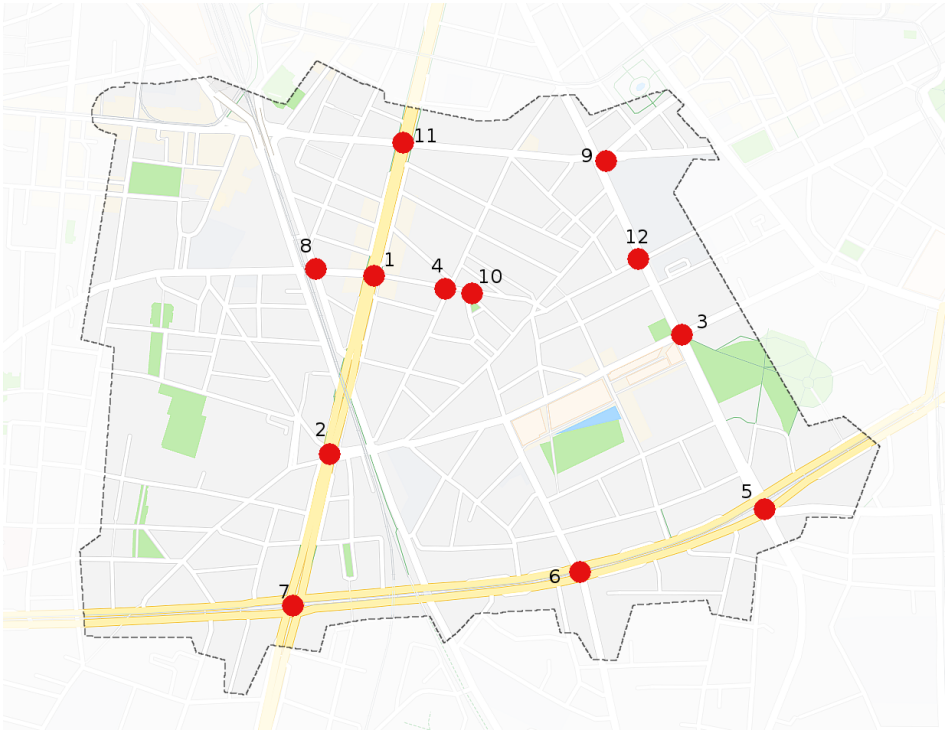


Figure 5.8: Intersections used to collect TR data using BT identifiers in red, with their location ID.

where Counts(i, j, t) is the number of counts from road i to road j via the BT during the time interval $[t - \Delta t, t]$, with $\Delta t = 1$ hour.

To evaluate the quality of the data, Fig. 5.10 shows the time series of the TR values for one road. A zoom of this figure for only one day (Oct 22) is shown in Figure 5.11. Note that during the day hours (6h00 to 20h00), the TR values are fairly constant, lying in a range of ± 0.07 around the mean. The peaks that can be seen between 23h00 and 6h00 are due to the low vehicular flow during these times, which results in a very low number of measurements and the estimation may not be good.

The mean TR values are computed using the total number of vehicle detections during the duration of the campaign,

$$r_{i,j}^{\mathrm{BT}} = \frac{\mathrm{TotalCounts}(i, j)}{\sum_k \mathrm{TotalCounts}(i, k)}. \tag{5.2}$$

For the rest of the document, the TR parameters for the selected intersections are given by (5.2).

As BT technology is used, data is provided only by a fraction of the vehicles in the network. Taking into account the daytime hours (6h00 to 20h00) for intersection 2, the turn with the minimum number of counts reported an average of 8 detections per hour, and the turn with

Figure 5.9: Location of BT devices around intersection 2, with their corresponding ID in database.

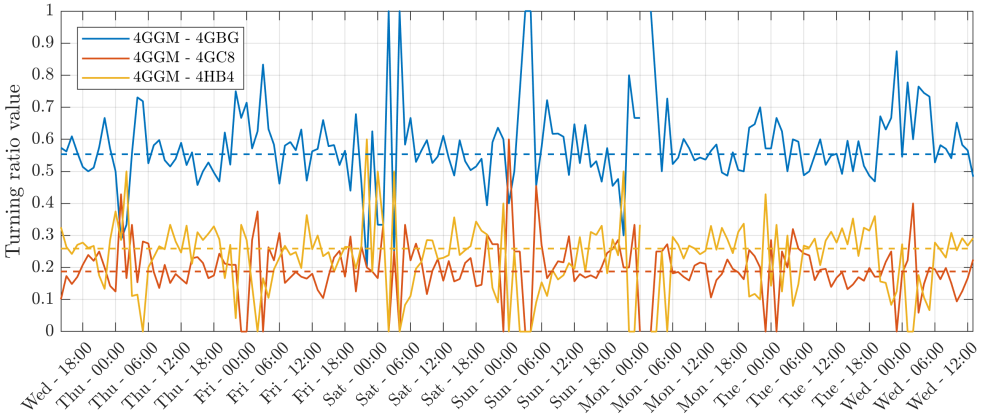


Figure 5.10: Time series for the estimated TRs for one road in intersection 2, from Oct 21 to Oct 28, 2020. Dashed lines are the mean values for each turn.

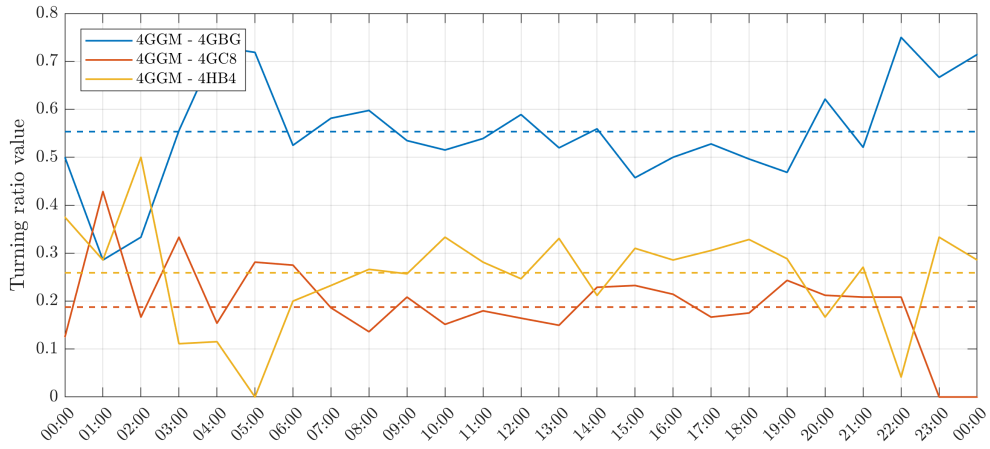


Figure 5.11: Time series for the estimated TRs for one road in intersection 2, for Thursday Oct 22, 2020. Dashed lines are the mean values for each turn.

the maximum number of counts reported an average of 92 detections per hour. Over all turns, the average is 26 detections per hour.

5.3 Validation of dynamic density-flow estimation

This section evaluates the performance of the flow and density estimator described in Chapter 3, which provides values for each road in the network. Real data from the GTL-Ville is used to deploy the estimation algorithms, and to cross-validate the results.

5.3.1 Sensor classification

From the stationary sensors present in the considered zone (induction loops and radars), three sets were created:

Input set Corresponds to the sensors that are in the boundaries of the zone and provide the external input demands to the network. This information is required as an input to the estimation algorithm. We refer as the input set only to the data oriented towards the inside of the network. Flow data provided by the input set is denoted by $\mathbf{u}(t)$.

Output set Correspond to sensors located at the boundaries of the network that take into account the outbound vehicles. This is provided by microwave radars which are capable of measuring the flow in multiple lanes and directions. Data from these locations are denoted by $\mathbf{y}(t)$.

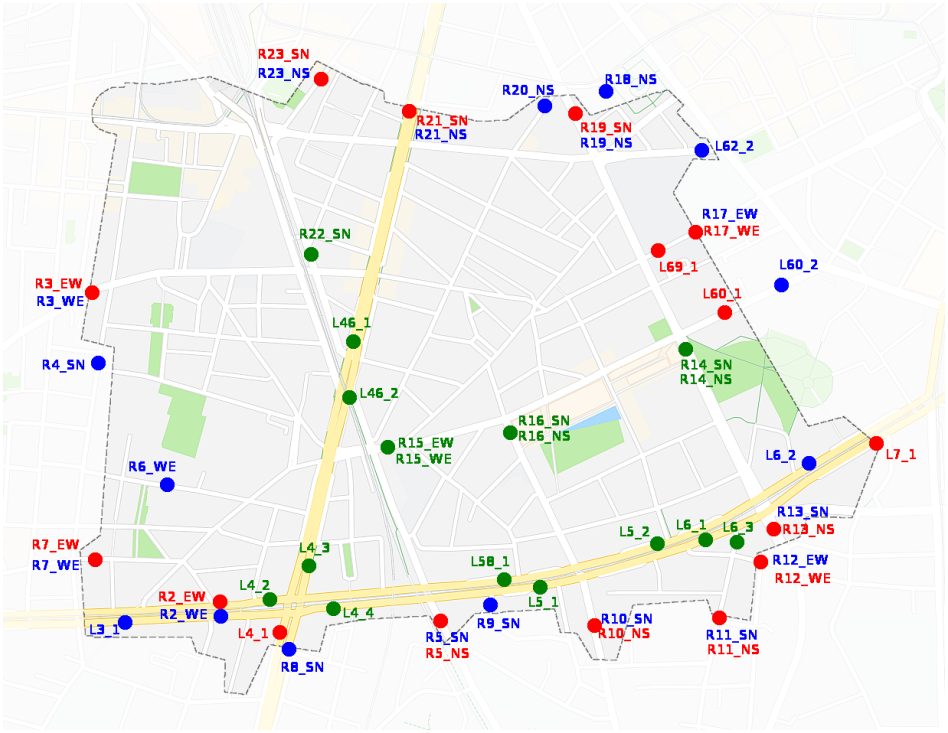


Figure 5.12: Classification of the sensors contained in the area of interest. The input set is shown in blue, the output set is shown in red, and the validation set is shown in green.

Validation set Correspond to sensors located in selected internal locations of the network, which will be used for the cross-validation of the estimator. Data from these sensors will not be used by the estimator, and provide ground truth data to compare with the estimates.

The location of sensors for each of this class is shown in Fig. 5.12. Note that some of the radar sensors locations present two identifiers belonging to different sets. These are the locations that allow the measurement of flow in both directions.

5.3.2 Deployment of the flow and density estimator

To estimate the flow and density for each road, we use the dynamic estimator proposed in Chapter 3,

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\boldsymbol{\rho}}(t) = L^{-1}(R^\top - \mathbb{I})\hat{\boldsymbol{\varphi}}^{\mathrm{out}}(t) + B\mathbf{u}(t) \tag{5.3}$$

$$\hat{\boldsymbol{\varphi}}^{\mathrm{out}}(t) = V_{FCD}(t)\hat{\boldsymbol{\rho}}(t) \tag{5.4}$$

where L is a diagonal matrix with the road lengths, and R is the TR matrix. The input vector $\mathbf{u}(t)$ contains the flow data from the sensors in the input group, and matrix B maps each input to its corresponding road index. The diagonal matrix $V_{FCD}(t)$ contains the speed data provided by TomTom. Roads with very few number of measurements which do not provide FCD speed data are assumed to have a low vehicular flow, so we use a speed value equal to

the speed limit. The discretized version of the estimator is done by using Algorithm 3.1. The output estimates are computed as

$$\hat{\mathbf{y}}(t) = C\hat{\boldsymbol{\varphi}}^{\text{out}}(t) \tag{5.5}$$

where C is a binary matrix that selects the road indexes corresponding to the output set. The estimation error is

$$\mathbf{e}(t) = \mathbf{y}(t) - \hat{\mathbf{y}}(t) \tag{5.6}$$

5.3.3 Parameter estimation

For the estimation, values of the TRs are required at every intersection. As described in Section 5.2.4, these parameters are measured for some set of intersections. To provide values for the remaining intersections, we propose the use of an heuristic that uses the FRC of each road to estimate the TRs.

An initial proposition was described in Appendix B, which is based on the hypothesis that TRs are proportional to the capacity of each outgoing road. Nevertheless, preliminary tests using this estimates in the Grenoble city center showed that this choice was not satisfactory. Although Appendix B provided an experimental validation of this method, this was originally done in a single Highway, where all segments were homogeneous. However, when applied to an urban setting where roads have different purposes and roles as specified by their FRC, this caused underestimation in major roads that shared intersections with minor roads, albeit with the same capacity.

As an alternative, we consider the use of the FRC information to estimate the TRs. For each FRC class in the set $\{1, 2, \ldots, 7\}$, we define a weight $\theta \in (0, 1]$. Let $\boldsymbol{\theta} \in (0, 1]^7$ be the vector of class weights. Suppose that the TRs at each intersection are distributed proportionally to the class weights of each of its outgoing roads. Thus, TRs are computed as

$$r_{i,j}^{FRC} = \frac{\theta_{\text{FRC}(j)}}{\sum_{k \in \mathcal{O}(n_i)} \theta_{\text{FRC}(k)}} \tag{5.7}$$

where $\text{FRC}(i)$ is the FRC class of road i, and $\mathcal{O}(n_i)$ is the set of outgoing roads from intersection n_i, to which the input i is adjacent. However, for the turns corresponding to intersections where BT sensors were installed, their TRs are instead computed from the collected data.

Let $\mathcal{B}$ be the set of intersections equipped with BT sensors. For the remaining intersections where data is not available, the value of the weight class vector is required. Consider the

Table 5.3: Value of FRC weights for the estimation of TR parameters. Class 2 has no assigned weight as no roads with this class are contained in the area.

Class index	1	2	3	4	5	6	7
Class weight θ	1.00	N/A	1.00	0.50	0.23	0.13	0.04

following optimization problem

$$\begin{aligned} \min_{\boldsymbol{\theta}} \quad & ||\bar{\mathbf{y}} - C(\mathbb{I} - R^\top(\boldsymbol{\theta}))^{-1} B\bar{\mathbf{u}}|| \\ \text{subject to} \quad & \boldsymbol{\theta} \in (0,1]^7, \\ & \theta_1 = 1, \\ & r_{i,j}(\boldsymbol{\theta}) = \begin{cases} r_{i,j}^{\mathrm{BT}} & \text{if} \quad n_i \in \mathcal{B} \\ \theta_{\mathrm{FRC}(j)} / \sum\limits_{k \in \mathcal{O}(n_i)} \theta_{\mathrm{FRC}(k)} & \text{if} \quad n_i \notin \mathcal{B} \end{cases}. \end{aligned} \tag{5.8}$$

where n_i is the destination intersection of road i, i.e., $i \in \mathcal{I}(n_i)$, $r_{i,j}^{\mathrm{BT}}$ is computed by (5.2), and

$$\bar{\mathbf{u}} = \frac{1}{T}\int_0^T \mathbf{u}(t) \quad , \quad \bar{\mathbf{y}} = \frac{1}{T}\int_0^T \mathbf{y}(t). \tag{5.9}$$

are the average flows from the input and output sets, respectively.

This optimization problem is based on the assumption that, over a long enough time period, the average flows in a network behave as if in steady-state, and are only subject to the flow conservation laws at intersections as described in Chapter 2. The estimated steady-state flow is computed as

$$\bar{\boldsymbol{\varphi}} = (\mathbb{I} - R^\top)^{-1} B\bar{\mathbf{u}}. \tag{5.10}$$

and the estimated outputs are $C\bar{\boldsymbol{\varphi}}$. Thus, the optimization problem tries to minimize the difference between the average validation flows and the corresponding steady-state flows, by changing the values of the TR matrix according to the weights of the FRC classes. The condition θ_1 is set arbitrarily without loss of generality, as only the relative differences between the weights are important.

Due to the limited number of parameters, this problem can be solved by using the well known gradient descent method. We obtained the values shown in Table 5.3, and in Fig. 5.13 for visualization purposes. Note that as the importance of the road decreases, so does the corresponding class weight as is to be expected.

5.3.4 Results

For evaluation purposes, we considered the data collected for January 8, 2021, from 6h00 to 23h00. Figures 5.14 and 5.15 show the time series for the real and estimated flow from the cross-validation sensors.

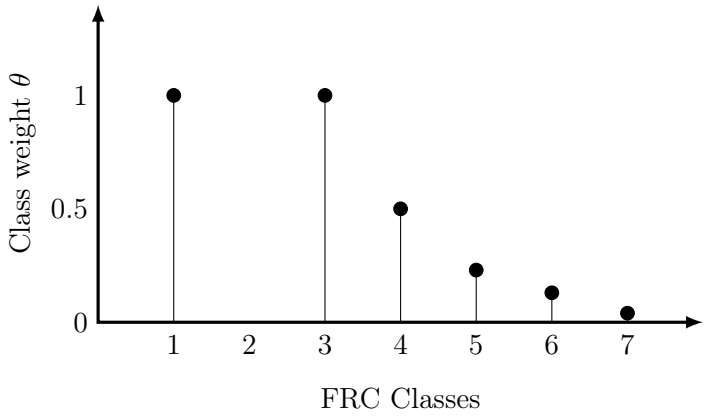


Figure 5.13: Visualization of the class weights for each FRC class.

Note that for most cases, the estimated and real values have a very similar trajectory. This shows that the proposed estimator is able to accurately reconstruct the propagation of flow throughout the network. Nevertheless, some of the locations present different trajectories for the real and estimated flows, showing that there are some mismatches in the estimation. The principal source of error considered to be the deviations of the TRs for some of the network's intersections. Furthermore, internal sinks and sources of vehicles (such as parkings and buildings) are neglected, which has an impact on the accuracy of the estimator.

The effect of deviations in the TR values can be seen in the behavior of sensors R14_SN, R15_EW, and R16_NS. This sensors are interconnected, as R15_EW and R16_NS can be reached from vehicles turning left after passing sensor R14_SN. Thus, error in the flow estimation for the latter has cascading effects that partially explains the deviations presented in the former. Thus, a better estimation of the flow at location R14_SN in the future could improve the overall accuracy of the proposed approach.

As error metrics, we use the Relative Mean Error (RME) and the Relative Absolute Error (RAE), defined as

$$\mathrm{RME}_i = \frac{\left|\int_0^T e_i(t)\mathrm{d}t\right|}{\int_0^T y_i(t)\mathrm{d}t} \quad , \quad \mathrm{RAE}_i = \frac{\int_0^T |e_i(t)|\mathrm{d}t}{\int_0^T y_i(t)\mathrm{d}t}, \tag{5.11}$$

and are shown in Fig. 5.16. The RME shows that the proposed estimator provides close estimates to the real values, as half of the validation locations present an error under 20%, and all of them have an error under 45%.

When considering the RAE, the error increases as this metric considers not only the differences between the mean trajectories, but also takes into account the dispersion of the real data. This indicates that the macroscopic model used for the estimation approach filters out high frequency traffic variations due to vehicle interactions. Despite this limitation, for half of the considered locations the RAE has low values between 20% and 30%.

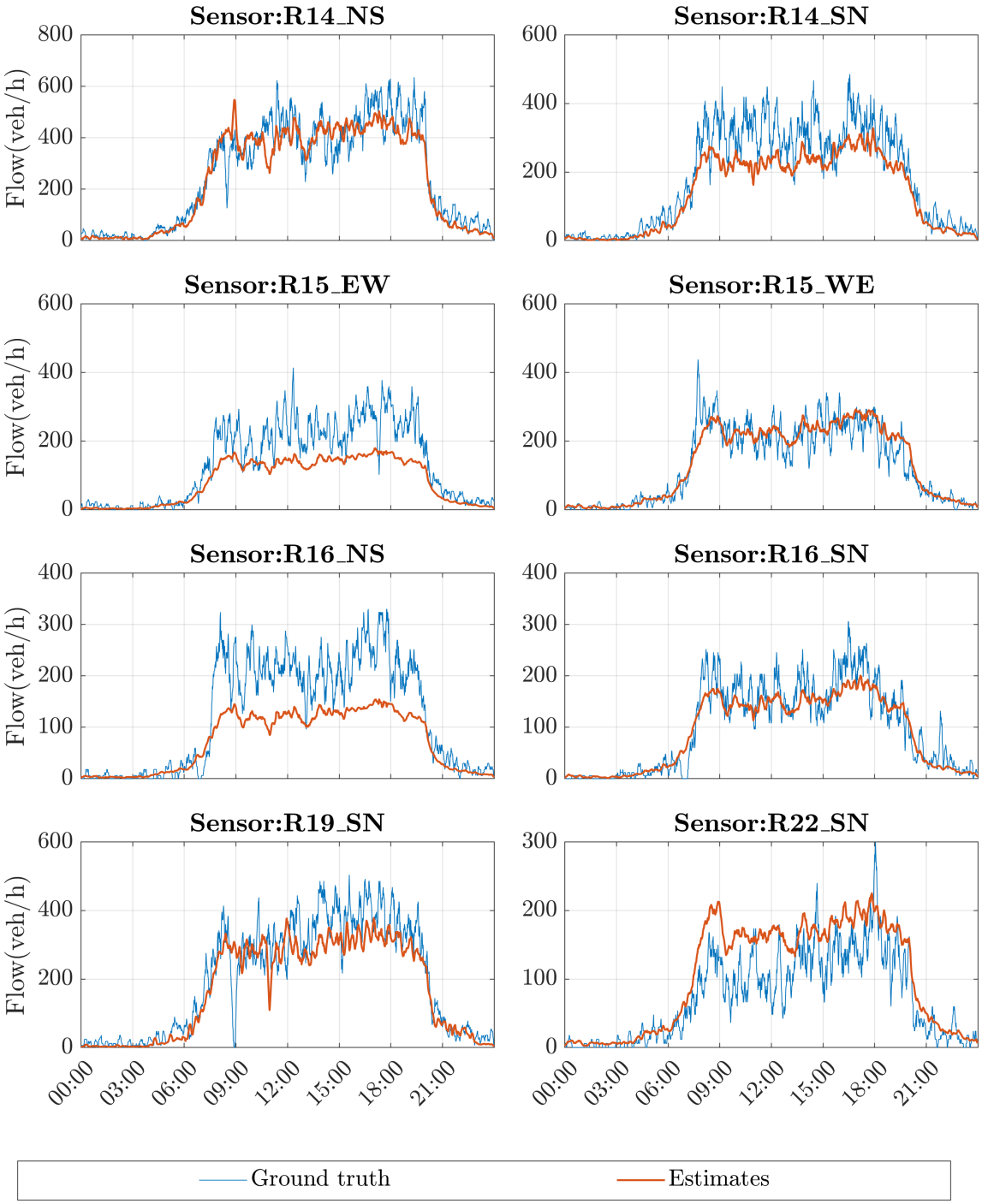


Figure 5.14: Values for the real and estimated flows.

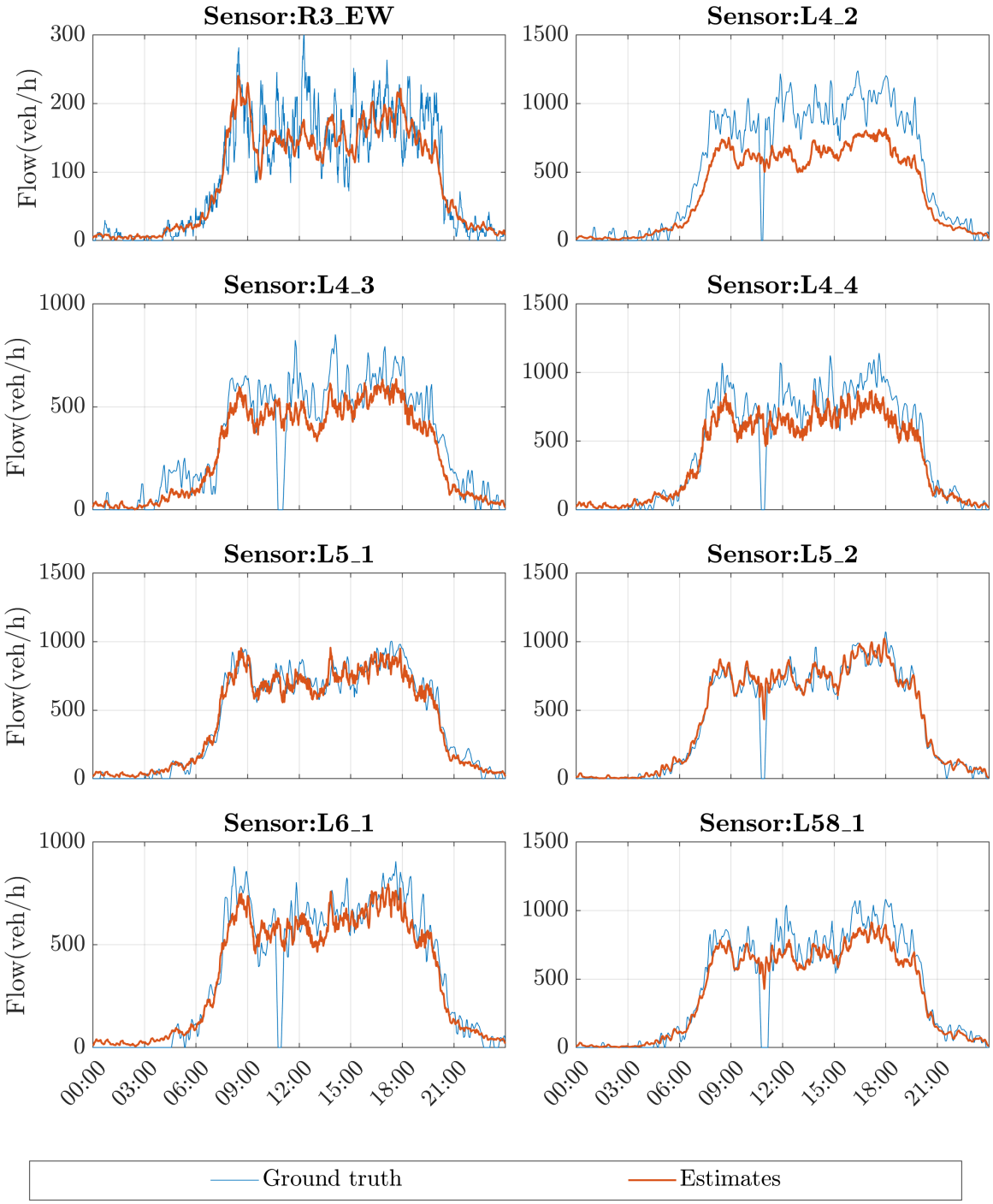


Figure 5.15: Values for the real and estimated flows.

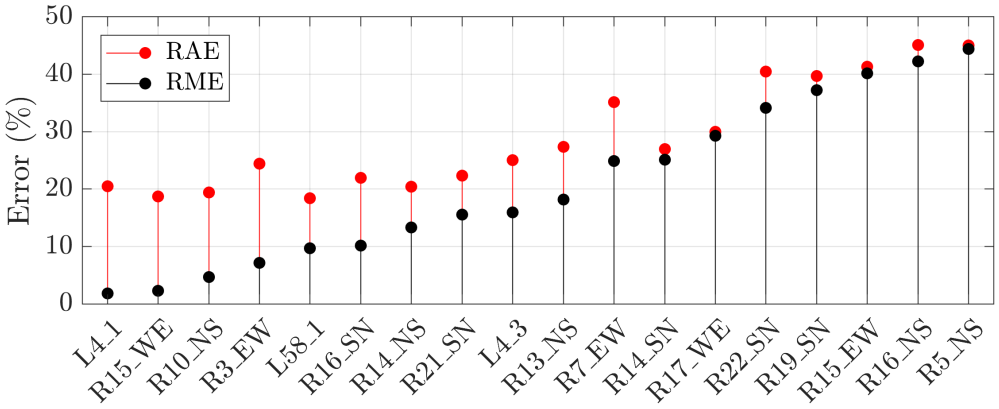


Figure 5.16: Relative estimation error for the validation sensors.

For locations at the right of Figure 5.16, the RME and RAE values are very close to each other. This is due to the fact that for these sensors, most of the error is due to the underestimation bias between the mean trajectories, and the contribution of the up and down peaks of the real data are not as important. All sensors have a RME and RAE less than 45%.

5.4 Validation of average density estimation

This section will evaluate the application of the one-dimensional open loop observer for the average density of an entire network described in Chapter 4, using real data from the Grenoble network. Following the same process described in Section 4.2.1, the roads of the network are divided into two partitions, one corresponding to the boundary roads where measurements are available (the measured roads), and the internal roads without measurements. The density vector is then partitioned as $\boldsymbol{\rho}(t) = [\boldsymbol{\rho}_1^\top(t) \quad \boldsymbol{\rho}_2^\top(t)]^\top$ where $\boldsymbol{\rho}_1(t)$ is the density of the internal roads, and $\boldsymbol{\rho}_2(t)$ is the density of the measured roads. We are interested in the estimation of the average density of the internal roads

$$\rho_{av}(t) = \frac{1}{m}\mathbf{1}^\top\boldsymbol{\rho}_1 \tag{5.12}$$

where m is the number of internal roads.

5.4.1 Estimation approach

As described in Section 4.4.2, the average-density estimator is of the form

$$\dot{\hat{\rho}}_{av}(t) = -\gamma\hat{\rho}_{av}(t) + \frac{\gamma}{\mathbf{n}^\top\mathbf{1}}\mathbf{n}^\top V_1^{-1}(\mathbb{I} - R_{11}^\top)^{-1}R_{21}\mathbf{u}(t). \tag{5.13}$$

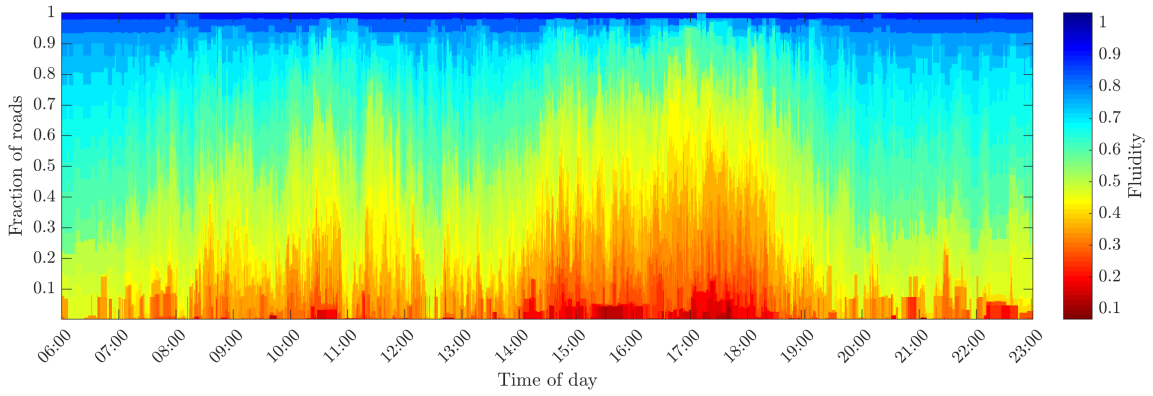


Figure 5.17: Fluidity for the Grenoble network during December 21, 2020.

where R_{11} and R_{21} are blocks of the TR matrix resulting from the partition of the network into the internal and measured roads, such that

$$R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}. \tag{5.14}$$

Similarly, V_1 is a diagonal matrix containing the speed of the internal roads. $\mathbf{n}$ corresponds to the division vector, i.e., each internal road is divided into a number of virtual cells specified by $\mathbf{n}$. γ is the observer gain. $\mathbf{n}$ and γ must be carefully selected to provided error bounds on the convergence of the observer. Chapter 4 proposes Algorithm 4.1 to compute both $\mathbf{n}$ and γ. This algorithm has as inputs the length of each road, its velocity, and the TR matrix.

However, this method assumes that the network's dynamics are linear, which implies that the speed of each road is constant and independent of the number of vehicles. This is not always the case, as phenomena such as congestion can cause great variations in the vehicle speed profile. Consider for instance the mean speed data from FCD for December 21, 2020 shown in Fig. 5.17. This figure shows the fluidity of each road, defined as

$$\text{Fluidity}_i = \frac{v_i(t)}{v_i^{max}} \tag{5.15}$$

where v_i^{max} is the speed limit of road i. The left axis in the figure corresponds to the fraction of roads which have a fluidity value equal to or less than the value indicated by the colorbar (at the right). Note the high variations in the speed values throughout the day, specially during the after noon rush between 15h00 and 19h00.

To deploy the estimator, it is necessary to define a nominal velocity value for each internal road, i.e., a constant matrix $\bar{V}_1$. For instance, Chapter 4 used the speed limit for each road. However, this is not a good choice for the current case, as Fig. 5.17 shows that the vast majority of roads exhibit speed values lower than 80% of the speed limit. To provide a more representative nominal value, we use the average speed computed as

$$\bar{V} = \frac{1}{T}\int_0^T V(t) \tag{5.16}$$

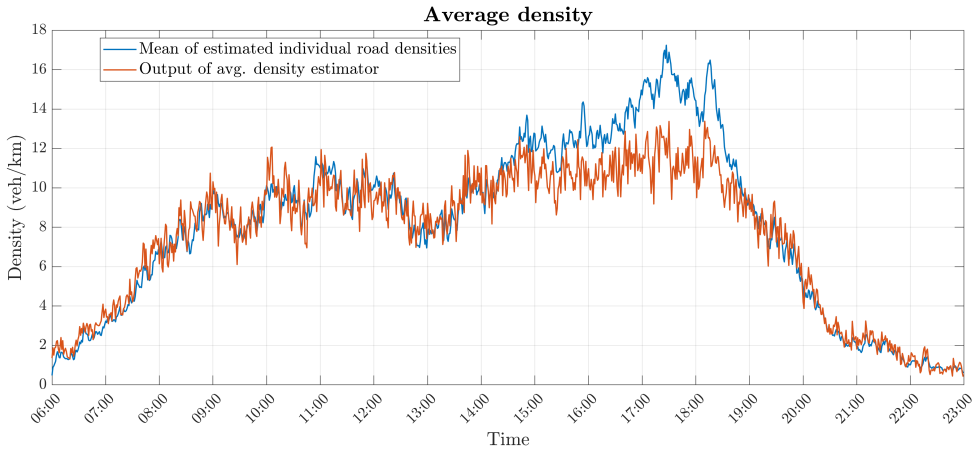


Figure 5.18: Time series for the average density of the Grenoble downtown network.

As input data for the observer, $\mathbf{u}(t)$, we use the data provided by the sensors of the input set as described in Section 5.3.1. TR values are the same as used in Section 5.3.3.

5.4.2 Validation methodology

To directly validate the results of the average-density observer, the ground truth value of the average density if needed. However, as previously described, only a handful of sensors placed in strategic locations are available, so there is no direct measure of the real average density. Therefore, this section considers as proxy for the ground truth the mean of the estimated densities of the individual roads,

$$\rho_{av}(t) = \frac{1}{m}\sum_{i=1}^{m}\hat{\rho}_i(t) \tag{5.17}$$

where $\hat{\boldsymbol{\rho}}(t)$ is calculated via eqs. (5.3) and (5.4) in Section 5.3. Therefore, the error in this case is defined as

$$e(t) = \rho_{av}(t) - \hat{\rho}_{av}(t) = \frac{1}{m}\sum_{i=1}^{m}\hat{\rho}_i(t) - \hat{\rho}_{av}(t) \tag{5.18}$$

Although these individual estimates present errors (Section 5.3.4), as both estimators use the same parameters for the TRs, they are expected to have similar results.

5.4.3 Results

The estimates for the average density are computed using data from 6h00 to 23h00, December 21 - 2020. Figure 5.18 shows the outputs of the observer, and the mean of the individual road estimated densities. Note that for the majority of the time, the estimated average density is

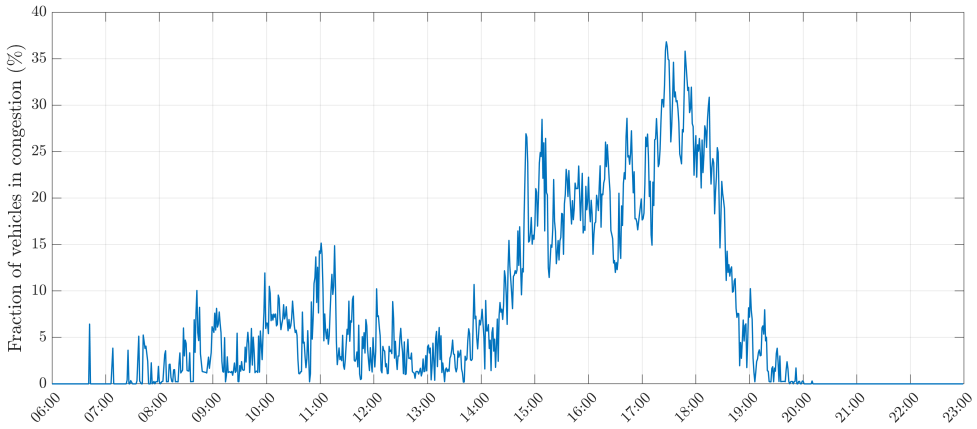


Figure 5.19: Fraction of total vehicles in the network which are in a congested road at every time.

very close to the ground truth values, except for a time window between 15h00 and 19h00, with a peak error at 17h30, where a the average density is underestimated.

To identify the reason for the increase in error, Fig. 5.19 shows the fraction of vehicles that are in a congested road at every time instant. A road i is said to be in congestion if its density $\hat{\rho}_i$ is greater than the critical value ρ^c. For the current analysis, we use the same value ρ^c as in Chapter 2. Note that during the period of 15h00 to 19h00, a large fraction of vehicles are congested, ranging from 15% to 35%. Thus, the error in the average density estimation occurs when the underlying assumptions are violated, namely during congestions.

Note that during the morning, between 9h00 and 12h00, the fraction of vehicles in congestion reaches up to 15%. Although of lower magnitude compared to the afternoon, this is still a noticeable congestion event. Nevertheless, during this period the average density estimates are very close to the real values, and there is no noticeable increase in the error. This indicates that when low-level of congestion are present in the network, the proposed estimator can still provide accurate results.

As an error metric, the total RAE is equal to 12%. During the afternoon congestion (15h00 to 19h00), the obtained RAE increases to 18.3%. For comparison purposes, the RAE for the rest of the time (from 6h00 to 15h00 and from 19h00 to 23h00), the RAE is 8.2%.

5.5 Concluding remarks

This Chapter described the use of the GTL-Ville platform that provides real traffic data from the city of Grenoble to validate the dynamic flow-density estimation approaches proposed in this manuscript. Although the problem of TSE in large networks is challenging, the obtained results are encouraging as the estimated flow for individual roads are very close to the ground truth data provided by sensors. For more than half of the validation locations, the mean

trajectories presented an error below 20%.

For some other locations, the estimated trajectories underestimated the real values. Nevertheless, even in this case the obtained errors were below 45%. We identify as the main error source the uncertainty in the values of the TRs. This is because direct measurements of TRs are only obtained for 12 intersections of the network, and for the rest of the intersections they are estimated based on an heuristic formula based on the properties of the network. However, this can be improved in the future by performing more measuring campaigns, so the estimation results in the real application are expected to improve significantly.

Considering the average density estimation, the results show that during a high portion of the day, the estimates and the ground truth values are very close, with an average error of 8%. However, it was evidenced that when the underlying assumptions of linear dynamics are not satisfied, the trajectories of the real and estimated densities diverged, increasing the error to almost 20%.

Conclusion and perspectives

Traffic state estimation in large scale urban networks is a challenging problem. This is due to several factors such as the difficulty to accurately model the outflow of intersections, to the large number of required parameters which are difficult to measure, and the to lack of sufficient data. In this thesis we have addressed this issues, and provided different solutions that can be used in real application to estimate the density and flow of large-scale traffic networks. Below we will first summarize the main contributions of this work, and then propose some possible future works.

Contributions

Sensor location

The availability of real traffic data is one of the most important elements in the deployment of any TSE application. Due to the high costs related to the construction and maintenance of a sensor network, we have proposed different methods that minimizes the required number of flow and TR sensors, while minimizing the estimation error. In Chapter 2 this is done for stationary networks by considering a trade-off between both sensing technologies. It was found that the inclusion of TR measurements for a subset of intersections can significantly reduce the number of flow sensors, where the reduction depends on the total out-degree on the measured intersections. When considering different costs for both technologies, we proposed a way to calculate the optimal number of sensors for each technology such that the total cost is minimized. Furthermore, an efficient sensor location algorithm is proposed which can be used for very large traffic networks without requiring a high computational cost. However, although useful, this method presents certain limitations. As it is design for stationary networks, the predicted sensor locations are not well-suited for dynamic scenarios. Additionally, as the underlying flow estimation method uses only conservation laws, the predicted number of sensors can still be high for certain applications.

To address these limitations, Chapter 3 proposes an alternative sensor location method that considers TR sensors only. This method assumes that a priori values for the TRs for all intersections are available, which may be subject to deviations from the real values. With this initial guess, a sensitivity weight is computed for each intersection. This weight measures the total error in the density estimation that is due to deviations of the TR estimates from the real values. Then, a small given number of TR sensors are placed in the intersections with the highest sensitivity weights, minimizing the estimation error. This method has the advantage of being flexible for different budgets, and additional information can be used to improve the resulting estimates. Nevertheless, this method is limited by the fact that it requires a priori TR values for all intersections, or a TR estimation methodology. The accuracy of these initial values can affect the quality of the results.

Parameter estimation

For the implementation of the sensor location and estimation methods proposed in this thesis, the TR parameters are a fundamental component. As their measurement for all intersections is unfeasible, we analyzed the use of TR estimation methods based on network parameters that are easily known, such as the number of lanes, the speed limit, and the FRC. We first considered the TR estimation method proposed in Appendix B, which uses only the number of lanes and speed limit of each road to compute the TR estimates. Tests in homogeneous networks such as Highways provided good results, but in heterogeneous urban networks the preliminary results were not satisfactory. As an alternative, Chapter 5 considers instead the use of the FRC of each road to estimate the TRs. This method was applied to real data in the Grenoble downtown, obtaining overall good results.

Although these methods are heuristic and introduce uncertainty in the parameters, the obtained results show that they are an interesting tool that can be used to provide an initial approximation with little economical cost.

Flow and density estimation approaches

To estimate the density and flow in urban traffic network, this thesis proposed three methods that can be used for different applications. The steady-state method proposed in Chapter 2 provided good results, but due to its underlying assumptions, is limited to specific scenarios and can require a large number of sensors and FD parameters.

The data-based dynamic estimation method in Chapter 3 can be applied to a wider range of applications, as it does not assumes steady-state. Furthermore, compared to other works in the literature, this method has the advantage of not using the FD, so fewer parameters are used. This is done by using the road speed from FCD to directly estimate the outflow of each road. As its main drawback, uncertainty in the TR were shown to significantly affect the estimation error. Nevertheless, the results of this method in the real case of the Grenoble downtown are encouraging, as the flow estimates were very close to the real values for a high fraction of the validation locations. Furthermore, inclusion of new TR measurements can be included to improve these initial results.

The third estimation approach proposed in Chapter 4 consists on estimating the average density of an entire region. This differs from the methods of Chapters 2 and 3 that estimate the density of each individual road. Instead, Chapter 4 proposes a one-dimensional observer, such that the average density is estimated directly. This offers the advantage of a much lower need for computational power, when only this aggregated indicator is required. However, this approach is designed for the case of linear dynamics, which is generally not the case in real traffic networks. Specifically, the presence of high levels of congestion were shown to cause errors in the average density estimation. Nevertheless, simulations and the application to the real case of Grenoble show that for most of the time this estimator can provide good results, even when mild congestions are present in the network.

Application in real networks

Chapter 5 of this document presented the experimental platform GTL-Ville, which consist of the analysis of the network of the city of Grenoble, where real data from FCD, induction loops, radars, and BT vehicle identifiers is available. This platform was used to validate the results of the different approaches proposed in this thesis. On one hand, part of the sensors and measurement campaigns that were realized in this area, were decided using the proposed sensor location methods. On the other hand, the real data was used to deploy the algorithms to estimate the flow and density for the roads of the network. All the information concerning the raw data and the estimated indicators are available to the public, providing value to the users of the network, traffic operators, and to the scientific community.

Perspective and extensions

Although the methods described in this research work have provided good initial results, there are still open problems and questions that can provide significant improvements to the accuracy of the estimations, and can be the base for future research.

One clear path is the improvement of the TR estimation methods. We have shown that using the FRC of each road to estimate these parameters can lead to useful estimations of flow propagation in the network. Nevertheless, the analysis we performed for this model is by no means extensive, and the consideration of other parameter such as the geometry of the network (e.g. connection angle between roads) can lead to more precise results. To do this, further measurements of TR in the network are required, which in turn will improve the dynamic flow and density estimation. Another possible improvement is to consider path-based rather than intersection-based approaches, as these take into account not only the network structure, but also additional information relating the possible routes between origins and destinations.

An important open question is how to deal with congestion for the case of the average density estimator. This problem is not trivial, as it is mainly concerned to the study of the detectability and observability of non-linear systems, for which the theoretical tools are limited. The main issue regarding this extension is that keeping tract of congestion requires to store the state of each individual road, which defeats the purpose of having a low-dimensional estimator. Nevertheless, further study of this problem could developed new theoretical tools able to provide a more general descriptor of the average traffic dynamics.

As a continuation of this PhD thesis, one of the main goals is to give further value to the real data from the city of Grenoble and the proposed estimation approaches to improve the GTL-Ville platform. For instance, this can be done by using the estimated flows and densities as input for methods that predict the energy consumption and vehicle emissions which are generated by the users of the network. This information is crucial for the development of ecological traffic management, which develop strategies to improve the sustainability of transportation networks.

Appendix A

Complementary proofs

A.1 Validity of Algorithm 2.2

Algorithm 2.2 presented in Section 2.3.2 constructs a basis for the null space of $L(\mathcal{R}^*)$ and then locates sensors in such a way that the dimension of $\ker L(\mathcal{R}^*)$ is reduced to 0. Recall that $C(\mathcal{S})$ can be written as the concatenation of rows $\mathbf{u}_s^T$ for $s \in \mathcal{S}$. It can be shown that if there exists any $\mathbf{v} \in \ker L(\mathcal{R}^*)$ such that $\mathbf{v}(i) \neq 0$, then $\mathbf{u}_i^T$ is linearly independent to the rows of $L(\mathcal{R}^*)$, and thus, this row reduces the dimension of the null space.

In the following propositions, we show that the proposed algorithm provides a solution to problem (2.15). First, we show that the algorithm locates exactly the optimal number of flow sensors n_s^*. Subsequently, we show that the collection of $\mathbf{u}_s^T$ for $s \in \mathcal{S}$ is linearly independent to the rows of $L(\mathcal{R}^*)$.

A.1.1 Number of sensors

Proposition A.1. *Given any feasible network* $\{\mathcal{C} \cup \mathcal{N}, \mathcal{E}\}$ *and any set* $\mathcal{R}^* \subset \mathcal{N}$*, the graph* $\{\mathcal{N}_T, \mathcal{E}_T\}$ *generated by Algorithm 2.2 is a spanning tree of* $\{\mathcal{N} \cup \{v_0\}, \mathcal{E}\}$.

Proof. From the original graph, we know that every edge is part of a directed path that begins with an element of $\mathcal{E}_{\text{in}}$ and ends with an element of $\mathcal{E}_{\text{out}}$. Because of this, step 1 of the algorithm generates a strongly connected graph: for any node $k \in \mathcal{N}$, there is a directed path from v_0 to k and another from k to v_0. As step 2 implies the removal of edges, it is possible that the result is a disconnected graph. For now, consider the case when the graph is connected. Step 3 performs a DFS over a connected graph, so every node is visited, and thus step 4 is not performed. Therefore, $\{\mathcal{N}_T, \mathcal{E}_T\}$ is a spanning tree of $\{\mathcal{N} \cup \{v_0\}, \mathcal{E}\}$.

Now consider the case when step 2 generates a disconnected graph. Assume that there are q connected components $\mathcal{G}_i = \{\mathcal{N}_i, \mathcal{E}_i\}$, $i = 0, 1, \ldots, q-1$, and let $\mathcal{G}_0$ be the subgraph containing node v_0. The DFS in step 3 will visit only the nodes in $\mathcal{G}_0$. In step 4, we begin by searching some $k \in \mathcal{R}$ that belongs to some $\mathcal{G}_i$, $i \neq 0$, such that originally there is an edge $j \in \mathcal{O}(k)$ that would connect $\mathcal{G}_i$ and $\mathcal{G}_0$. As the original graph is strongly connected, every node is part of a directed path connecting to $v_0 \in \mathcal{G}_0$, therefore such k and j must always exist. Next, the single remaining outgoing edge of k is also removed. If this causes $\mathcal{G}_i$

to become disconnected, denote $\mathcal{G}_q$ as the new connected component, which will be treated in the following iterations. The DFS starting from k will generate a spanning tree of $\mathcal{G}_i$. Then, the addition of edge j connects subgraphs $\mathcal{G}_i$ and $\mathcal{G}_0$ without creating any cycles. This process is carried out iteratively until all nodes haven been visited, creating a spanning tree of $\{\mathcal{N} \cup \{v_0\}, \mathcal{E}\}$. □

Corollary A.1. *For any set of intersections $\mathcal{R}^*$, with cardinality $|\mathcal{R}^*| = n_{\mathcal{R}}$, the described algorithm locates n_s^* sensors, where n_s^* follows Corollary 2.1.*

Proof. From step 1, we start with a graph that has $n_{\mathcal{N}} + 1$ nodes and $n_{\mathcal{E}}$ edges. At the end of step 4, the algorithm obtains a spanning tree of this graph, which is known to have $n_{\mathcal{N}}$ edges. It follows that there are $n_{\mathcal{E}} - n_{\mathcal{N}}$ edges not contained in the tree. However, step 2 removes $\deg^{\text{out}}(k) - 1$ edges for each $k \in \mathcal{R}^*$, for a total of $\sum_{k\in\mathcal{R}^*} \deg^{\text{out}}(k) - n_{\mathcal{N}}$ removed edges. As sensors are to be located in edges not belonging to the tree nor the set of removed edges, we end up with $n_{\mathcal{E}} + n_{\mathcal{R}}^* - \sum_{k\in\mathcal{R}^*} \deg^{\text{out}}(k) - n_{\mathcal{N}}$ locations, which is the same result as Corollary 2.1. □

A.1.2 Linearly independence of sensor locations

The algorithm must also find the location of these sensors such that the rows of $C(\mathcal{S})$ and $L(\mathcal{R}^*)$ are linearly independent. To do this, we first discuss the relationship of the null space of matrix $L(\mathcal{R}^*)$ with the cycles of the graph. Then, we show that by placing sensors in edges that "break" the cycles we obtain a set of linearly independent rows to matrix $L(\mathcal{R}^*)$.

Definition A.1. Given a path $\mathcal{P} \subset \mathcal{E}$, $\mathcal{P} = \{e_1, e_2, \ldots, e_l\}$ such that there are no repeated edges, the corresponding path vector is defined as $\mathbf{v} \in \{-1, 0, 1\}^{n_{\mathcal{E}}\times 1}$ where $\mathbf{v}(e_1) = 1$, $\mathbf{v}(e_i) = 0$ if $e_i \notin \mathcal{P}$, and $\mathbf{v}(e_i) = \mathbf{v}(e_{i-1})$ if edges e_i, e_{i-1} have the same direction or $\mathbf{v}(e_i) = -\mathbf{v}(e_{i-1})$ else.

Lemma A.1. *Let $\mathcal{P}$ be a cycle that does not include any node belonging to $\mathcal{R}^*$. The corresponding path vector $\mathbf{v}$ belongs to the null space of $L(\mathcal{R}^*)$.*

Proof. If $\mathcal{R}^* = \varnothing$, then $L(\varnothing) = B(\mathcal{N})$ is the incidence matrix of the graph $\{\mathcal{N} \cup \{v_0\}, \mathcal{E}\}$ and the proposition has already been proved by [Cas+14].

For an arbitrary $\mathcal{R}^*$, as the nodes comprised by $\mathcal{P}$ do not belong to $\mathcal{R}^*$, then each one of them is associated to a row of $B(\mathcal{U})$. As this matrix is just a reduced version of $B(\mathcal{N})$, $B(\mathcal{U})\mathbf{v} = \mathbf{0}$ must hold.

We can see that $j \in \mathcal{P} \iff j \notin \bigcup_{k\in\mathcal{R}^*} \mathcal{O}(k)$ and $j \notin \bigcup_{k\in\mathcal{R}^*} \mathcal{I}(k)$. Therefore, $A(\mathcal{R}^*)\mathbf{v} = \mathbf{0}$. With these two results, $L(\mathcal{R}^*)\mathbf{v} = \mathbf{0}$. □

Lemma A.2. *Let $k \in \mathcal{R}^*$ such that $e_0 \in \mathcal{I}(k)$ and $\mathcal{O}(k) = \{e_1, e_2, \ldots, e_q\}$. Let $\mathcal{P}_i = \{e_i, \ldots, e_0\}$ for $i = 1, \ldots, q$ be a feasible path, with associated path vector $\mathbf{v}_i$, such that it does not include any node from $\mathcal{R}^*$ other than k. Then $\mathbf{v} = \sum_i r_{e_0,e_i}\mathbf{v}_i$ belongs to the null space of $L(\mathcal{R}^*)$.*

Proof. Note that each of the paths $\mathcal{P}_i$ is a cycle: the path ends with e_0 which is connected via k to the starting edge e_i. We have $B(\mathcal{U})\mathbf{v}_i = \mathbf{0}$. Also, $A(\mathcal{R}^* \setminus \{k\})\mathbf{v}_i = \mathbf{0}$, as there are no shared edges between $\mathcal{P}_i$ and $\bigcup_{h\in\mathcal{R}^*, h\neq k} \mathcal{O}(h)$ or $\bigcup_{h\in\mathcal{R}^*, h\neq k} \mathcal{I}(h)$. It can be seen that

$$A(\{k\})\mathbf{v}_i = \mathbf{u}_i - \begin{bmatrix} r_{e_0,e_1} \\ r_{e_0,e_2} \\ \vdots \\ r_{e_0,e_q} \end{bmatrix}.$$

Thus,

$$\begin{aligned} A(\{k\})\mathbf{v} &= \sum_{i=1}^{q} r_{e_0,e_i} A(\{k\})\mathbf{v}_i \\ &= \sum_{i=1}^{q} r_{e_0,e_i}\mathbf{u}_i - \begin{bmatrix} r_{e_0,e_1} \\ r_{e_0,e_2} \\ \vdots \\ r_{e_0,e_q} \end{bmatrix} \sum_{i=1}^{q} r_{e_0,e_i} \\ &= \begin{bmatrix} r_{e_0,e_1} \\ r_{e_0,e_2} \\ \vdots \\ r_{e_0,e_q} \end{bmatrix} - \begin{bmatrix} r_{e_0,e_1} \\ r_{e_0,e_2} \\ \vdots \\ r_{e_0,e_q} \end{bmatrix} \\ A(\{k\})\mathbf{v} &= \mathbf{0} \end{aligned}.$$

Thus, $A(\mathcal{R}^*)\mathbf{v} = \mathbf{0}$, which implies that $L(\mathcal{R}^*)\mathbf{v} = \mathbf{0}$ finalizing the proof. □

In general, for each $j \in \mathcal{I}(k)$ for some $k \in \mathcal{R}^*$, the cycles that connect every outgoing edge of k to j must appear together in the elements of $\ker L(\mathcal{R}^*)$.

Theorem A.1
The rows of $C(\mathcal{S})$ calculated via the described algorithm and the rows of $L(\mathcal{R}^*)$ form a linearly independent set.

Proof. The algorithm generates a set $\mathcal{S}$ of sensor locations such that these edges do not belong to a constructed spanning tree or a set of removed edges. It is well known that adding an edge $s \in \mathcal{S}$ to the tree generates a single cycle $\mathcal{P}_s$. Additionally, if the corresponding path vector to $\mathcal{P}_s$ is $\mathbf{v}_s$, the collection of $\mathbf{v}_s$ for $s \in \mathcal{S}$ is a linearly independent set. Consider a partition of $\mathcal{S}$ in $\mathcal{S}_\mathcal{U}$ and $\mathcal{S}_\mathcal{R}$, such that $\mathcal{S}_\mathcal{U}$ consists of the edges that when added to the tree will generate cycles that do not include any node from $\mathcal{R}^*$.

Consider one element $s_0 \in \mathcal{S}_\mathcal{U}$. From Lemma A.1, $\mathbf{v}_{s_0} \in \ker L(\mathcal{R}^*)$. As $s_0 \in \mathcal{P}_{s_0}$ then $\mathbf{v}_{s_0}(s_0) \neq 0$, therefore $\mathbf{u}_{s_0}^T$ is linearly independent to the rows of $L(\mathcal{R}^*)$. Now define $L' = [L(\mathcal{R}^*)^T \quad \mathbf{u}_{s_0}]^T$. Consider another $s_1 \in \mathcal{S}_U$, $s_0 \neq s_1$. Because $s_1 \notin \mathcal{P}_{s_0}$, it is clear that

$\mathbf{u}_{s_0}(s_1) = 0$, hence $\mathbf{v}_{s_1} \in \ker L'$. Following the previous reasoning we obtain that $\mathbf{u}_{s_1}^T$ is linearly independent to the rows of L'. Repeating this process iteratively, it can be seen that the collection of $\mathbf{u}_s^T$ for all $s \in \mathcal{S}_{\mathcal{U}}$ is linearly independent to the rows of $L(\mathcal{R}^*)$.

Now consider one element $s_0 \in \mathcal{S}_{\mathcal{R}}$ which forms a cycle $\mathcal{P}_{s_0}$ that includes some $k \in \mathcal{R}^*$ when added to the tree. From Lemma A.2, there is at least one vector $\mathbf{v}$ in the null space of $L(\mathcal{R})$ such that $\mathbf{v}(s_0) \neq 0$, so $\mathbf{u}_{s_0}^T$ is linearly independent to $L(\mathcal{R}^*)$. Consider another $s_1 \in \mathcal{S}_{\mathcal{R}}$ associated with cycle $\mathcal{P}_{s_1}$ such that $\mathcal{P}_{s_1}$ also includes node k. If there exists $i \in O(k) \cap \mathcal{P}_{s_0}$ and $j \in O(k) \cap \mathcal{P}_{s_1}$ with $i \neq j$, then Lemma A.2 implies that the path vectors $\mathbf{v}_{s_0}, \mathbf{v}_{s_1}$ form part of the same $\mathbf{v} \in \ker L(\mathcal{R}^*)$. Therefore, $\{\mathbf{u}_{s_0}^T, \mathbf{u}_{s_1}^T\}$ and the rows of $L(\mathcal{R}^*)$ do not form a linearly independent set.

Nevertheless, step 2 of the algorithm removes all but one of the outgoing edges for the intersections in $\mathcal{R}^*$, implying that it is not possible to form two cycles $\mathcal{P}_{s_0}, \mathcal{P}_{s_1}$ that include two different outgoing edges from the same $k \in \mathcal{R}^*$ by only adding elements from $\mathcal{S}$ to the tree. Thus, all the path vectors $\mathbf{v}_s$ for all $s \in \mathcal{S}_{\mathcal{R}}$ are related to linearly independent vectors in the null space of $L(\mathcal{R}^*)$. Hence, the collection of $\mathbf{u}_s^T$ for all $s \in \mathcal{S}_{\mathcal{R}}$ are linearly independent to the rows of $L(\mathcal{R}^*)$.

Furthermore, note that as the cycles generated by the elements of $\mathcal{S}_{\mathcal{U}}$ do not include any node from $\mathcal{R}^*$, Lemma A.2 does not apply to them, and thus, they are independent to the cycles formed by the elements of $\mathcal{S}_{\mathcal{R}}$. Hence, the rows of $C(\mathcal{S})$ and $L(\mathcal{R}^*)$ form a linearly independent set. □

A.2 Proof of Theorem 3.1

Consider the linear transformation $\mathbf{x}(t) = L\mathbf{e}(t)$, subject to

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = (R^\top - \mathbb{I})L^{-1}V(t)\mathbf{x}(t) \tag{A.1}$$

note that L is positive definite, and therefore (3.10) is asymptotically stable if and only if (A.1) is asymptotically stable. Consider the Lyapunov candidate function

$$f(\mathbf{x}, t) = \mathbf{x}^\top P^{-1}(t)\mathbf{x} \tag{A.2}$$

where

$$P(t) = \mathrm{diag}(\mathbf{p}(t)) \tag{A.3}$$

and

$$\mathbf{p}(t) = V^{-1}(t)L(\mathbb{I} - R^\top)^{-1}\mathbf{1} \tag{A.4}$$

As $\mathbb{I} - R^\top$ is an invertible M-matrix, its elements are non-negative. Moreover, as $V(t)$ has positive diagonal entries and is bounded, $P(t)$ and $P^{-1}(t)$ are bounded diagonal matrices with positive diagonal entries, so f is positive definite. The derivative of f is

$$\frac{\mathrm{d}}{\mathrm{d}t}f(\mathbf{x}, t) = \mathbf{x}^\top \left(V(t)L^{-1}(R - \mathbb{I})P^{-1}(t) + P^{-1}(t)(R^\top - \mathbb{I})L^{-1}V(t) + \frac{\mathrm{d}}{\mathrm{d}t}P^{-1}(t) \right) \mathbf{x}. \tag{A.5}$$

Consider

$$Q(t) = P(t)V(t)L^{-1}(R - \mathbb{I}) + (R^\top - \mathbb{I})L^{-1}V(t)P(t) - \frac{\mathrm{d}}{\mathrm{d}t}P(t) \tag{A.6}$$

Note that

$$\frac{\mathrm{d}}{\mathrm{d}t}f(t) = \mathbf{x}^\top(t)P^{-1}(t)Q(t)P^{-1}(t)\mathbf{x}(t) \tag{A.7}$$

where we have use the fact that

$$\frac{\mathrm{d}}{\mathrm{d}t}P^{-1}(t) = -P^{-1}(t)\left(\frac{\mathrm{d}}{\mathrm{d}t}P(t)\right)P^{-1}(t) \tag{A.8}$$

Furthermore, as $P(t)$ is positive definite, $\dfrac{\mathrm{d}}{\mathrm{d}t}f(\mathbf{x},t)$ is negative-definite if and only if $Q(t)$ is negative-definite.

Consider the first two terms of $Q(t)$,

$$Q_1(t) = P(t)V(t)L^{-1}(R - \mathbb{I}) + (R^\top - \mathbb{I})L^{-1}V(t)P(t) \tag{A.9}$$

Let $q_{i,j}(t)$ be the element in the (i,j) position of $Q_1(t)$. According to the Gershgorin circle theorem, the i-th eigenvalue of $Q_1(t)$, $\lambda_i(t)$, is contained in a disk with center $q_{i,i}(t)$ and radius

$$\sum_{j\neq i}|q_{i,j}(t)|. \tag{A.10}$$

In this specific case, we have

$$q_{i,i} = -2\frac{p_i(t)v_i(t)}{\ell_i} \tag{A.11}$$

and

$$\begin{aligned}\sum_{j\neq i}|q_{i,j(t)}| &= \sum_j\left(\frac{p_i(t)v_i(t)}{\ell_i}r_{i,j}\right) + \sum_j\left(\frac{p_j(t)v_j(t)}{\ell_j}r_{j,i}\right)\\ &\leq \frac{p_i(t)v_i(t)}{\ell_i} + \sum_j\left(\frac{p_j(t)v_j(t)}{\ell_j}r_{j,i}\right)\end{aligned} \tag{A.12}$$

where we have used the property

$$\sum_j r_{i,j} = \begin{cases} 1 & \text{if} \quad i \notin \mathcal{E}_{\text{out}} \\ 0 & \text{if} \quad i \in \mathcal{E}_{\text{out}} \end{cases} \tag{A.13}$$

and $r_{i,i} = 0$ for all i.

As the eigenvalues of $Q_1(t)$ are real, they can be upper-bounded by

$$\lambda_i(t) \leq z_i(t) \tag{A.14}$$

where

$$z_i(t) = -\frac{p_i(t)v_i(t)}{\ell_i} + \sum_{j\neq i}\left(\frac{p_j(t)v_j(t)}{\ell_j}r_{j,i}\right) \tag{A.15}$$

is the sum of the center and radius of the Gershgorin disk. This results in a system of equations of the form

$$\mathbf{z}(t) = -(\mathbb{I} - R^\top)L^{-1}V(t)\mathbf{p}(t) \tag{A.16}$$

Substitution of (A.4) into (A.16) gives $\mathbf{z}(t) = -\mathbf{1}$. This implies that all eigenvalues of $Q_1(t)$ are bounded above by -1, and thus,

$$Q_1(t) \leq -\mathbb{I} \tag{A.17}$$

Now consider $\frac{\mathrm{d}}{\mathrm{d}t}P(t)$, whose i-th diagonal entry is

$$\frac{\mathrm{d}}{\mathrm{d}t}p_i(t) = -\frac{\ell_i}{v_i^2(t)}\frac{\mathrm{d}}{\mathrm{d}t}v_i(t)\sum_j m_{i,j} \tag{A.18}$$

where $m_{i,j}$ is the (i,j) entry of $(\mathbb{I} - R^\top)^{-1}$. Using (3.12),

$$\frac{\mathrm{d}}{\mathrm{d}t}p_i(t) \geq -\frac{v_{min}^2}{v_i^2(t)}\frac{\ell_i}{\ell_{max}}\frac{\sum_j m_{i,j}}{||(\mathbb{I} - R^\top)^{-1}||_\infty}(1-\epsilon) \geq -(1-\epsilon). \tag{A.19}$$

The last inequality follows from the fact that each of the fractions in the equation are less than or equal to 1. This implies

$$\frac{\mathrm{d}}{\mathrm{d}t}P(t) \geq -(1-\epsilon)\mathbb{I}. \tag{A.20}$$

Substitution of (A.17) and (A.20) into

$$Q(t) = Q_1(t) - \frac{\mathrm{d}}{\mathrm{d}t}P(t) \tag{A.21}$$

gives

$$Q(t) \leq -\epsilon\mathbb{I} \tag{A.22}$$

Thus, the Lyapunov function $f(\mathbf{x}, t)$ is positive-definite with negative-definite derivative, which implies that (A.1) (and by extension (3.10)) is asymptotically stable.

A.3 Proof of Theorem 4.5

Because of the independence between the vectors $\mathbf{u}$ and $\mathbf{v}$, we can rewrite the desired probability in terms of the conditional probability

$$P_{\mathbf{u},\mathbf{v}}(|\mathbf{u}^\top\mathbf{v}| < \epsilon) = P_{\mathbf{u}}(|\mathbf{u}^\top\mathbf{v}| < \epsilon | \mathbf{v} = \mathbf{y}) \tag{A.23}$$

where $\mathbf{y} \in \mathbb{S}^m$ is any arbitrary vector.

Consider the set $\mathcal{P}_\epsilon = \{\mathbf{x} \in \mathbf{S}^m : |\mathbf{x}^\top\mathbf{y}| < \epsilon\}$. This can be visualized in the unit m-sphere as all points within a distance ϵ from the equator, where the pole is $\mathbf{y}$. Let $\mathcal{A} = \{\mathbf{x} \in \mathbb{S}^m : \mathbf{x}^\top\mathbf{y} > 0\}$ be the hemisphere of $\mathbb{S}^m$ closest to $\mathbf{y}$, and $\mathcal{B} = \{\mathbf{x} \in \mathbb{S}^m : \mathbf{x}^\top\mathbf{y} \leq 0\}$ be the opposite hemisphere. Note that $P_{\mathbf{u}}(\mathcal{A}) = P_{\mathbf{u}}(\mathcal{B}) = 1/2$.

Define the ϵ-extension $\mathcal{A}_\epsilon = \{\mathbf{x} \in \mathbb{S}^m : d(\mathbf{x}, \mathcal{A}) < \epsilon\}$, and $\mathcal{B}_\epsilon$ defined analogously. It follows that $\mathcal{P}_\epsilon = \mathcal{A}_\epsilon \cap \mathcal{B}_\epsilon$. Due to the fact that $\mathbf{u}$ is independent from the choice of $\mathbf{y}$, the events $\mathcal{A}_\epsilon$ and $\mathcal{B}_\epsilon$ are independent, and thus

$$P_{\mathbf{u}}(\mathcal{P}_\epsilon) = P_{\mathbf{u}}(\mathcal{A}_\epsilon) P_{\mathbf{u}}(\mathcal{B}_\epsilon) \tag{A.24}$$

Because of the concentration of measure phenomenon, (see Theorem 2.3 in [Led05]), we have that

$$\begin{aligned} P_{\mathbf{u}}(\mathcal{A}_\epsilon) &\geq 1 - e^{-(m-1)\epsilon^2/2}, \\ P_{\mathbf{u}}(\mathcal{B}_\epsilon) &\geq 1 - e^{-(m-1)\epsilon^2/2}. \end{aligned} \tag{A.25}$$

Substitution of (A.24) and (A.25) into (A.23) gives

$$P_{\mathbf{u},\mathbf{v}}(|\mathbf{u}^\top \mathbf{v}| < \epsilon) \geq 1 - 2e^{-(m-1)\epsilon^2/2}, \tag{A.26}$$

thus, finalizing the proof.

Appendix B

Turning ratio topology-based model

This section proposes a model to give estimate values for the turning proportions for a given intersection of a traffic network based on a topology-based method. These techniques use only physical parameters of the network at each intersection (e.g.: angle between incoming and outgoing roads, number of lanes, maximum capacity, etc.) to determine the most likely turns. Consider the following assumption,

Assumption B.1. For any intersection, the TR from road i to road j, $r_{i,j}$, is proportional to the maximum capacity of the outgoing road j.

The capacity of a road relies on most cases on parameters such as the critical density, which may be difficult to obtain in practice. We simplify this assumption by estimating the capacity in terms of the maximum vehicle speed $v_i^{\max}$ and the number of lanes Γ_i. Thus,

$$r_{i,j} = \frac{v_j^{\max}\Gamma_j}{\sum_{k\in\mathcal{O}(n)} v_k^{\max}\Gamma_k} \tag{B.1}$$

The values for the maximum velocity can be calculated using FCD, or by using the speed limit of each road. It can be seen that the main advantage of these type of approaches is that they require few parameters to be known, so they can be applied to the entire network. However, because of their simplicity they are more prone to errors and can provide estimates with high error mean and variance. These values can be used as a working initial value for the sensor location scheme described in Section 3.3.3. This flexible approach can improve estimates from an initial cheap alternative.

To evaluate this choice of model for the TRs, data recollected by the Grenoble Traffic Lab (GTL)[1] was used. This data consists of inductive magnetic loop sensors installed in 21 locations along the Rocade highway at the outskirts of the city of Grenoble in the direction North-South. Figure B.1 shows the 21 locations equipped with sensors. To measure TRs, data from sensors located at off-ramps was taken. The 8 locations of the off-ramps with available data are shown in the figure with red stars. Figure B.2 shows an aerial view of the an intersection, corresponding to the label 1 in Figure B.1. The top location marked with an "O" corresponds to an off-ramp, whereas the locations in the middle and bottom marked with "I"s are in-ramps from the East-West and West-East directions, respectively. Note that at

[1] https://gtl.inrialpes.fr

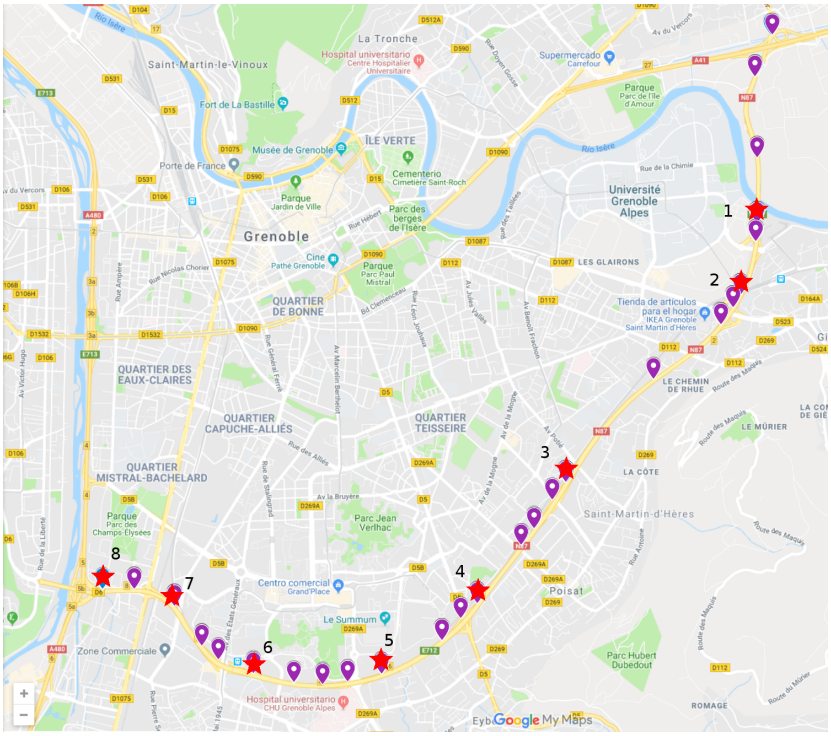


Figure B.1: Sensing locations in the Rocade highway. Purple markers signal each of the 21 locations. Red stars show a subsets of the sensors which correspond to off-ramps such that TRs can be measured.

Figure B.2: Aerial photo of intersection 1. Location of the magnetic loops are shown with dark-green dots. The off-ramp is labeled with a light green marker with an "O".

Location ID	1	2	3	4	5	6	7
Number of lanes Rocade	2	2	2	2	2	2	2
Max. Speed Rocade (km/h)	97	95	98	97	97	94	88
Number of lanes Off-ramp	1	1	1	1	1	1	1
Max. Speed Off-ramp (km/h)	60	50	44	26	58	37	28
Estimated ratio	0.24	0.21	0.18	0.12	0.33	0.17	0.12

Table B.1: Physical parameters and estimated TRs for off-ramps 1 to 7.

Figure B.3: Aerial photo of intersection 8. There are three diverging directions: right, center, left.

each location there are three sensors corresponding to the off-ramp (or in-ramp), the regular lane and the fast-lane. The setup for the remaining 7 locations are similar.

The estimated values for the TRs are obtained using only the topology and physical properties of the network. Table B.1 shows the parameters for the off-ramps 1 to 7. For these intersections there are only 2 diverging directions, namely continuing in the highway or taking the off-ramp. Intersection 8 has a different setup, as it splits into 3 directions as shown in Fig. B.3. The physical parameters for the three directions are shown in Table B.2 with the corresponding TR estimates.

Vehicle count and speed data is available for each sensor with an aggregation time of 5 minutes. Let $\varphi_i^j[k]$ be the vehicle count during the time interval k for the off-ramp $i \in \{1, 2, \ldots, 8\}$ and lane $j \in \{1, 2, 3\}$ (where 1 identifies the off-ramp, 2 the regular-lane, and 3 the fast-lane). For this particular set-up, as there is only one incoming road to the intersection,

Direction	Right	Center	Left
Number of lanes	1	1	1
Max. Speed (km/h)	67	58	76
Estimated ratio	0.33	0.29	0.38

Table B.2: Physical parameters for the outgoing roads at intersection 8.

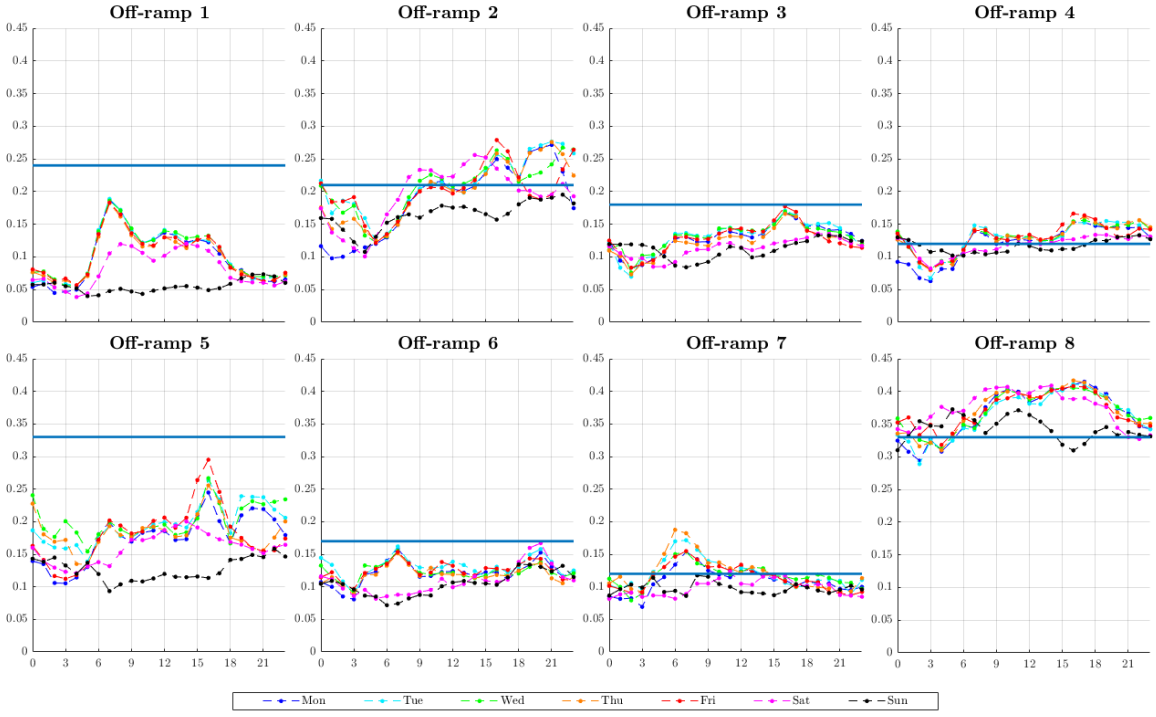


Figure B.4: Time series of the measured TRs for the days of the week for each intersection. For intersection 8 only the right turn ratio is shown.

the TR for the off-ramp i is calculated as

$$r_i[k] = \frac{\sum_{\tau=0}^{T-1} \varphi_i^1[k-\tau]}{\sum_{j=1}^{3}\sum_{\tau=0}^{T-1} \varphi_i^j[k-\tau]} \tag{B.2}$$

where $T \geq 0$ is the length of a sliding window to smooth out the measures of the TRs.

We used 1 year of data from March 2014 to February 2015, with a sliding window of 20 minutes. As expected, the TRs present a variation in time, specifically in terms of the hour of the day and the day of the week. Figure B.4 shows the time evolution of the measured TRs for each time interval and each day of the week. Each point is calculated by taking the median of all data points of the same day of the week at the same time.

Note that for 6 out of 8 intersections, the proposed topology-based model yields a good approximation of the actual values for the TRs.

Bibliography

[AG13] Konstantinos Aboudolas and Nikolas Geroliminis. "Perimeter and boundary flow control in multi-reservoir heterogeneous networks." In: *Transportation Research Part B: Methodological* 55 (2013), pp. 265–281 (cit. on pp. 6, 11).

[AI+04] Luis Alvarez-Icaza et al. "Adaptive observer for traffic density estimation." In: *Proceedings of the 2004 American Control Conference*. Vol. 3. IEEE, 2004, 2705–2710 vol.3 (cit. on pp. 9, 33).

[AjSB13] Michael Abbott-jard, Harpal Shah, and Ashish Bhaskar. "Empirical evaluation of Bluetooth and Wifi scanning for road transport." In: *Australasian Transport Research Forum*. October. Brisbane, Australia, 2013, pp. 1–14 (cit. on p. 7).

[AR00] A. Aw and M. Rascle. "Resurrection of 'second order' models of traffic flow." In: *SIAM Journal on Applied Mathematics* 60.3 (2000), pp. 916–938 (cit. on p. 4).

[AS72] Milton Abramowitz and Irene Stegun. *Handbook of Mathematical Function with Formulas, Graphs, and Mathematical Tables*. Ed. by Milton Abramowitz. 10th. United States Department of Commerce, National Institute of Standards and Technology, 1972 (cit. on p. 60).

[AZG17] Konstantinos Ampountolas, Nan Zheng, and Nikolas Geroliminis. "Macroscopic modelling and robust control of bi-modal multi-region urban road networks." In: *Transportation Research Part B: Methodological* 104 (2017), pp. 616–637 (cit. on p. 6).

[Ban+95] M. Bando et al. "Dynamical model of traffic congestion and numerical simulation." In: *Physical Review E* 51.2 (1995), pp. 1035–1042 (cit. on p. 2).

[BC05] Jaime Barceló and Jordi Casas. "Dynamic Network Simulation with AIMSUN." In: *Simulation Approaches in Transportation Analysis*. New York: Springer-Verlag, 2005, pp. 57–98 (cit. on p. 32).

[BC13] Ashish Bhaskar and Edward Chung. "Fundamental understanding on the use of Bluetooth scanner as a complementary transport data." In: *Transportation Research Part C: Emerging Technologies* 37 (2013), pp. 42–72 (cit. on p. 7).

[BCG06] Lucio Bianco, Giuseppe Confessore, and Monica Gentili. "Combinatorial aspects of the sensor location problem." In: *Annals of Operations Research* 144.1 (2006), pp. 201–234 (cit. on p. 8).

[BCR01] Lucio Bianco, Giuseppe Confessore, and Pierfrancesco Reverberi. "A Network Based Model for Traffic Sensor Location with Implications on O/D Matrix Estimates." In: *Transportation Science* 35.1 (2001), pp. 50–60 (cit. on p. 8).

[BF85] Richard L. Burden and J. Douglas Faires. *Numerical Analysis*. 3rd. Boston, MA: Prindle, Weber & Schmidt, 1985 (cit. on p. 66).

[BLRP16] Nikolaos Bekiaris-Liberis, Claudio Roncoli, and Markos Papageorgiou. "Highway Traffic State Estimation with Mixed Connected and Conventional Vehicles." In: *IFAC-PapersOnLine* 49.3 (2016), pp. 309–314 (cit. on pp. 9, 54).

[BQC15] Ashish Bhaskar, Ming Qu, and Edward Chung. "Bluetooth vehicle trajectory by fusing bluetooth and loops: Motorway travel time statistics." In: *IEEE Transactions on Intelligent Transportation Systems* 16.1 (2015), pp. 113–122 (cit. on p. 8).

[Bra+17] A. Brandi et al. "Model predictive control with state estimation for freeway systems." In: *Proceedings of the American Control Conference*. Seattle, USA, 2017, pp. 3536–3541 (cit. on p. 9).

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge: Cambridge University Press, 2004 (cit. on p. 33).

[Car+12] Malachy Carey et al. "A Review of Properties of Flow – Density Functions." In: *Transport Reviews* 32.1 (2012), pp. 49–73 (cit. on pp. 3, 33).

[Cas+08] Enrique Castillo et al. "The observability problem in traffic models: Algebraic and topological methods." In: *IEEE Transactions on Intelligent Transportation Systems* 9.2 (2008), pp. 275–287 (cit. on p. 8).

[Cas+13] Enrique Castillo et al. "Deriving the upper bound of the number of sensors required to know all link flows in a traffic network." In: *IEEE Transactions on Intelligent Transportation Systems* 14.2 (2013), pp. 761–771 (cit. on p. 8).

[Cas+14] Enrique Castillo et al. "Non-planar hole-generated networks and link flow observability based on link counters." In: *Transportation Research Part B: Methodological* 68 (2014), pp. 239–261 (cit. on pp. 8, 102).

[Cas+15] Enrique Castillo et al. "A State-of-The-Art Review of the Sensor Location, Flow Observability, Estimation, and Prediction Problems in Traffic Networks." In: *Journal of Sensors* 2015 (2015) (cit. on p. 8).

[D'A+14] Antonino D'Andrea et al. "A functional road classification with data mining techniques." In: *TRANSPORT* 29.4 (2014), pp. 419–430 (cit. on p. 79).

[Dag94] Carlos F. Daganzo. "The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory." In: *Transportation Research Part B* 28.4 (1994), pp. 269–287 (cit. on p. 3).

[Dag95] Carlos F. Daganzo. "The cell transmission model, part II: Network traffic." In: *Transportation Research Part B* 29.2 (1995), pp. 79–93 (cit. on p. 5).

[FHJ10] Tyrone Lucius Fernando, Hieu Minh Trinh, and Les Jennings. "Functional Observability and the Design of Minimum Order Linear Functional Observers." In: *IEEE Transactions on Automatic Control* 55.5 (2010), pp. 1268–1273 (cit. on p. 53).

[FSS18] Antonella Ferrara, Simona Sacone, and Silvia Siri. *Freeway Traffic Modelling and Control*. 1st Editio. Advances in Industrial Control. Springer International Publishing, 2018 (cit. on p. 9).

[Fu+16] Chenyi Fu et al. "Heterogeneous sensor location model for path reconstruction." In: *Transportation Research Part B: Methodological* 91 (2016), pp. 77–97 (cit. on p. 8).

[Fur90] Peter G. Furth. "Model of Turning Movement Propensity." In: *Transportation research record* 1287 (1990), pp. 195–204 (cit. on p. 41).

[GCZ17] Yuqi Guo, Yangzhou Chen, and Chiyuan Zhang. "Decentralized State-Observer-Based Traffic Density Estimation of Large-Scale Urban Freeway Network by Dynamic Model." In: *Information* 8.3 (2017), p. 95 (cit. on p. 9).

[GD08] Nikolas Geroliminis and Carlos F. Daganzo. "Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings." In: *Transportation Research Part B: Methodological* 42.9 (2008), pp. 759–770 (cit. on p. 6).

[GM12] M. Gentili and P. B. Mirchandani. "Locating sensors on traffic networks: Models, challenges and research opportunities." In: *Transportation Research Part C: Emerging Technologies* 24 (2012), pp. 227–255 (cit. on p. 8).

[GM18] M. Gentili and Pitu B. Mirchandani. "Review of optimal sensor location models for travel time estimation." In: *Transportation Research Part C: Emerging Technologies* 90.January (2018), pp. 74–96 (cit. on p. 7).

[GQZ17] Yiming Gu, Zhen (Sean) Qian, and Guohui Zhang. "Traffic State Estimation for Urban Road Networks Using a Link Queue Model." In: *Transportation Research Record: Journal of the Transportation Research Board* 2623.1 (2017), pp. 29–39 (cit. on p. 9).

[Gre+34] B D Greenshields et al. "A study of traffic capacity." In: *Highway Research Board*. Washington, USA, 1934, pp. 448–477 (cit. on p. 2).

[HB10] Juan C. Herrera and Alexandre M. Bayen. "Incorporation of Lagrangian measurements in freeway traffic state estimation." In: *Transportation Research Part B: Methodological* 44.4 (2010), pp. 460–481 (cit. on p. 9).

[He13] Sheng Xue He. "A graphical approach to identify sensor locations for link flow inference." In: *Transportation Research Part B: Methodological* 51.516 (2013), pp. 65–76 (cit. on pp. 8, 9, 23, 24).

[HL14] Shou Ren Hu and Han Tsung Liou. "A generalized sensor location model for the estimation of network origin-destination matrices." In: *Transportation Research Part C: Emerging Technologies* 40 (2014), pp. 93–110 (cit. on p. 8).

[HPC09] Shou Ren Hu, Srinivas Peeta, and Chun Hsiao Chu. "Identification of vehicle sensor locations for link-based network traffic applications." In: *Transportation Research Part B: Methodological* 43.8-9 (2009), pp. 873–894 (cit. on pp. 8, 28).

[Hua+20] Y. P. Huang et al. "A dynamic user equilibrium model for multi-region macroscopic fundamental diagram systems with time-varying delays." In: *Transportation Research Part B: Methodological* 131 (2020), pp. 1–25 (cit. on p. 6).

[Jab16] Saif Eddin Jabari. "Node modeling for congested urban road networks." In: *Transportation Research Part B: Methodological* 91 (2016), pp. 229–249 (cit. on p. 6).

[KD17] Victor L. Knoop and Winnie Daamen. "Automatic fitting procedure for the fundamental diagram." In: *Transportmetrica B: Transport Dynamics* 5.2 (2017), pp. 129–144 (cit. on p. 33).

[KMG06] Lawrence A Klein, Milton K Mills, and David R P Gibson. *Traffic Detector Handbook*. Tech. rep. Turner-Fairbank Highway REsearch Center, 2006, p. 462. arXiv: `arXiv:1011.1669v3` (cit. on pp. 7, 8).

[KTK21] Masao Kuwahara, Atsushi Takenouchi, and Katsuya Kawai. "Traffic state estimation by backward moving observers: An application and validation under an incident." In: *Transportation Research Part C: Emerging Technologies* 127.April (2021), p. 103158 (cit. on p. 9).

[Lad+18] Andres Ladino et al. "Density and flow reconstruction in urban traffic networks using heterogeneous data sources." In: *IEEE European Control Conference (ECC)*. Limassol, Ciprus, 2018, pp. 1679–1684 (cit. on p. 10).

[Led05] Michel Ledoux. *The Concentration of Measure Phenomenon*. Ed. by Peter Landweber et al. Vol. 89. Mathematical Surveys and Monographs. Providence, Rhode Island: American Mathematical Society, 2005. Chap. 89, p. 181 (cit. on p. 107).

[LHP17] Han Tsung Liou, Shou Ren Hu, and Srinivas Peeta. *Estimation of Time-Dependent Intersection Turning Proportions for Adaptive Traffic Signal Control under Limited Link Traffic Counts from Heterogeneous Sensors*. Tech. rep. NEXTRANS Center, Purdue University, 2017 (cit. on p. 6).

[LW55] M J Lighthill and G B Whitham. "On kinematic waves II. A theory of traffic flow on long crowded roads." In: *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 229.1178 (1955), pp. 317–345 (cit. on p. 2).

[LWK16] E. Lovisari, Carlos Canudas-de Wit, and A. Y. Kibangou. "Density/Flow reconstruction via heterogeneous sources and Optimal Sensor Placement in road networks." In: *Transportation Research Part C: Emerging Technologies* 69 (2016), pp. 451–476 (cit. on pp. 8, 10).

[MBH07] L Mihaylova, R Boel, and A Hegyi. "Freeway Traffic Estimation Within Particle Filtering Framework." In: *Automatica* 43.2 (2007), pp. 290–300 (cit. on p. 9).

[ML19] Guilhem Mariotte and Ludovic Leclercq. "Flow exchanges in multi-reservoir systems with spillbacks." In: *Transportation Research Part B: Methodological* 122 (2019), pp. 327–349 (cit. on p. 6).

[MN19] J. R. Magnus and H. Neudecker. *Matrix Differential Calculus with Applications in Statistics and Econometrics*. Ed. by John Wiley & Sons. 3rd. 2019 (cit. on p. 43).

[Mun+03] L. Munoz et al. "Traffic density estimation with the cell transmission model." In: *Proceedings of the 2003 American Control Conference, 2003.* Vol. 5. IEEE, 2003, pp. 3750–3755 (cit. on p. 9).

[New02] G. F. Newell. "A simplified car-following theory: A lower order model." In: *Transportation Research Part B: Methodological* 36.3 (2002), pp. 195–205 (cit. on p. 1).

[Ng12] Man Wo Ng. "Synergistic sensor location for link flow inference without path enumeration: A node-based approach." In: *Transportation Research Part B: Methodological* 46.6 (2012), pp. 781–788 (cit. on pp. 8, 28, 30).

[NWK19] Muhammad Umar B Niazi, Carlos Canudas-de Wit, and Alain Y Kibangou. "Average observability of large-scale network systems." In: *2019 18th European Control Conference (ECC)* 694209 (2019), pp. 1506–1511 (cit. on pp. 11, 53).

[PL20] Mahendra Paipuri and Ludovic Leclercq. "Bi-modal macroscopic traffic dynamics in a single region." In: *Transportation Research Part B: Methodological* 133 (2020), pp. 257–290 (cit. on p. 6).

[PN06] R. Pueboobpaphan and T. Nakatsuji. "Real-Time Traffic State Estimation on Urban Road Network: The Application of Unscented Kalman Filter." In: *Applications of Advanced Technology in Transportation*. Reston, VA: American Society of Civil Engineers, 2006, pp. 542–547 (cit. on p. 9).

[Ric56] Paul I. Richards. "Shock Waves on the Highway." In: *Operations Research* 4.1 (1956), pp. 42–51 (cit. on p. 2).

[Ros+20] Majid Rostami Shahrbabaki et al. "State estimation in urban traffic networks: A two-layer approach." In: *Transportation Research Part C: Emerging Technologies* 115 (2020), p. 102616 (cit. on pp. 1, 10).

[RV17] Marco Rinaldi and Francesco Viti. "Exact and approximate route set generation for resilient partial observability in sensor location problems." In: *Transportation Research Part B: Methodological* 105 (2017), pp. 86–119 (cit. on p. 8).

[RVWF19] Martin Rodriguez-Vega, Carlos Canudas-de Wit, and Hassen Fourati. "Location of turning ratio and flow sensors for flow reconstruction in large traffic networks." In: *Transportation Research Part B: Methodological* 121 (2019), pp. 21–40 (cit. on pp. 42, 61).

[Sae+20] Ramin Saedi et al. "Estimating network travel time reliability with network partitioning." In: *Transportation Research Part C: Emerging Technologies* 112.January (2020), pp. 46–61 (cit. on p. 6).

[SC74] P. N. Shivakumar and Kim Ho Chew. "A Sufficient Condition for Nonvanishing of Determinants." In: *Proceedings of the American Mathematical Society* 43.1 (1974), p. 63 (cit. on p. 22).

[Seo+17] Toru Seo et al. "Traffic state estimation on highway: A comprehensive survey." In: *Annual Reviews in Control* 43 (2017), pp. 128–151 (cit. on pp. 1, 8, 9).

[SII17] Tomonori Sadamoto, Takayuki Ishizaki, and Jun Ichi Imura. "Average state observers for large-scale network systems." In: *IEEE Transactions on Control of Network Systems* 4.4 (2017), pp. 761–769 (cit. on p. 11).

[SKA15a] Toru Seo, Takahiko Kusakabe, and Yasuo Asakura. "Estimation of flow and density using probe vehicles with spacing measurement equipment." In: *Transportation Research Part C: Emerging Technologies* 53 (2015), pp. 134–150 (cit. on p. 9).

[SKA15b] Toru Seo, Takahiko Kusakabe, and Yasuo Asakura. "Traffic State Estimation with the Advanced Probe Vehicles Using Data Assimilation." In: *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC* 2015-Octob (2015), pp. 824–830 (cit. on p. 9).

[Sug99] Yuki Sugiyama. "Optimal velocity model for traffic flow." In: *Computer Physics Communications* 121 (1999), pp. 399–401 (cit. on p. 2).

[SW14] Ye Sun and Daniel B. Work. "A distributed local Kalman consensus filter for traffic estimation." In: *Proceedings of the IEEE Conference on Decision and Control* 2015-Febru.February (2014), pp. 6484–6491 (cit. on p. 9).

[TI07] Chris M.J. Tampère and L. H. Immers. "An extended Kalman filter application for traffic state estimation using CTM with implicit mode switching and dynamic parameters." In: *IEEE Intelligent Transportation Systems Conference*. Seattle, USA, 2007, pp. 209–216 (cit. on p. 9).

[TK13] Martin Treiber and Arne Kesting. *Traffic flow dynamics: Data, models and simulation*. Ed. by Springer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 1–503 (cit. on pp. 2, 8, 9, 33).

[Tra11] Transportation Research Board. "75 Years of the Fundamental Diagram for Traffic Flow Theory. Greenshields Symposium." In: *Transportation Research Circular* E-C149.June (2011), pp. 45–62 (cit. on p. 3).

[Vit+14] Francesco Viti et al. "Assessing partial observability in network sensor location problems." In: *Transportation Research Part B: Methodological* 70 (2014), pp. 65–89 (cit. on pp. 8, 30).

[Viv+15] Carlos Vivas et al. "Distributed consensus-based switched observers for freeway traffic density estimation." In: *2015 IEEE 54th Annual Conference on Decision and Control (CDC)*. Osaka, Japan, 2015, pp. 3445–3450 (cit. on p. 9).

[WK+15] Femke van Wageningen-Kessels et al. "Genealogy of traffic flow models." In: *EURO Journal on Transportation and Logistics* 4.4 (2015), pp. 445–473 (cit. on p. 1).

[WOK12] Carlos Canudas-de Wit, Luis Leon Ojeda, and Alain Y. Kibangou. "Graph constrained-CTM observer design for the Grenoble south ring." In: *IFAC Proceedings Volumes* 45.24 (2012), pp. 197–202 (cit. on p. 9).

[Xu+16] Xiangdong Xu et al. "Robust network sensor location for complete link flow observability under uncertainty." In: *Transportation Research Part B: Methodological* 88 (2016), pp. 1–20 (cit. on p. 8).

[Zha02] H. M. Zhang. "A non-equilibrium traffic model devoid of gas-like behavior." In: *Transportation Research Part B: Methodological* 36.3 (2002), pp. 275–290 (cit. on p. 4).